IMPROVEMENTS TO BCL::FOLD DE NOVO PROTEIN STRUCTURE PREDICTION

By

Sten Heinze

Dissertation

Submitted to the Faculty of the
Graduate School of Vanderbilt University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY
in
Chemistry

May, 2015
Nashville, Tennessee

Approved:
Professor Jens Meiler
Professor Terry P. Lybrand
Professor Prasad Polavarapu
Professor B. Andes Hess
Professor Jarrod Smith

To Jessica

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Jens Meiler for his encouragement and support during my graduate career. His unshakable confidence in me allowed me to overcome obstacles and grow as a researcher and person. I would also like to thank my committee members Dr. Terry Lybrand, Dr. Prasad Polavarapu, Dr. Andes Hess, Dr. Jarrod Smith, and Dr. David Tabb for their guidance and support.

The last years have been an invaluable experience. With current and former members of the Meiler lab I have spent countless hours asking many and answering some research questions, and lots of other enjoyable lab activities. Thank you for the help you provided, and the fun times we had.

Graduate school has a lot of ups and downs, and it has not always been easy. I am fortunate that I could always count on my friends, my sister, and my parents, even when they were far away. Finally, I would like to thank my wife Jessica. She has been understanding, encouraging and challenging me; she has always been there for me. This dissertation would not have been possible without her.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This dissertation enhances the BCL::Fold protein structure prediction method by directly improving aspects of the BCL::Fold algorithm and scoring itself, and by laying the ground work for future extensions. Chapter I introduces the importance of understanding protein structure. It gives an overview of the field of protein structure prediction and describes the significance of our approach. Chapter I was written for this dissertation.

Chapter II describes the BCL::Align sequence alignment and fold recognition method. It is based on the publication entitled “BCL::Align - sequence alignment and fold recognition with a custom scoring function online” by Elizabeth Nguyen, Jarrod Smith, Sten Heinze, Nathan Alexander, and Jens Meiler.

Chapter III investigates the quality of BCL::Fold models predicted for the tenth Critical Assessment of Structure Prediction (CASP) experiment, and implements two new criteria to measure if loops and sheet arrangements are native-like. This chapter is based on the manuscript with the title “CASP10 - BCL::Fold efficiently samples topologies of large proteins” by Sten Heinze, Daniel K. Putnam, Axel W. Fischer, Tim Kohlmann, Brian E. Weiner, and Jens Meiler.

Chapter IV presents the incorporation of small angle X-ray scattering (SAXS) experimental data as restraints into BCL::Fold. The discriminatory power of SAXS data to distinguish protein models of different topology is measured, and the improvements in model quality for filtering and building models with SAXS restraints was benchmarked. Chapter IV is based on a manuscript in preparation with the title “BCL::Fold protein topology prediction guided by SAXS profiles” with contributions by Daniel K. Putnam, Sten Heinze, and Jens Meiler.

Chapter V provides a summary of the major conclusions for this dissertation and relates the findings to other work in the field. This chapter was written for this dissertation.

The Appendix captures the details of the protocols used for the protein structure prediction and analysis of BCL models for Chapter III.

CHAPTER I.

INTRODUCTION

Proteins play a vital role in the cell

Proteins are the largest non-aqueous fraction in cells and account for about half of a cell's dry mass (Freitas, 1999; Oliveira, Nielsen, & Forster, 2005). They play a pivotal role in many of the diverse functions of a cell, including maintenance of the structure and shape (Wickstead & Gull, 2011); mechanical functions like motion (Wickstead & Gull, 2011); catalyzing metabolic reactions; and transport of molecules and signals. Because of this pervasive involvement in essential functions of the cell, any changes to the stability or structure of a protein can disrupt its function which has an effect on the fitness of the cell and the organism the cell is part of (Worth, Preissner, & Blundell, 2011). Adverse changes like that can be caused by mutations in the sequence of amino acids a protein is constructed of.

Such disruptive mutations have been shown for all functions that proteins have. Mutations of neuronal intermediate filament (IF) and microtubule-associated protein Tau, both associated with the cytoskeleton, have been linked to neurodegenerative diseases like Alzheimer's disease (AD), amyotrophic lateral sclerosis (ALS), and Parkinson's diseases (PD), among others (Cairns, Lee, & Trojanowski, 2004). More than 400 mutations are known for phenylalanine hydroxylase (PAH) making it unable to catalyze the conversion of phenylalanine to tyrosine, which causes high levels of phenylalanine in the blood and affecting the brain by phenylalanine competing for the large neutral amino acid transporter (LNAA) to cross the blood-brain barrier, which leads to insufficient amounts of the other amino acids transported by the LNAA (Fölling, 2009; Pietz et al., 1999); Cystic fibrosis (CF) is caused by the deletion of F508 (phenylalanine at position 508) in the cystic fibrosis transmembrane

regulator (CFTR), an ATP-binding cassette (ABC) transporter that transports chloride ions (Ratjen, 2009); Both K-Ras (mutations in 50% of colon cancers) and B-Raf (mutations in 66% of malignant melanomas) are part of ERK signaling pathways, promoting cancer cell migration by phosphorylation of myosin light chain kinase and others, and controlling the apoptosis regulator Bcl-2 family of proteins increasing survival of cancer cells (Kim & Choi, 2010). Despite its length, this enumeration is far from complete and many more examples have been found (Chaudhuri & Paul, 2006).

The protein sequence defines structure and function

Proteins molecules are created as unbranched polymers, sequences, of amino acids. The information of which amino acids make up a particular protein and in which order to connect them, i.e. the genetic information, is stored in deoxyribonucleic acid (DNA) molecules. To produce proteins, a flow of information needs to occur from the DNA molecules to the protein; this is known as the central dogma of molecular biology (F. Crick, 1970).

DNA molecules consist of a sequence of 4 nucleobases, adenine, guanine, cytosine, and thymine, in a specific order. Each triplet of nucleobases, called a codon, encodes one of 20 (+2) amino acids or a stop signal; this set of translation rules from codons into amino acids is called the genetic code (F. H. Crick, Barnett, Brenner, & Watts-Tobin, 1961; Jukes & Osawa, 1990). The amino acids corresponding to each codon are connected by forming a peptide bond between the carboxyl group of one amino acid with the amino group of the following amino acid. All proteinogenic amino acids have the same handedness and are L-amino acids, except Glycine, which has no stereo center. Such an amino acid sequence is the primary structure of a protein.

In its native environment in the cell, the sequence of a protein is thought to fold by itself into the 3D structure of its lowest energy, commonly referred to as the protein's native state. This was first shown

for ribonuclease, which after being denatured refolds itself when placed into a suitable, non-denaturing solution (Anfinsen, 1973). Thus the amino acid sequence defines the secondary structure, which is the set of hydrogen bonds between the backbone atoms (amide-hydrogen and carboxyl-oxygen), and the tertiary structure, which are the bonds between side chain atoms and other side chain atoms as well as other backbone atoms.

The structure of a protein, as defined by its amino acid sequence, is able to perform the function it is intended for. Consequently, an understanding of the structure is necessary to understand the functioning or non-functioning of a protein's structure. The easiest method to investigate protein structure is by conducting experiments on the protein.

The structure of many proteins was determined experimentally

Several experimental methods have been developed to determine the three-dimensional structure of a protein to atomic detail. The most frequently used ones are X-ray crystallography, nuclear magnetic resonance (NMR) spectroscopy, and cryo-electron microscopy (Berman, 2000). X-ray crystallography uses crystallized protein to diffract a monochromatic beam of X-rays. The X-rays are elastically scattered by the electrons in the crystal. The periodic arrangement of the crystal causes the scattered X-rays to constructively add for certain scattering angles and to destructively interfere in others according to Bragg's law (Bragg & Bragg, 1913). From the angles and intensities of the diffracted X-rays, the position of the electron density can be constructed, from which the positions of the atoms can be derived.

NMR spectroscopy exploits that some atomic nuclei have a nuclear spin and an associated magnetic moment. When the atomic nuclei is placed in an external magnetic field generated e.g. in an NMR instrument, the magnetic moment precesses around the direction of the external magnetic field with a characteristic frequency, the Larmor frequency. Since the external magnetic field is influenced locally by

the magnetic and electronic environment, the Larmor frequency differs slightly even for nuclei of the same type. By measuring the Larmor frequencies, the surrounding atoms can be determined and thus the structure of a molecule (Howard, 1998).

Cryo-electron microscopy takes many images of the protein sample frozen at cryogenic temperature (Kuhlbrandt, 2014c; Milne et al., 2013). Since the sample protein is contained in different orientations in the frozen solution, the images are taken from different perspectives, i.e. with different projection angles. These images are combined computationally into a 3D molecular volume (Scheres, 2014). Even though a transmission electron microscope (TEM) works similar to a light microscope in that it has an illumination source and lenses to focus, it is able to achieve much higher resolution because it is using electrons instead of photons. Since the electrons can e.g. break bonds and thus hamper with the structure determination, keeping the biological samples at low temperatures reduces the electron beam damage. The cryogenic sample preparation became only possible with the understanding of extremely rapid freezing that avoids crystallization of water and creates vitrified ice, which preserves the sample (Adrian, Dubochet, Lepault, & McDowall, 1984).

While all these methods are able to determine protein structures in atomic or near-atomic detail, and despite their continuous improvement, they are not applicable to all types of proteins equally.

Certain proteins are challenging for experimental methods

It remains difficult to determine the structure for large macromolecular assemblies and membrane proteins. This is reflected in the distribution of structures in the PDB; only ~2% of the determined protein structures are of membrane proteins (Kozma, Simon, & Tusnady, 2013), only ~2% are assemblies larger than 500kDa (Dutta & Berman, 2005).

X-ray crystallography requires the protein of interest to be crystallized, placing it in a non-physiological environment (Billeter, 1992). Generally, larger crystals with more copies in each dimension diffract to higher intensity and allow better resolution than microcrystals; however, especially for the underrepresented groups of proteins, even crystals large in their physical size contain only few copies because of the large protein or assembly size and the resulting large unit size. Additionally, crystallization can introduce crystal contacts between copies that are not biologically relevant (Kobe et al., 2008). Flexible parts of a protein that are present in different conformations in the crystal do not diffract to high intensity and no position will be determined for those electrons and atoms. Further developments of X-ray crystallography to reduce the crystal size can reduce but not completely avoid the problems caused by the required crystallization of a protein of interest; for example, it is now possible to determine structures from nano-crystals using femtosecond pulses from a free electron laser source (Chapman et al., 2011).

NMR spectroscopy might be better suited to determine the structure of a protein that cannot be crystallized or is flexible. By nature of its mechanism, NMR is limited to nuclei that have a magnetic moment, which are often isotopes that are not the most abundant in nature (e.g. ^{13}C has an abundance of 1.1 %, ^{15}N has an abundance of only 0.4 %). Thus, to prepare for NMR experiments, specifically prepared media have to be used to produce protein sample containing NMR active isotopes. For large samples with high numbers of NMR active atoms, the complexity of an NMR spectrum increases, making it difficult to determine the structure. By using segment- and site-selective labeling (Weigelt, Wikstrom, Schultz, & van Dongen, 2002; Yu, 1999), the maximum size for proteins whose structure can be determined by a solution NMR experiment has been increased to 100 kDa (Yu, 1999). This is still insufficient for large assemblies.

Cryo-electron microscopy on the other hand has a lower size limit, which was 170 kDa so far (Scheres, 2014), but it was recently estimated to be about 38 kDa for images of ideal quality (Kuhlbrandt, 2014c).

Despite recent improvements in detector technology and computational algorithms, the best resolution of 3-3.2 Å (Kuhlbrandt, 2014a; Zhou, 2011) is lower than what X-ray crystallography or NMR spectroscopy are able to achieve and barely allows to determine side chain conformations. Structural homogeneity of samples is required; therefore structures that are not sufficiently robust, e.g. lipid bilayers, or flexible, are causing problems when trying to determine a structure at atomic resolution. Similarly, very large complexes increase errors due to focus variation across the depth of the particle, making high resolution structure determination difficult (Zhou, 2011).

Even though experimental protein structure elucidation methods still advance swiftly, a number of conditions exist for which no atomic resolution structure can be experimentally determined by any of these methods. In some cases limited experimental data can still be obtained, but is insufficient to create an atomic resolution model. In these cases, a computational approach might be able to provide some insight.

Structure prediction can guide experiments

In situations of limited experimental information insufficient for creating a model, or in absence of any experimental information, computational methods can be applied to predict the tertiary structure of a protein. Such predicted structure models are often not as reliable as models based on experimental results, but they allow to create hypothesis that can be tested in further experiments. Insights gained in this way can be used to refine the computational method and the model that describes the structure or function of the protein in an iterative fashion.

Depending on how much information about the protein of interest is available, different approaches are able to predict a model with the highest accuracy. Available information also includes identified similar

proteins, because even two proteins of low sequence similarity can share a number of structural properties like secondary structure composition or protein fold.

De novo structure prediction

De novo structure prediction predicts a tertiary structure model solely from the sequence information of the protein. Therefore it can be used when no or insufficient experimental information is available and when no structural information from structures of similar proteins can be derived.

Predicting protein structures de novo is an incredibly complex problem because of the extremely large number of degrees of freedom proteins have. Even for small proteins the number of degrees of freedom is so large that it is impossible to sample and evaluate all conformations computationally. Despite this, proteins fold into their native state within short amounts of time, usually correlating with their size (Ivankov & Finkelstein, 2004; Naganathan & Munoz, 2005), from about ~2 microseconds for a 35 amino acid protein (Kubelka, Hofrichter, & Eaton, 2004) to an hour for adenylate kinase (H. Zhang, Sheng, Pan, & Zhou, 1998). This is known as Levinthal's paradox. Thus structure prediction methods must employ simplifications in order to predict a protein's conformation and avoid to try to sample and evaluate all possible ones (Zwanzig, Szabo, & Bagchi, 1992).

Such simplifications have been done at varying degrees, from molecular dynamics simulation with molecular mechanics force fields that include all atoms (Hagler, Huler, & Lifson, 1974; Hagler & Lifson, 1974; S. J. Weiner et al., 1984), to Rosetta, which, in coarse grained mode, includes all amino acids with only their C-beta centroid (Simons, Kooperberg, Huang, & Baker, 1997), to BCL::Fold, which represents a model only by its secondary structure parts (Karakas et al., 2012). Since protein structure prediction methods are limited in the protein size they can predict by the compute time necessary, higher amounts of simplifications are desirable to enable structure predictions for larger proteins.

Even limited experimental information that is by itself insufficient to create a model can often be used in conjunction with de novo methods. Such information limits the sampling space, i.e. the conformations that the prediction algorithm creates and evaluates, to conformations that are in agreement with the experimental data. Among other methods, the Rosetta and BCL protein structure prediction methods can utilize limited experimental information (Mao, Tejero, Baker, & Montelione, 2014; B. E. Weiner et al., 2014).

Template based structure prediction

That proteins of similar sequences fold into similar structures (Finn et al., 2008; Murzin, Brenner, Hubbard, & Chothia, 1995; Orengo et al., 1997), opens up another approach for protein structure prediction. If a protein of known structure with a sequence similar to the protein of interest can be identified, the known structure can be used as a template for the structure of the protein of interest. The similarity of sequences can be determined by alignment methods like Blast (Altschul, Gish, Miller, Myers, & Lipman, 1990) or BCL::Align (Dong, Smith, Heinze, Alexander, & Meiler, 2008).

Compared to de novo methods, template based structure prediction is fast and has no size limitation. But the need for a template limits this approach to proteins for which a suitable template protein with the same fold can be identified, which is only possible if the sequence similarity of two proteins is higher than 20-25 % (Chung & Subbiah, 1996; Rost, 1999).

Some methods try to evade the need for a full template by finding smaller templates that exhibit similarity only for parts of the protein of interest, and the combining of several partial templates to get a complete prediction. I-Tasser is one such method (Y. Zhang, 2008). It threads the sequence of the protein of interest onto potential templates, which are split into template fragments and reassembled into the final predicted model.

To measure how well all the different approaches and methods predict protein structures they have to be evaluated.

Evaluation of structure prediction methods

To evaluate the performance of a structure prediction method, and allow comparisons between different methods, each prediction method is benchmarked to measure how accurate their predictions are. For these comparisons, proteins are used, whose structure was already experimentally determined. The models predicted by a method are then compared to the experimentally determined structure, called the native structure, and a measure of how similar the models are to the native can be calculated. The results for different methods can then be compared.

Several different measures exist to compare two protein structures. The most frequently used ones are the Root Mean Square Deviation (RMSD) (Carugo & Pongor, 2001; Coutsias, Seok, & Dill, 2005; Kabsch, 1976), and the Global Distance Test (GDT) (Zemla, Venclovas, Moult, & Fidelis, 2001; Zemla, Venclovas, Moult, & Fidelis, 1999). RMSD calculates the deviation of all corresponding atoms after superimposing the two structures. For the GDT, the largest set of C_α atoms is determined, whose distance to the native is less than a certain cutoff. The GDT is then calculated as the sum of the fractions of atoms that deviate less than the cutoffs of 1, 2, 4 and 8 Å.

To simulate the experiment of predicting a protein's structure without having its native structure to compare, every two years the Critical Assessment of Structure Prediction (CASP) (Moult, Fidelis, Kryshtafovych, Schwede, & Tramontano, 2014) experiment is conducted. In this experiment, the organizers obtain proteins that just have been or are in the process of being experimentally determined, but for which the structural biologists are holding back with publishing the experimental data. It is then the task of the predictors to create structure predictions in a truly blind fashion. At the end, all

predictions are evaluated by the RMSD and GDT measures, among others. The CASP experiment does not only ensure blind testing, it also creates a consistent environment that allows for direct comparisons of the different methods on the same data set.

CHAPTER II.

BCL::ALIGN - SEQUENCE ALIGNMENT AND FOLD RECOGNITION WITH A CUSTOM SCORING FUNCTION ONLINE

This work is based on the publication Dong et al. (2008). S. H. contributed to the sequence alignment algorithm and its implementation in the BCL.

Summary

BCL::Align is a multiple sequence alignment tool that utilizes the dynamic programming method in combination with a customizable scoring function for sequence alignment and fold recognition. The scoring function is a weighted sum of the traditional PAM and BLOSUM scoring matrices, position-specific scoring matrices output by PSI-BLAST, secondary structure predicted by a variety of methods, chemical properties, and gap penalties. By adjusting the weights, the method can be tailored for fold recognition or sequence alignment tasks at different levels of sequence identity. A Monte Carlo algorithm was used to determine optimized weight sets for sequence alignment and fold recognition that most accurately reproduced the SABmark reference alignment test set. In an evaluation of sequence alignment performance, BCL::Align ranked best in alignment accuracy (Cline score of 22.90 for sequences in the Twilight Zone) when compared with Align-m, ClustalW, T-Coffee, and MUSCLE. ROC curve analysis indicates BCL::Align's ability to correctly recognize protein folds with over 80% accuracy. The flexibility of the program allows it to be optimized for specific classes of proteins (e.g. membrane proteins) or fold families (e.g. TIM-barrel proteins). BCL::Align is free for academic use and available online at http://www.meilerlab.org/.

Introduction

Sequence alignment and fold recognition are key computational tools for predicting the evolutionary history of proteins and detecting structurally related proteins from their amino acid sequence. The importance of these methods continues to increase with the exponential growth of sequence databases driven by various genome projects (Benson, Karsch-Mizrachi, Lipman, Ostell, & Wheeler, 2007; Mewes et al., 1999). With the help of these tools, relationships are being determined between newly discovered sequences and existing sequence databases (Bairoch & Apweiler, 1998; Benson, Karsch-Mizrachi, Lipman, Ostell, & Wheeler, 2006) along with proteins of known structure collected in the protein data bank (Berman, 2000). While sequence similarity frequently accompanies structural similarity as well as evolutionary relation to a common ancestor (Castillo-Davis, Kondrashov, Hartl, & Kulathinal, 2004; Phillips, Janies, & Wheeler, 2000), one major goal of these comparisons is the assignment of a function to newly discovered sequences.

Yet it is known that many structurally homologous proteins can have very low sequence identity (Rychlewski, Jaroszewski, Li, & Godzik, 2000), and in these cases sequence alignment methods alone provide little information. Threading algorithms (Jones, 1999a; Lindahl & Elofsson, 2000) and sequence-only methods (Karplus, Barrett, & Hughey, 1998; Rychlewski et al., 2000) for fold recognition have been specifically developed to predict structural similarity. However, the accuracy of most sequence alignment methods as well as the reliability of fold recognition methods is greatly diminished when comparing sequences in the so-called “Twilight Zone” with less than 25% sequence identity (Rost & Sander, 1993; Thompson, Plewniak, & Poch, 1999).

Approaches to improve the accuracy of automatic sequence alignments start with the introduction of common substitution matrices such as PAM (Dayhoff, Schwartz, & Orcutt, 1978) or BLOSUM (Henikoff & Henikoff, 1992). The progressive algorithm (Feng & Doolittle, 1987; Hogeweg & Hesper, 1984)

implemented in MUSCLE (Edgar, 2004) uses probabilities derived from the PAM 240 matrix and position-specific gap penalties with iterative score refinement. ClustalW (Thompson, Higgins, & Gibson, 1994) also uses a progressive alignment method and improves its accuracy by weighting sequences, customizing substitution matrix usage and changing gap penalties depending on the surrounding residues. Align-m (Van Walle, Lasters, & Wyns, 2004) uses a non-progressive local approach to guide a global alignment. T-Coffee (Notredame, Higgins, & Heringa, 2000) combines information from global and local sequence alignments to determine an optimized alignment. However, BLAST (Altschul et al., 1990) and PSI-BLAST (Altschul et al., 1997) continue to dominate the field of sequence alignment tools with their rapid word-based algorithm and the iterative search using position-specific score matrices.

While there is some overlap between the tools used for sequence alignment and fold recognition, there is significant emphasis on secondary structure prediction in fold recognition methods. Recent sequence-based methods (Lindahl & Elofsson, 2000; Rychlewski et al., 2000) include predicted structural information when generating the sequence-structure alignment. ORFeus (Ginalski, Pas, et al., 2003) uses a scoring matrix based on the PSI-BLAST profile and secondary structure prediction from PSIPRED (Jones, 1999c). Threading-based algorithms like THREADER (Jones, 1999a) evaluate template-based models of the target sequence using residue contact and hydrophobicity scores in a double dynamic programming algorithm. K*sync (Chivian & Baker, 2006) is a recent hybrid of both approaches that uses various weight sets to create an ensemble of sequence–sequence alignments. Based on this ensemble a library of models is created from which the optimal model is selected by tertiary structure analysis and energy prediction.

It was shown that fold recognition can be improved by incorporating the output of several primary fold recognition approaches in a secondary approach. Such meta-servers work by forming the consensus of several primary methods using either artificial neural networks (P-Cons) (Lundstrom, Rychlewski,

Bujnicki, & Elofsson, 2001) or more straight-forward structure comparison tools (3D-Jury) (Ginalski, Elofsson, Fischer, & Rychlewski, 2003).

With the growing number of sequence analysis and fold recognition tools being developed, it became clear that different scoring schemes can perform quite differently depending on the protein class, sequence identity level, or type of problem (fold recognition vs. sequence alignment). In turn, researchers often need to invoke multiple tools to accomplish these tasks, and it is difficult to determine which method produces the most accurate result given a particular scenario.

In the present study we seek to address this shortcoming by introducing BCL::Align. The program gives the user maximum flexibility in tailoring the scoring function to fit the specific problem. The effective scoring function used by BCL::Align is a linear combination of various substitution matrices, position-specific scoring matrices, secondary structure predictions, chemical properties, and gap penalties. Here, the algorithms implemented in BCL::Align are described, and optimized parameter sets for four typical tasks are presented (sequence alignment and fold recognition in the 0-25% and 25-50% sequence identity regime). Results for the SABmark benchmark database (Van Walle, Lasters, & Wyns, 2005) are compared with other leading sequence alignment tools. The significance of the weights is discussed in terms of their importance for sequence alignment and fold recognition at different levels of sequence identity.

Materials and methods

Needleman and Wunsch algorithm is employed for generation of optimal pairwise sequence alignment

BCL::Align uses a standard dynamic programming algorithm (Needleman & Wunsch, 1970) to optimally align two sequences A and B of length m and n. In order to execute the alignment, a scoring scheme for matches as well as gaps needs to be provided (see next section). The dynamic programming algorithm will output the optimal score $S_{m,n}$ together with the alignment.

