New computational and data-driven methods for protein homology modeling

by

Sutanu Bhattacharya

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
August 7, 2021

Keywords: dynamic programming, protein threading, protein homology, residue-residue interaction maps, distance maps, inter-residue orientation

Approved by

Debswapna Bhattacharya, Chair, Assistant Professor of Computer Science and Software Engineering
James H. Cross II, Professor of Computer Science and Software Engineering
Yang Zhou, Assistant Professor of Computer Science and Software Engineering
Scott R. Santos, Professor of Department of Biological Sciences
Robert J. Pantazes, Assistant Professor of Chemical Engineering Department

Abstract

Threading a query protein sequence onto a library of weakly homologous structural templates remains challenging. In light of the recent advancements in residue-residue contact prediction technologies powered by sequence co-evolution and deep learning, we propose a new contact-assisted threading method. In particular, the method integrates the residue-residue contact information with various sequential and structural features to improve the threading scoring function for the better template selection. A large-scale benchmarking result on 500 targets demonstrates that our contact-assisted threading method attains a statistically significantly better threading performance than a baseline contact-free threading acting as a control. Our study further reveals contact-assisted threading using high-quality contacts with the Matthews Correlation Coefficient (MCC) ≥ 0.5 improves the threading performance in nearly 30% of the cases, while low-quality contacts with the MCC <0.35 degrades the performance for 50% of the cases. Moreover, instead of leveraging binary contacts, we move one step further by developing a new distance- and orientation-based covariational threading method called DisCovER by effectively integrating information from inter-residue distances and orientations along with the topological network neighborhood of a query-template alignment. Multiple large-scale benchmarking results on query proteins classified as weakly homologous from the Continuous Automated Model Evaluation (CAMEO) experiment and from the current literature show that our method outperforms several existing state-of-the-art threading approaches. It also shows that the integration of the neighborhood effect with the inter-residue distance and orientation information synergistically contributes to the improved performance of DisCovER.

Acknowledgments

I would like to take this opportunity to thank my supervisor, Dr. Debswapna Bhattacharya, for his invaluable guidance throughout my Ph.D. at Auburn University. His participation with his uniqueness has nurtured my intellectual talent, from which I will benefit for a long time. He has provided unwavering encouragement and support to me in both academic and non-academic settings. His presence in my life has inspired and enriched my growth as a student and a researcher since the first day of my Ph.D. I am heartily thankful to him. My deepest gratitude also goes to all members of my committee for their support and helpful advice. Dr. James H. Cross II has always been my role model, and the numerous interaction I had with him enlightened me in many ways. I am also grateful to Dr. Scott R. Santos for providing me valuable suggestions and tips. I am also thankful to Dr. Yang Zhou for being a member of my committee. I am also thankful to Dr. Robert J. Pantazes for being my University Reader. I would like to convey my special thanks to Dr. Xiao Qin, Director of Computer Science and Software Engineering Graduate Programs, for his kind help and support throughout my Ph.D. study.

Special thanks also goes to every present and past member of Bhattacharya Lab for their support, inspiration, and fruitful collaborations. These include Mr. Rahul Alapati, Mr. Md Hossain Shuvo, Mr. Rahmatuallah Roche, Mr. Andrew J. McGehee, Mr. Muhammad Gulfam, Mr. Bernard Moussad, and all other members of Bhattacharya Lab.

Finally, I am grateful to my family members.

OM Namo Bhagavate Vasudevaya

Table of Contents

List of Tables

List of Figures

List of Abbreviations

PDB	Protein Data Bank
TBM	Template-Based Modeling
CASP	Critical Assessment of Techniques for Protein Structure Prediction
BLAST	Basic Local Alignment Search Tool
CMO	Contact Map Overlap
MSA	Multiple Sequence Alignment
PSI-BLAST	Position-Specific Iterated BLAST
SCOP	Structural Classification of Protein
MCC	Matthews Correlation Coefficient
3D	Three Dimensional
2D	Two Dimensional
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
CAMEO	Continuous Automated Model Evaluation

Chapter 1

Introduction

1.1 Protein structure prediction problem

The computational prediction of the three-dimensional (3D) structure of a protein from its amino acid sequence remains elusive (Baker and Sali 2001, 200; Dill and MacCallum 2012; D. T. Jones, Taylort, and Thornton 1992; Moult et al. 2014). Depending on the availability of relevant proteins in the Protein Data Bank (PDB) (Berman et al. 2000), protein structure prediction methods are broadly classified in two categories: (1) *ab initio* folding, and (2) template-based modeling (TBM). Methods based on *ab initio* folding predict protein 3D structures from the scratch by using the sequence information alone. Despite the encouraging recent progress in *ab initio* protein structure prediction (S. Wang et al. 2016; J. Yang et al. 2020; Greener, Kandathil, and Jones 2019; Adhikari et al. 2015; Adhikari and Cheng 2018, 2; Roche, Bhattacharya, and Bhattacharya 2021; Jinbo Xu 2019), template-based modeling (TBM) remains one of the most reliable approaches in protein structure prediction, especially when homologous templates are available in the PDB. TBM approaches can be broadly classified into homology modeling and protein threading based on the degree of homology. Homology modeling or comparative modeling is the process of building a structure of a query protein from a homologous template with a high degree of sequence similarity (Moult 1996). The success of homology modeling critically depends on the ability to identify the homologous template and generate accurate query-template alignment. Intuitively, the performance of these methods sharply deteriorates when the direct evolutionary relationship between the query and templates becomes very low, the so-called remote-homology modeling scenarios (Bowie, Luthy, and Eisenberg 1991; Petrey and Honig 2005). Protein threading, an

advanced homology modeling technique, aims to address the challenge by leveraging multiple sources of information by mining the evolutionary profile of the query and templates to reveal the potential distant homology and perform low-homology modeling to predict the 3D structure of the query protein.

1.2 Overview of protein threading

The general principle behind protein threading is that there exists a finite number of unique folds in nature, and many proteins (~90% (Jinbo Xu et al. 2003)) share the same folds (Kinch and Grishin 2002; Y. Zhang and Skolnick 2005a), even though their sequences differ, illustrating that, in theory, the structure of most proteins can be successfully predicted by threading a query protein sequence onto a library of structural templates (Jinbo Xu et al. 2003). Therefore, the goal of protein threading is to optimally align a query sequence to a known structural template (Bienkowska and Lathrop 2005). This requires identifying the correct or best-fit template from a library of templates and the optimal query-template alignment from the space of all possible query-template alignments. The query-template alignment represents a correspondence between each query residue and the spatial positioning of the aligned template residues. Overall, protein threading can be mainly considered to involve three components: (1) a threading scoring function that evaluates the fitness of query-template alignments; (2) identification of the best-fit structural template from the library of templates; and (3) an optimal alignment of the query sequence to the template. In the following, we discuss each component in more detail:

1.2.1 Threading scoring function

The scoring function plays an important role to quantitatively assess the fitness of query-template alignments (Jinbo Xu et al. 2003). The scoring function normally consists of the profile similarity score, the structural consistency score, and the gap penalty. The profile similarity score

can be calculated by comparing the query and template profiles. It quantifies how the query is evolutionary related to the template. The structural consistency score contains two components: consistency of local structures such as secondary structure and solvent accessibility compatibility, and consistency of global structures or pairwise inter-atomic interactions. Weights can be used in the scoring function to control the relative importance of different scoring terms.