Dynamic programming solves optimization problems by dividing the problem into independent subproblems. Since the sequence alignment problem has optimal substructure, a subproblem can be defined as aligning prefixes of two sequences up to a point (i, j) with $0 \leq i \leq m$ and $0 \leq j \leq n$. To find the alignment with the highest score $S_{m,n}$, a two-dimensional matrix with the dimensions m and n is filled at each position (i, j) with the best score $S_{i,j}$ of these prefix sequences ("matrix filling"). The optimal score $S_{i,j}$ builds upon the best score computed so far. The second part of the algorithm - so-called "trace back" - starts at the lower right corner of the matrix which now contains the best possible score $S_{m,n}$. It traces back step-by-step the pathway through the matrix that lead to this optimal score, thereby generating the optimal alignment of the two sequences.

Setup of parametric scoring function as a sum of weighted Z-scores

The scoring function of BCL::Align is a weighted sum of multiple scoring schemes that have been successfully used in prior sequence alignment and fold recognition approaches (discussed in Section "Scoring function components were chosen from successful sequence alignment benchmarks and can be

easily extended"). The user can choose the individual weight of each scheme and BCL::Align will recalibrate them to add up to 100 %.

Raw scores obtained from each of the different scoring schemes are not directly comparable. Therefore all scores are first translated into Z-scores. For every scoring scheme, a random distribution was created by computing the score S for 10^6 arbitrarily chosen pairs of amino acids out of a representative database consisting of 1,800 protein sequences. This database was created by culling the PDB (Berman, 2000) for sequences with less than 25 % sequence identity (Wang & Dunbrack, 2003). For each of the different scores an average S_{av} and a standard deviation S_{sd} were derived (Table 1), which are used within BCL::Align to rescale all scores into Z-scores with $Z = (S - S_{\mathrm{av}})/S_{\mathrm{sd}}$.

Therefore, positive scores larger than 1 indicate that two positions align with a score that is at least one standard deviation above the average. Since the total score is a sum of weighted Z-scores, this statement holds not only for the individual scores but also for the total score, which makes all scores obtained with BCL::Align directly comparable even if the composition of the scoring function was altered.

Use of the affine gap penalty is essential for alignment of distant sequence homologs

The affine gap penalty approach (Barton & Sternberg, 1987) improves sequence alignment by customizing gap penalties to the sequence, which makes them length- and location-dependent. BCL::Align distinguishes gap open penalties P_{open} from gap extension penalties $P_{\mathrm{extension}}$. It also distinguishes boundary gaps at the beginning or end of an alignment P^{B} from enclosed gaps P^{E}. In turn, a total of four gap penalties are defined that can be chosen by the user. The total penalty for a gap is computed using $P = P_{\mathrm{open}} + \mathrm{length}{\cdot}P_{\mathrm{extension}}$.

Scoring function components were chosen from successful sequence alignment benchmarks and can be easily extended

Table 1 lists the parameter options available to the user. While substitution matrices of various sequence identity are available, the PAM250 (Dayhoff et al., 1978) and BLOSUM45 (Henikoff & Henikoff, 1992) matrices were used for sequence alignment because these matrices are most suitable for aligning sequences with low sequence identity. The logarithm of the probability of replacing amino acid i with j is used as the score.

Table 1. Adjustable parameters and gap penalties.

Description	Parameters	S_{av}[a]	S_{sd}[b]
Amino acid identity	Identity		
Substitution matrices	PAM 100, 120, 160, 250 (Dayhoff et al., 1978)	−0.0824	0.2498
	BLOSUM 90, 80, 62, 45 (Henikoff and Henikoff, 1992)	−0.0821	0.2273
Position-specific scoring matrix	BLAST profile (Altschul et al., 1997)	−0.0072	0.0881
Secondary structure predictions	PSIPRED (Jones, 1999)	−0.1431	0.4728
	JUFO (Meiler and Baker, 2003)	−0.0388	0.2451
	SAM (Karplus et al., 1998; Hughey and Korgh, 1996)	−0.0056	0.2076
Chemical properties	Steric parameter	−1.1514	0.8981
	Polarizability	−0.1061	0.0814
	Volume	−1.9938	1.5660
	Hydrophobicity	−1.0737	0.7871
	Isoelectric point	−1.6180	1.8058
Gap penalties	Open gap		
	Extension gap		
	Open boundary gap		
	Extension boundary gap		

[a] Average score for Z-score correction.
[b] Standard deviation for Z-score correction.

The BLAST profile is iteratively built from members of the homologous family by scanning a sequence database (Altschul et al., 1997). In this work, the BLAST profile was determined by 3 PSI-BLAST iterations at an E-value cutoff of 0.001. The logarithm of the scalar product of the probability vectors for position i

and j is used as the score. One advantage of using these parameters is that the scoring matrix obtained can be used directly for running PSIPRED and JUFO (see below).

The secondary structure predictions used in BCL::Align include PSIPRED (Jones, 1999c), JUFO (Meiler & Baker, 2003) and SAM (Hughey & Krogh, 1996; Karplus et al., 1998). The logarithm of the scalar product of the 3-state (helix, strand, coil) probability vectors for position i and j is used as the score.

The chemical properties used include sterical parameters, polarizability, volume, hydrophobicity, and the isoelectric point, which are also used as input for JUFO (Meiler & Baker, 2003). For scoring, the negative absolute difference for amino acids i and j is computed. After Z-score normalization, all five properties were combined with equal weights into a single score for weight optimization.

The SABmark benchmark database

For parameter optimization, we chose to use a subset of the 1.65 version of the SABmark reference alignment database (Van Walle et al., 2005), which is itself divided into two subsets. Sequences in the Superfamily subset have 25-50 % sequence identity and are divided into test groups that represent different SCOP superfamilies. The Twilight Zone subset has sequences with 0-25 % sequence identity and each test group represents a different SCOP fold.

SABmark also includes a second set of Twilight Zone and Superfamily subsets with the same sequences, plus the addition of up to the same number of false positive sequences. These false positives differ in fold from the true positives. They were selected from a BLAST search of the original sequences against a 70 % identity subset of SCOP. The database covers the entire known fold space, and each pairwise reference alignment is a consensus structural alignment provided by SOFI (Boutonnet, Rooman, Ochagavia, Richelle, & Wodak, 1995) and CE (Shindyalov & Bourne, 1998).

Because SABmark contained pairwise sequence alignments as well as fold information, we were able to use the benchmark to optimize the parameters for both the sequence alignment and fold recognition methods.

Optimizing the Cline score avoids over- and underprediction in sequence alignment

A total of eleven parameters and four gap penalties were optimized in our experiment (Table 1). For sequence alignment parameter and gap penalty optimization, we chose to maximize the Cline score (Cline, Hughey, & Karplus, 2002) as a measure of alignment quality, which is in agreement with previous publications that maximizing the developer's score (f_{d}) alone leads to overprediction while maximizing modeler's score (f_{m}) leads to underprediction (Edgar & Sjolander, 2004; Sauder, Arthur, & Dunbrack, 2000). Scores were calculated using the qscore program (http://www.drive4.com/qscore/) (Edgar, 2004).

ROC curve analysis predicts accuracy of fold recognition

For fold alignment parameter optimization, we performed a receiver operating characteristic (ROC) curve analysis on the rate of correct versus incorrect fold assignment. A ROC curve plots the false positive rate against the true positive rate. Calculating of the area underneath the ROC curve provides a measure of fold alignment accuracy, where an area of 50 % would represent a program with no ability to recognize folds. The area underneath the ROC curve was maximized during parameter optimization.

Parameter and gap penalty optimization using a Monte Carlo algorithm

For both the sequence alignment and fold recognition methods, we performed two different optimizations, one with Twilight Zone sequences with low (0-25 %) sequence identity and one with Superfamily sequences with intermediate (25-50 %) sequence identity. For sequence alignment, the parameter and gap penalty optimization was performed on 50 % of the Twilight Zone subset and 36 % of the Superfamily subset. For fold recognition, 45 % of the Twilight Zone subset and 22 % of the Superfamily subset were used for the training set.

Using a Monte Carlo approach, we started the optimization with random values between 0 and 1 for the parameters and values between -2 and 0 for the gap penalties. For 100 Monte Carlo iterations, we adjusted the weights for the parameters and gap penalties by a random value between -0.2 and 0.2, maximizing the Cline score for sequence alignment and the area under the ROC curve for fold recognition. Fifteen rounds of this optimization procedure were carried out on each subset and weights from the top ten scoring rounds were averaged to determine the optimal weight set. The most favorable range for a particular weight is defined by the average and standard deviation of the top ten scoring rounds of each trained subset.

Cross validation was used to avoid over-training

Since a subset of the SABmark database was used to determine the weight sets, we had to verify that the scores resulting from the parameter and gap penalty optimization were not affected by over-training. To do so, the scores for the trained and untrained subset were compared with each other. They were found to be within the standard deviation (Table 1), which validates that the scores taken from the weight optimization can be directly compared with other leading methods.

Performance assessment

We assessed the sequence alignment performance of BCL::Align using the entire SABmark database. The average Cline scores for pairwise alignments in a group were calculated, and those scores were averaged to determine the final Cline score for each subset of SABmark: Twilight Zone, Superfamilies, Twilight Zone with False Positives, and Superfamilies with False Positives.

Implementation

The benchmarking and testing methods were written in C (using MPI for automated load balancing across a number of processors), with additional scripts written in Perl. Parameter optimization and performance assessment were performed on the PowerPC Linux cluster of the Vanderbilt University Advanced Computing Center for Research and Education (ACCRE).

Results and discussion

Optimal parameters and gap penalties

Details of the training set are given in Table 2, along with the average of the top ten scoring rounds of the Monte Carlo optimization. The optimized sequence alignment training set had an average Cline score of 27 for Twilight Zone sequences and 49 for Superfamily sequences. For fold recognition, the area underneath the ROC curve for the optimized training set scored an average of 82 for both subsets.

Table 2. Training set on SABmark for parameter optimization.

Problem	Sequence identity level [a]	Fraction of SABmark database used [b]		Score [c]
Sequence alignment	Twilight Zone	50%	873 of 1740 seq.	27
	Superfamilies	36%	1197 of 3280 seq.	49
Fold recognition	Twilight Zone	45%	1552 of 3458 seq.	82
	Superfamilies	22%	1460 of 6526 seq.	82

[a] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.

[b] The fraction of the SABmark database used for weight optimization is given as a percentage and in absolute sequences.

[c] Cline scores are reported for sequence alignment methods and the area under the ROC curve is reported for fold recognition methods. All scores are multiplied by 100. The maximum for both scores is 100.

Table 3 and Table 4 give the distribution of the sequence-identity dependent optimal weight sets for BCL::Align parameters and gap penalties for sequence alignment and fold recognition. The standard deviation on most weights is five percentage points or less, which demonstrates the robust nature of the Monte Carlo optimization. However, we find that there is flexibility in the use of secondary structure elements for sequence alignment, particularly PSIPRED and JUFO. PSIPRED weights can vary up to 8 percentage points for the alignment of Twilight Zone sequences and 5 for Superfamily sequences. JUFO weights can vary up to 11 percentage points for Twilight Zone sequences and 8 for Superfamily sequences. The increase in standard deviation may be due to the various methods of secondary

structure prediction compensating for each other in weight value, making their individual weights vary from one round to another.

Table 3. Distribution of weights for parameters.

Problem	Sequence identity level [b]	PAM 250	BLOSUM 45	BLAST	PSIPRED	JUFO	SAM	Chemical properties [c]
Sequence alignment	Twilight Zone	0±0%	1±2%	36±5%	33±8%	16±11%	2±3%	11±3%
	Superfamilies	1±1%	2±2%	40±1%	35±5%	14±8%	1±2%	7±1%
Fold recognition	Twilight Zone	1±0%	19±6%	33±4%	30±4%	5±5%	5±5%	8±2%
	Superfamilies	0±1%	13±5%	18±5%	20±3%	18±3%	7±6%	24±4%

[a] Weight values, varying from 0 to 1.0, were normalized to calculate percentage of weight value out of 100%. Scores may not add to 100% due to rounding.
[b] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.
[c] Chemical properties include sterical parameters, polarizability, volume, hydrophobicity, and the isoelectric point.

For the gap penalties, we find that the same score is given by a consistent set of weights and the only range larger than 0.5 is found in the weight for the extension boundary gap for the alignment of sequences in the Twilight Zone subset at 0.6.

Table 4. Optimized weights for gap penalties.

Problem	Sequence identity level [a]	Open gap	Extension gap	Open boundary gap	Extension boundary gap
Sequence Alignment	Twilight Zone	−1.4±0.3	−0.1±0.1	−0.7±0.4	−0.3±0.6
	Superfamilies	−1.9±0.1	−0.1±0.1	−0.9±0.2	0.0±0.1
Fold recognition	Twilight Zone	−1.2±0.2	−1.3±0.2	−0.6±0.4	−0.2±0.1
	Superfamilies	−0.8±0.4	−1.4±0.4	−1.7±0.3	−0.1±0.1

[a]The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.

The relative weight of the parameters, expressed as percentages in Table 3, suggest that the BLAST profile and PSIPRED secondary structure information carry equal weight, within the standard deviation, for each of the four tasks. For sequence alignment, the BLAST profile has the highest average weight at 36 % for the Twilight Zone subset and 40 % for the Superfamily subset. This reiterates the power of position-specific scoring matrices created with PSI-BLAST as tools for sequence analysis. Amongst the secondary structure elements weights for alignment and fold recognition, we find that PSIPRED

consistently carries the largest weight, with JUFO and SAM following behind. Only in the fold recognition of the Twilight Zone sequences do we find that JUFO and SAM carry equal weight at an average of 5 %. For all other tasks, we find that JUFO outweighs SAM by over 10 %. It is remarkable that the sum of the three secondary structure prediction weights is the largest contribution to the composite scoring function for all four benchmark cases.

The chemical properties of amino acids carry more weight in aligning sequences from the Twilight Zone at 11 % compared to the 7 % for Superfamily sequences. However, we find that the chemical properties are even more important in fold recognition, carrying 8 % of the weight for the fold recognition of Twilight Zone sequences and 24 % of the weight for the Superfamily subset. The relative importance of the PAM and BLOSUM substitution matrices is minimal in sequence alignment with weights below 2 %, but we find that the BLOSUM matrix carries considerable weight in fold recognition at an average of 19 % for Twilight Zone sequences and 13 % for Superfamily sequences.

Large open gap and open boundary gap penalties were generally favored during parameter optimization of both the Twilight Zone and Superfamily subsets. The open gap penalty was -0.8 or more, and the open boundary gap penalty was greater than -0.6 for all fold recognition and sequence alignment tasks. Generally, the extension gap and extension boundary gaps were penalized less, which demonstrates the importance of the use of an affine gap penalty. We find that the extension boundary gap was penalized less than -0.3 for sequence alignment and fold recognition, as well as the extension gap for both sequence alignment tasks. However, for fold recognition there is a -1.3 penalty for Twilight Zone sequences and -1.4 for Superfamily sequences, that indicates a particular emphasis on a penalty of the extension gap for fold recognition.

Cross-validation confirms absence of over-training

The scores for the trained and untrained subsets of SABmark for each of the four tasks are given in Table 5. In the Twilight Zone subset, the untrained subset had a Cline score of 24 whereas the trained subset had a score of 23. For the Superfamily subset, the untrained subset scored 51, while the trained subset had a score of 49. The scores for the untrained subsets of SABmark for sequence alignment are higher than those of the trained subset, which provides evidence that the Monte Carlo optimization did not over-train the weight set, and thus the method is not biased towards this particular subset. Although the scores of the untrained subset are lower than those of the trained subset for fold recognition, the difference is still within 2 percentage points. Nevertheless, BCL::Align would still benefit from future benchmarking tests on fold recognition benchmark databases such as the Lindahl Benchmark for fold recognition sensitivity (Lindahl & Elofsson, 2000).

Table 5. Scores on trained and untrained subsets of SABmark with optimal weight set.

Problem	Sequence identity level [a]	Score for trained subset[b]	Score for test subset[b]
Sequence alignment	Twilight Zone	23	24
	Superfamilies	49	51
Fold recognition	Twilight Zone	87	86
	Superfamilies	88	86

[a] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.

[b] Cline scores are reported for sequence alignment methods and the area under the ROC curve is reported for fold recognition methods. All scores are multiplied by 100. The maximum for both scores is 100.

Comparison of sequence alignment methods

We compared the results of BCL::Align sequence alignment with Align-m (Van Walle et al., 2004), ClustalW (Thompson et al., 1994), T-Coffee (Notredame et al., 2000) and MUSCLE (Edgar, 2004) on the

SABmark benchmark database using the Cline score. Scores for the methods listed above are from Blackshields, Wallace, Larkin, and Higgins (2006). BCL::Align results on the entire SABmark benchmark database are shown in Table 6. In each subset, BCL::Align ranks the highest in alignment accuracy, which shows the superiority of BCL::Align's scoring function and the power of weight flexibility when compared to other programs that also use the dynamic programming algorithm (Figure 1). According to the data provided by Blackshields et al. (2006), ProbCons (Do, Mahabhashyam, Brudno, & Batzoglou, 2005) was the only program that consistently scored somewhat higher than BCL::Align. This is likely due to the fact that ProbCons does not employ dynamic programming but combines posterior-probabilities from pair-hidden Markov models (HMM) with a consistency-based method to determine scoring matrices.

Table 6. Performance comparison of multiple sequence alignment programs on SABmark[a].

	Superfamilies[b]		Twilight Zone[b]	
	No FP[c]	With FP[c]	No FP[c]	With FP[c]
Align-m	44.75	41.53	15.93	13.72
ClustalW	47.60	47.82	18.57	17.98
T-Coffee	50.20	45.58	20.80	16.94
MUSCLE	44.52	40.38	15.45	12.44
BCL Align	**50.74**	**50.80**	**23.02**	**23.66**

[a] Cline scores are reported for each multiple sequence alignment program. The highest score in each subset is displayed in bold. Scores for all methods except BCL::Align are from Blackshields et al. (2006).
[b] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.
[c] Subsets include the addition of up to the same number of false positive sequences. False positives differ in fold from the true positives and were selected from a BLAST search of the original sequences against a 70% identity subset of SCOP.

Performance in fold recognition

There is not a universal score for measuring fold recognition accuracy. To determine the fold recognition accuracy of BCL::Align on the SABmark benchmark database subsets that included false positives, an ROC curve analysis was performed. We find that BCL::Align has a strong performance, predicting the correct structure with 86 % accuracy for the Superfamily subset and 83 % accuracy for the Twilight Zone subset (Figure 2). However, the limiting factor for BCL::Align's ability to perform fold recognition is in the length of time it takes for the program to scan large databases in search of a matching fold, family and superfamily. Future improvements to increase the speed of BCL::Align using a word-based algorithm will allow for a more comprehensive study of the program's ability to perform fold recognition.

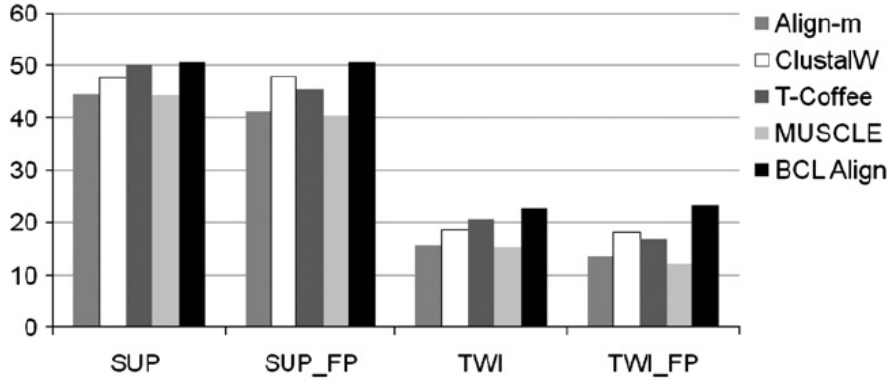

Figure 1. Performance comparison of multiple sequence alignment programs on SABmark. Cline scores are reported for each multiple sequence alignment program. Score for all methods except BCL::Align are from Blackshields et al. (2006).

Conclusions

Sequence alignment and fold recognition at varying levels of sequence identity benefits from the use of customized weight sets because of the emphasis of different parameters for each situation. For the Superfamily subset, fold recognition puts an average of 12 % more weight on chemical properties than sequence alignment. The BLOSUM45 substitution matrix carries over 10 % more weight in fold

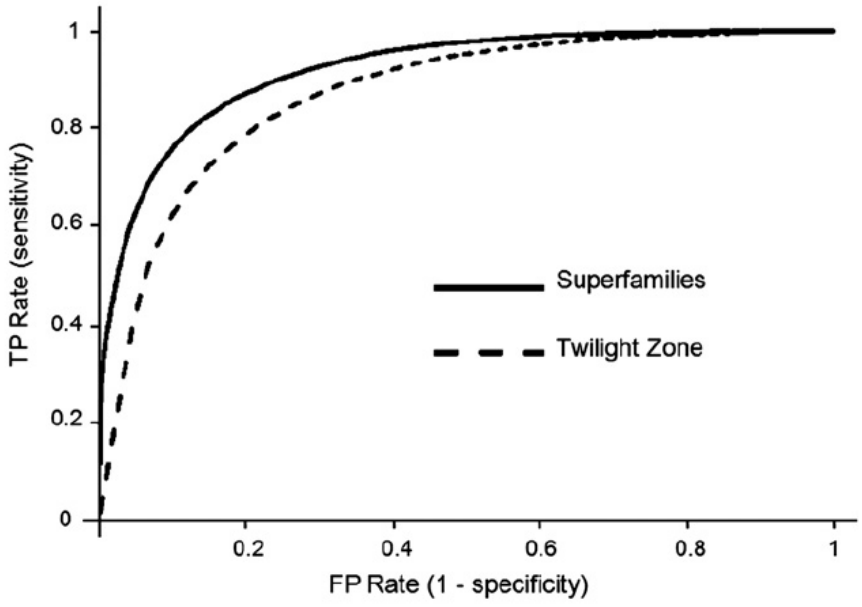

Figure 2. ROC curve analysis of fold recognition on SABmark.

recognition than sequence alignment. Of the secondary structure predictions, PSIPRED carries the most weight with 20-30 % on average for all categories. JUFO follows behind with weights between 5 and 18 %, and SAM has minimal involvement at less than 10 % weight in all categories. In all cases, however, large weights for the BLAST profile and affine gap penalties provide optimal alignment and fold recognition. With its optimal customized weight set, BCL::Align performed better than other dynamic-programming based methods with the highest rank in sequence alignment accuracy. With the future implementation of a faster word-based algorithm and the incorporation of HMM, we expect BCL::Align to have the efficiency to quickly align multiple sequences at once and perform fold recognition over

large databases of protein structures. BCL::Align is available on an online web server at http://www.meilerlab.org/.

CHAPTER III.

CASP10 – BCL::FOLD EFFICIENTLY SAMPLES TOPOLOGIES OF LARGE PROTEINS

This work is based on the publication Heinze et al. (2015). S. H. created the BCL::Fold CASP10 prediction pipeline; he predicted and submitted models for the targets; he analyzed most of the targets and identified potential improvements. He implemented and tested the loop angle score.

Summary

During CASP10 (the 10th Critical Assessment of protein Structure Prediction) in summer 2012 we tested BCL::Fold for prediction of free modeling (FM) and template-based modeling (TBM) targets. BCL::Fold assembles the tertiary structure of a protein from predicted secondary structure elements (SSEs) omitting more flexible loop regions early on. This approach enables the sampling of conformational space for larger proteins with more complex topologies. In preparation of CASP11 we analyzed the quality of CASP10 models throughout the prediction pipeline to understand BCL::Fold's ability to sample the native topology, identify native-like models by scoring and/or clustering approaches, and our ability to add loop regions and side chains to initial SSE-only models. The standout observation is that BCL::Fold sampled topologies with a GDT_TS score > 33 % for 12 of 18 and with a topology score > 0.8 for 11 of 18 test cases de novo. Despite the sampling success of BCL::Fold, significant challenges still exist in clustering and loop generation stages of the pipeline. The clustering approach employed for model selection often failed to identify the most native-like assembly of SSEs for further refinement and submission. It was also observed that for some β-strand proteins model refinement failed as β-strands were not properly aligned to form hydrogen bonds removing otherwise accurate models from the pool.

Further, BCL::Fold samples frequently non-natural topologies that require loop regions to pass through the center of the protein.

Introduction

Experimental structures in the protein data bank are biased towards small soluble proteins

The tertiary structure of a protein provides essential insights to its biological function in living organisms. Accordingly, experimental methods are applied to ascertain protein structure including X-ray crystallography and nuclear magnetic resonance (NMR) spectroscopy. Currently the Protein Data Bank (PDB), contains more than 89,258 proteins (December 2013) of which 79,585 (89 %) were elucidated by X-ray crystallography, 8,971 (10 %) by NMR, and the remainder by other technologies (Berman, 2000). Despite these efforts, the structures represented in the PDB are biased; 87,004 of the proteins in the PDB are soluble while only 2,254 (2.5 %) of the proteins represent membrane proteins (Berman, 2000). Further, the size distribution of proteins in the PDB is biased towards small proteins omitting many large macromolecular assemblies greater than 500,000 Da (2.0 %) (Berman, 2000, 2008; Dutta & Berman, 2005). This bias is due to the limitations of experimental methods for structure determination. Membrane proteins are underrepresented in the PDB because they are too large for NMR and their embedding in the two-dimensional membrane complicates formation of three-dimensional crystals required in X-ray crystallography (Bill et al., 2011). For membrane proteins up to approximately 1,000 folds remain to be determined (Hopf et al., 2012; B. E. Weiner, Woetzel, Karakas, Alexander, & Meiler, 2013). Large macromolecular assemblies are also underrepresented in the PDB because its protomers do not fold in isolation, they are difficult to crystallize, and they are too large for NMR spectroscopic methods (Alber et al., 2007). Thus, for many biologically relevant proteins only limited experimental

data can be collected with a combination of experimental techniques such as solid state NMR, cryo-electron microscopy (cryo-EM), electron paramagnetic resonance (EPR), mass spectrometry (MS), and small angle x-ray scattering (SAXS). On their own these datasets are insufficient for atomic-detail structure determination. One major justification to develop *de novo* protein structure prediction algorithms is to complement such limited experimental datasets.

De novo protein structure prediction needs a reduced search space

The cornerstone of *de novo* protein structure prediction methods is based on the assumption that (most) folded proteins exist in their lowest energy conformation (Anfinsen, 1973). Protein folding becomes an energy minimization process that depends on interaction of amino acids with the environment and other amino acids in the sequence. Finding the global minimum of the energy function on the energy landscape is challenging for several reasons including that the energy landscape contains many local minima. Currently, no universal method of identifying the global minimum of the energy function exists (Crippen, 1975). In practice, the conformational space of a protein is also far too large to be comprehensively searched with a highly accurate and therefore slow to compute energy function. Therefore, the conformational space is reduced by working with simplified protein representations, at least in the initial folding simulation. In effect this reduces the resolution of the energy function which allows more rapid calculation but decreases its accuracy to the point where the global energy minimum cannot be unambiguously detected and several local energy minima need to be considered.

Competing de-novo structure prediction software reduces the search space similarly. Rosetta addresses the sampling challenge by assembling protein models from three and nine residue peptide fragments (Chivian et al., 2005; Leaver-Fay et al., 2011; Simons et al., 1997). These fragments are determined from peptides of similar sequence and secondary structure extracted from other proteins in the PDB. For

proteins smaller than 80 residues Rosetta was able to predict atomic detail models in the absence of any experimental restraints for about 30 % of the test cases (Bradley, Misura, & Baker, 2005). For larger proteins up to around 150-180 residues Rosetta samples the correct topology about 50 % of the time (Bradley, Malmstrom, et al., 2005; Bradley, Misura, et al., 2005; Kaufmann, Lemmon, Deluca, Sheehan, & Meiler, 2010). Generally, Rosetta tends to perform better for α-helical proteins which is related to their reduced fold complexity. The complexity of a fold can be measured by contact order (CO) which is defined as the average sequence separation of residues in contact, i.e. residues whose C_β atoms are less than 8 Å apart (Baker, 2000; Grantcharova, Alm, Baker, & Horwich, 2001). As the complexity of protein topology increases (high contact order) the accuracy of the Rosetta prediction decreases (Baker, 2000; Bonneau, Ruczinski, Tsai, & Baker, 2002).

I-Tasser threads the target sequences through a library of PDB structures with a pair-wise sequence identity cut-off of 70 % to search for plausible protein folds. Rather than using a fixed set of three and nine residue peptide fragments, I-Tasser uses fragments of variable size that are identified by threading (Wu, Skolnick, & Zhang, 2007; Y. Zhang, 2007, 2013). The fragments are used to reassemble full-length models while the loop regions between fragments being constructed *de novo*. Critical to the success of I-Tasser is the identification of suitable templates to create the peptide fragments - a Pearson correlation coefficient of 0.89 for RMSD and 0.95 for TM-score (Y. Zhang, 2007). Generally, I-Tasser samples the correct topology about a third of the time for proteins up to 155 residues long with RMSD < 6.5 Å (Y. Zhang, 2007). I-Tasser shares the most critical limitation with Rosetta, the ready formation of long-range interactions between residues.

BCL::Fold was designed to overcome size and complexity limitations in de novo protein structure prediction methods

BCL::Fold is a *de novo* protein structure prediction algorithm based on the placement of disconnected secondary structure elements (SSEs) in three-dimensional space as previously published (Karakas et al., 2012; B. E. Weiner et al., 2013; Woetzel et al., 2012). This algorithm was developed to test the hypothesis that for many proteins the core responsible for thermodynamic stability is largely formed by SSEs. In this case, likely protein topologies could be detected from SSE-only models. Thereby, the size and contact order restrictions in protein structure prediction can be overcome by assembling disconnected, rather rigid SSEs, which reduces the search space substantially and allows the ready formation of non-local contacts (Karakas et al., 2012). A coarse grained knowledge based energy function identifies native-like SSE arrangements using a Monte Carlo simulated annealing sampling algorithm with Metropolis criterion (Karakas et al., 2012; B. E. Weiner et al., 2013; Woetzel et al., 2012). In contrast to I-Tasser or Rosetta, this algorithm is truly *de novo* as no fragments from the PDB are used. Loop regions between SSEs and side chains atoms are added to the model in subsequent steps using for example Rosetta (Baker & Sali, 2001; Rohl, Strauss, Chivian, & Baker, 2004; Sali & Blundell, 1993).

BCL::Fold uses a consensus of secondary structure prediction technologies to identify SSEs

Critical to the success of the BCL::Fold algorithm is the correct prediction of SSEs: α-helices, β-strands, coil regions, and trans-membrane spans from sequence. These predictions are obtained from a consensus prediction from PHD (Rost, 1996; Rost & Sander, 1994), PsiPred (Jones, 1999c; Ward, McGuffin, Buxton, & Jones, 2003), and Jufo9D (Leman, Mueller, Karakas, Woetzel, & Meiler, 2013; Meiler & Baker, 2003; Meiler, Muller, Zeidler, & Schmaschke, 2001) for soluble proteins. In addition to these methods we used Octopus (Viklund, Bernsel, Skwark, & Elofsson, 2008; Viklund & Elofsson, 2008)

and Jufo9D (Leman et al., 2013) for the trans-membrane span region of membrane proteins. The consensus prediction is used to build a pool of SSEs, which is input for protein folding.

A Monte Carlo Metropolis sampling algorithm positions SSEs in space

Protein models are assembled using a Monte Carlo sampling algorithm. Each iteration of the algorithm consists of a randomly selected modification to the current model. Modifications include the addition of an SSE from the SSE pool to the model; the removal of an SSE from the model; translational and rotational transformations of SSEs in the model; swapping of two SSEs; modifications of groups of SSEs (domains) consist of translating the domain; flipping; and shuffling the different SSEs.

After each modification, the model is evaluated by a knowledge based scoring function (Woetzel et al., 2012). This coarse grained scoring function is designed to evaluate the arrangement of SSEs in Euclidean space. It is a weighted sum of scoring terms that represent different aspects of SSEs of protein structures as observed in experimental structures like the preferred environment of amino acid types (buried or solvent exposed); the radius of gyration; an SSE packing and a strand pairing potential; a loop length potential; clash terms for amino acids and SSEs; and a loop closure penalty. The loop closure penalty limits the Euclidean distance between two consecutive SSEs to the maximum length a stretched out amino acid chain can bridge and applies a steep penalty for longer loop distances.

The evaluation with the Metropolis criterion results in one of four possible outcomes: 1) improved and accepted, if the calculated energy score is lower than the energy of the previous model; 2) accepted by the Metropolis criterion with a function taking the energy difference and the simulated temperature into account; 3) rejected if the score is higher than the previous model and rejected by the Metropolis criterion; 4) skipped, if the modification is not applicable to the model, for example swapping SSEs if the model contains only a single SSE. The probability of a step being accepted with higher energy is based on

the temperature used by the Metropolis criterion. BCL::Fold adjusts the temperature to achieve a ratio of accepted steps that reduces from 50 % to 20 % in the course of the simulation.

All scoring terms (except for the clash terms and the loop closure penalty) are statistically derived using Bayes' theorem from a divergent high resolution subset of the Protein Data Bank generated by the PISCES server with a maximum sequence identity of 25 % (Wang & Dunbrack, 2003, 2005), and then energies were approximated using the inverse Boltzmann relation.

The algorithm will continue generating modified models and evaluating them until a maximum number of 2000 steps is completed or no improvement in the score is found for 400 steps; this constitutes one folding stage. The folding process of one model has five stages which employ a decreasing number of modifications for large scale perturbations (e.g. swapping SSEs) and an increasing amount of small scale perturbations (e.g. bending an SSE). The lowest energy model within the trajectory will be saved as the resulting model for this run.

The algorithm will continue generating modified models and evaluating them until a maximum number of 2,000 steps is completed or no improvement in the score is found for 400 consecutive steps; this constitutes one folding stage. The folding process of one model has five assembly stages and one refinement stage; the assembly stages use modifications for large scale perturbations (for example, swapping SSEs) while the refinement stage replaces them with modifications of small scale perturbations (for example, bending an SSE). The lowest energy model within the trajectory will be saved as resulting model for this run.

The CASP10 experiment – a critical tool for development of techniques for protein structure prediction

To evaluate the accuracy of BCL::Fold in *de novo* protein structure prediction, we participated in the Critical Assessment of protein Structure Prediction (CASP10) experiment, which is held every two years (Moult, 2005; Moult, Fidelis, Kryshtafovych, & Tramontano, 2011). The double-blind experiment tests protein structure prediction methods objectively, because the experimentally determined structure is withheld from predictors, organizers and the assessors until the experiment is finished. After protein predictions have been made, the experimentally determined structures are revealed and the results are assessed. CASP10 contained the following categories: 1) Tertiary structure prediction which can be classified as: a) Template Based Modeling (TBM) – starting from a homologous protein template in the PDB. b) Free Modeling (FM) – no homologous template exists in the PDB; 2) Tertiary structure prediction with limited experimental information, e.g. amino acids in contact (Taylor, Bai, Tai, & Lee, 2013); 3) Residue-residue contact prediction (Monastyrskyy, D'Andrea, Fidelis, Tramontano, & Kryshtafovych, 2013); 4) Model refinement (Nugent, Cozzetto, & Jones, 2013); 5) Identification of disordered regions; 6) Function prediction; 7) Quality assessment (Kryshtafovych et al., 2013).

To maximally leverage CASP10 for testing BCL::Fold we assume all CASP10 targets to be FM targets

For some targets, templates can be found, i.e. proteins with similar sequence and known structure that can guide the prediction. Based on whether templates can be found and how similar the template structure is to the target structure, measured by the Global Distance Test/Total Score (GDT_TS) (Zemla et al., 1999), prediction for CASP10 targets is categorized as easy or hard Template Based Modeling (TBM easy if the maximal GDT_TS ≥ 50, hard if the maximal GDT_TS < 50), Free Modeling (FM) or a

combination of both (TBM/FM). The GDT_TS could obviously only be employed after the target structures were available; in the prediction process other measures like sequence similarity to proteins in the PDB were used to classify targets. To maximize the assessment of the BCL::Fold *de novo* protein structure prediction algorithm in CASP10 we treated all targets as FM targets, i.e. no homologous template from the PDB was used at any point of prediction.

Material and Methods

Secondary Structure and Trans-Membrane Span Prediction

In the first step, the secondary structure is predicted for soluble proteins using Jufo9D (Leman et al., 2013; Meiler & Baker, 2003; Meiler et al., 2001), PsiPred (Jones, 1999c; Ward et al., 2003), and ProfPHD (Rost, 1996; Rost & Sander, 1994). For membrane proteins Jufo9D (Leman et al., 2013) and Octopus (Viklund et al., 2008; Viklund & Elofsson, 2008) are used to detect secondary structure and trans-membrane spans. From the predicted secondary structures a pool is created for use by BCL::Fold as described before (Karakas et al., 2012). The pool is manually examined to ensure a complete as possible set of SSEs.

Fold Recognition and Domain Identification

Fold recognition methods combined in bioinfo.pl were used to see if the target sequence contains multiple domains (Ginalski, Elofsson, et al., 2003), and if proteins of those folds have been experimentally determined. If the fold recognition result indicated that the target has multiple domains, the SSE pool is split up into sub pools according to the domain boundaries.

BCL::Fold Folding Simulation

BCL::Fold is run next to produce 12,000 models for each domain of one target. Depending on the target, the soluble or membrane protocol is employed. For each model, a completeness estimate is calculated as a fraction of the sum of the sequence lengths of all SSEs in the models to the total sequence length of the target. Models that are 2 % less complete than the average model produced are removed.

Clustering to Identify Topologies that reside in wide Energy Funnels

After filtering the 12,000 models per target by completeness score, models were selected by three criteria for further refinement. The first method for selection was clustering by average RMSD linkage between models where the clusters ideally only contain models with the same fold. Cluster sizes varied with the largest clusters having a few hundred models and the smallest clusters containing a few or even a single model. Cluster radii leafs were between 0 and 18 Å with most at 10 Å. The RMSD cutoff was manually adjusted based on protein size and model similarity. Up to five models from each cluster were selected for further refinement. The second method for selection was ranking by the BCL scoring function. All filtered models were sorted by BCL sum score and the lowest scoring models were selected. The third method was only used if we successfully identified a template model of the target protein and models were pooled into a separate set. In this case the RMSD between the template and BCL generated models were computed. The models with the highest similarity (lowest RMSD) were selected for further refinement. Furthermore, in some cases the selected models were visually inspected in PyMOL to evaluate sequence length and Euclidean distances for later loop reconstruction. In this step some models were removed from further processing if loops went through the center of the protein core (Table 7).

Combining Domains into Complete Models

If the target consisted of multiple domains, models of all possible combinations of domains are created either by arranging the domains in space close to each other or, if possible, by aligning the domain models to a template. The domains do not have to be connected by creating a loop at this point, because all models consist of only SSEs; loops will be built in the next step.

Table 7. Clustering Statistics of CASP10 Targets folded by BCL::Fold.

Target	Folded Models	After Filtering	Top Cluster	Top Scoring	Top Homology
T0644	9980	4485	2	0	0
T0649	10000	5135	3	5	0
T0655	9980	4335	1	3	2
T0663	12000	6495	3	2	3
T0666	12000	5979	3	3	0
T0676	12000	6341	3	0	1
T0678	12000	6371	5	1	2
T0682	12000	5554	0	3	4
T0684	12000	5884	16	0	1
T0686	12000	6230	2	1	0
T0691	12000	6083	4	1	2
T0700	12000	6605	1	2	3
T0704	12000	5932	1	3	1
T0720	12000	6345	2	2	1
T0722	12000	8747	1	2	1
T0724	11999	5886	3	1	0
T0743	12000	6374	2	2	4
T0745	12000	6108	2	2	0

Loop construction using Cyclic Coordinate Decent

Once secondary structure elements (SSEs) have been placed, loop regions between SSEs must be built. This is accomplished by adding loop residues using 1) knowledge based potentials, 2) likely phi and psi backbone angles, and 3) cyclic coordinate descent. The first step is to dynamically add missing residues in the loop region. Residues are added with initial phi and psi angles derived from a probability distribution of experimentally observed angles. They are then perturbed and scored using a knowledge

based potential for native like angles. This potential has scoring terms that penalize clashes between atoms using van der Waals radii, compares the sequence length with the Euclidean distance, measures the gap between adjacent SSEs, incorporates angles derived from Ramachandran plots, and scores the likelihood that the distance between the SSEs can be closed by a loop. Once the initial residue coordinates of the loop region have been placed, cyclic coordinate descent (CCD) (Canutescu & Dunbrack, 2003) is used to minimize the distance between a freely moving and fixed set of coordinates to close a loop. In this step an additional penalty term is added to the scoring function that scores how close the residue at the loop end is to the pseudo residue at the N terminus of the target SSE. Adding loops is a two-step process of inserting the missing amino acids in a model and creating coordinates for them by Cyclic Coordinate Decent (CCD). Between 200 and 8,400 loop models were built depending on model size and complexity to achieve a sufficiently low BCL sum score, i.e. in a similar score range than the non-loop start model. Models with loops difficult to close were either modified to allow an easier loop closure by shortening the SSEs adjacent to the loop, or they were removed from further modeling. The best scoring loop models according to the BCL sum score were further processed.

Addition of Side Chains and Model Relaxation

One of two methods was used: either side chains were added with a relax step in which the amino acids were restrained to their starting position; or, if the first method fails because of misaligned β-strands, by adding and repacking side chains. Between 10 and 200 side chain models were built to obtain an optimal overall Rosetta score.

Model Selection for Submission

From the lowest scoring side chain models for each loop model, the ones deemed most native-like by visual inspection were selected for CASP10 submission. If a template model and a similar BCL model were found before, it was selected as the fifth submitted model.

Topology score to evaluate protein models

To evaluate if BCL::Fold can sample the folding space required for our target proteins, we introduce a new measure that focuses on SSE contacts instead of comparing atom positions like RMSD100 (Fischer et al., 2014 (Submitted)) or GDT (Zemla et al., 1999). This new measure computes the similarity of a model to a native protein by calculating the fraction of SSE contacts of the native that are present in a given model and the total number of SSE contacts of the native (true positive rate, sensitivity). An SSE contact is assumed if distance of the central axis of two SSEs is less than a certain threshold. An SSE can be represented by its central axis for the purpose of the distance calculation, because all SSEs in a BCL model are idealized. The threshold below which two SSEs are assumed in contact depends on the type of SSE contact (helix-helix: 16 Å; helix-sheet: 16 Å; strand-strand: 5.5 Å; sheet-sheet: 14 Å) and was derived from native protein structures from the Protein Data Bank. These thresholds were chosen to be large to be as inclusive as possible. The strength of the interaction is represented by line thickness of the connecting lines (Figure 3).

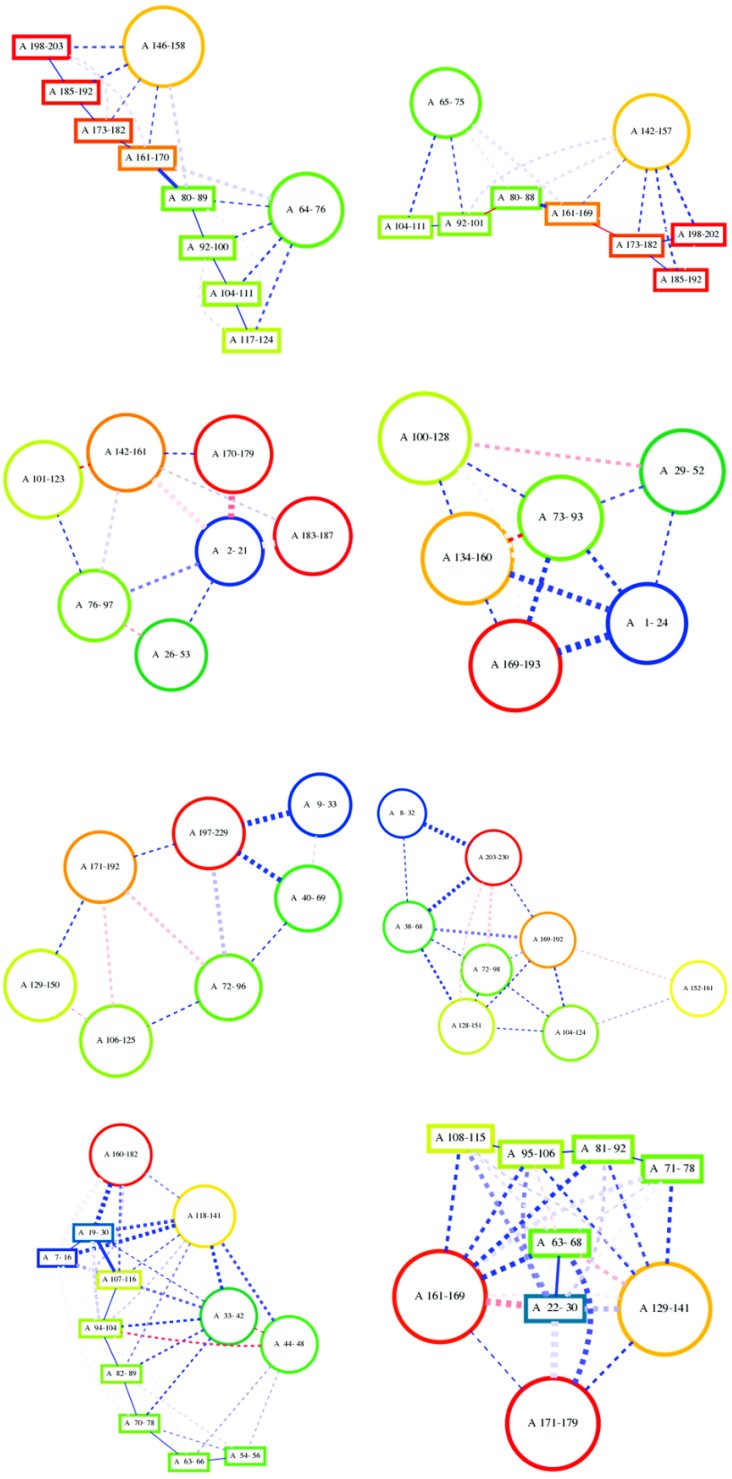

Figure 3. Visualization of the topologies for native and best scoring model according to topology score for selected target showing both successful (T0663, T0666, and T0682 in order from the top down with topology scores of 0.81, 0.82, and 1.00 for the respective best scoring model) and unsuccessful cases (T0655 at the bottom with a topology score of 0.44).

Results

Eighteen targets included in the present analysis

During CASP10 a total of 53 targets were released for human predictors. 18 of these had at least one domain in the FM category. To focus our efforts we excluded proteins that were very small (< 50 residues) or very large (> 400 residues). Further, for some targets calculations did not finish in time for submission. For 21 targets models were submitted, five of them in the FM category. For two targets files were corrupted on our server, for one target no experimental structure has been released. This leaves 18 targets, three in the FM category, for analysis (Table 8). Accordingly, treatment of the TBM targets as

Table 8. Statistics on 18 CASP10 targets predicted with BCL::Fold.

Target	PDB ID	Length	NCO	Category	Oligomeric State	Domains	α-Helices	TM α-helices	β-strands
T0644	4FR9	166	22.1	TBM-easy	Monomer	1	2	0	8
T0649	4F54	210	58.9	TBM-hard	Monomer	1	4	0	9
T0655	2LUZ	182	44.2	TBM-easy	Monomer	3	4	0	8
T0663	4EXR	205	28.4	FM	Monomer	2	2	0	8
T0666	3UX4	195	64.9	FM	Trimer	1	6	6	0
T0676	4E6F	204	45.2	TBM-hard	Dimer	1	4	0	7
T0678	4EPZ	161	30.5	TM-hard	Monomer	1	7	0	0
T0682	4JQ6	235	63.5	TMB-easy	Trimer	1	7	7	0
T0684	4GL6	270	36.9	FM	Dimer	2	8	0	8
T0686	4HQO	259	55.7	TMB-easy	Dimer	3	4	0	5
T0691	4GZV	163	25.7	TMB-easy	Monomer	3	0	0	8
T0700	4HFX	86	18.0	TMB-easy	Tetramer	2	3	0	0
T0704	4HG2	254	55.4	TMB-easy	Dimer	3	9	0	8
T0720	4IC1	202	47.5	TMB-easy	Monomer	1	7	0	6
T0722	4FLA	152	44.1	Cancelled	Tetramer	Cancelled	4	0	0
T0724	4FMR	265	42.6	TMB-easy	Tetramer	2	4, 5	0	16
T0743	4HYZ	149	36.9	TMB-easy	Monomer	1	4	0	5
T0745	4FMW	185	49.4	Cancelled	Dimer	Cancelled	6	0	6

FM targets substantially increased the number of proteins that could be included in the study beyond the small number of FM targets. One consequence of this procedure is that BCL::Fold will not rank among the top methods for the TBM section as we do not expect BCL::Fold to predict protein structure more accurately than comparative modeling.