1.2.2 Template Selection

Identifying the best-fit template inevitably requires using the alignment score of query-template alignments. The raw query-template alignment score cannot be directly used to rank templates due to the biases introduced by the protein length (Jinbo Xu et al. 2003). Both machine learning-based methods and Z-score are used to mitigate the bias. Several protein threading methods (David T. Jones 1999; Y. Xu and Xu 2000; Y. Xu, Xu, and Uberbacher 1998; Akutsu and Miyano 1999; Zhu et al. 2018) use machine learning models such as the neural network for the template ranking by formulating the template selection as a classification problem, even though a majority of the threading methods (S. Wu and Zhang 2008; Zheng, Zhang, et al. 2019; Du et al. 2020) rely on the Z-score for the template selection. Z-scores of the query-template pair are computed from the means and standard deviations of the scores of the query sequence with all the templates of the template library. However, it cannot cancel out all the biases introduced by protein length. A large protein appears to have a high Z-score. It is also difficult to interpret the Z-score, particularly when the scoring function is the weighted sum of different scoring terms (Jinbo Xu et al. 2003).

1.2.3 Optimal query-template alignment

The optimal query-template alignment is the alignment that optimally aligns residues in the query sequence homologous to residues in the template. While a threading scoring function

can be effective in selecting the homologous template, the query-template alignment may be suboptimal (Petrey and Honig 2005; Venclovas 2003; Du et al. 2020; Zheng, Zhang, et al. 2019; Zhu et al. 2018), which might result in less accurate template-based models built from such an alignment. That is, the sensitivity of the query-template alignment directly affects the overall performance of template-based modeling.

1.2.4 Dynamic programming

Dynamic programming, one of the most popular pairwise alignment methods, is used to find the optimal query-template (global or local) alignment. The Needleman-Wunsch algorithm (Needleman and Wunsch 1970) is used to produce the optimal global alignment, whereas the Smith-Waterman algorithm (Smith and Waterman 1981) is used for local alignments. In particular, the Needleman-Wunsch algorithm, developed in 1970, is a widely used nonlinear global optimization method. It divides a large task (e.g. alignment between the full query and the full template sequence) into a set of smaller sub-tasks and uses the solution to the sub-tasks to find the optimal solution to the larger task. The worst-case time complexity is $O(L_q, L_t)$, where L_q and L_t are the length of the query and the template protein, respectively. However, obtaining the global alignments might not useful for distantly related sequences because of the noise added by low similarity regions. Local alignments are useful to focus more on conserved motifs, where the Smith-Waterman algorithm, developed in 1981, guarantees to find the optimal local query-template alignment. One of the main differences to the Needleman-Wunsch algorithm is the traceback procedure. It also has the worst-case time complexity of $O(L_q, L_t)$, where L_q and L_t are defined earlier. Both of these algorithms are guaranteed to find an optimal alignment using a scoring function, however, a robust scoring function plays a vital role in the threading performance, as discussed in Section 1.2.1.

1.3 Protein threading assisted by inter-residue interaction maps

1.3.1 Granularities of protein inter-residue interaction maps

Protein inter-residue interaction maps are predicted at various resolutions ranging from binary contact maps to finer-grained distance and orientation maps as well as their combination. A low-resolution version of inter-residue interaction is a contact map, which is a square, symmetric matrix with binary entries, where a contact indicates the spatial proximity of a residue pair at a given cutoff distance, typically set to 8Å between the Cα or Cβ carbons of the interacting residue pairs. Here, the set of contacts between residue pair (i, j) is defined as:

$$C(i,j) = \begin{cases} 1 & if \;\; d_{ij} \leq 8\text{Å} \\ 0 & otherwise \end{cases}$$

where d_{ij} is the distance between the residue pair (i, j).

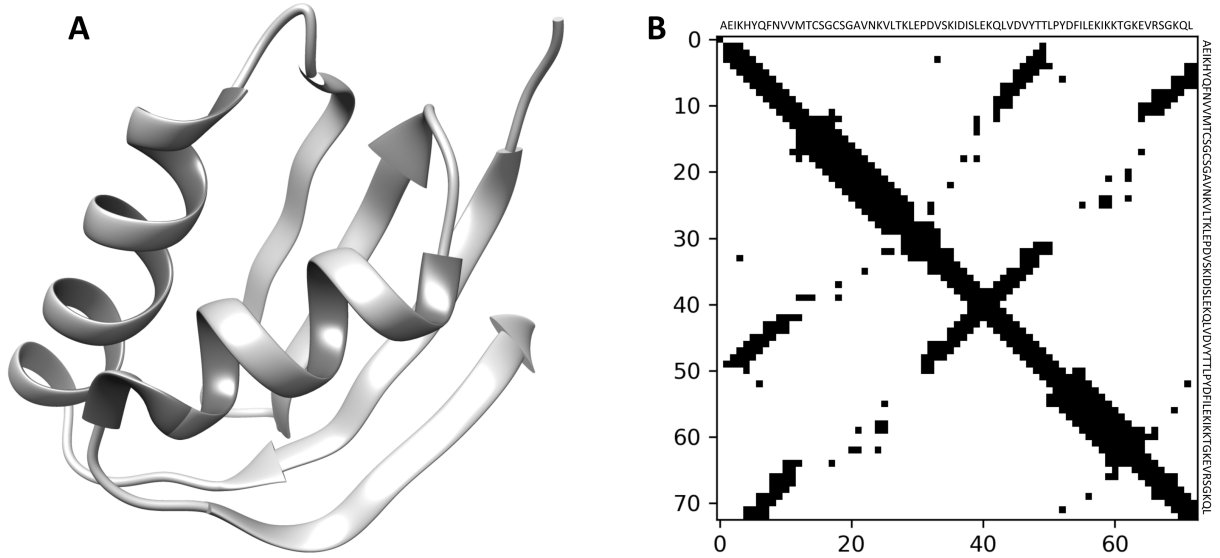


Figure 1.1 A representative protein 3D structure and its corresponding 2D binary contact map. (A) 3D structure of a representative protein (PDB ID 1cc8A), (B) the corresponding 2D residue-residue contact map, considering C_α atoms and a distance threshold of 8Å.

Figure 1.1 shows a representative protein 3D structure and its corresponding 2D residue-residue contact map. The inter-residue distance map is finer-grained in that it captures the distribution of real-valued inter-residue spatial proximity information rather than the binary

contacts at a fixed cutoff distance. Recent studies (Jinbo Xu and Wang 2019; Jinbo Xu 2019) have demonstrated the advantage of using distance maps in protein structure prediction over binary contacts as distances carry more physical constraints information of protein structures than contacts. The granularities of predicted distance maps vary from distance histograms to real-valued distances (Greener, Kandathil, and Jones 2019; J. Yang et al. 2020; Ding and Gong 2020; Adhikari 2020; T. Wu et al. 2021; J. Li and Xu 2020). Very recently, trRosetta(J. Yang et al. 2020) has introduced inter-residue orientations in addition to distances to capture not only the spatial proximity information of the interacting pairs but also their relative angles and dihedrals. Collectively, inter-residue distances and orientations encapsulate the spatial positioning of the interacting pairs much better than only distances let alone binary contacts.

1.3.2 Contact map alignment

Contact map alignment is a way of measuring the similarity between two contact maps. The maximum contact map overlap problem tries to evaluate the similarity of the two proteins by calculating the maximum overlap between their contact maps while preserving the ordering of the residues of both sequences, leading to a pairwise sequence alignment as illustrated in **Figure 1.2**. Since the direct contact map alignment is computationally expensive (Zheng, Wuyun, et al. 2019), several approximation algorithms (Yosi Shibberu, Holder, and Lutz 2010; Y. Shibberu and Holder 2011; Teichert, Bastolla, and Porto 2007; Di Lena et al. 2010; Teichert et al. 2010; Malod-Dognin and Pržulj 2014; Ovchinnikov et al. 2017) have been developed to address the contact map alignment problem including the eigen-decomposition-based strategy, the graphlet degree-based approach, and the iterative double dynamic programming-based approach. The eigen-decomposition decomposes a contact map into eigenvectors and corresponding eigenvalues. This approach compares two proteins by comparing their contact map eigenvectors, which can be

performed in polynomial time. For example, approaches such as EIGAs (Yosi Shibberu, Holder, and Lutz 2010), SABERTOOTH (Teichert, Bastolla, and Porto 2007), and Al-Eigen (Di Lena et al. 2010) use the eigen-decomposition to approximate contact maps using the top eigenvectors and use the global alignment of key eigenvectors to find the similarity between two contact maps. GR-Align (Malod-Dognin and Pržulj 2014) is a fast contact map alignment approach based on the graphlet degree distribution. Moreover, (Skolnick and Zhou 2017) proposes a contact map alignment algorithm C-Align based on C_α atoms using a dynamic programming. Recent methods such as map_align (Ovchinnikov et al. 2017) employ an iterative double dynamic programming to calculate contact map alignments, with the goal of optimizing the number of contact overlaps while minimizing the number of gaps.

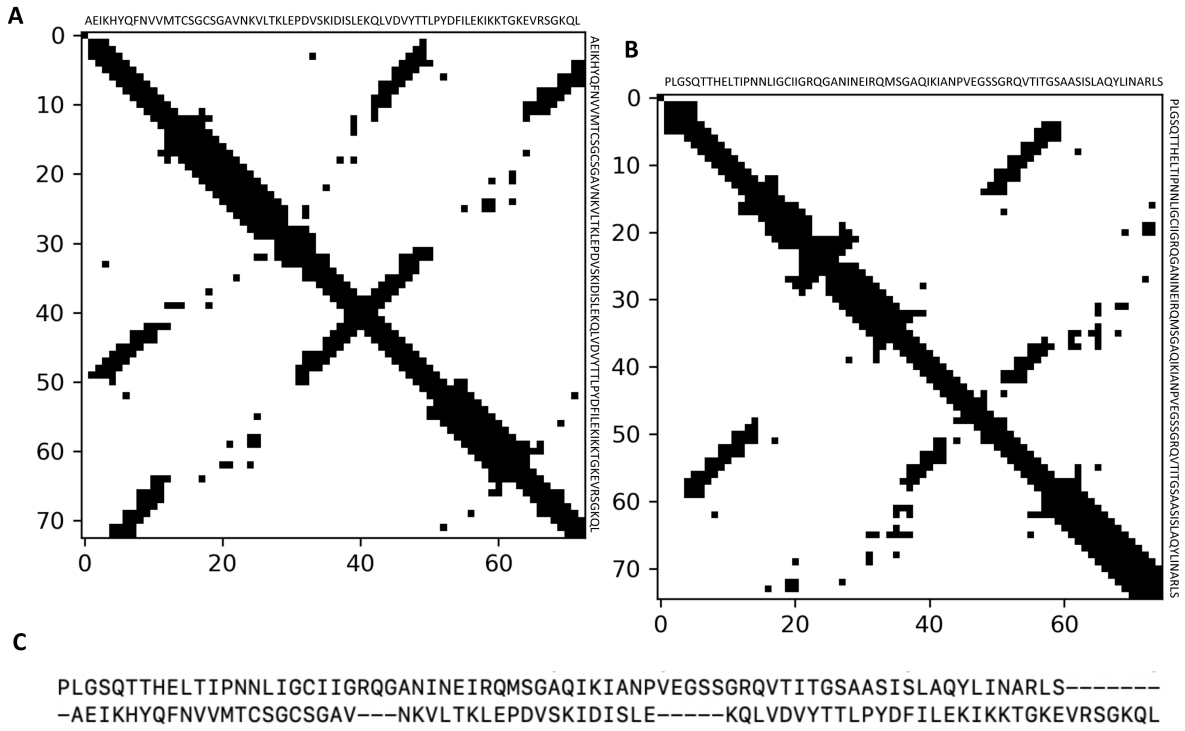


Figure 1.2 Contact map alignment. (A) contact map of a representative protein (PDB ID 1cc8A), (B) contact map of another representative protein (PDB ID 1wvnA), (C) sequence alignment of 1cc8A and 1wvnA using Al-Eigen. In both cases, C_α atoms and the distance threshold of 8Å are considered.

1.3.3 Inter-residue interaction map threading

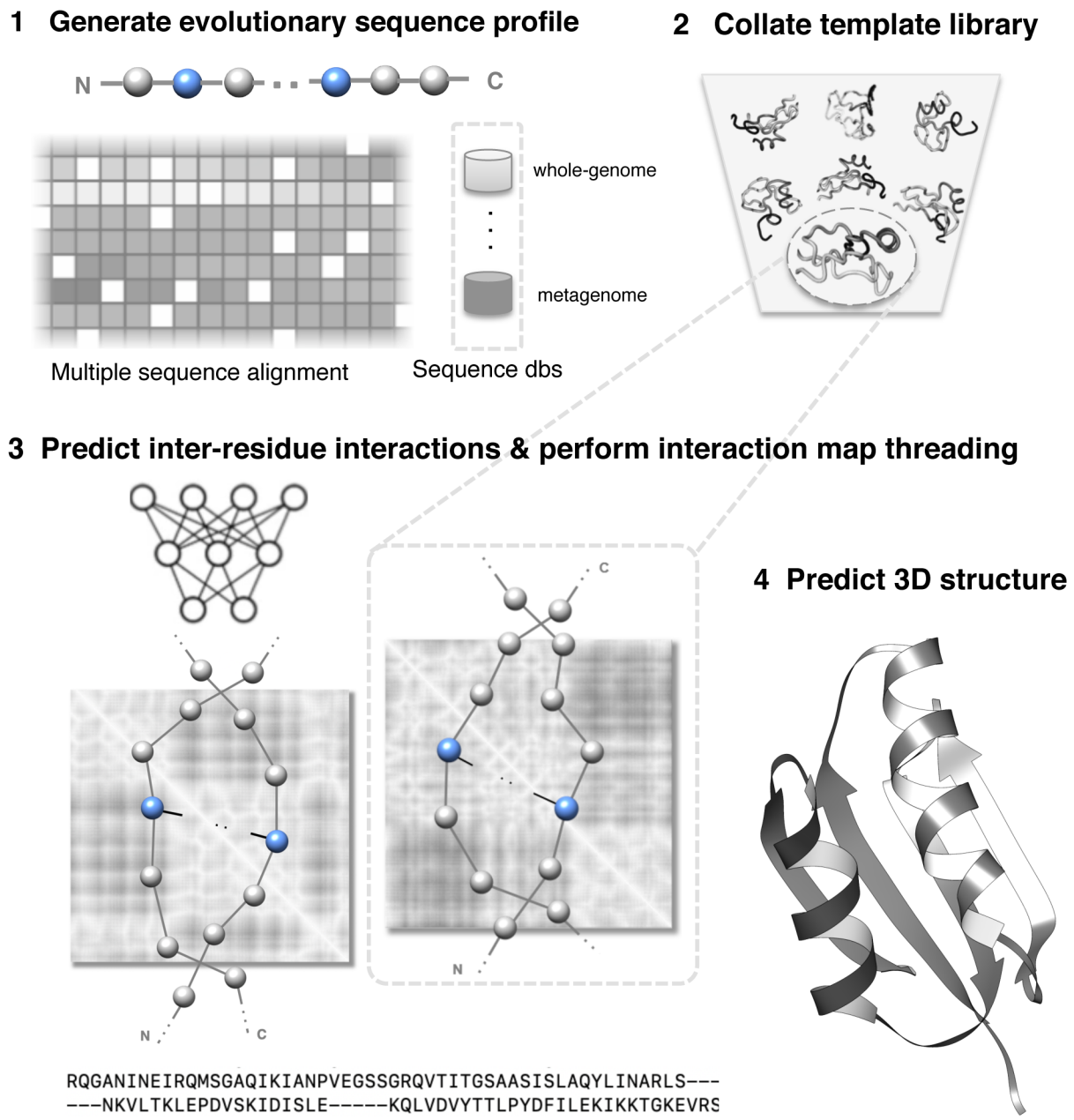


Figure 1.3 Illustration of protein interaction map threading.