An automated pipeline with minimal human intervention was setup

Here we give an overview of the overall protocol (Figure 4). A detailed description of the individual steps is given in the methods section. The folding pipeline starts with the downloaded target sequence from CASP10 Prediction center. In the first step, secondary structure and trans-membrane spanning regions are predicted and stored in a "pool" using the consensus SSE prediction results. The SSE pool is manually examined to ensure that weakly predicted SSEs are available. Domain boundaries were identified with bioinfo.pl - a consensus fold recognition Meta server (Ginalski, Elofsson, et al., 2003). At this stage of folding, templates were identified for TBM targets and comparative models were constructed using the Modeler (Baker & Sali, 2001; Rohl et al., 2004; Sali & Blundell, 1993) link of the bioinfo.pl server. The homology model was saved for later analysis or prioritization of the *de novo* folded models. It was not used to bias the folding simulation. If the fold recognition result from bioinfo.pl indicated that the target consisted of multiple domains, the SSE pool was split into sub-pools according to the domain boundaries. Next, each domain was folded 12,000 times with BCL::Fold. The resulting models were filtered for completeness before entering the clustering protocol. The completeness estimate is the total number of residues in secondary structure elements divided by the total number of residues in the protein model. The filtering cutoff is the average of all the completeness estimates reduced by 0.01. After filtering, cluster centers of the 10 to 20 largest clusters were selected for further processing. In addition we included the five best scoring models measured by the BCL total score. If templates were

identified, best-scoring models that were similar to the template by Mammoth Z-score (Ortiz, Strauss, & Olmea, 2002) were retained in a separate pool of models. If the target was split into multiple domains, these were recombined at this stage. The backbone of the resulting models was completed using a Cyclic Coordinate Decent (CCD) (Canutescu & Dunbrack, 2003) loop closure algorithm within the BCL. Subsequently side chain coordinates were constructed, and the model was relaxed using Rosetta. From the resulting set of up to 200 models five were chosen for submission by Rosetta energy. If a template

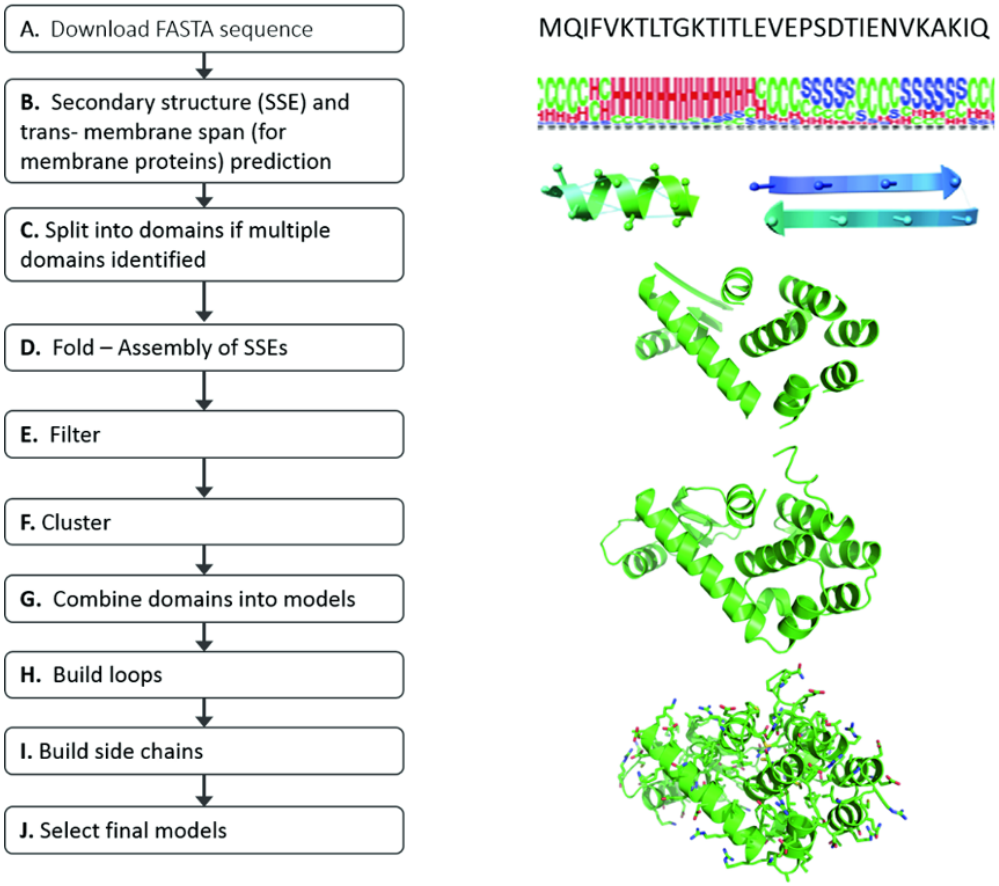

Figure 4. CASP10 Pipeline. Obtain target sequence from CASP10 prediction center (A); Perform SSE prediction (B); Split multimeric proteins into individual domains (C); Assemble SSEs in Folding algorithm, analyze fold models, compare generated models with native secondary structure, evaluate loop closure potential and beta sheet register shift (D); Filter erroneous models from further analysis (E); Cluster predicted folds and analyze cluster centers (F); Combine domains if previously split (G); Reconstruct loop regions and analyze models (H); Build side chains with Rosetta or other high resolution refinement software (I); Select final models and analyze final model selection (J).

has been identified, the fifth model submitted was chosen from the second pool as the one most similar to the template, in order to assess BCL::Fold's sampling capability independent from scoring.

Accuracy of Secondary Structure and Trans-Membrane Span Prediction

Table 9 depicts Q3 accuracies (a measure of the accuracy for predicting per residue secondary structure), the percentage of native secondary structures correctly predicted and the average shifts for the SSE pools of the 18 CASP10 protein targets. The shift values are the sum of the deviations in the first and last residues of the predicted SSEs when compared with native SSEs. The overall average percentage of native secondary structures correctly predicted (% found) using PHD (Rost, 1996; Rost & Sander, 1994), PSIPRED (Jones, 1999c; Ward et al., 2003), and JUFO9D (Leman et al., 2013; Meiler & Baker, 2003; Meiler et al., 2001) was 91.8 %. In the original benchmark of BCL, the overall average % found was 96.6 % (Karakas et al., 2012). We achieved the highest overall accuracy by combining multiple secondary

Table 9. Secondary structure pool statistics for CASP10 targets.

		PHD			PSIPRED			JUFO9D			Combined	
Target	**PDB ID**	**Q3**	**% found**	**shift**	**Q3**	**% found**	**shift**	**Q3**	**% found**	**shift**	**% found**	**shift**
T0644	4FR9	68.7	80.0	1.8	80.1	100.0	1.1	77.7	100.0	1.5	100.0	0.9
T0649	4F54	63.8	53.8	6.0	71.9	69.2	5.2	65.2	76.9	5.3	61.5	2.9
T0655	2LUZ	54.9	75.0	5.4	76.4	91.7	3.7	70.9	91.7	4.4	66.7	3.5
T0663	4EXR	54.1	80.0	2.9	80.5	100.0	1.5	69.3	100.0	1.8	90.0	0.9
T0666	3UX4	50.3	57.1	9.5	74.9	85.7	8.3	81.0	85.7	7.0	85.7	4.8
T0676	4E6F	66.7	80.0	8.9	77.9	90.0	1.9	57.4	90.0	7.3	90.0	1.3
T0678	4EPZ	72.0	85.7	11.7	83.2	100.0	2.7	78.9	100.0	3.4	100.0	1.3
T0682	4JQ6	62.6	100.0	14.1	71.1	100.0	10.6	79.1	100.0	11.6	100.0	4.0
T0684	4GL6	72.2	81.3	3.4	73.7	87.5	2.9	67.8	75.0	3.2	100.0	1.9
T0686	4HQO	64.1	72.2	5.0	74.5	66.7	2.8	67.6	88.9	4.3	94.4	3.4
T0691	4GZV	47.9	75.0	4.7	69.9	100.0	3.6	59.5	100.0	4.8	100.0	3.3
T0700	4HFX	74.4	100.0	5.0	75.6	100.0	4.3	72.1	100.0	3.3	100.0	2.7
T0704	4HG2	63.0	58.8	3.1	74.8	88.2	3.3	72.0	88.2	2.8	94.1	2.1
T0720	4IC1	70.3	84.6	5.5	84.2	92.3	3.6	79.2	92.3	3.8	92.3	3.3
T0722	4FLA	87.5	100.0	30.0	89.5	100.0	9.0	80.3	100.0	16.5	100.0	7.0
T0724	4FMR	71.3	84.2	2.9	86.0	89.5	1.8	78.5	94.7	1.7	89.5	1.2
T0743	4HYZ	72.5	77.8	3.7	77.2	77.8	2.7	67.8	77.8	6.6	88.9	1.8
T0745	4FMW	65.9	75.0	2.8	77.3	100.0	1.9	67.6	83.3	2.9	100.0	1.8
Average		65.7	78.9	7.0	77.7	91.0	3.9	71.8	91.4	5.1	91.8	2.7
Std Dev		9.6	13.4	6.6	5.3	10.7	2.7	7.2	8.8	3.8	11.3	1.6

structure prediction methods to create the SSE pool, rather than relying on a single secondary structure prediction method. For example, the % found values for PHD, PSIPRED, and JUFO9D are 78.9 %, 91.0 %, and 91.4 % respectively. In the original BCL benchmark these values for PSIPRED and JUFO are 96.1 % and 90.3 % respectively. This indicates that the secondary structure prediction is more challenging for the CASP10 targets than the original BCL benchmark. In addition, during a folding run, BCL::Fold can merge, grow, or shrink SSEs based on the predicted probabilities.

Quality of CASP10 FM Models submitted by other Research Groups

There were 20 free modeling targets in CASP10. For all participating methods the average GDT_TS score ranged from 7.0 % – 36.0 % with a mean GDT_TS score of 21.7 % and a standard deviation of 7.2 %. The maximum GDT_TS score ranged from 16.5 – 44.0 % with a mean GDT_TS score for 32.8 % and a standard deviation of 8.1 %. For the three targets attempted with BCL::Fold (T0663, T0666, T0684) the

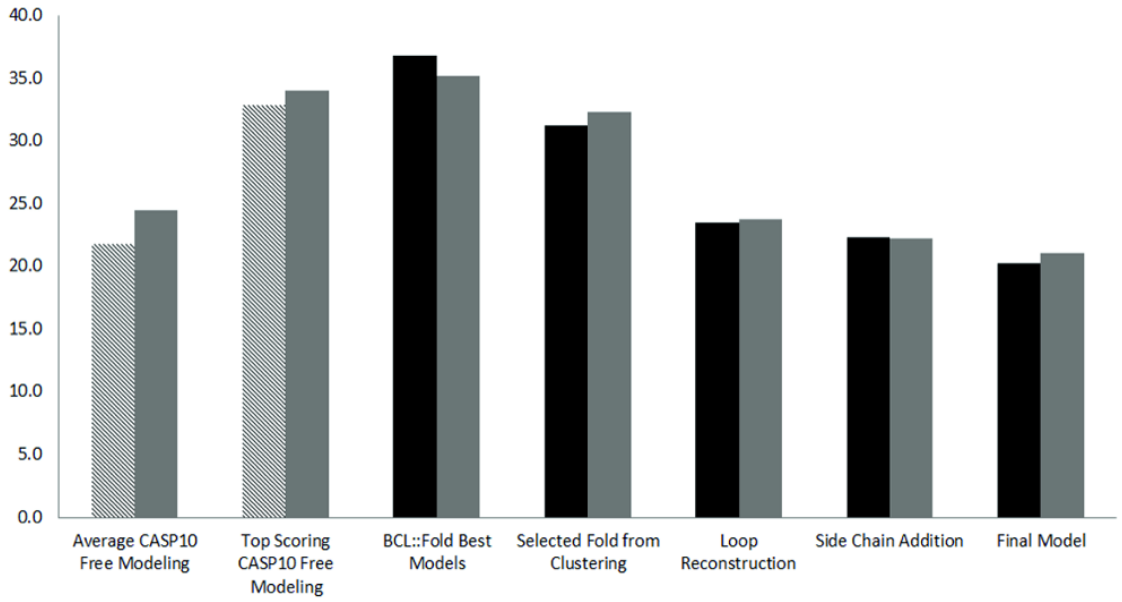

Figure 5. GDT_TS score analysis. 20 free modeling targets from CASP10 (left two bars, pattern). Three targets folded also by BCL::Fold from free modeling category in CASP10 (left two bars, gray). All 18 targets folded by BCL::Fold (black). Three FM targets folded by BCL::Fold (right five bars, gray). The y-axis represents GDT_TS score.

average GDT_TS score submitted by CASP10 participants was 24.5 % with a standard deviation of 10.2 %. The mean of the maximum GDT_TS scores for these targets was 34 % with a standard deviation of 9.5 % (Figure 5).

Quality of BCL::Fold models and sampling of the Topology Space

We assess the quality of BCL::Fold models in two ways. The GDT_TS score allows for comparison with other results; the topology score focuses its evaluation criteria specifically on SSE contacts which tests BCL::Fold's method of assembly.

GDT_TS scores for the best model generated by BCL::Fold ranged in from 23.8 % – 53.5 % with a mean GDT_TS score of 34.7 % and a standard deviation of 8.7 %. Using the mean GDT_TS score of 33 % as a comparative measure between other methods, BCL::Fold was able to sample models above this threshold in 12 out of 18 cases. Comparisons of the BCL models with the experimentally determined structure by measuring RMSD100 (Carugo & Pongor, 2001) and GDT_TS show efficient sampling of the correct topology (**Table 10**, Figure 6).

BCL::Fold's sampling performance was evaluated previously with soluble and membrane proteins and compared to Rosetta. BCL::Fold was able to sample the correct topology in 61 of 66 soluble benchmark proteins (Karakas et al., 2012) and in 32 of 38 membrane benchmark proteins (B. E. Weiner et al., 2013). The correct topology was defined as the ability to fold models with an RMSD100 of less than 8 Å to the native.

Table 10. Comparison of the GDT_TS score and RMSD100 score with the native showing the best model produced during folding with BCL::Fold (A); The selected models from Clustering (B); The models after loop reconstruction (C); The models after side chain addition (D); The final submitted model (E).

Target	PDB ID	GDT_TS					RMSD 100				
		A	B	C	D	E	A	B	C	D	E
T0644	4FR9	41.7	32.1	19.1	21.1	21.1	7.7	12.4	11.5	10.5	10.5
T0649	4F54	38.5	29.5	19.5	16.7	12.6	9.6	13.5	14.8	14.9	14.9
T0655	2LUZ	37.4	25.6	18.1	18.0	17.0	9.6	13.4	10.4	11.2	11.2
T0663	4EXR	43.0	39.7	26.0	24.7	24.5	5.8	7.1	10.2	13.1	13.3
T0666	3UX4	38.8	35.0	29.9	28.6	25.6	5.1	7.2	6.9	7.1	8.3
T0676	4E6F	31.9	26.2	24.0	21.3	20.0	9.7	11.7	11.6	13.1	13.1
T0678	4EPZ	40.0	29.1	30.7	29.2	20.9	8.0	10.3	7.9	11.5	11.8
T0682	4JQ6	37.4	28.8	37.1	36.3	33.0	4.8	8.3	4.5	4.6	5.4
T0684	4GL6	23.8	22.2	15.3	13.5	13.1	12.0	12.0	12.8	13.7	13.7
T0686	4JQ6	29.1	29.1	13.8	12.0	12.0	10.4	10.4	12.2	16.9	16.9
T0691	4GZV	34.8	26.8	19.2	17.4	13.7	10.9	12.7	10.9	12.3	15.2
T0700	4HFX	64.5	57.6	38.6	38.6	31.3	7.2	10.4	14.1	13.3	15.4
T0704	4HG2	25.0	17.9	12.6	11.3	10.5	10.7	11.9	10.3	13.5	13.5
T0720	4IC1	26.1	24.2	19.8	19.8	15.6	10.6	10.8	10.3	10.8	13.8
T0722	4FLA	53.5	46.0	38.7	40.7	38.9	5.1	6.9	20.7	20.1	21.7
T0724	4FMR	23.3	21.9	13.6	12.4	12.4	11.8	14.1	14.4	17.3	17.3
T0743	4HYZ	38.4	37.2	25.7	25.2	23.5	8.3	10.6	9.4	9.8	10.6
T0745	4FMW	35.1	33.1	21.8	18.7	18.5	8.5	10.2	10.4	11.5	14.0

FM Target	Average GDT_TS	Best GDT_TS
T0663	36	43.5
T0666	21	34
T0684	16.5	24.5

While RMSD100 is suitable to assess Rosetta models, it is not as helpful for BCL::Fold models that are focusing on sampling long-range contacts between SSEs. Figure 3 shows how well BCL::Fold samples the different protein topologies, measured the topology score. Its applicability is limited foremost by the number of SSE contacts. For targets with very few contacts (T0722 has a single contact) many models achieve a high score, and the discriminative value of the topology score is reduced. While the topology score does currently not consider specific types of interactions between SSEs, it does include the secondary structure type; thus, an incorrectly predicted secondary structure type leads to all contacts of this incorrect SSE to be evaluated as false. The thresholds to assume a contact between two SSEs are

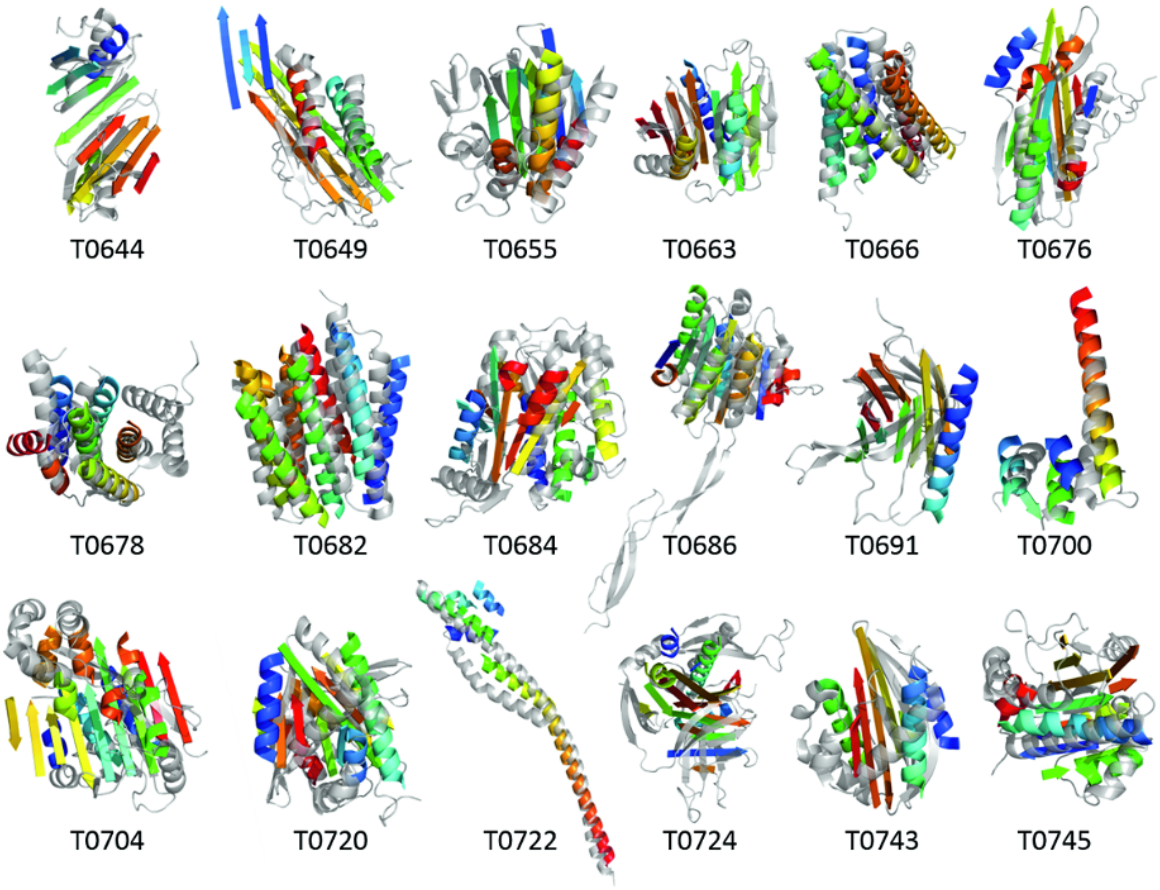

Figure 6. Highest GDT_TS models sampled with BCL::Fold (rainbow) overlaid with experimental protein structure (gray).

derived from idealized, native protein models and therefore fairly large; this can lead to detection of SSE contacts even for SSEs that are only indirectly in contact but still a very short Euclidean distance apart, like the first and third strand of a sheet. Additionally, the value of the topology visualization is narrowed by the projection of three-dimensional protein structures into two dimensions which reaches its limits for complex topologies. While the topology score has some caveats, overall it captures the protein topology quite well.

For the topology score, which measures the true positive contact ratio, we set the threshold to 0.8. At this level, two topologies share an overwhelming number of SSE contacts. Furthermore, we observe similarities when visually inspecting the topology plots of protein models (Figure 3).

BCL::Fold samples models above the threshold of 0.8 for 11 out of 18 targets (Figure 7). All targets with a native SSE contact count up to 20 have a topology score above the threshold. With increasing native SSE contact count and complexity, the topology score decreases expectedly.

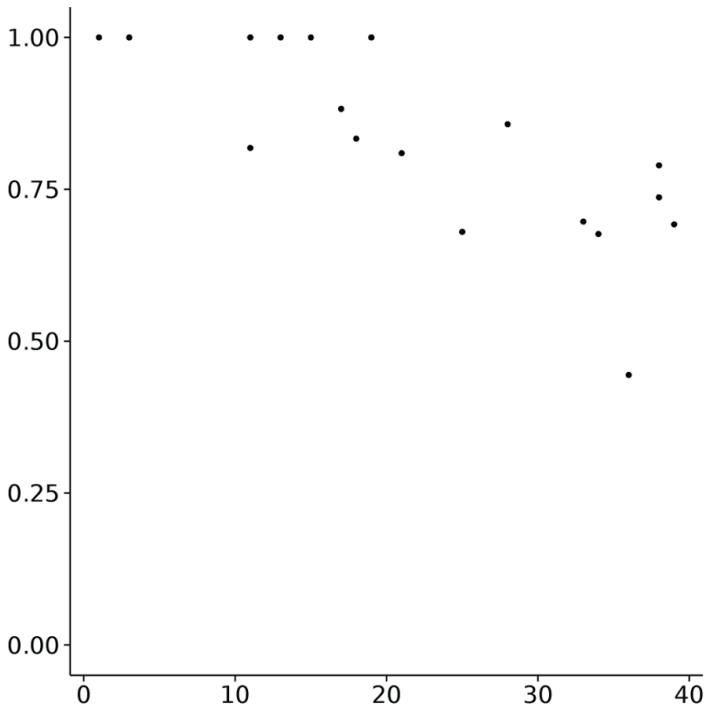

Figure 7. Comparison of true positive rate vs. protein complexity. True positive rate (precision) is shown on the y-axis, the protein complexity (number of SSE contacts in the native) on the x-axis. The true positive rate of BCL::Fold models decreases with increasing complexity.

Selection of Models for Loop and Side Chain Construction

However, the selection of models for the subsequent refinement steps proved to be difficult. During CASP10 we attempted selecting the best models by BCL sum score, the centers of the largest clusters, and the best scoring model in each cluster. However, no method enriched for high GDT_TS and consequently the models most similar to the native were consistently lost. For model T0700 we sampled a topology with an overall GDT_TS score of 64.5. We selected a model with a GDT_TS score of 57.6 for

further refinement. After loop and side chain reconstruction, our model drifted further from the true native structure with a GDT_TS score of 38.6. Our final submitted model for this target had a GDT_TS score of 31.3. Most of the targets folded with BCL::Fold had this attrition pattern. Interestingly, model T0682 improved substantially after loop reconstruction from a GDT_TS score of 28.8 to 37.1. Our final submitted to CASP10 for this target had an RMSD100 score of 5.4 and GDT_TS Score of 33.0 (Figure 5, Figure 8).