Figure 1.3 shows an overview of an interaction map threading of a query protein. Generally, threading has three components: (1) an effective scoring function to evaluate the fitness of query-template alignments; (2) an efficient templates searching strategy; and (3) generating optimal query-template alignments (Bhattacharya et al. 2021). One of the most important components of threading approaches is the scoring function, which is composed of standard threading features ranging from sequential features such as secondary structures, solvent accessibility, and sequence profiles to non-linear features such as pairwise potentials (Brylinski and Skolnick 2010; Bienkowska and Lathrop 2005). Weights control the relative importance of different terms. An efficient scoring function should reliably differentiate a homologous template from the alternatives because the accuracy of the predicted model significantly depends on the evolutionary relatedness of the identified template. The inter-residue interaction map helps to improve the sensitivity of the threading scoring function by augmenting the standard scoring terms with additional contributions from the predicted interactions. Specifically, the score to align the ith residue of the query protein to the jth residue of the template can be defined as:

$$E(i,j) = w_1 E_{map}^{interaction}(i,j) + \sum_{\substack{k \in standard \\ threading\ features}} w_k E_k^{feature}(i,j)$$

where the first term accounts for the contribution of the interaction map and the second term accounts for the standard threading features with w_i being their relative weights. Typically, the similarity between the predicted inter-residue interaction map of the query protein and that derived from the template structure informs the interaction map term in the threading scoring function. It is worth noting here that the raw alignment score is biased to the protein length (Jinbo Xu et al. 2003). As such, most threading methods use a normalized alignment score in standard deviation units relative to the mean score of all templates in the template library.

1.4 Overview of existing threading methods

Current threading strategies are based on various techniques ranging from dynamic programming to profile-profile comparison based on hidden Markov models to more advanced machine learning approaches (S. Wu and Zhang 2008; Lobley, Sadowski, and Jones 2009; Y. Yang et al. 2011; Ma et al. 2012; 2013; Peng and Xu 2010; Söding 2005; Peng and Xu 2009; Jaroszewski et al. 2005; Rychlewski et al. 2000; Cheng and Baldi 2006; Marti-Renom, Madhusudhan, and Sali 2004; Ginalski et al. 2003; Zhou and Zhou 2005; David T. Jones 1999; S. Wu and Zhang 2007; Gniewek et al. 2014; Rost, Schneider, and Sander 1997; Olmea, Rost, and Valencia 1999; Peng and Xu 2011; Y. Xu and Xu 2000; Ma et al. 2014; Yan et al. 2013; Lee and Skolnick 2010). Some of these methods use only sequence-based features, while others (Jinbo Xu et al. 2003; S. Wu and Zhang 2008; 2010; Söding 2005) use sequence and structure-based features for calculating the fitness score between the query and template. Popular profile-based threading methods include MUSTER (S. Wu and Zhang 2008), SEGMER (S. Wu and Zhang 2010), HHsearch (Söding 2005), GenTHREADER (David T. Jones 1999), SPARKS-X (Y. Yang et al. 2011), CNFpred (Ma et al. 2012; 2013), PROSPECT (Y. Xu and Xu 2000), RAPTOR (Jinbo Xu et al. 2003), and MRFalign (Ma et al. 2014). However, there is still room for improvement, particularly with the rapid growth in both sequence and structure databases (Yan et al. 2013).

With the recent progress in protein residue-residue interaction map prediction powered by sequence co-evolution and machine learning (David T. Jones et al. 2015; S. Wang et al. 2017; He et al. 2017; Adhikari, Hou, and Cheng 2018, 2; David T. Jones et al. 2012; Kandathil, Greener, and Jones 2019; Kaján et al. 2014; Seemayer, Gruber, and Söding 2014; Hanson et al. 2018; Greener, Kandathil, and Jones 2019; Jinbo Xu 2019), the contact or distance maps may become a valuable additional structural feature that can assist protein threading. While there exist pure

contact driven *ab initio* folding methods such as CONFOLD (Adhikari et al. 2015), CoinFold (S. Wang et al. 2016), CONFOLD2 (Adhikari and Cheng 2018), and DConStruct (Roche, Bhattacharya, and Bhattacharya 2021) based on distance geometry; cutting-edge threading methods including EigenTHREADER (Buchan and Jones 2017), map_align (Ovchinnikov et al. 2017), CEthreader (Zheng, Wuyun, et al. 2019), CATHER (Du et al. 2020), and DeepThreader (Zhu et al. 2018) are increasingly integrating contact or distance information along with other features to boost threading accuracies. The usefulness of these cutting-edge contact- or distance-assisted threading methods are particularly noteworthy in low-homology protein modeling scenarios (Zheng, Wuyun, et al. 2019; Du et al. 2020; Zheng, Zhang, et al. 2019). Here, we provide an overview of existing contact- or distance-assisted threading methods, highlighting some of the recent advances in low-homology protein modeling. We also discuss some of the current limitations of contact- or distance-assisted threading.

1.5 Existing contact-assisted threading methods

Table 1.1 shows several publicly available contact-assisted threading methods. These approaches can be broadly subdivided into two classes: (1) methods that implicitly use contact information via pairwise-contact potential such as PROSPECT (Y. Xu and Xu 2000), PROSPECTOR (Skolnick and Kihara 2001; Skolnick, Kihara, and Zhang 2004), and RAPTOR (Jinbo Xu et al. 2003); and (2) methods that explicitly use contact information via predicted residue-residue contacts including the current state-of-the-art contact-assisted threading methods such as EigenTHREADER, map_align, CEthreader, CATHER, ThreaderAI (H. Zhang and Shen 2020), and our in-house threading method (Bhattacharya and Bhattacharya 2019a). We briefly discuss them below.

Table 1.1 Selected publicly accessible threading methods that implicitly or explicitly use contact information.

Name (Reference)	Method	Availability
PROSPECT (Y. Xu and Xu 2000)	Divide-and-Conquer algorithm	http://compbio.ornl.gov/structure/prospect/
PROSPECTOR (Skolnick and Kihara 2001; Skolnick, Kihara, and Zhang 2004)	Hierarchical approach	http://bioinformatics.danforthceneter.org/services/threading.html
RAPTOR (Jinbo Xu et al. 2003)	Linear programming	http://www.cs.uwaterloo.ca/~j3xu/RAPTOR_form.htm
EigenTHREADER (Buchan and Jones 2017)	Dynamic programming and eigen-decomposition	http://bioinfadmin.cs.ucl.ac.uk/downloads/eigenTHREADER/
map_align (Ovchinnikov et al. 2017)	Iterative double dynamic programming	https://github.com/sokrypton/map_align
CEthreader (Zheng, Wuyun, et al. 2019)	Dynamic programming and eigen-decomposition	https://zhanglab.ccmb.med.umich.edu/CEthreader/
CATHER (Du et al. 2020)	Iterative double dynamic programming	https://yanglab.nankai.edu.cn/CATHER/
ThreaderAI (H. Zhang and Shen 2020)	Deep residual neural network and dynamic programming	https://github.com/ShenLab/ThreaderAI

1.5.1 Threading methods that implicitly use the contact information via pairwise-contact potentials

PROSPECT (PROtein Structure Prediction and Evaluation Computer Toolkit) is one of the earliest protein threading methods, which makes use of pairwise contact potentials by introducing a contact term into its scoring function. This study considers that pairwise contact potentials are measured only between core secondary structures. The contact cutoff is set at 7Å between the C_β atoms. Additionally, the method uses a Divide-and-Conquer algorithm for the alignment searching procedure. Another method, PROSPECTOR (PROtein Structure Predictor Employing Combined Threading to Optimize Results) uses a "partly thawed" technique to assess the contact potential based on the previous alignment iterations. RAPTOR (RApid Protein Threading by Operation Research technique) (Jinbo Xu et al. 2003) is another protein threading method that introduces the contact capacity score. It considers only contacts between two core residues where the spatial distance between the C_α atoms is 7Å with a sequence separation of 4. It addresses threading as a problem of wide-scale integer programming, relaxes it to a problem of linear programming, and uses a branch-and-bound approach to solve the integer program. However, the performance contribution of pairwise contact potentials in the above methods is not significant compared to that of sequence profiles, particularly for distantly related proteins. The underlying reason may be the noisy contacts that do not hold any extra signal, yielding just a modest improvement.