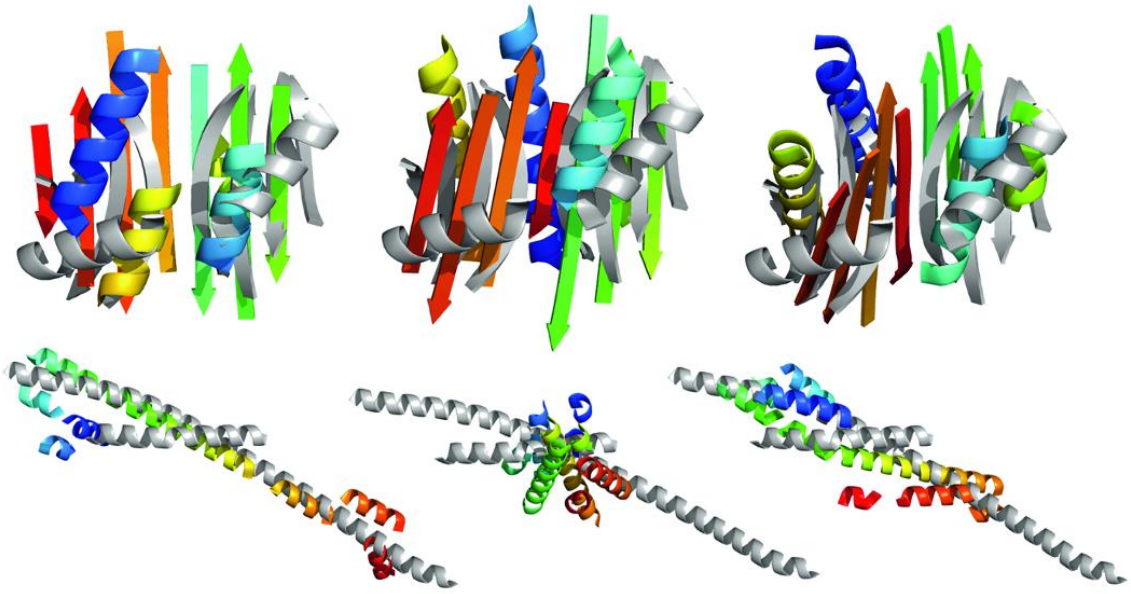

Figure 8. Comparison of example BCL models with the native target structure for T0663 (top) and T0722 (bottom). The experimental structures without loops are shown in gray (based on PDBIDs 4EXR and 4FLA, respectively). The predicted models (rainbow) show the highest scoring model produced by BCL (A, D, with a GDT_TS of 43.0 and 53.5, respectively); The best scoring model by BCL energy function (B, E, with a GDT_TS of 28.9 and 26.9); The best scoring model in largest cluster (C, F, with a GDT_TS of 22.1 and 32.6).

Addition of loop and side chain coordinates

While adding loops to the cluster centers decreased the average GDT_TS scores from 31.2 to 23.5, the GDT_TS average dropped again from 23.5 to 22.4 when the side chains were added with Rosetta version 3.3. To rebuild side chains, the models were relaxed. To limit movement of the backbone constraints for

every C_α-C_α bond distance below a cutoff of 8 Å were applied using a harmonic function with a standard deviation of 0.5. During side-chain reconstruction with Rosetta, 12 of the 18 CASP10 targets had a radius of gyration score larger than 1,100 for approximately 30 % of all models indicating unfolding despite the constraint used (T0644, T0649, T0655, T0663, T0666, T0684, T0691, T0704, T0720, T0722, T0743, and T0745). This unfolding-like event was triggered because the BCL models scored poorly in the Rosetta energy function (Figure 9). Models that were unfolded were not considered further. As a method of last resort, Rosetta was used to add side chains without relaxing the backbone but only repacking the side chains.

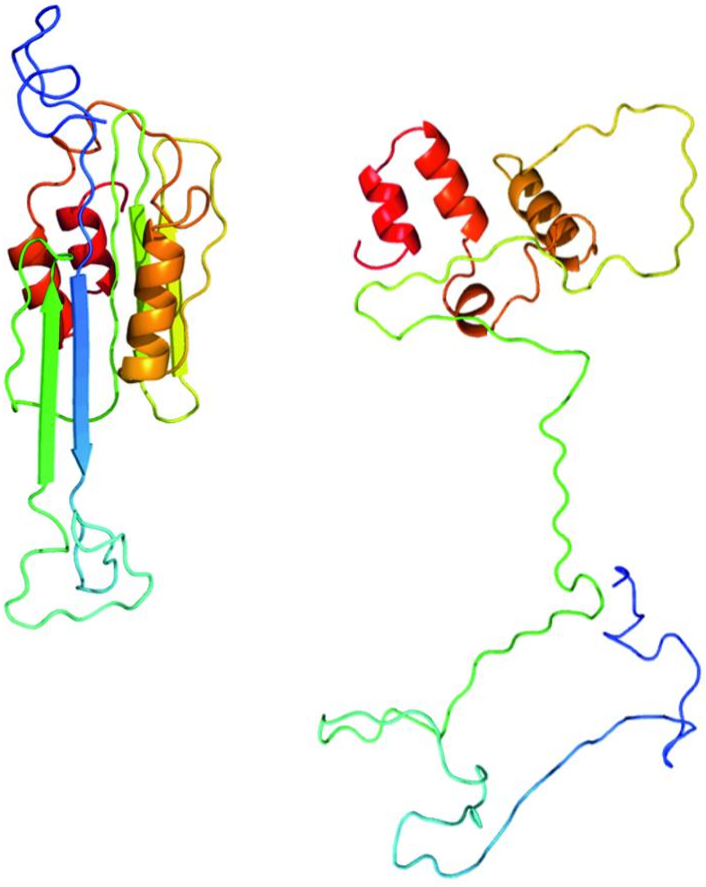

Figure 9. Unfolding of a BCL model for target T0655 by side chain addition and relaxation with Rosetta (right) compared to the input model (left).

Discussion

BCL::Fold fails to sample to correct topology in 7 cases

In 7 out of 18 cases the best scoring BCL::Fold model had a topology score of less than 0.8, which means the correct topology was not found. Investigating the reasons for these failures, we found that the target with the lowest topology scores had SSEs missing in the secondary structure prediction and subsequently in the SSE pool. T0655 had a topology score of 0.44 and had two helices missing; T0649 had a score of 0.68 and had one helix missing.

Models for T0724 have an incorrect strand topology because BCL::Fold models were created as protomers while the native exists as dimer in which strands from both monomers form a sheet.

The remainder of four incorrect targets failed to sample the correct topology because of a combination of reasons, most notably for two reasons. Long SSEs were split into two smaller ones, either by DSSP when assigning secondary structure to the natives, or by the secondary structure prediction methods that we employed. The correct topology was simply not sampled and recognized as a best scoring model, often with the order of strand SSEs in sheets being incorrect.

BCL::Fold models have loops that are impossible to close

BCL::Fold assembles tertiary structure from disconnected SSEs. Because of this, we must ensure that the distance between the end of one SSE and the beginning of the next SSE can be bridged by a loop. Two components of the BCL::Fold scoring function control this requirement: First, there is a penalty if the Euclidean distance between two SSEs is longer than the maximal Euclidean distance that can be bridged by the number of amino acids in the loop. Models that violate this rule are heavily penalized during

Monte Carlo sampling and likely rejected. The second component is designed to place SSEs so that loops between them match a loop score potential that reflects native loop conformations from the PDB (PISCES dataset, see Methods). This loop score potential evaluates the Euclidean distance probability in dependence of number of residues (Woetzel et al., 2012). As this score is a function of only Euclidean distance and sequence distance, it neglects the spatial arrangement of SSEs. Analysis of CASP10 models revealed that BCL::Fold constructs models where loops cannot be closed without passing through SSEs. Figure 10 depicts a model produced by BCL::Fold for target T0663. The Euclidean distance between residues ASN55 of helix 1 and TYR65 of helix 2 is 25.5 Å. To bridge this distance with 9 amino acids, each amino acid has to be 2.8 Å on average, which is less than the average C_α-C_α distance of 3.3 Å. However, with the placement of strand SSEs between the loop ends, all paths to close the loop between helices 1 and 2 pass through the strand SSEs. Overall, 76 % of BCL::Fold models produced during CASP10 folding simulations contain non-closable loops because of this behavior.

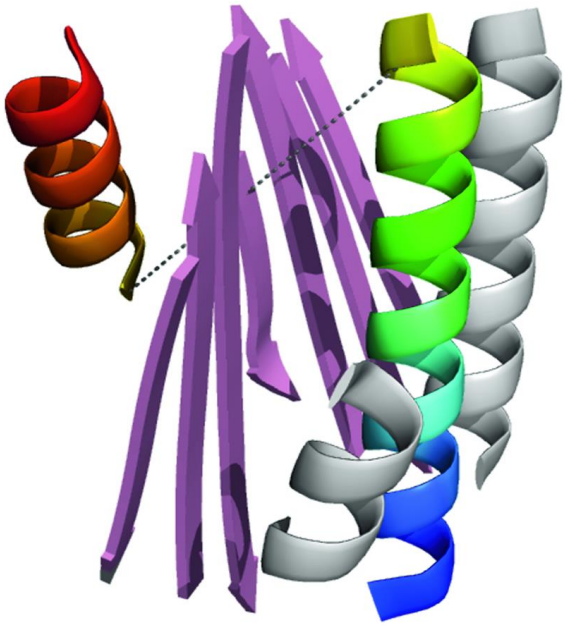

Figure 10. A model for CASP10 target T0663 folded by BCL with a loop that cannot be closed. The Euclidean distance between residues ASN55 in helix 1 (rainbow colored on the right) and TYR65 in helix 2 (rainbow colored on the left) is 25.5 Å. Without the central sheet (pink) the loop could be closed; it is impossible to close the loop if the connecting amino acids have to be position around the sheet.

The BCL::Fold Loop Potential is often violated for consecutive SSEs

Loops found in native proteins bridge preferable Euclidean distances d_e depending on the loop's sequence length d_s. The current loop potential of BCL::Fold mirrors this preference. It is a sequence independent score that contributes to the overall energy function. The PISCES data set used to create this potential includes all possible loops, i.e. loops between consecutive and non-consecutive SSEs. Because BCL::Fold does not assemble SSEs in sequence order, the potential must evaluate incomplete protein models with unplaced SSEs. Therefore non-consecutive SSEs were included in the loop scoring potential.

To test the loop potential accuracy, we compare the CASP10 models produced by BCL::Fold to structures from the PISCES pdb set. Because the Euclidean distance that a loop spans depends on the sequence length of the loop, we normalize the Euclidean distance by the logarithm of the sequence length, $d_e/\log d_s$; this results in homogeneous distributions independent of loop length. The all-loop distributions (i.e. consecutive and non-consecutive loops) for $d_e/\log d_s$ for CASP10 models, CASP10 natives, and PISCES are alike (Figure 11A). The means of the distributions are 6.2 Å, 6.6 Å, and 6.5 Å respectively and confirm their similarity. Thus we conclude that this weighted potential distinguishes native-like sequence and distance length of loops from non-native configurations in terms of sequence length and corresponding Euclidean distance.

However, when evaluating the CASP10 models with the consecutive-only loop distribution (i.e. only loops between consecutive SSEs are included), we find a substantial bias between CASP10 models and both CASP10 natives and PISCES structures (Figure 11B). Their means are 8.1 Å, 5.8 Å, and 5.7 Å, respectively. The sequence length d_s of a loop is not changing as it is defined by the secondary structure (prediction) of the particular protein and only used for normalization. Therefore, the difference between the distributions can only be caused by differences in the Euclidean distances d_e. Creating models with

loops of longer Euclidean distances d_e than found in native structures for a given sequence length causes BCL::Fold to produce non-native like loop arrangements. Thus, the loop potential is not a sufficient metric to generate native-like models from disconnected SSEs. Furthermore, the current loop potential does not consider the spatial positioning of other SSEs and does not account for potential clashes between these SSEs and a loop (Figure 10).

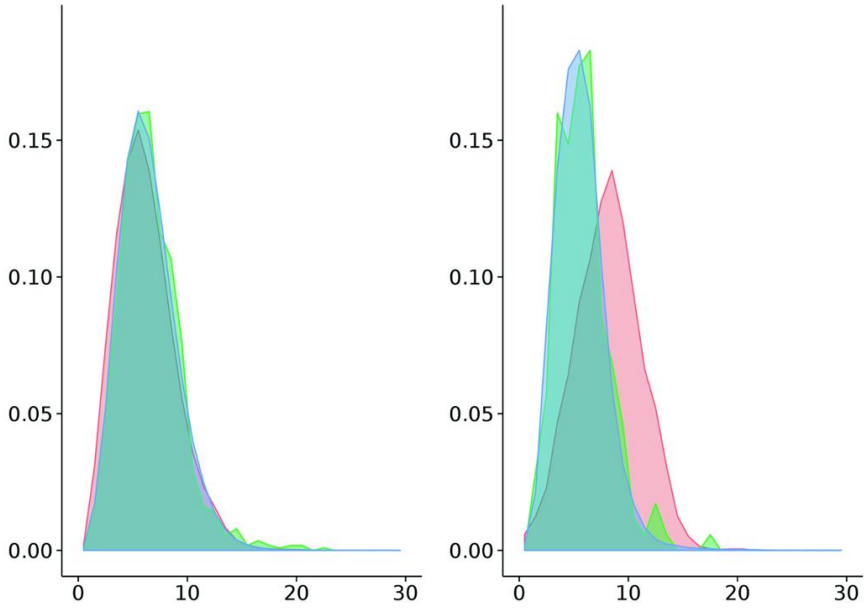

Figure 11. The density distribution of the BCL loop score displaying Euclidean distance over the logarithm of the sequence separation for loop regions between all SSEs (A) and consecutive SSEs only (B). While the distributions of BCL models (red), CASP10 natives (green) and PISCES dataset (blue) match each other for loops between all SSEs (A), the distribution of BCL models shows a shift when only loops between consecutive SSEs are considered (B).

A small loop angle favors more native-like loops

To address the shortcoming we devised a loop measure that reflects this difference between consecutive and non-consecutive SSEs more drastically. For native proteins, we observe that loops between consecutive SSEs are positioned locally on a protein structure, i.e. consecutive loops tend to begin and end on the same side of the structure and do not connect through the center. Geometrically this can be measured at as the angle between the end of one helix, the center of the protein, and the

start of the next helix (Figure 12A). In native protein structures, consecutive loops overwhelmingly favor small angles, as shown for the CASP10 native and PISCES pdb sets, of which 75 % are smaller than 40 degrees (Figure 12B, green and blue, respectively). Models with loops that would clash with other parts of the protein frequently have large angles of close to 180 degrees (Figure 12B, red). We can use this information to discriminate native like arrangements from models with large angles.

When including non-consecutive loops, the distribution of loop angles exhibits two frequently occurring angles, small ones for loops connecting consecutive SSEs, and large ones for connecting non-consecutive SSEs (Figure 12C). To evaluate the loop angles of a protein model, we must differentiate between loops that connect consecutive and non-consecutive SSEs.

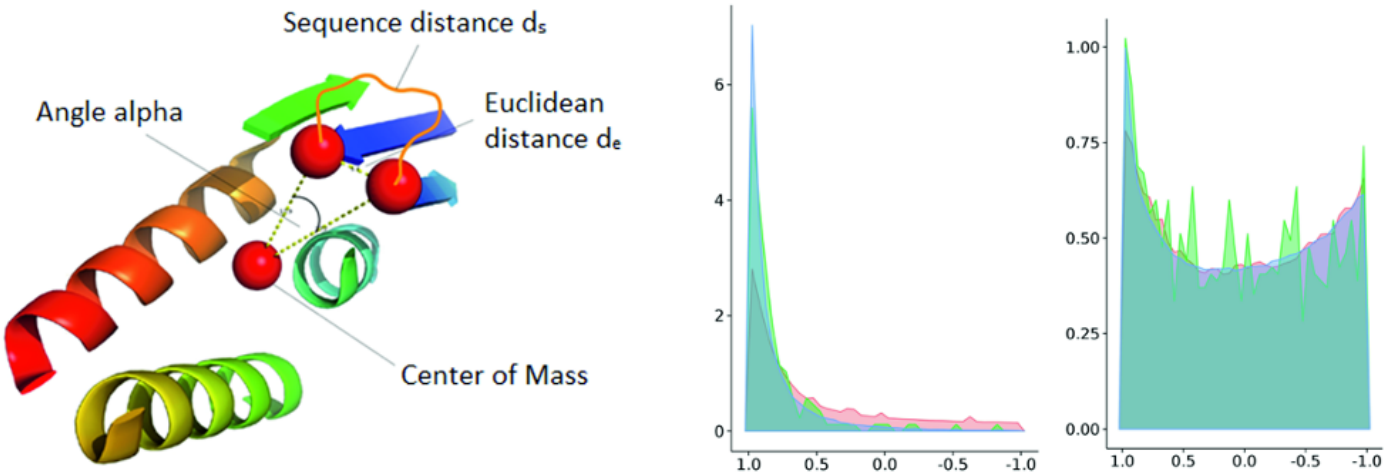

Figure 12. Visualization of loop angle metric, which measures the angle α between the end of one SSE (dark blue), the center of gravity, and the beginning of the next SSE (light blue) (A). The density distribution of the cos(α) metric for loop regions between consecutive SSEs only is concentrated to acute angles for PISCES and CASP10 natives (B, blue and red, respectively). BCL models exhibit a higher number of large angles for consecutive loops (B, red). The density distribution of the cos(α) metric for loop regions between all possible SSEs shows two frequently found angles, small ones and large ones, for all sets, BCL models (red), CASP10 natives (green) and PISCES (blue) (C).

To test whether filtering by the new loop angle measure would select for lower RMSD models compared to the existing loop score, we folded models for eight CASP10 targets (1000 models for T0655, T0663, T0676, T0678, T0684, T0700, T0745; 700 models for T0722). The RMSD cutoff was set to 10th percentile.

Both, the existing loop score and the loop angle score were then used to select the best 50 % according to each score. The existing loop score filtered on average 50 % of the models below the RMSD cutoff and in three cases decreased the number of models below the RMSD cutoff by more than the expected 50 % (T0684, T0700, and T0722). The loop angle score filtered on average 61 % of the models below the RMSD cutoff and only in one case, T0722, it selected less than 50 % of the models below the RMSD cutoff. Thus, the loop angle score is selecting more native-like models and can improve the BCL scoring function moving forward (Table 11).

Table 11. The percentage of models below the RMSD cutoff kept when filtering models for each target with the existing loop score and the loop angle score, showing that the loop angle score keeps in all cases more low RMSD models.

Target	% models kept by existing loop score	% models kept by loop angle score
T0655	70	70
T0663	67	76
T0676	52	57
T0678	52	63
T0684	44	57
T0700	37	57
T0722	16	43
T0745	59	62
Average	50	61

BCL::Fold misaligns β-Strand Registers

Carbonyl and amide groups in parallel and antiparallel strands of native proteins are aligned to allow the formation of stabilizing hydrogen bonds. A hydrogen bond is formed between the carbonyl-oxygen (hydrogen-bond acceptor) of one amino acid with the amide hydrogen (donor). In a sheet with the antiparallel strands i and j, the following pairs of atoms form hydrogen-bonds, here denotes as (acceptor, donor): $(\mathrm{C}_i, \mathrm{C}_j), (\mathrm{C}_j, \mathrm{C}_i), (\mathrm{C}_{i+2}, \mathrm{C}_{j-2}), (\mathrm{C}_{j-2}, \mathrm{C}_{i+2}), (\mathrm{C}_{i+4}, \mathrm{C}_{j-4}), (\mathrm{C}_{j-4}, \mathrm{C}_{i+4})$, … (Figure 13A); the pattern for parallel strands i and j is: $(\mathrm{C}_i, \mathrm{C}_{j+1}), (\mathrm{C}_{j+1}, \mathrm{C}_{i+2}), (\mathrm{C}_{i+2}, \mathrm{C}_{j+3})$, … (Figure 13C).

BCL::Fold does not control for this alignment in order to simplify the folding energy landscape. It only controls for distance and relative orientation of β-strands within β-sheets. We hypothesized that misalignment of hydrogen bonds within β-sheets might cause clashes that are responsible for the large fraction of models that unfolds during Rosetta refinement.

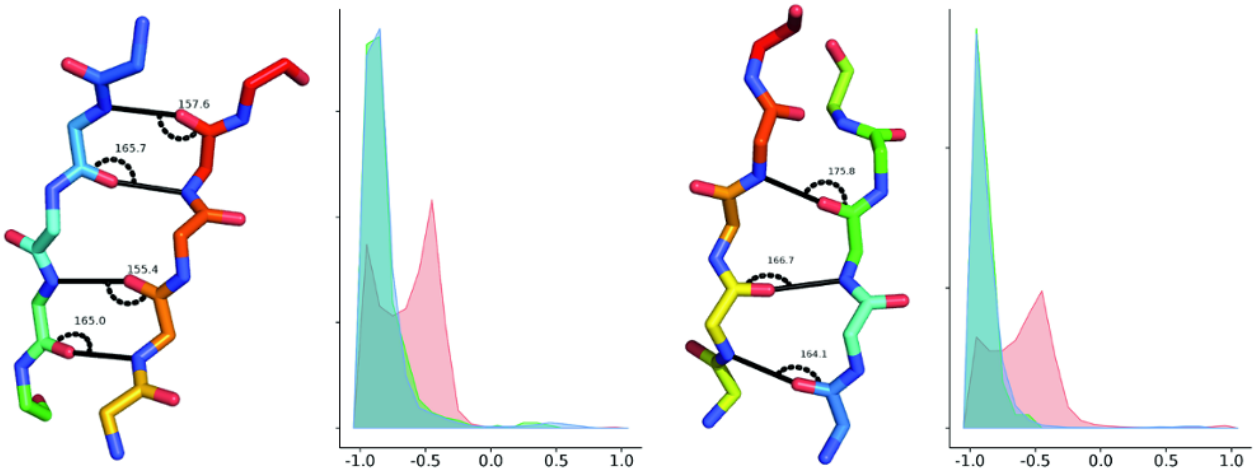

Figure 13. Hydrogen bonds and hydrogen bond angle distribution. Hydrogen-bond pattern and angles between the carbonyl-carbon, carbonyl-oxygen and amide-hydrogen in antiparallel (A) and parallel strands (C). Comparison of the hydrogen-bond angle for BCL models (red), CASP10 natives (green), and PISCES (blue) for antiparallel (B) and parallel strands (D). While the angles for CASP10 native and PISCES sets match, BCL models deviate. The x-axis shows the cosine of the hydrogen-bond angle, the y-axis the normalized density.

To evaluate the strand register alignment of BCL models and compare them to natives, we measured the angle between carbonyl-carbon, the carbonyl-oxygen and the amide-hydrogen, and the distance from the carbonyl-oxygen to the amide-hydrogen. While in native proteins a hydrogen bond rarely has a Euclidean distance longer than 2.1 Å, we measured putative hydrogen bond atom pairs that were in paired β-strand SSEs and within a relaxed cutoff of 4.5 Å. The hydrogen-bonds in aligned strands of elucidated proteins have characteristic angles close to 180° and distances of 1.9 Å to 2 Å. Analysis of CASP10 BCL::Fold models, CASP10 experimental structures, and the PISCES is summarized in Figure 13. In BCL models we find substantial deviations to smaller angles and larger distances up to 4 Å for more than half of the models for both antiparallel and parallel sheets. The deviation in hydrogen bond angle

and distance is correlated in BCL models. Additionally, BCL models exhibit a slightly shorter hydrogen bond distance of 1.8 Å to 1.9 Å even for hydrogen bonds with a native-like angle (Figure 14). This points to an incorrect placement of SSEs.