1.5.2 Threading methods that explicitly use the contact information via predicted residue-residue contacts

Recent successful applications of deep learning have resulted in significantly improved inter-residue contact prediction methods (David T. Jones et al. 2015; S. Wang et al. 2017). As such, the newest contact-assisted threading methods have been explicitly integrating predicted

residue-residue contact information to improve the threading performance. EigenTHREADER, developed in 2017, extends Al-Eigen (Di Lena et al. 2010) to enable threading by predicting a protein's contact map using a classical neural network-based predictor MetaPSICOV (David T. Jones et al. 2015), and then searching a library of templates' contact maps. Despite the superior performance of EigenTHREADER over other profile-based threading methods for low homology threading, it can be further improved by integrating other linear features such as sequence profiles along with inter-residue contact maps. map_align, developed in 2017, proposes an iterative double dynamic programming algorithm (Taylor 1999) that aligns contact maps, predicted by a pure co-evolutionary based predictor GREMLIN (Kamisetty, Ovchinnikov, and Baker 2013), in combination with metagenomics sequences of microbial DNA (Söding 2017). The elevated performance of map_align can be attributed to the contribution of contact maps in low-homology threading. However, considering that the outcomes rely on the initial estimate of the similarity matrix, which is not always the optimal, this approach does not necessarily guarantee optimal solutions. CEthreader (Contact Eigenvector-based threader), developed in 2019, uses contact maps predicted from a deep residual neural-network based predictor ResPRE (Y. Li et al. 2019). Similar to Al-Eigen, this work uses the eigen-decomposition technique to approximate contact maps by the cross product of single body eigenvectors. CEthreader introduces a dot-product scoring function by incorporating the contact information along with secondary structures and sequence profiles to align contact eigenvectors and uses dynamic programming to generate the query-template alignments. However, the method can be further strengthened by considering negative eigenvalues in addition to positive eigenvalues, since the incorporation of both positive and negative eigenvalues restores the contact map. Another new contact-assisted threading algorithm CATHER (Du et al. 2020) (Contact Assisted THreadER), developed in 2020, uses both

conventional sequential profiles and contact maps predicted by a deep learning-based method MapPred (Q. Wu et al. 2020). A very recent method, ThreaderAI (H. Zhang and Shen 2020) integrates the deep learning-based contact information with traditional sequential and structural features by formulating the task of threading as the classical computer vision's classification problem. This work introduces a deep residual neural network to predict query-template alignments. Based on the reported results of the above methods, contact-assisted threading methods significantly outperform profile-based threading methods by a large margin, particularly for low-homology targets.

Our in-house threading method (Bhattacharya and Bhattacharya 2019a), developed in 2019, integrates the standard threading technique along with the inter-residue contact information predicted by the state-of-the-art ultra-deep learning-based method RaptorX (S. Wang et al. 2017). First, our method applies the standard threading technique to select the top templates based on the Z-score and then applies the contact map overlap score using AlEigen along with the Z-score to calculate the final score for selecting the best-fit template. Based on large-scale benchmarking results, this method outperforms the profile-based threading method MUSTER as well as other contact-assisted threading methods EigenTHREADER and map_align.

1.6 Existing distance-assisted threading methods

Building on the successes of contact-assisted threading methods, Xu and coworkers developed a distance-based threading method called DeepThreader (Zhu et al. 2018). The method predicts distance maps by employing deep learning, and then incorporates the predicted inter-residue distance information along with sequential features into threading through Alternating Direction Method of Multipliers (ADMM) algorithm. The inter-residue distance is binned into 12 bins: <5Å, 5-6Å, .., 14-15Å, >15Å. Based on their reported results as well as the performance

evaluation in the 13th Critical Assessment of protein Structure Prediction (CASP13), incorporating distance information boosts the threading performance, particularly for low homology targets, outperforming contact-assisted threading methods by a large margin (Jinbo Xu and Wang 2019). Zhang and coworkers have recently extended CEthreader to develop a distance-assisted threading method DEthreader introduced during the recently concluded CASP14 experiment by incorporating distance-based scoring term into the scoring function. The method uses the C_α-C_α and C_β-C_β distance distribution, both are binned into 38 bins: 1 bin of <2Å, 36 bins of 2-20Å with a width of 0.5Å, and 1 bin of ≥20Å. Similarly, Yang and coworkers have extended CATHER into a distance-based threading approach by replacing contacts with distances in CASP14.

To further improve the threading performance, NDthreader (F. Wu and Xu 2021) (New Deep-learning Threader) and ProALIGN (Kong et al. 2021) move one step further by integrating deep learning to optimally predict query-template alignments using distance potentials. Their reported state-of-the-art performance demonstrate the effectiveness of deep learning in distant homology protein modeling.

1.7 The role of sequence databases in interaction map threading

The prediction of inter-residue interaction maps depends heavily on the availability of homologous sequences. As such, the role of the sequence databases is becoming increasingly important in protein homology detection via interaction map threading. In addition to the well-established whole-genome sequence databases such as the nr database from the National Center for Biotechnology Information (NCBI), UniRef (Suzek et al. 2015), UniProt (The UniProt Consortium 2019), and Uniclust (Mirdita et al. 2017); emerging metagenome sequence databases from the European Bioinformatics Institute (EBI) Metagenomics (Mitchell et al. 2018; Markowitz et al. 2014) and Metaclust (Steinegger and Söding 2018) are playing a prominent role. For

example, (Y. Wang et al. 2019) have demonstrated the applications of marine metagenomics for the improved protein structure prediction. map_align uses the Integrated Microbial Genomes (IMG) database (Markowitz et al. 2014), containing around 4 million unique protein sequences, to reliably predict high-quality models for low-homology Pfam families of unknown structures. Another recent method for generating protein multiple sequence alignments, DeepMSA (C. Zhang et al. 2020), combines whole-genome and metagenome sequence databases and reports improved threading performance, particularly for distant-homology proteins. Newer sequence databases are getting larger and diverse. For example, BFD (Steinegger, Mirdita, and Söding 2019), a recent sequence database, is one of the largest sequence databases containing 2 billion protein sequences from soil samples and 292 million sequences of marine samples. Another very recent sequence database MGnify (Mitchell et al. 2020) contains around 1 billion non-redundant protein sequences. As such, the availability of the evolutionary information of low-homology proteins is getting enriched, likely leading to the improved prediction accuracy of inter-residue interaction maps and hence more accurate interaction map threading for low-homology protein modeling.

1.8 Structural similarity measure

Measuring the structural similarity between the predicted and the native protein 3D structure is of utmost importance for evaluating the quality of the predicted 3D model of the query protein. The most commonly used scores are the template modeling score (TM-score) (Y. Zhang and Skolnick 2004), the root mean square deviation (RMSD) (W. Kabsch 1976), the global distance test (GDT) (Zemla 2003), and the local distance difference test (lDDT) (Mariani et al. 2013). Here, we discuss the widely used scoring metric TM-score. It is calculated by:

$$TM-score = \frac{1}{L}\sum_{i=1}^{L_{ali}} \frac{1}{1+\frac{d_i^2}{\left(1.24\sqrt[3]{L-15}-1.8\right)^2}}$$

where d_i is the distance between the ith residues of the query and the template after an optimal superimposition, L is the length of the query sequence, and L_{ali} is the length of the aligned regions. TM-score measures the similarity between two protein 3D structures and gives a score in the range (0,1], where a higher score means a better similarity. TM-score > 0.5 indicates the pair of proteins share the same fold, whereas a score < 0.17 indicates a random fold (Jinrui Xu and Zhang 2010).

1.9 Dissertation outline and contributions

The contents of Chapter 1 are mostly from the following manuscripts:

> *S. Bhattacharya, R. Roche, MH. Shuvo, and D. Bhattacharya, "Recent advances in protein homology detection propelled by inter-residue interaction map threading", **Frontiers in Molecular Biosciences**, 8, 377, 2021. https://doi.org/10.3389/fmolb.2021.643752*
>
> *S. Bhattacharya, R. Roche, MH. Shuvo, and D. Bhattacharya, "Contact-assisted threading in low-homology protein modeling", Methods in Molecular Biology by **Springer Nature**, 2021 (Under revision).*

The remainder of this dissertation is structured as follows. In Chapter 2, we analyze whether the addition of residue-residue contact information helps increase the accuracy of protein threading by proposing a new threading approach that integrates sequence and structural information with residue-residue contacts in order to examine how much more accuracy gain can be obtained by incorporating contact information than a pure threading-based approach (Bhattacharya and Bhattacharya 2019a). The contents of Chapter 2 are mostly from the manuscript published as:

> *S. Bhattacharya and D. Bhattacharya, "Does inclusion of residue-residue contact information boost protein threading?," **Proteins: Structure, Function, and Bioinformatics**, vol. 87, no. 7, pp. 596–606, 2019, doi: 10.1002/prot.25684.*

In Chapter 3, we explore the significance of diverse quality of contact maps in contact-assisted threading performance by seamlessly customizing our new method to perform contact-assisted or contact-free threading modes (Bhattacharya and Bhattacharya 2020). The contents of Chapter 3 are mostly from the manuscript published as:

*S. Bhattacharya and D. Bhattacharya, "Evaluating the significance of contact maps in low-homology protein modeling using contact-assisted threading," **Scientific Reports**, vol. 10, no. 1, Art. no. 1, Feb. 2020, doi: 10.1038/s41598-020-59834-2.*

In Chapter 4, we present DisCovER (Bhattacharya, Roche, and Bhattacharya 2020), a new distance- and orientation-based threading method that effectively integrates information from inter-residue distances and orientations along with the topological network neighborhood of a query-template alignment. It demonstrates the integration of the neighborhood effect with the inter-residue distances and orientations information synergistically contributes to the improved threading performance. The contents of Chapter 4 are mostly from the manuscript:

*S. Bhattacharya, R. Roche, and D. Bhattacharya, "DisCovER: distance- and orientation-based covariational threading for weakly homologous proteins", **bioRxiv**, 2020 (Under revision). https://doi.org/10.1101/2020.01.31.923409*

The final chapter, Chapter 5, outlines the conclusion.

Moreover, throughout this work, TM-score (Y. Zhang and Skolnick 2004) is used to evaluate the performance of competing threading methods. A TM-score of more than 0.5 is considered as the correct fold against the native structure. It is worth highlighting a higher TM-score illustrates a better topological similarity between a pair of protein structures.

Chapter 2

Does Inclusion of Residue-Residue Contact Information Boost Protein Threading?

2.1 Abstract

Template-based modeling is considered as one of the most successful approaches for protein structure prediction. However, reliably and accurately selecting the optimal template proteins from a library of known protein structures having similar folds as the target protein and making correct alignments between the target sequence and the template structures, a template-based modeling technique known as threading, remains challenging, particularly for non- or distantly-homologous protein targets. With the recent advancement in protein residue-residue contact map prediction powered by sequence co-evolution and machine learning, here we systematically analyze the effect of inclusion of residue-residue contact information in improving the accuracy and the reliability of protein threading. We develop a new threading algorithm by incorporating various sequential and structural features, and subsequently integrate residue-residue contact information as an additional scoring term for the threading template selection. We show that the inclusion of the contact information attains statistically significantly better threading performance compared to a baseline threading algorithm that does not utilize the contact information when everything else remains the same. Experimental results demonstrate that our contact-based threading approach outperforms the popular threading method MUSTER, the contact-assisted ab initio folding method CONFOLD2, and recent state-of-the-art contact-assisted protein threading methods EigenTHREADER and map_align on several benchmarks. Our study illustrates that the inclusion of contact maps is a promising avenue in protein threading to ultimately help improve the accuracy of protein structure prediction.

2.2 Introduction

With the recent progress in protein residue-residue contact prediction powered by sequence co-evolution and machine learning (David T. Jones et al. 2015; S. Wang et al. 2017; Adhikari, Hou, and Cheng 2018; David T. Jones et al. 2012; Kaján et al. 2014), contact maps may become a valuable additional structural feature that can assist protein threading. Cutting-edge threading methods, including EigenTHREADER (Buchan and Jones 2017), map_align (Ovchinnikov et al. 2017), and DeepThreader (Zhu et al. 2018), are increasingly integrating the contact or distance information along with other features to boost threading accuracies. For example, EigenTHREADER integrates predicted contact maps from MetaPSICOV (David T. Jones et al. 2015) and the sequential information whereas map_align uses a pure contact driven threading approach by maximizing the overlap of the co-evolutionary predicted contact map of the target protein to the template's true contact map. Very recently, DeepThreader proposes to select top templates by combining sequential features and predicted inter-residue distances for generating a query-template alignment.

Here, we analyze whether the addition of the residue-residue contact information helps increase the accuracy of protein threading. We develop a new threading approach that integrates the sequence and the structural information with residue-residue contacts in order to examine how much accuracy gain can be obtained by incorporating the contact information. First, we investigate whether incorporating the contact information into a threading-based approach helps in predicting the top ranked models with better accuracy than a pure threading-based approach. We further explore whether the residue-residue contact information along with a threading-based approach is more promising than purely contact driven *ab initio* folding methods. Finally, we compare the performance of our work with the state-of-the-art contact-assisted protein structure prediction

methods in a blind manner using protein targets from the recently concluded 13th Critical Assessment of protein Structure Prediction (CASP13) experiment.

2.3 Materials and methods

2.3.1 Alignment scoring function for threading

We develop a threading approach (**Figure 2.1**), which uses sequential and structural features to search a library of templates. Specifically, we use structure profiles, native secondary structures, native solvent accessibility, and native torsion angles as features for the template. For the query protein, sequence profiles, predicted secondary structures, predicted solvent accessibility, and predicted torsion angles are used as features. We use DSSP (Wolfgang Kabsch and Sander 1983) for extracting native secondary structures, native solvent accessibility, and native torsion angles for each template whereas SPIDER3 (Heffernan et al. 2017) is used to predict the corresponding features from the sequence of the query protein.