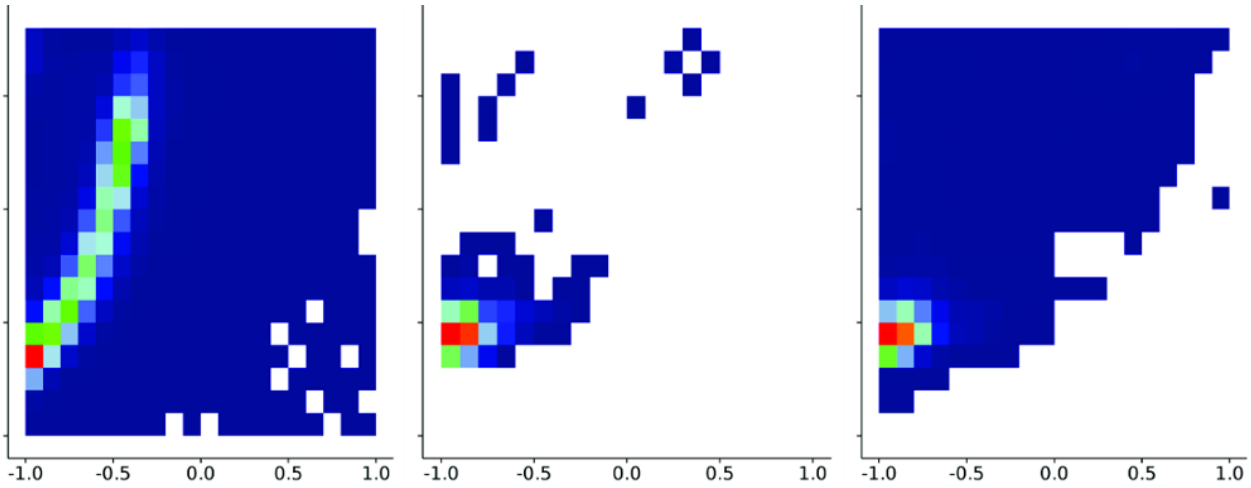

Figure 14. Heat map for hydrogen-bond angles in sheets showing the cosine of the angle (x-axis) vs. the distance between carbonyl-oxygen and amide-hydrogen for BCL models (A), CASP natives (B), and PISCES (C).

Misaligned β-Strands are cause clashes in Rosetta

The misaligned β-strands result in a high positive contribution from the repulsive score term (fa_rep) and no attractive contribution from the hydrogen bond score term (hbond_lr_bb), which leads to an unfavorable Rosetta score overall. The fa_rep term is the repulsive component of the van der Waals force, for example originating from carbonyl-oxygen of two strands being positioned too close to each other. The hbond_lr_bb term evaluates backbone-backbone hydrogen bonds distant in the primary sequence as they appear in sheets. Due to the misalignment of strands, the hbond_lr_bb term is zero and does not contribute to the overall Rosetta score (Figure 15).

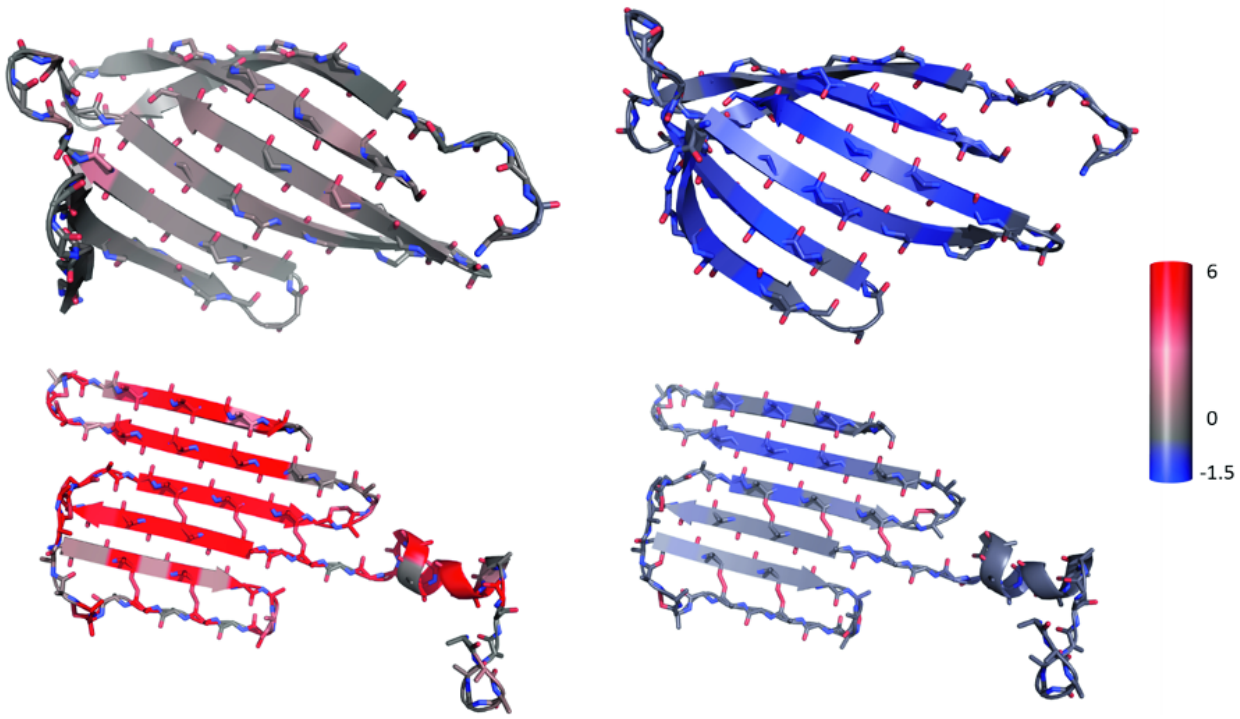

Figure 15. The analysis of Rosetta energy scoring terms for the native and a BCL model of target T0655 (shown is only the sheet part of native and model). The native shows no penalty from the repulsive score (A, fa_rep Rosetta score term) and a beneficial contribution from the hydrogen bonding score term (B, hbond_lr_bb Rosetta score term). Contrary, the BCL model exhibits a very high repulsive score (C, fa_rep) and little benefit from the hydrogen bonding term (D, hbond_lr_bb). The color scale stretches from blue representing -1.5 Rosetta energy units (REU) through grey (0 REU) to red (6 REU); the scale was chosen to red depicting a value further from zero than blue to account for the bigger range of the repulsive score.

This causes Rosetta to unfold BCL models, despite constraints (Figure 9), in the last step of our CASP10 pipeline, which adds side chains and structurally refines the protein by cycling through repack and minimization steps.

β-Strand placement in BCL::Fold models needs to be refined to align hydrogen bond donors and acceptors

The assembly of disconnected SSEs allows BCL::Fold to sample different sheet topologies and register positions without being restricted by the residues connecting the two strand SSEs. For this reason β-strand placement is controlled only by a mutate-function that places one strand next to another in the

preferred angle and distance(Woetzel et al., 2012). However, the placement of β-strands only by the distance and torsion angle within the β-sheet is insufficient to produce BCL::Fold models that can be refined with other programs. We plan to add a refinement stage into BCL::Fold that translates β-strands along their z-axis and evaluates a scoring term that controls the angle α introduced above. This will result in an improved scoring function that selects for more native-like models. We expect that improved alignment of β-strands will reduce the unfolding events observed during Rosetta refinement.

Conclusion

Despite inaccuracies in secondary structure prediction, BCL::Fold was able to sample the correct fold for all 18 cases studied herein. The best methods in CASP10 submitted models with an average GDT_TS of around 33 % in the FM category. BCL::Fold achieves this threshold in initial models after folding for 12 of 18 targets. Similarly, BCL::Fold is able to produce models with a topology score of at least 0.8 for 11 of 18 targets. However, the post folding filtering and refinement strategies removed correctly folded models from consideration in almost all cases, mostly for structural artefacts present in the BCL::Fold models. This result shows that BCL::Fold has the potential to compete with the best *de novo* structure prediction algorithms if a) unrealistic geometries in loops and β-strands can be removed and thereby the attrition of accurate topologies during model refinement can be stopped and b) an approach can be found that recognizes the most accurate models within the BCL::Fold ensemble. However, with this analysis and planned work to address the recognized weaknesses, future versions of BCL::Fold should produce more native-like models without incorporating templates or experimental data.

CHAPTER IV.

BCL::FOLD PROTEIN TOPOLOGY PREDICTION GUIDED BY SAXS PROFILES

This work is based on a manuscript in preparation for publication by Putnam et. al. S.H. contributed to the implementation of the SAXS score and the loop approximation; he designed the benchmark for BCL::Fold guided by SAXS experimental data.

Summary

SAXS experiments can provide low resolution information that does not allow the structural determination of proteins at atomic resolution, but can be useful in conjunction with other techniques. Here we combine SAXS data with the BCL::Fold de-novo structure prediction algorithm. The information about a protein's shape contained in SAXS data is sufficient to identify the correct protein from a test set of 455 non-redundant proteins. In a ROC curve analysis, we found an area under the curve (AUC) of 0.9995 when using an all atom representation for the proteins. Even for reduced representations as used during BCL::Fold's folding process, with loop regions and side chains approximated, the AUC was 0.9264, allowing for good discrimination of correct from incorrect models. We created a protein folding benchmark to show in which cases low resolution SAXS data contains protein-specific information that adds to the information in the BCL::Fold scoring function. As expected, where the SAXS data adds information to the scoring function, filtering models built with the default scoring function by SAXS data, enriches for correct models; if in such cases the SAXS data is used during the folding by extending the scoring function with a term based on the SAXS profile, the resulting models are more native like and have a lower RMSD100 to the native. For proteins for which SAXS data does not contribute any

information to the scoring function, we observe no improvement neither in enrichment after filtering, nor in RMSD100 when folding with the SAXS information.

Introduction

X-ray crystallography and NMR spectroscopy are challenged by large proteins that do not crystallize

Protein structure determination is challenging. X-ray crystallography, while accounting for 89 % of the determined structures in the Protein Data Bank (PDB), requires the protein of choice to be crystallized, which places it in a non-physiological environment. This causes problems for flexible proteins as well as proteins that are not sufficiently stable outside of their native environment (Bill et al., 2011). The applicability of x-ray crystallography is also limited by possible protein-protein contacts caused by the arrangement in the crystal lattice; by potential non-native conformations of proteins that were co-crystallized for better stability, e.g. by using antibodies; and most importantly by the reduced diffraction intensity for large proteins and large unit cells.

Nuclear magnetic resonance (NMR) spectroscopy is the second most frequently used method for experimental structure determination and accounts for 10 % of the structures in the PDB. While it has advantages compared to X-ray crystallography in allowing it to work with soluble proteins in a more physiological environment and also being able to capture flexibility, NMR spectroscopy is difficult to use with larger proteins, because of overlapping data points in the recorded spectra and resulting problems in assigning these data points to specific atoms (Skrisovska, Schubert, & Allain, 2010).

SAXS determines limited experimental information

While both of these methods, X-ray crystallography and NMR spectroscopy, cannot determine the structure of a large protein in solution, other experimental methods exist and allow to gather at least some limited amount of information (Koch, Vachette, & Svergun, 2003; Putnam, Hammel, Hura, & Tainer, 2007; D. I. Svergun & Koch, 2003; D. I. Svergun, Petoukhov, & Koch, 2001; Tsuruta & Irving, 2008). Small angle X-ray scattering (SAXS) and the related small angle neutron scattering (SANS) are two such methods.

In a SAXS experiment, a monochromatic X-ray beam is directed at the protein sample in solution. The beam's photons are scattered elastically by proteins as well as by the solution, and angles and intensities of scattered photons are recorded by a detector. Because the solution contains multiple orientations of the protein of interest (isotropic scattering), the pattern is radially averaged into a one-dimensional scattering plot of the scattering intensity I versus the scattering vector q. By recording scattering data for the solution only and for the solution containing the protein, the difference between the two, which is the scattering contribution of the protein, can be calculated. From such a scattering plot, the maximum diameter of the protein $D_{\max}$ and a pair-distance distribution function $P(r)$ can be derived. This is insufficient for determining the protein's structure; it does however provide information to verify the correctness of a model (Forster et al., 2008; Mertens & Svergun, 2010).

Experimental information helps BCL::Fold

BCL::Fold is a de-novo protein structure prediction algorithm that is based on the assumption that native proteins exist in their lowest energy conformation; it uses a Monte Carlo optimization algorithm to build protein models (Karakas et al., 2012). Because of the vast conformational space that needs to be

searched to find the lowest energy state of a protein, BCL::Fold uses a simplified representation for its protein models to allow to sample larger and more complex topologies.

Protein models are assembled from disconnected helical and strand secondary structure building blocks that are based on secondary structure prediction results from a number of methods, PSIPRED (Jones, 1999c) and JUFO (Leman et al., 2013), among others.

Each change to a protein model by the Monte Carlo algorithm during the optimization is evaluated by a knowledge-based scoring function. The ability of the scoring function to find the protein's lowest energy conformation is limited by the reduced representation of the model. Thus only an unlikely conformation will be rejected, but a native cannot be distinguished from a fairly low energy conformation.

In this situation, experimental information, even if limited, can help BCL::Fold to rule out some low-energy conformations as incorrect models, and determine the correct fold. This can be done either by filtering models that were built with the default scoring function, or by adding a restraint-based scoring term to the scoring function thus limiting the search space to models that fulfill the experimental data (Woetzel et al., 2012).

Here we specify the discriminatory power of SAXS experimental information, in particular the scattering intensities, for distinguishing different protein shapes, and describe the improvements possible by using such data as restraints for predicting protein models with BCL::Fold.

Results

PISCES benchmark set to verify the simulation of profiles

From the PISCES protein culling server (Wang & Dunbrack, 2005), a set of 455 non-redundant proteins was obtained to verify the correctness of our SAXS scattering profiles. We used a 20 % identify cutoff, 1.6 Å resolution cutoff, and 0.25 R-factor cutoff value to define the protein subset.

Simulation of scattering profiles agrees with other methods

First, to ensure the correctness of our simulation of SAXS scattering profiles, we compared our profiles with the ones computed by CRYSOL (D. Svergun, Barberato, & Koch, 1995). This allows the use of a much larger set of proteins rather than be limited to proteins for which published experimental results are available. Simulating experimental profiles also removes the influence of experimental errors in these comparisons.

For each pair of protein models, we compared the computed SAXS scattering profile to the "experimental" profile generated by CRYSOL. If a computed profile had the best agreement with the correct "experimental" profile, and this "experimental" profile also had the best agreement with the SAXS scattering profile computed from the correct protein, then this was labeled a true positive; otherwise, it was categorized as false positive.

For the full atom protein models, this criteria of matching our computed scattering profile to CRYSOL's was fulfilled in almost all cases. Only for three of the 455 proteins, the computed SAXS profile did not score best when compared to the "experimental" profile, but it scored as second and third best instead. When plotted as a receiver operating characteristic (ROC) curve, the AUC is 0.9995, which is extremely

close to the optimal result of 1.0, and far above the AUC value for randomly assigning profiles to protein models of 0.5.

Approximating instead of omitting atoms improves profiles

Next, we used the PISCES benchmark set described above to test the discriminatory power of the SAXS profiles for protein models with reduced representation.

Approximating atom positions led to a drop in the ability to assign the correct profile and protein model to each other; this drop is higher if more atom positions are approximated. When only side chain atoms are approximated onto the C_β position, the AUC drops only very slightly to 0.9962; approximating side chain atoms onto the C_β position and loop amino acids on a parabolic path between the previous and the next SSE, the AUC drops to 0.9264 (Figure 16), which is still substantially above the random AUC of 0.5 and allows for selecting the correct protein model in most cases.

Not approximating side chain and loop atom positions and simply omitting them, results in a significant decrease of an AUC to 0.7085 (Figure 16).

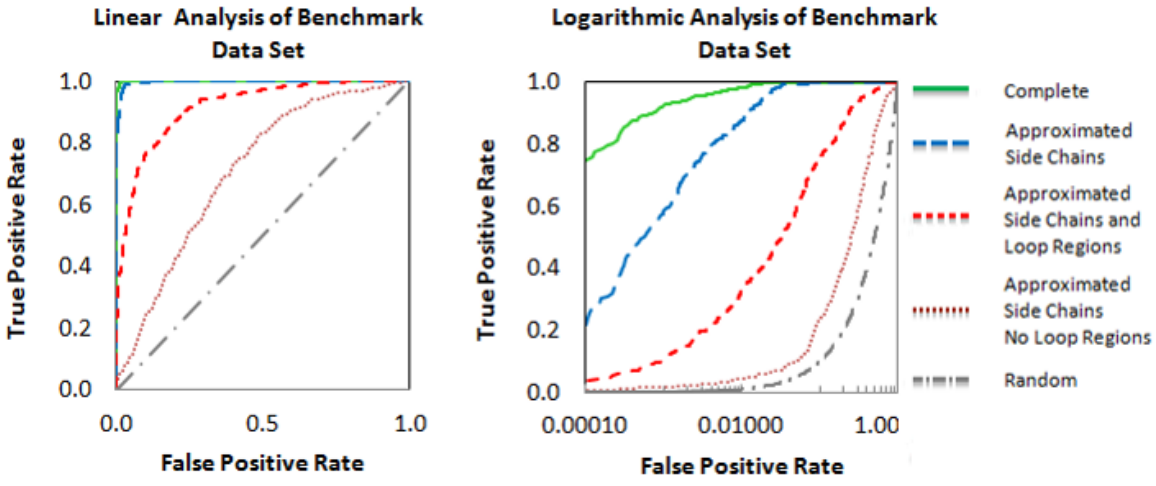

Figure 16. ROC curves to determine discriminatory power of SAXS data for complete (full atom representation) and different levels of reduced representations.

SAXS distinguishes shapes

We compared the derivative score for two SAXS scattering profiles with the MAMMOTH score (Ortiz et al., 2002) of the two corresponding protein models to test whether SAXS can differentiate protein folds. A high MAMMOTH score indicates similar proteins and, as before, a low derivative score indicates similar SAXS scattering profiles. We again used the PISCES set of 455 proteins.

Three groups of protein pairs can be identified in this comparison; pairs with high MAMMOTH and low derivative score, which are proteins of similar structure and shape; pairs of low MAMMOTH and high derivative score, i.e. proteins that are dissimilar in structure and shape; and pairs with low MAMMOTH and derivative score, which are proteins that are dissimilar in their structure, but have a similar shape. Missing from this are protein pairs that exhibit a dissimilar structure while having a shape that resembles the other (Figure 17).

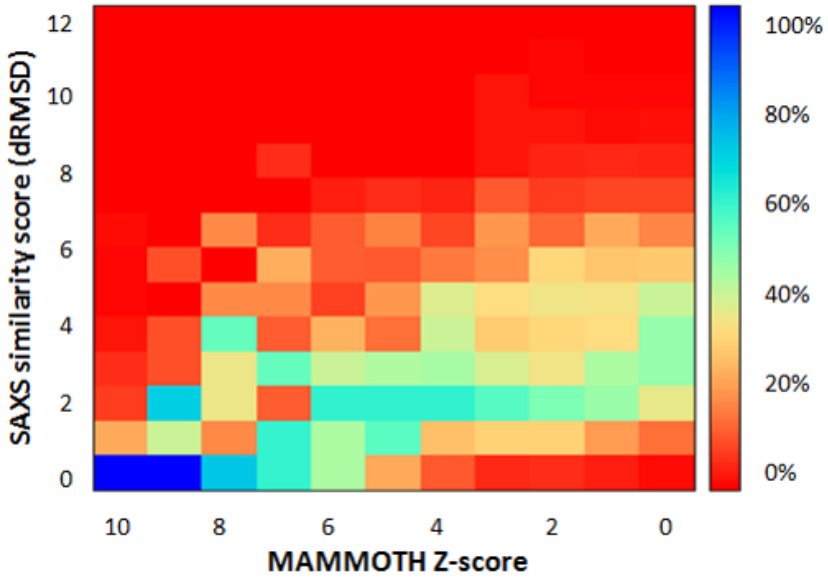

Figure 17. The correlation between SAXS similarity score and MAMMOTH Z-score. No models were found to be dissimilar by SAXS similarity score (high score) and similar by MAMMOTH Z-score (high score) indicating the SAXS information is necessary but insufficient to identify the correct protein.

Comparing SAXS scattering profiles, one can distinguish proteins of different shape; however, structurally different proteins with similar shape cannot be distinguished (Figure 17).

Folding benchmark set with shapes not ideal for BCL::Score

Four proteins were selected for benchmarking BCL::Fold, 1CI6A, 3PDYA, 1PBVA, and 1J27A. They were selected based on the amount of information that the SAXS scattering profile could add to the BCL::Fold scoring function, e.g. can the SAXS profile complement the radius of gyration scoring term with experimental information. For two proteins, 1CI6A and 3PDYA, the native is not its most compact shape and the scattering profile could complement the scoring function; the other two proteins, 1PBVA, and 1J27A, have the shape of a globular protein, similar to what the radius of gyration score is reflecting, and the scattering profile is expected to not improve the predicted models.

Filtering and Folding with SAXS data improves models

For each protein in the folding benchmark set, 1,000 models were created each without and with applying the scoring term based on the derivative score of the SAXS scattering profile.

The first set of models built without the SAXS scoring term were subsequently sorted and filtered by the derivative score. The enrichment for the top 10 % of the models was calculated, resulting in possible enrichments between 0 and 10, with 0 denoting the maximal de-enrichment (non-models were sorted first), 1 meaning no enrichment i.e. same probability to find a native-like model as before the filtering, and 10 indicating that native-like models were preferred by the SAXS score and sorted first. The enrichments for the proteins not in the most compact globular shape were 3.6 and 4.8, showing that the SAXS scattering profile adds information to find models in the protein's native conformation. For the other two proteins, the enrichments were 1.4 and 0.9, indicating no improvement (Figure 18).

The second set of models were used to test the improvement in model quality for models built with the SAXS scoring term active versus models built without it. The RMSD100 of a predicted model to the native structure was used as the quality measure. Similar to the enrichment results, the two proteins with a non-compact shape showed improvements in RMSD100, while the two globular proteins showed no improvement (Figure 18).

The information contained in a SAXS scattering profile helps in both the folding as well as the filtering of existing models. The improvements vary, however, with the amount of information that is added to the scoring function by the scattering profile.

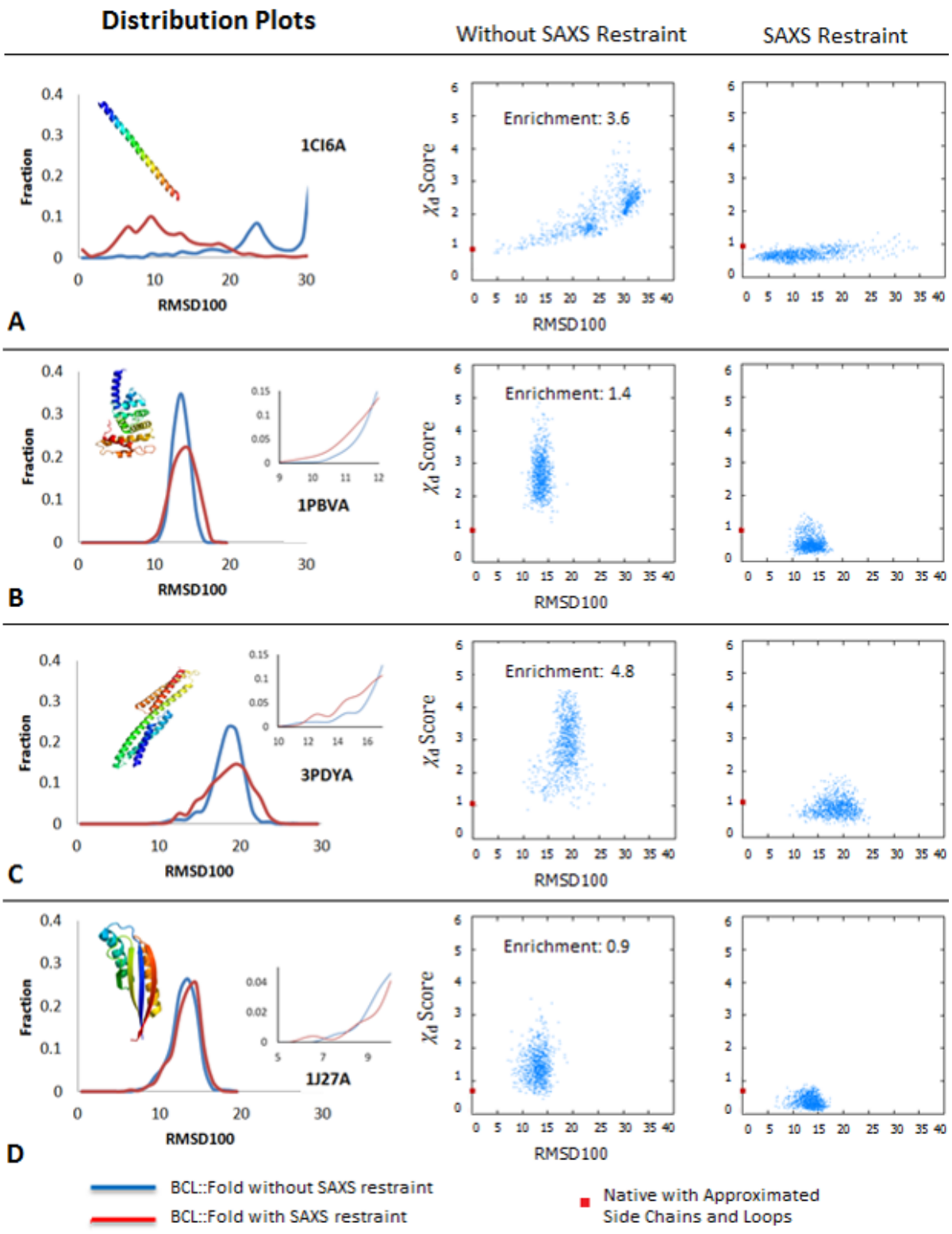

Figure 18. Model RMSD100 distributions and enrichments for the four proteins in the BCL::Fold benchmark set (A-D). Proteins A and C show improvements in RMSD100 and enrichment, for proteins B and D the SAXS data does not affect either. For each protein, the RMSD100 distribution for folding with and without SAXS restraint (left), and the correlations between SAXS similarity score and RMSD100 for models built without (center) and with SAXS restraint score (right) is shown.