The alignment score for aligning the query position i with the template position j is:

$$\begin{aligned} Score(i,j) &= S_{seq\_prof} + S_{ss} + S_{struc\_prof} + S_{sa} + S_{psi} + S_{phi} + S_{hydro} + S_{shift} \qquad (1) \\ &= \sum_{k=1}^{20} \frac{\left(Pc_q(i,k) + Pd_q(i,k)\right) L_t(j,k)}{2} + w_1 \delta\left(s_q(i), s_t(j)\right) \\ &+ w_2 \sum_{k=1}^{20} Ps_t(j,k) L_q(i,k) + w_3\left(1 - 2\left|SA_q(i) - SA_t(j)\right|\right) \\ &+ w_4\left(1 - 2\left|\varphi_q(i) - \varphi_t(j)\right|\right) + w_5\left(1 - 2\left|\Phi_q(i) - \Phi_t(j)\right|\right) \\ &+ w_6 M\left(AA_q(i), AA_t(j)\right) + w_7 \end{aligned}$$

where "q" stands for the query and "t" stands for the template protein. The first term in Equation (1) is the sequence profile-profile alignment between the query and the template. The frequency of the kth residue at the ith position of the multiple sequence alignment (MSA), $Pc_q(i,k)$ and $Pd_q(i,k)$, are obtained by two iterations of PSIBLAST (Altschul et al. 1997) search against a non-redundant (nr) sequence database with an e-value cutoff of 0.001 and 1.0 to get "close" and

"distant" homologues, respectively. The sequence-derived log-odds profile of the template, $L_t(j,k)$, is obtained by PSIBLAST with an e-value cutoff of 0.001. The second term in Equation (1) compares the predicted secondary structure with the native secondary structure. The term $\delta(s_q(i), s_t(j))$ is 1 if $s_q(i) = s_t(j)$ and -1 otherwise. Both $s_q(i)$ and $s_t(j)$ have three distinct states: helix, strand, and coil. The third term S_{struc_prof} is a structure-derived profile where $Ps_t(j,k)$ denotes the kth amino acid's frequency at the jth position of the template, $L_q(i,k)$ denotes the log-odds profile of the query, and both are obtained by PSIBLAST search with an e-value cutoff of 0.001. The fourth term S_{sa} computes the difference between the predicted solvent accessibility and the native solvent accessibility, where $SA_q(i)$ is for the ith residue of the query and $SA_t(j)$ is for the jth residue of the template. The next two terms (fifth and sixth) account for the match between the predicted torsion angles of the query ($\varphi_q(i)$ and $\Phi_q(i)$ of the ith position of the query) and the native torsion angles of the template ($\varphi_t(j)$ and $\Phi_t(j)$ of the jth position of the template). Both psi and phi angles are normalized by 360^0. The seventh term S_{hydro} matches the hydrophobic residues (V, I, L, F, Y, W, M) of the query and the template. $M(AA_q(i), AA_t(j))$ = 1 if the ith position of the query ($AA_q(i)$) and the jth position of the template ($AA_t(j)$) are both hydrophobic; $M(AA_q(i), AA_t(j))$ = 0.7 if $AA_q(i)$ and $AA_t(j)$ are identical; otherwise, $M(AA_q(i), AA_t(j))$= 0. Finally, Needleman-Wunsch (Needleman and Wunsch 1970) dynamic programming algorithm is used to get the best query-template alignment. A position-specific gap penalty is also employed. The last term of Equation (1) is a constant, w_7, which is used to discourage the alignment of unrelated residues. Seven weight parameters and two gap penalty parameters (gap opening g_o and gap extension g_e) are used (S. Wu and Zhang 2008): w_1 = 0.66, w_2 = 0.39, w_3 = 1.60, w_4 = 0.19, w_5 = 0.19, w_6 = 0.31, w_7 = 0.99, g_o = 7.01, and g_e = 0.55.

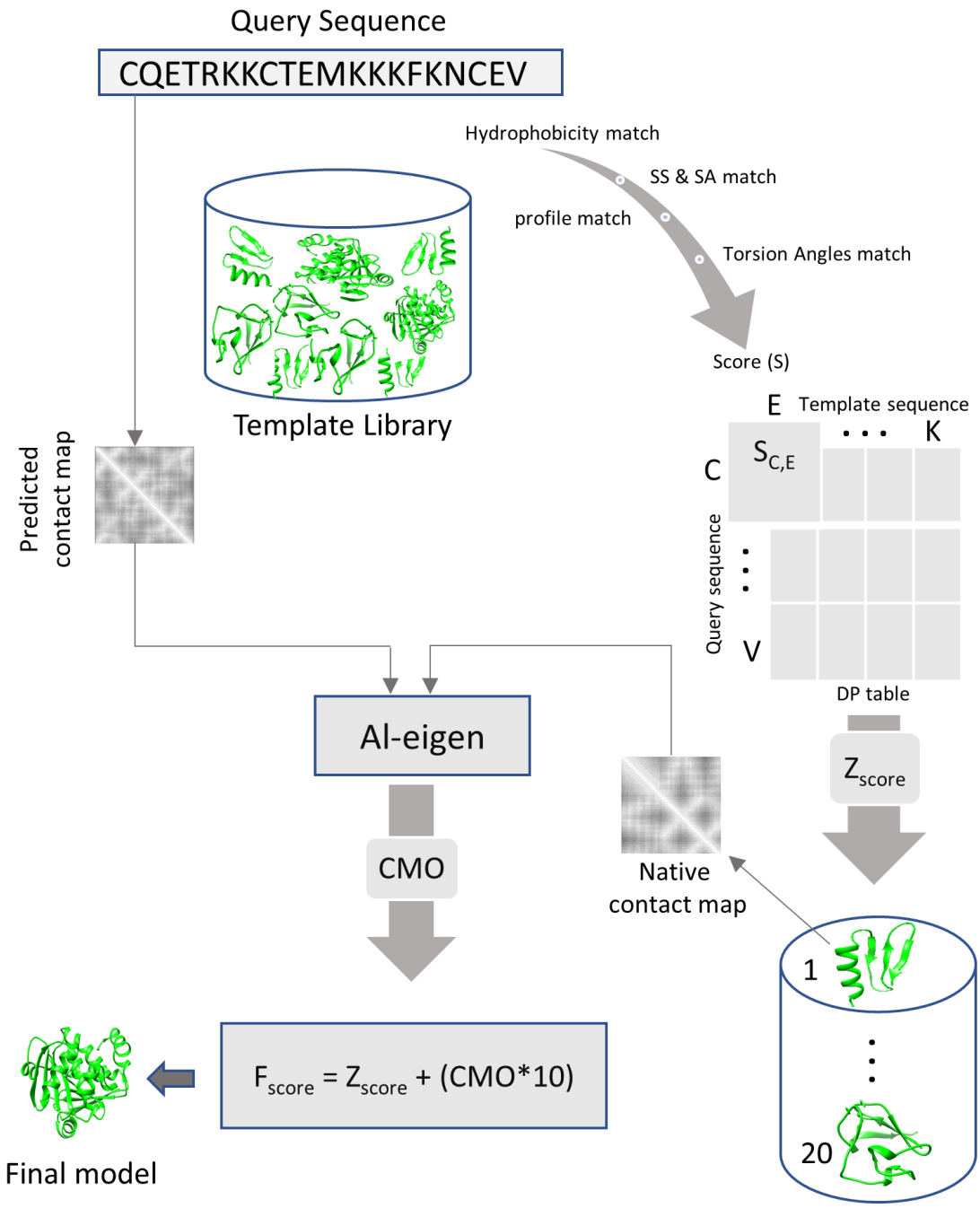


Figure 2.1 Flowchart of our work. The query sequence is threaded against the template library by calculating the similarity score using sequential and structural features. Subsequently, contact map overlap (CMO) score between the contact map of each template and the predicted contact map of the query is calculated, and integrated to the threading-based similarity score in a weighted manner to select the best-fit template for predicting the 3D structure of the query.

Initially, the templates are ranked by the following Z_{score}

$$Z_{score} = \frac{(R'_{score} - < R'_{score} >)}{\sqrt{< R'^2_{score} > - < R'_{score} >^2}} \quad (2)$$

where R'_{score} is the score normalized by the greater one of the raw alignment score with L (full alignment length) and L' (partial alignment length), and $< \cdots >$ denotes the average of all templates in the library.

2.3.2 Inclusion of residue-residue contact information

A residue-residue contact map is a binary, symmetric matrix that provides a two-dimensional (2D) view of the inter-residue spatial distances in a protein 3D structure with contacts denoted as 1, and non-contacts as 0. That is, whenever the distance between any two residues in the 3D structure within a distance threshold value, typically considered between 6 and 16 Å, considering some specific atoms (mostly C_α or C_β) of the residue pairs, the corresponding residue pairs are said to be in contact. In this work, we use 8 Å as the distance threshold for contact maps. Contact map overlap (CMO) is used to find the similarity between two contact maps, where the higher CMO score means there is a higher likelihood of being similar. The state-of-the-art method for CMO, Al-eigen (Di Lena et al. 2010), applies a heuristic approach to obtain a set of principal weighted eigenvectors by using the eigenvalue decomposition of symmetric matrices or contact maps. Finally, the overlap score between two contact maps is obtained by calculating the optimal global alignment between two sets of weighted eigenvectors by using the Needleman-Wunsch global alignment algorithm. In our present work, we run Al-eigen using seven eigenvectors to get the CMO score between a pair of contact maps.