Discussion

The results from the comparison of MAMMOTH score with the derivative score based on SAXS scattering profiles show that we can simulate a scattering profile from a protein model with sufficient accuracy to distinguish different shapes, even from a model having only a reduced representation as long as missing atoms are approximated.

However, it is not possible to discriminate different folds based on scattering profiles. This means, if the derivative score is high, indicating a different shape, the protein model is incorrect (or two proteins are different); if the derivative score is low, the protein model can be correct, but it does not have to be; the agreement of SAXS scattering profiles is necessary, but not sufficient for a correct model. However, this is exactly the use case for BCL::Fold together with sparse experimental data; the search space is limited and the search can be focused on the space that is in agreement with the data.

This holds true for both filtering and folding protein models. If the SAXS scattering profile contains information that limits the likelihood of a conformation beyond what the default BCL scoring function does, this can be used to either filter a disagreeing model, or reject changes to a protein model during the folding process that increase the disagreement of this model with the SAXS data.

While the information that a SAXS experiment can provide is useful and can improve the models BCL::Fold can build as well as help identifying the correct models, it cannot narrow down the number of models to just a small number of agreeing models, because the shape information is insufficient.

Multiple others ways exist to use this SAXS information. The shape information is in particular very informative for determining the multimeric state of a protein. Currently, the multimeric state and symmetry are user inputs into BCL::Fold when using the protocol for multimeric folding; the protocol could use shape information derived from SAXS data instead.

The SAXS scattering profile can be converted into a distribution of pairwise distances $P(r)$. This experimentally determined distribution can be compared to the distance distribution of a protein model. Unlike a SAXS scattering profile, which is limited to complete models, the distance distribution can always be to evaluate the correctness of a model. This should allow for a better restriction of the folding space that is explored by BCL::Fold during the folding process; it could also be more accurate, because the model's distribution can be directly compared without any conversions or approximations that are necessary for the simulation of a scattering profile.

Methods

Simulate SAXS data

We simulate a SAXS experiment in silico for a given BCL protein model to compare the simulated data with the experimentally acquired data, and evaluate their agreement.

Knowing the positions of the atoms in the protein model, we can apply the Debye formula to calculate a scattering profile. Debye's formula (Debye, 1915) assumes only a single scattering event, only elastic scattering, and freely rotating particles, with all orientations averaging (isotropic scattering). Debye's formula describes the intensity I, which is a function of the momentum transfer vector q, as the sum of the product of the form factors $F(q)$, the momentum transfer vector q, and the atom pair distance r for all pairs of m atoms.

$$I(q) = \sum_{i=1}^{m} \sum_{j=1}^{m} F_i(q) F_j(q) \frac{\sin(q r_{ij})}{q r_{ij}}$$

The momentum transfer vector q depends on the scattering angle ϕ and the wavelength λ.

$$q = 4\pi\lambda \sin\phi$$

The form factors F describe the shape of the atoms. To account for the displaced solvent contribution, the form factor F is calculated as difference between the form factor in vacuo f_v and the form factor of the displaced solvent f_s.

$$F(q) = f_v(q) - f_s(q)$$

The form factor in vacuo f_v is approximated by a combination of relativistic Dirac-Slater wave functions and numerical Hartree-Fock wave functions. The Hartree-Fock scattering factors were computed from $q = 0$ to 1.5 at intervals of 0.01 Å^{-1}. These scattering factors were fit to the 5-gaussian Cromer-Mann analytic function with the nine atom specific constants a_i, b_i, and c, with $i = 1$ to 4.

$$f_v(q) = \sum_{i=1}^{4} a_i \cdot \exp\left(-b_i \left(\frac{q}{4\pi}\right)^2\right) + c$$

This approximation is valid only for range of $q = 0$ to 2.0 Å^{-1}, but this is sufficient for simulating SAXS scattering experiments, which measure only up to $q \approx 0.33$ Å^{-1}. The displaced solvent form factor f_s is approximated by the excluded solvent volume V of the volume displaced by the atoms.

These equations are used to compute a scattering profile with Q values of different momentum transfer vectors q and associated intensities $I(q)$ for a protein with atoms in known positions (Cromer & Mann, 1968; Cromer & Waber, 1965; Debye, 1915; Fraser, MacRae, & Suzuki, 1978; Schneidman-Duhovny, Hammel, & Sali, 2010; Stovgaard, Andreetta, Ferkinghoff-Borg, & Hamelryck, 2010).

Simulate for BCL::Fold models with reduced representation

BCL::Fold uses a reduced representation when assembling protein models that does not contain loop regions or side chains. For computing a SAXS scattering profile, these missing atoms have to be

approximated, since leaving them out of the intensity summation leads to substantial differences in the profiles.

The positions of amino acids in loop regions between two secondary structure elements (SSEs) are approximated by placing them on a parabolic path that connects the vector through the N-terminal (previous) SSE with the vector through the C-terminal (following) SSE. The amino acids were placed along the path 3.8 Å apart, which is the average distance between amino acids.

Amino acids in N- and C-termini are not summed up in the computation of the scattering plot.

The form factors for the side chain atoms were placed at the C_β position of an amino acid, except for Glycine which does not have any side chain atoms. This is similar to the treatment of hydrogens in CRYSOL.

Derivative score for comparing SAXS profiles

To compare two SAXS scattering profiles with the same number of Q values, both are first normalized to the range $[0,1]$ and converted to log scale, to increase the importance of larger q. The simulated scattering profile is scaled by the scaling factor c to overlap both profiles. The scaling factor is based on the intensities of both scattering profiles.

$$c = \frac{\sum_{k=1}^{Q} I_s(q_k) \cdot I_e(q_k)}{\sum_{k=1}^{Q} I_e(q_k)^2}$$

Here, I_s and I_e are the simulated and experimentally measured intensities for a specified momentum transfer vector q_k out of all Q values.

After applying the scaling factor to the simulated scattering profile, the derivative of both profiles is computed using cubic splines. The derivative profile difference d is then computed from the differences between the derived intensities.

$$d = \sqrt{\frac{1}{Q}\sum_{k=1}^{Q}\left(I_e'(q_k) - I_s'(q_k)\right)^2}$$

Thus, the derivative score d calculates the difference between two scattering profiles returning zero for identical profiles and values larger than zero for differing profiles.

SAXS restraints extend BCL::Score

The BCL::Fold scoring function is a weighted sum of terms, which are either statistically derived from experimentally determined protein structures, or penalty terms. The statistically derived terms evaluate the arrangement of SSEs (pairing and packing of helices and strands) and amino acids, and the compactness of a model by its radius of gyration; the penalty terms ensure that clashes are avoided.

Additional terms can be added to the weighted sum to evaluate a BCL model against experimental data like a SAXS scattering profile. A score term is added to the BCL scoring function penalizing models whose simulated scattering profile does not agree with experimentally determined scattering profile based on the described derivative score. The weight is set to a value that results in the experimental terms accounting for about half of the total score sum.

Because the experimental scattering profile represents the shape of the complete protein, the SAXS derivative score is not used in the first two of six stages of BCL::Fold (assembly stages). In these stages, the model's shape can differ substantially from the experimentally determined information because of the incompleteness of the model; however, penalizing an incomplete model for its incorrect shape is

counterproductive. Therefore we only utilize the SAXS derivative score once a model contains a complete set of SSEs (in the last assembly stages and in the refinement stage) and modifications to the model, like rearranging the SSEs, would cause it to change its overall shape.

CHAPTER V.

CONCLUSIONS

Summary of this work

The body of the work in this dissertation thesis focuses on improving BCL::Fold using three different approaches. The first part focuses on the development of BCL::Align; the second and third parts review the improvements to the general scoring function based on the CASP10 results and the utilization of SAXS experimental data to extend the BCL scoring function to guide the folding process. The following sections will summarize the results and implications, and entertain some thoughts towards future improvements.

BCL::Align

BCL::Align is an alignment algorithm producing an optimal alignment given the weights of its customizable scoring function. With this unique scoring function, BCL::Align's recognition of sequence features can be specifically tuned to the task and to the level of sequence similarity. This allows it to perform sequence alignment, fold recognition, and alignments for similarity of other features. Weights optimized for different tasks cannot only be combined; it is also possible to add new scoring terms for new features.

Despite its deficiencies in efficiency, BCL::Align can be used for a wide variety of tasks and sequence similarities. On top, it can be easily tuned for new purposes.

One possible application lies within BCL::Fold. Using BCL::Align's fold recognition, it is possible to find pieces of already known structure to use in a fragment-based assembly during the folding process. Depending on the sequence of the target protein and its environment, BCL::Align can align sequences to find patterns in charge, hydrophobicity, or interactions.

BCL::Fold in CASP10

We tested BCL::Fold in CASP10. To evaluate the quality of our models, we designed the topology score. It specifically evaluates models with the level of abstraction that BCL::Fold uses during the folding process. Using both the topology score as well as GDT_TS, the majority of targets had the correct fold. In addition, we found that a substantial fraction of loops and sheets arrangements were not native-like, and we suggested and implemented improvements.

While our criteria for finding the correct fold are acceptable for measuring if two protein models have a similar topology, the bar has to be raised to allow subsequent refinement steps to produce models open to more implications. Currently our criteria for GDT_TS is > 33 %, and while the topology score has a higher cutoff (> 0.8), is not considering false positives.

To produce better models, BCL::Fold's scoring has to use the protein's sequence more and consequently increase the weight for this scoring term (currently 0.35 per amino acid, with often total scores > 5,000). The secondary structure, which is heavily used to sample and predict protein structure, is not fully defining the tertiary structure; the side chains, which depend on the amino acid types, and their interactions define the tertiary structure.

BCL::Fold with SAXS experimental data

A new scoring term to use SAXS scattering profile data in the BCL::Fold scoring function has been implemented. The comparison of a simulated SAXS profile to experimental data is able to discriminate correct and incorrect protein shapes. We have shown that for non-globular protein shapes, SAXS data can help to enrich for correct models when used for filtering as well as improve the RMSD100 to the native when used during the folding process.

The simulation of a SAXS scattering profile using the Debye formula is a computationally expensive step and has to be executed for every step of the folding process. Converting the SAXS scattering profile into a distribution of pairwise distances avoids this computational cost. It also makes the experimental data more usable, because a distance distribution is quickly computed; it can also be directly evaluated from the model without any scaling. This would allow for a better restriction of the folding space that is explored by BCL::Fold; it also avoids approximations during the use of Debye's formula and thus can be more accurate.

In summary it can be concluded that BCL::Align is an alignment algorithm that can be very flexibly tailored to different tasks, including potential new ones e.g. in BCL::Fold's scoring function. BCL::Fold is able to quickly sample large topologies which are problematic for other structure prediction algorithms. Future changes in sampling and scoring, like the ones outlined above, may increase prediction accuracy and extend the possibilities for applying BCL::Fold to larger folding problems that can only be tackled because of its unique simplified model representation.

CHAPTER VI.

REFERENCES

Adrian, M., Dubochet, J., Lepault, J., & McDowall, A. W. (1984). Cryo-electron microscopy of viruses. *Nature, 308*(5954), 32-36.

Alber, F., Dokudovskaya, S., Veenhoff, L. M., Zhang, W., Kipper, J., Devos, D., . . . Sali, A. (2007). Determining the architectures of macromolecular assemblies. *Nature, 450*(7170), 683-694. doi: 10.1038/nature06404

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *J Mol Biol, 215*(3), 403-410. doi: 10.1016/S0022-2836(05)80360-2

Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W., & Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res, 25*(17), 3389-3402.

Anfinsen, C. B. (1973). Principles that Govern the Folding of Protein Chains. *Science, 181*(4096), 223-230. doi: 10.1126/science.181.4096.223

Bairoch, A., & Apweiler, R. (1998). The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1998. *Nucleic Acids Res, 26*(1), 38-42.

Baker, D. (2000). A surprising simplicity to protein folding. *Nature, 405*(6782), 39-42.

Baker, D., & Sali, A. (2001). Protein structure prediction and structural genomics. *Science, 294*(5540), 93-96.

Barton, G. J., & Sternberg, M. J. (1987). Evaluation and improvements in the automatic alignment of protein sequences. *Protein Eng, 1*(2), 89-94.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., & Wheeler, D. L. (2006). GenBank. *Nucleic Acids Res, 34*(Database issue), D16-20. doi: 10.1093/nar/gkj157

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., & Wheeler, D. L. (2007). GenBank. *Nucleic Acids Res, 35*(Database issue), D21-25. doi: 10.1093/nar/gkl986

Berman, H. M. (2000). The Protein Data Bank. *Nucleic Acids Res, 28*(1), 235-242. doi: 10.1093/nar/28.1.235

Berman, H. M. (2008). The Protein Data Bank: a historical perspective. *Acta Crystallographica Section A, 64*(Pt 1), 88-95. doi: 10.1107/S0108767307035623

Bill, R. M., Henderson, P. J., Iwata, S., Kunji, E. R., Michel, H., Neutze, R., . . . Vogel, H. (2011). Overcoming barriers to membrane protein structure determination. *Nat Biotechnol, 29*(4), 335-340. doi: 10.1038/nbt.1833

Billeter, M. (1992). Comparison of protein structures determined by NMR in solution and by X-ray diffraction in single crystals. *Q Rev Biophys, 25*(3), 325-377.

Blackshields, G., Wallace, I. M., Larkin, M., & Higgins, D. G. (2006). Analysis and comparison of benchmarks for multiple sequence alignment. *In Silico Biol, 6*(4), 321-339.

Bonneau, R., Ruczinski, I., Tsai, J., & Baker, D. (2002). Contact order and ab initio protein structure prediction. *Protein Sci, 11*(8), 1937-1944. doi: 10.1110/ps.3790102

Boutonnet, N. S., Rooman, M. J., Ochagavia, M. E., Richelle, J., & Wodak, S. J. (1995). Optimal protein structure alignments by multiple linkage clustering: application to distantly related proteins. *Protein Eng, 8*(7), 647-662.

Bradley, P., Malmstrom, L., Qian, B., Schonbrun, J., Chivian, D., Kim, D. E., . . . Baker, D. (2005). Free modeling with Rosetta in CASP6. *Proteins, 61 Suppl 7*, 128-134. doi: 10.1002/prot.20729

Bradley, P., Misura, K. M., & Baker, D. (2005). Toward high-resolution de novo structure prediction for small proteins. *Science, 309*(5742), 1868-1871.

Bragg, W. H., & Bragg, W. L. (1913). *The Reflection of X-rays by Crystals* (Vol. 88).

Cairns, N. J., Lee, V. M., & Trojanowski, J. Q. (2004). The cytoskeleton in neurodegenerative diseases. *J Pathol, 204*(4), 438-449. doi: 10.1002/path.1650

Canutescu, A. A., & Dunbrack, R. L., Jr. (2003). Cyclic coordinate descent: A robotics algorithm for protein loop closure. *Protein Sci, 12*(5), 963-972. doi: 10.1110/ps.0242703

Carugo, O., & Pongor, S. (2001). A normalized root-mean-square distance for comparing protein three-dimensional structures. *Protein Sci, 10*(7), 1470-1473. doi: 10.1110/ps.690101

Castillo-Davis, C. I., Kondrashov, F. A., Hartl, D. L., & Kulathinal, R. J. (2004). The functional genomic distribution of protein divergence in two animal phyla: coevolution, genomic conflict, and constraint. *Genome Res, 14*(5), 802-811. doi: 10.1101/gr.2195604

Chapman, H. N., Fromme, P., Barty, A., White, T. A., Kirian, R. A., Aquila, A., . . . Spence, J. C. (2011). Femtosecond X-ray protein nanocrystallography. *Nature, 470*(7332), 73-77. doi: 10.1038/nature09750

Chaudhuri, T. K., & Paul, S. (2006). Protein-misfolding diseases and chaperone-based therapeutic approaches. *FEBS J, 273*(7), 1331-1349. doi: 10.1111/j.1742-4658.2006.05181.x

Chivian, D., & Baker, D. (2006). Homology modeling using parametric alignment ensemble generation with consensus and energy-based model selection. *Nucleic Acids Res, 34*(17), e112. doi: 10.1093/nar/gkl480

Chivian, D., Kim, D. E., Malmstrom, L., Schonbrun, J., Rohl, C. A., & Baker, D. (2005). Prediction of CASP6 structures using automated Robetta protocols. *Proteins, 61 Suppl 7*, 157-166. doi: 10.1002/prot.20733

Chung, S. Y., & Subbiah, S. (1996). A structural explanation for the twilight zone of protein sequence homology. *Structure, 4*(10), 1123-1127.

Cline, M., Hughey, R., & Karplus, K. (2002). Predicting reliable regions in protein sequence alignments. *Bioinformatics, 18*(2), 306-314.

Coutsias, E. A., Seok, C., & Dill, K. A. (2005). Rotational superposition and least squares: the SVD and quaternions approaches yield identical results. Reply to the preceding comment by G. Kneller. *J Comput Chem, 26*(15), 1663-1665. doi: 10.1002/jcc.20316

Crick, F. (1970). Central dogma of molecular biology. *Nature, 227*(5258), 561-563.

Crick, F. H., Barnett, L., Brenner, S., & Watts-Tobin, R. J. (1961). General nature of the genetic code for proteins. *Nature, 192*, 1227-1232.

Crippen, G. M. (1975). Global optimization and polypeptide conformation. *Journal of Computational Physics, 18*(2), 224-231. doi: 10.1016/0021-9991(75)90030-3

Cromer, D. T., & Mann, J. B. (1968). X-ray scattering factors computed from numerical Hartree-Fock wave functions. *Acta Crystallographica Section A, 24*(2), 321-324. doi: doi:10.1107/S0567739468000550

Cromer, D. T., & Waber, J. T. (1965). Scattering factors computed from relativistic Dirac–Slater wave functions. *Acta Crystallographica, 18*(1), 104-109. doi: 10.1107/S0365110X6500018X

Dayhoff, M. O., Schwartz, R. M., & Orcutt, B. C. (1978). A model of evolutionary change in proteins. *Atlas of protein sequence and structure, 5*(suppl 3), 345-351. doi: citeulike-article-id:4442167

Debye, P. (1915). Zerstreuung von Röntgenstrahlen. *Annalen der Physik, 351*(6), 809-823. doi: 10.1002/andp.19153510606

Do, C. B., Mahabhashyam, M. S., Brudno, M., & Batzoglou, S. (2005). ProbCons: Probabilistic consistency-based multiple sequence alignment. *Genome Res, 15*(2), 330-340. doi: 10.1101/gr.2821705

Dong, E., Smith, J., Heinze, S., Alexander, N., & Meiler, J. (2008). BCL::Align-sequence alignment and fold recognition with a custom scoring function online. *Gene, 422*(1-2), 41-46. doi: 10.1016/j.gene.2008.06.006

Dutta, S., & Berman, H. M. (2005). Large macromolecular complexes in the Protein Data Bank: a status report. *Structure, 13*(3), 381-388. doi: 10.1016/j.str.2005.01.008

Edgar, R. C. (2004). MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res, 32*(5), 1792-1797. doi: 10.1093/nar/gkh340

Edgar, R. C., & Sjolander, K. (2004). A comparison of scoring functions for protein sequence profile alignment. *Bioinformatics, 20*(8), 1301-1308. doi: 10.1093/bioinformatics/bth090

Feng, D. F., & Doolittle, R. F. (1987). Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J Mol Evol, 25*(4), 351-360.

Finn, R. D., Tate, J., Mistry, J., Coggill, P. C., Sammut, S. J., Hotz, H. R., . . . Bateman, A. (2008). The Pfam protein families database. *Nucleic Acids Res, 36*(Database issue), D281-288. doi: 10.1093/nar/gkm960

Fischer, A., Alexander, N., Woetzel, N., Karakas, M., Weiner, B., & Meiler, J. (2014 (Submitted)). BCL::MP-Fold: membrane protein structure prediction guided by EPR restraints. *Structure*.

Fölling, A. (2009). Über Ausscheidung von Phenylbrenztraubensäure in den Harn als Stoffwechselanomalie in Verbindung mit Imbezillität. *Hoppe-Seyler´s Zeitschrift für physiologische Chemie, 227(1-4)*, 169-181. doi: 10.1515/bchm2.1934.227.1-4.169

Forster, F., Webb, B., Krukenberg, K. A., Tsuruta, H., Agard, D. A., & Sali, A. (2008). Integration of small-angle X-ray scattering data into structural modeling of proteins and their assemblies. *J Mol Biol, 382*(4), 1089-1106. doi: 10.1016/j.jmb.2008.07.074

Fraser, R. D. B., MacRae, T. P., & Suzuki, E. (1978). An improved method for calculating the contribution of solvent to the X-ray diffraction pattern of biological molecules. *Journal of Applied Crystallography, 11*(6), 693-694. doi: 10.1107/S0021889878014296

Freitas, R. A. (1999). *Nanomedicine*. Austin, TX: Landes Bioscience.

Ginalski, K., Elofsson, A., Fischer, D., & Rychlewski, L. (2003). 3D-Jury: a simple approach to improve protein structure predictions. *Bioinformatics, 19*(8), 1015-1018. doi: 10.1093/bioinformatics/btg124

Ginalski, K., Pas, J., Wyrwicz, L. S., von Grotthuss, M., Bujnicki, J. M., & Rychlewski, L. (2003). ORFeus: Detection of distant homology using sequence profiles and predicted secondary structure. *Nucleic Acids Res, 31*(13), 3804-3807.

Grantcharova, V., Alm, E. J., Baker, D., & Horwich, A. L. (2001). Mechanisms of protein folding. *Curr Opin Struct Biol, 11*(1), 70-82. doi: 10.1016/s0959-440x(00)00176-7

Hagler, A. T., Huler, E., & Lifson, S. (1974). Energy functions for peptides and proteins. I. Derivation of a consistent force field including the hydrogen bond from amide crystals. *J Am Chem Soc, 96*(17), 5319-5327.

Hagler, A. T., & Lifson, S. (1974). Energy functions for peptides and proteins. II. The amide hydrogen bond and calculation of amide crystal properties. *J Am Chem Soc, 96*(17), 5327-5335.

Heinze, S., Putnam, D. K., Fischer, A. W., Kohlmann, T., Weiner, B. E., & Meiler, J. (2015). CASP10-BCL::Fold efficiently samples topologies of large proteins. *Proteins, 83*(3), 547-563. doi: 10.1002/prot.24733

Henikoff, S., & Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A, 89*(22), 10915-10919.

Hogeweg, P., & Hesper, B. (1984). The alignment of sets of sequences and the construction of phyletic trees: an integrated method. *J Mol Evol, 20*(2), 175-186.

Hopf, T. A., Colwell, L. J., Sheridan, R., Rost, B., Sander, C., & Marks, D. S. (2012). Three-dimensional structures of membrane proteins from genomic sequencing. *Cell, 149*(7), 1607-1621. doi: 10.1016/j.cell.2012.04.012

Howard, M. J. (1998). Protein NMR spectroscopy. *Curr Biol, 8*(10), R331-333.

Hughey, R., & Krogh, A. (1996). Hidden Markov models for sequence analysis: extension and analysis of the basic method. *Comput Appl Biosci, 12*(2), 95-107.

Ivankov, D. N., & Finkelstein, A. V. (2004). Prediction of protein folding rates from the amino acid sequence-predicted secondary structure. *Proc Natl Acad Sci U S A, 101*(24), 8942-8944. doi: 10.1073/pnas.0402659101

Jones, D. T. (1999a). GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. *J Mol Biol, 287*(4), 797-815. doi: 10.1006/jmbi.1999.2583

Jones, D. T. (1999c). Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol, 292*(2), 195-202. doi: 10.1006/jmbi.1999.3091

Jukes, T. H., & Osawa, S. (1990). The genetic code in mitochondria and chloroplasts. *Experientia, 46*(11-12), 1117-1126.

Kabsch, W. (1976). Solution for Best Rotation to Relate 2 Sets of Vectors. *Acta Crystallographica Section A, 32*(Sep1), 922-923. doi: Doi 10.1107/S0567739476001873

Karakas, M., Woetzel, N., Staritzbichler, R., Alexander, N., Weiner, B. E., & Meiler, J. (2012). BCL::Fold--de novo prediction of complex and large protein topologies by assembly of secondary structure elements. *PLoS One, 7*(11), e49240. doi: 10.1371/journal.pone.0049240

Karplus, K., Barrett, C., & Hughey, R. (1998). Hidden Markov models for detecting remote protein homologies. *Bioinformatics, 14*(10), 846-856.

Kaufmann, K. W., Lemmon, G. H., Deluca, S. L., Sheehan, J. H., & Meiler, J. (2010). Practically useful: what the Rosetta protein modeling suite can do for you. *Biochemistry, 49*(14), 2987-2998. doi: 10.1021/bi902153g

Kim, E. K., & Choi, E. J. (2010). Pathological roles of MAPK signaling pathways in human diseases. *Biochim Biophys Acta, 1802*(4), 396-405. doi: 10.1016/j.bbadis.2009.12.009

Kobe, B., Guncar, G., Buchholz, R., Huber, T., Maco, B., Cowieson, N., . . . Forwood, J. K. (2008). Crystallography and protein-protein interactions: biological interfaces and crystal contacts. *Biochem Soc Trans, 36*(Pt 6), 1438-1441. doi: 10.1042/BST0361438

Koch, M. H., Vachette, P., & Svergun, D. I. (2003). Small-angle scattering: a view on the properties, structures and structural changes of biological macromolecules in solution. *Q Rev Biophys, 36*(2), 147-227.

Kozma, D., Simon, I., & Tusnady, G. E. (2013). PDBTM: Protein Data Bank of transmembrane proteins after 8 years. *Nucleic Acids Res, 41*(Database issue), D524-529. doi: 10.1093/nar/gks1169

Kryshtafovych, A., Barbato, A., Fidelis, K., Monastyrskyy, B., Schwede, T., & Tramontano, A. (2013). Assessment of the assessment: Evaluation of the model quality estimates in CASP10. *Proteins*. doi: 10.1002/prot.24347

Kubelka, J., Hofrichter, J., & Eaton, W. A. (2004). The protein folding 'speed limit'. *Curr Opin Struct Biol, 14*(1), 76-88. doi: 10.1016/j.sbi.2004.01.013

Kuhlbrandt, W. (2014a). Biochemistry. The resolution revolution. *Science, 343*(6178), 1443-1444. doi: 10.1126/science.1251652

Kuhlbrandt, W. (2014c). Cryo-EM enters a new era. *Elife, 3*, e03678. doi: 10.7554/eLife.03678

Leaver-Fay, A., Tyka, M., Lewis, S. M., Lange, O. F., Thompson, J., Jacak, R., . . . Bradley, P. (2011). ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. *Methods Enzymol, 487*, 545-574. doi: B978-0-12-381270-4.00019-6 [pii]

10.1016/B978-0-12-381270-4.00019-6

Leman, J. K., Mueller, R., Karakas, M., Woetzel, N., & Meiler, J. (2013). Simultaneous prediction of protein secondary structure and transmembrane spans. *Proteins, 81*(7), 1127-1140. doi: 10.1002/prot.24258

Lindahl, E., & Elofsson, A. (2000). Identification of related proteins on family, superfamily and fold level. *J Mol Biol, 295*(3), 613-625. doi: 10.1006/jmbi.1999.3377

Lundstrom, J., Rychlewski, L., Bujnicki, J., & Elofsson, A. (2001). Pcons: a neural-network-based consensus predictor that improves fold recognition. *Protein Sci, 10*(11), 2354-2362.

Mao, B., Tejero, R., Baker, D., & Montelione, G. T. (2014). Protein NMR structures refined with Rosetta have higher accuracy relative to corresponding X-ray crystal structures. *J Am Chem Soc, 136*(5), 1893-1906. doi: 10.1021/ja409845w

Meiler, J., & Baker, D. (2003). Coupled prediction of protein secondary and tertiary structure. *Proc Natl Acad Sci U S A, 100*(21), 12105-12110. doi: 10.1073/pnas.1831973100

Meiler, J., Muller, M., Zeidler, A., & Schmaschke, F. (2001). Generation and evaluation of dimension-reduced amino acid parameter representations by artificial neural networks. *J Mol Model, 7*(9), 360-369. doi: DOI 10.1007/s008940100038

Mertens, H. D., & Svergun, D. I. (2010). Structural characterization of proteins and complexes using small-angle X-ray solution scattering. *J Struct Biol, 172*(1), 128-141. doi: 10.1016/j.jsb.2010.06.012

Mewes, H. W., Heumann, K., Kaps, A., Mayer, K., Pfeiffer, F., Stocker, S., & Frishman, D. (1999). MIPS: a database for genomes and protein sequences. *Nucleic Acids Res, 27*(1), 44-48.

Milne, J. L., Borgnia, M. J., Bartesaghi, A., Tran, E. E., Earl, L. A., Schauder, D. M., . . . Subramaniam, S. (2013). Cryo-electron microscopy--a primer for the non-microscopist. *FEBS J, 280*(1), 28-45. doi: 10.1111/febs.12078

Monastyrskyy, B., D'Andrea, D., Fidelis, K., Tramontano, A., & Kryshtafovych, A. (2013). Evaluation of residue-residue contact prediction in CASP10. *Proteins*. doi: 10.1002/prot.24340

Moult, J. (2005). A decade of CASP: progress, bottlenecks and prognosis in protein structure prediction. *Curr Opin Struct Biol, 15*(3), 285-289. doi: 10.1016/j.sbi.2005.05.011

Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T., & Tramontano, A. (2014). Critical assessment of methods of protein structure prediction (CASP)--round x. *Proteins, 82 Suppl 2*, 1-6. doi: 10.1002/prot.24452

Moult, J., Fidelis, K., Kryshtafovych, A., & Tramontano, A. (2011). Critical assessment of methods of protein structure prediction (CASP)--round IX. *Proteins, 79 Suppl 10*, 1-5. doi: 10.1002/prot.23200

Murzin, A. G., Brenner, S. E., Hubbard, T., & Chothia, C. (1995). SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol, 247*(4), 536-540. doi: 10.1006/jmbi.1995.0159

Naganathan, A. N., & Munoz, V. (2005). Scaling of folding times with protein size. *J Am Chem Soc, 127*(2), 480-481. doi: 10.1021/ja044449u

Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol, 48*(3), 443-453.

Notredame, C., Higgins, D. G., & Heringa, J. (2000). T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol, 302*(1), 205-217. doi: 10.1006/jmbi.2000.4042

Nugent, T., Cozzetto, D., & Jones, D. T. (2013). Evaluation of predictions in the CASP10 model refinement category. *Proteins*. doi: 10.1002/prot.24377

Oliveira, A. P., Nielsen, J., & Forster, J. (2005). Modeling Lactococcus lactis using a genome-scale flux model. *BMC Microbiol, 5*, 39. doi: 10.1186/1471-2180-5-39

Orengo, C. A., Michie, A. D., Jones, S., Jones, D. T., Swindells, M. B., & Thornton, J. M. (1997). CATH--a hierarchic classification of protein domain structures. *Structure, 5*(8), 1093-1108.

Ortiz, A. R., Strauss, C. E., & Olmea, O. (2002). MAMMOTH (matching molecular models obtained from theory): an automated method for model comparison. *Protein Sci, 11*(11), 2606-2621. doi: 10.1110/ps.0215902

Phillips, A., Janies, D., & Wheeler, W. (2000). Multiple sequence alignment in phylogenetic analysis. *Mol Phylogenet Evol, 16*(3), 317-330. doi: 10.1006/mpev.2000.0785

Pietz, J., Kreis, R., Rupp, A., Mayatepek, E., Rating, D., Boesch, C., & Bremer, H. J. (1999). Large neutral amino acids block phenylalanine transport into brain tissue in patients with phenylketonuria. *J Clin Invest, 103*(8), 1169-1178. doi: 10.1172/JCI5017

Putnam, C. D., Hammel, M., Hura, G. L., & Tainer, J. A. (2007). X-ray solution scattering (SAXS) combined with crystallography and computation: defining accurate macromolecular structures, conformations and assemblies in solution. *Q Rev Biophys, 40*(3), 191-285. doi: 10.1017/S0033583507004635

Ratjen, F. A. (2009). Cystic fibrosis: pathogenesis and future treatment strategies. *Respir Care, 54*(5), 595-605.

Rohl, C. A., Strauss, C. E., Chivian, D., & Baker, D. (2004). Modeling structurally variable regions in homologous proteins with rosetta. *Proteins, 55*(3), 656-677. doi: 10.1002/prot.10629

Rost, B. (1996). PHD: predicting one-dimensional protein structure by profile-based neural networks. *Methods Enzymol, 266*, 525-539.

Rost, B. (1999). Twilight zone of protein sequence alignments. *Protein Eng, 12*(2), 85-94.

Rost, B., & Sander, C. (1993). Improved prediction of protein secondary structure by use of sequence profiles and neural networks. *Proc Natl Acad Sci U S A, 90*(16), 7558-7562.

Rost, B., & Sander, C. (1994). Combining evolutionary information and neural networks to predict protein secondary structure. *Proteins, 19*(1), 55-72. doi: 10.1002/prot.340190108

Rychlewski, L., Jaroszewski, L., Li, W., & Godzik, A. (2000). Comparison of sequence profiles. Strategies for structural predictions using sequence information. *Protein Sci, 9*(2), 232-241. doi: 10.1110/ps.9.2.232

Sali, A., & Blundell, T. L. (1993). Comparative protein modelling by satisfaction of spatial restraints. *J Mol Biol, 234*(3), 779-815. doi: 10.1006/jmbi.1993.1626

Sauder, J. M., Arthur, J. W., & Dunbrack, R. L., Jr. (2000). Large-scale comparison of protein sequence alignment algorithms with structure alignments. *Proteins, 40*(1), 6-22.

Scheres, S. H. (2014). Beam-induced motion correction for sub-megadalton cryo-EM particles. *Elife, 3*, e03665. doi: 10.7554/eLife.03665

Schneidman-Duhovny, D., Hammel, M., & Sali, A. (2010). FoXS: a web server for rapid computation and fitting of SAXS profiles. *Nucleic Acids Res, 38*(Web Server issue), W540-544. doi: 10.1093/nar/gkq461

Shindyalov, I. N., & Bourne, P. E. (1998). Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Eng, 11*(9), 739-747.

Simons, K. T., Kooperberg, C., Huang, E., & Baker, D. (1997). Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. *J Mol Biol, 268*(1), 209-225. doi: 10.1006/jmbi.1997.0959

Skrisovska, L., Schubert, M., & Allain, F. H. (2010). Recent advances in segmental isotope labeling of proteins: NMR applications to large proteins and glycoproteins. *J Biomol NMR, 46*(1), 51-65. doi: 10.1007/s10858-009-9362-7

Stovgaard, K., Andreetta, C., Ferkinghoff-Borg, J., & Hamelryck, T. (2010). Calculation of accurate small angle X-ray scattering curves from coarse-grained protein models. *BMC Bioinformatics, 11*, 429. doi: 10.1186/1471-2105-11-429

Svergun, D., Barberato, C., & Koch, M. H. J. (1995). CRYSOL - a Program to Evaluate X-ray Solution Scattering of Biological Macromolecules from Atomic Coordinates. *Journal of Applied Crystallography, 28*(6), 768-773. doi: doi:10.1107/S0021889895007047

Svergun, D. I., & Koch, M. H. J. (2003). Small-angle scattering studies of biological macromolecules in solution. *Reports on Progress in Physics, 66*(10), 1735-1782. doi: Pii S0034-4885(03)12688-7

Doi 10.1088/0034-4885/66/10/R05

Svergun, D. I., Petoukhov, M. V., & Koch, M. H. (2001). Determination of domain structure of proteins from X-ray solution scattering. *Biophys J, 80*(6), 2946-2953. doi: 10.1016/S0006-3495(01)76260-1

Taylor, T. J., Bai, H., Tai, C. H., & Lee, B. (2013). Assessment of CASP10 contact-assisted predictions. *Proteins*. doi: 10.1002/prot.24367

Thompson, J. D., Higgins, D. G., & Gibson, T. J. (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res, 22*(22), 4673-4680.

Thompson, J. D., Plewniak, F., & Poch, O. (1999). A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Res, 27*(13), 2682-2690.

Tsuruta, H., & Irving, T. C. (2008). Experimental approaches for solution X-ray scattering and fiber diffraction. *Curr Opin Struct Biol, 18*(5), 601-608. doi: 10.1016/j.sbi.2008.08.002

Van Walle, I., Lasters, I., & Wyns, L. (2004). Align-m--a new algorithm for multiple alignment of highly divergent sequences. *Bioinformatics, 20*(9), 1428-1435. doi: 10.1093/bioinformatics/bth116

Van Walle, I., Lasters, I., & Wyns, L. (2005). SABmark--a benchmark for sequence alignment that covers the entire known fold space. *Bioinformatics, 21*(7), 1267-1268. doi: 10.1093/bioinformatics/bth493

Viklund, H., Bernsel, A., Skwark, M., & Elofsson, A. (2008). SPOCTOPUS: a combined predictor of signal peptides and membrane protein topology. *Bioinformatics, 24*(24), 2928-2929. doi: btn550 [pii]

10.1093/bioinformatics/btn550

Viklund, H., & Elofsson, A. (2008). OCTOPUS: improving topology prediction by two-track ANN-based preference scores and an extended topological grammar. *Bioinformatics, 24*(15), 1662-1668. doi: 10.1093/bioinformatics/btn221

Wang, G., & Dunbrack, R. L., Jr. (2003). PISCES: a protein sequence culling server. *Bioinformatics, 19*(12), 1589-1591.

Wang, G., & Dunbrack, R. L., Jr. (2005). PISCES: recent improvements to a PDB sequence culling server. *Nucleic Acids Res, 33*(Web Server issue), W94-98.

Ward, J. J., McGuffin, L. J., Buxton, B. F., & Jones, D. T. (2003). Secondary structure prediction with support vector machines. *Bioinformatics, 19*(13), 1650-1655.

Weigelt, J., Wikstrom, M., Schultz, J., & van Dongen, M. J. (2002). Site-selective labeling strategies for screening by NMR. *Comb Chem High Throughput Screen, 5*(8), 623-630.

Weiner, B. E., Alexander, N., Akin, L. R., Woetzel, N., Karakas, M., & Meiler, J. (2014). BCL::Fold--protein topology determination from limited NMR restraints. *Proteins, 82*(4), 587-595. doi: 10.1002/prot.24427

Weiner, B. E., Woetzel, N., Karakas, M., Alexander, N., & Meiler, J. (2013). BCL::MP-fold: folding membrane proteins through assembly of transmembrane helices. *Structure, 21*(7), 1107-1117. doi: 10.1016/j.str.2013.04.022

Weiner, S. J., Kollman, P. A., Case, D. A., Singh, U. C., Ghio, C., Alagona, G., . . . Weiner, P. (1984). A new force field for molecular mechanical simulation of nucleic acids and proteins. *Journal of the American Chemical Society, 106*(3), 765-784. doi: 10.1021/ja00315a051

Wickstead, B., & Gull, K. (2011). The evolution of the cytoskeleton. *J Cell Biol, 194*(4), 513-525. doi: 10.1083/jcb.201102065

Woetzel, N., Karakas, M., Staritzbichler, R., Muller, R., Weiner, B. E., & Meiler, J. (2012). BCL::Score--knowledge based energy potentials for ranking protein models represented by idealized secondary structure elements. *PLoS One, 7*(11), e49242. doi: 10.1371/journal.pone.0049242

Worth, C. L., Preissner, R., & Blundell, T. L. (2011). SDM--a server for predicting effects of mutations on protein stability and malfunction. *Nucleic Acids Res, 39*(Web Server issue), W215-222. doi: 10.1093/nar/gkr363

Wu, S., Skolnick, J., & Zhang, Y. (2007). Ab initio modeling of small proteins by iterative TASSER simulations. *BMC Biol, 5*, 17. doi: 10.1186/1741-7007-5-17

Yu, H. (1999). Extending the size limit of protein nuclear magnetic resonance. *Proc Natl Acad Sci U S A, 96*(2), 332-334.

Zemla, A., Venclovas, Moult, J., & Fidelis, K. (2001). Processing and evaluation of predictions in CASP4. *Proteins, Suppl 5*, 13-21.

Zemla, A., Venclovas, C., Moult, J., & Fidelis, K. (1999). Processing and analysis of CASP3 protein structure predictions. *Proteins, Suppl 3*, 22-29.

Zhang, H., Sheng, X. R., Pan, X. M., & Zhou, J. M. (1998). Refolding of urea-denatured adenylate kinase. *Biochem J, 333 (Pt 2)*, 401-405.

Zhang, Y. (2007). Template-based modeling and free modeling by I-TASSER in CASP7. *Proteins, 69 Suppl 8*, 108-117. doi: 10.1002/prot.21702

Zhang, Y. (2008). I-TASSER server for protein 3D structure prediction. *BMC Bioinformatics, 9*, 40. doi: 10.1186/1471-2105-9-40

Zhang, Y. (2013). Interplay of I-TASSER and QUARK for template-based and ab initio protein structure prediction in CASP10. *Proteins*. doi: 10.1002/prot.24341

Zhou, Z. H. (2011). Atomic resolution cryo electron microscopy of macromolecular complexes. *Adv Protein Chem Struct Biol, 82*, 1-35. doi: 10.1016/B978-0-12-386507-6.00001-4

Zwanzig, R., Szabo, A., & Bagchi, B. (1992). Levinthal's paradox. *Proc Natl Acad Sci U S A, 89*(1), 20-22.

APPENDIX A.

PROTOCOL CAPTURE FOR CHAPTER "CASP10 – BCL::FOLD EFFICIENTLY SAMPLES TOPOLOGIES OF LARGE PROTEINS"

Overview

This protocol capture lists the steps necessary to obtain the results presented in Heinze et al. (2015).

Background

During CASP10 in summer 2012, we tested BCL::Fold for prediction of free modeling (FM) and template-based modeling (TBM) targets. BCL::Fold assembles the tertiary structure of a protein from predicted secondary structure elements (SSEs) omitting more flexible loop regions early on. This approach enables the sampling of conformational space for larger proteins with more complex topologies. In preparation of CASP11, we analyzed the quality of CASP10 models throughout the prediction pipeline to understand BCL::Fold's ability to sample the native topology, identify native-like models by scoring and/or clustering approaches, and our ability to add loop regions and side chains to initial SSE-only models. The standout observation is that BCL::Fold sampled topologies with a GDT_TS score > 33% for 12 of 18 and with a topology score > 0.8 for 11 of 18 test cases de novo. Despite the sampling success of BCL::Fold, significant challenges still exist in clustering and loop generation stages of the pipeline. The clustering approach employed for model selection often failed to identify the most native-like assembly of SSEs for further refinement and submission. It was also observed that for some β-strand proteins model refinement failed as β-strands were not properly aligned to form hydrogen bonds removing otherwise

accurate models from the pool. Further, BCL::Fold samples frequently non-natural topologies that require loop regions to pass through the center of the protein.

Protocol

Generate BCL::Fold models for CASP targets

1. Prepare the data for a target or for all targets
    a. For each requested target, create the directory structure and download the target sequence, and if available, restraints.
    b. Command for preparing all available (new) targets:

       ./run_preparation.pl --casp Casp10
    c. Command for preparing a particular target:

       ./run_preparation.pl --casp Casp10 --target T0669
2. Predict the secondary structure and transmembrane helix locations
    a. Depending on the target type (soluble, membrane), different methods are used to predict the secondary structure content (JUFO9D, PSIPRED, ProfPHD) and the positions of transmembrane helices (JUFO9D, OCTOPUS) for the specified target.
    b. Command:

       ./run_sspred.pl --target T0669

3. Predict the tertiary structure using BCL::Fold
    a. This creates the pool for the specified (sub-) sequence; it creates the cluster scripts which contain the commands to run BCL::Fold; and it runs the cluster scripts.
    b. Command:

       ./run_fold.pl --target T0669 --begin 1 --end 109
4. Filter and cluster the predicted models
    a. Filter the predicted models to exclude incomplete models; Compare all models to all others using RMSD; Cluster the models.
    b. Command (for all three steps, clustering RMSD cutoff is 9.0A by default):

       ./run_cluster.pl --target T0669
    c. Command with a specific clustering RMSD cutoff:

       ./run_cluster.pl --target T0669 --cluster-threshold 8.0
5. Inspect the clustered models
    a. Display the five best scoring models overall and for each cluster with the missing loop visualized in PYMOL
    b. Command:

       run_cluster_pymol.pl --target T0669
6. Copy the selected models to the cluster subfolder
7. Create loop models
    a. Create loop models for the all or a specific input model; the best scoring of the loop models will be returned (copied back)

b. Command for all models:

run_loops.pl --target T0669

c. Command for a specific model:

run_loops.pl --target T0669 --model model0_run0_0A.pdb

8. Copy the models with closed loops into the sidechains folder
9. Add sidechains using Rosetta
    a. Add sidechains using Rosetta relax with constraints
    b. Command:

    run_add_sc_relax.sh

    c. Add sidechains using Rosetta fixbb (repack)
    d. Command:

    run_add_sc_repack.sh

10. Copy the best scoring sidechain models the final subfolder
11. Prepare and submit the final models
    a. Adjust the file format to the CASP submission format
    b. Submit the models to the CASP website

Calculate the statistics used for the Loop Angle and Sheet Alignment scores

1. Create lists of pdb files to evaluate
    a. Create list files containing the input pdbs for the CASP10 BCL models, for the CASP natives, and for the PISCES set (one pdb file per line with full path).

b. The CASP10 BCL model list was created from the pdb lists and tables with the following commands:

```
cat *_fold.ls > caspmodels-fold18.ls
cat *_fold_filtered.table | grep -v "bcl::storage::Table" | awk '{print $1}' \
  > caspmodels-fold-filtered18.ls.tmp
cat caspmodels-fold-filtered18.ls.tmp | awk '{system("grep "substr($0, 45, 23)" \
  caspmodels-fold18.ls")}' > caspmodels-fold-filtered18.ls
```

c. The CASP natives list was created manually.

d. The PISCES file list was copied, and then had folders added:

```
cp PISCES/2010_06_22/soluble/current_soluble_5.ls .
cat current_soluble_5.ls | \
  awk '{ print"PISCES/data/"tolower(substr($0,2,2))"/" tolower(substr($0, 1, 4)) \
  substr($0,5,1)".pdb"'} > pisces_20100622_soluble_5.ls
```

2. Loop Distance and Loop Angle statistics

a. Creates the loop angle statistic from the input list.

b. Command:

```
bcl.exe bioutil:ProteinStatistics -pdb_list input.ls -loop_angle
```

3. Sheet Alignment statistics

a. Creates the Sheet Alignment from the input list.

b. Command:

```
bcl.exe bioutil:ProteinStatistics -pdb_list input.ls -strand_alignment
```