For the threading template scoring, we integrate the CMO score along with the Z_{score}

described above to calculate the final score for selecting the best-fit template. Since the range of CMO is [0,1], we use the weight 10 as the weight of CMO in calculating the final score. Consequently, the final score for the threading template selection is:

$$F_{score} = Z_{score} + (CMO \times 10) \quad (3)$$

After selecting the top template using Equation (3), we build the 3D model of the query protein using the query-template alignment by copying the coordinate of aligned residues from the template.

2.3.3 Template libraries, benchmark data, and programs to compare

We use a representative non-redundant template library collected from https://zhanglab.ccmb.med.umich.edu/library/ (J. Yang et al. 2015), which contains 70,670 template structures.

We benchmark against three datasets. The first data set is the Test500 (S. Wu and Zhang 2008), which contains 500 test proteins. It is a set of nonhomologous proteins with sequence identity < 25% and having length from 50 to 633 residues. On Test500 dataset, we compare the performance of our work against a popular threading-based method, MUSTER (S. Wu and Zhang 2008). MUSTER (Multi-Source ThreadER) is a threading algorithm that uses various structural and sequential single-body features to generate the query-template alignment using the Needleman-Wunsch dynamic programming algorithm. We also benchmark our approach against our in-house baseline threading approach that does not use the contact information but utilizes the same alignment scoring function described above. For a fair comparison, we use the same template library for all competing methods where templates with sequence identity > 30% to the query protein are excluded. As the source of contact, we use both true contact maps extracted from the native structures of the query proteins as well as contact maps predicted from their sequences by

RaptorX (S. Wang et al. 2017), a state-of-the-art contact prediction method that integrates sequence co-evolution and deep learning. To reduce the noise in RaptorX predicted contact maps, residue pairs with predicted contact probability < 0.5 are excluded.

The second test set is the 150 proteins in the PSICOV (David T. Jones et al. 2012) dataset, which contains 150 single chains and single domain monomeric proteins. On this dataset, we benchmark our work against the state-of-the-art contact guided *ab initio* folding method CONFOLD2(Adhikari and Cheng 2018), which builds 3D protein structures using predicted contact maps and secondary structures. It constructs a pool of models by exploring the fold space using different subsets of contacts and then selects the top five models through clustering. As the source of contacts, CONFOLD2 uses contact maps predicted from MetaPSICOV (David T. Jones et al. 2015), another state-of-the-art contact predictor that integrates sequence co-evolution and machine learning. The published work of CONFOLD2 fails to report results for 4 targets from the PSICOV dataset. We, therefore, consider 146 targets for the current benchmarking. For a fair comparison, we use the same MetaPSICOV predicted contact maps after excluding homologous templates. To do this, we use three different increasingly stringent homology cutoffs as follows. First, we exclude all templates from the template library with sequence identity > 30% to the query proteins (referred to as Cutoff-1). In addition to sequence identity cutoff, to make the template selection cutoff more stringent, we exclude templates in the same SCOP (Hubbard et al. 1997) (Version 1.75) family of the query proteins (referred to as Cutoff-2); and templates in both SCOP family and superfamily (referred to as Cutoff-3).

The third test set is CASP13 targets officially released in December 2018. We consider only 20 full-length targets resulting in a total of 32 domains that CASP has officially released so far. Here, we benchmark against two state-of-the-art contact-assisted methods: EigenTHREADER

(Buchan and Jones 2017), and map_align (Ovchinnikov et al. 2017). EigenTHREADER is a fold recognition method, which uses MetaPSICOV contact maps for searching the template library of contact maps. Similar to Al-eigen (Di Lena et al. 2010), it uses the eigenvector decomposition and dynamic programming to generate alignment between two sets of weighted eigenvectors. Since EigenTHREADER produces three kinds of alignment scores for each template, we use the contact map overlap (CMO) to rank templates, as it gives the best result among the three. map_align uses a pure co-evolutionary based contact maps to find analogous folds from the library of templates. To make a fair comparison, we use the same template library curated before CASP13 started on May 1, 2018 containing 69,041 template structures as well as the same contact maps predicted by RaptorX (S. Wang et al. 2017) for all three methods.

2.3.4 Evaluation criteria

TM-score (Y. Zhang and Skolnick 2004) is used to evaluate the performance of each competing method. TM-score gives a score in the range (0,1], where a higher score means a better structural similarity. TM-score > 0.5 indicates the pair of proteins share the same fold (Jinrui Xu and Zhang 2010).

2.4 Results and discussion

2.4.1 Performance on Test500 set

As shown in **Table 2.1**, our threading method that includes residue-residue contact information (referred to as 'This work') outperforms MUSTER as well as our baseline threading implementation that does not include contacts (referred to as 'This work$_{NOCONTACT}$') in predicting the top ranked model both when using native contact maps and RaptorX predicted contact maps. On an average, our method that includes the contact information predicts the top ranked model with an average TM-score of 0.528 and 0.524 by using native contact maps and RaptorX predicted

contact maps respectively. In terms of average TM-score of top ranked models, our work (with native contacts) outperforms MUSTER as well as our baseline threading method by 0.011 and 0.01 TM-score points respectively, whereas the differences are 0.007 and 0.006 TM-score points respectively when we evaluate our work (with RaptorX contacts) with MUSTER and our baseline threading method. It is worth mentioning that the higher average TM-score means the higher topological similarity of the query protein with the native structure. Moreover, an average TM-score >0.5 over 500 targets illustrates that the corresponding method predicts correct folds (TM-score >0.5) for the majority of the cases. **Table 2.1** also shows that the performance of our baseline threading method is comparable to that of MUSTER. Our work, therefore, delivers consistently better average TM-score compared to MUSTER as well as our baseline threading method, indicating that the inclusion of the residue-residue contact information helps to boost the average accuracy of the top ranked predicted model.

Table 2.1 Performance comparison on Test500 dataset[a] based on the average TM-score of top ranked models.

Contact Source	**MUSTER (*p*-value*)**	**This work$_{\text{NOCONTACT}}$ (*p*-value*)**	**This work**
Native contact	0.517 (0.001)	0.518 (0.002)	**0.528**
RaptorX	0.517 (0.007)	0.518 (0.012)	**0.524**

[a] excluding templates with sequence identity > 0.30 to the query protein.

*one sample t-test's *p*-value of the TM-score difference compared to This work.

To examine whether the performance boost attained by our work is statistically significant, we perform t-test of the TM-score improvements. On Test500 dataset, our work (with native contacts) is statistically significantly better at 95% confidence level compared to MUSTER (*p*-value = 0.001) and our baseline threading method (*p*-value = 0.002). Furthermore, the performance

improvement of our work (with RaptorX contacts) is also statistically significant at 95% confidence level compared to MUSTER and our baseline threading method with p-values < 0.05 (**Table 2.1**). Overall, the results indicate that incorporating the contact information yields statistically significantly better threading performance in terms of the top ranked model using both native and predicted contact maps.

Figures 2.2 and **2.3** show a head-to-head comparison of our work (referred to as 'This work') with MUSTER and our baseline threading method (referred to as 'This work$_{\text{NOCONTACT}}$') respectively. In **Figures 2.2(A)** and **2.3(B)**, there are 12% and 11% more points, respectively below the diagonal line, which indicates that our work identifies better alignment than MUSTER regardless of using native contacts or RaptorX predicted contacts. We also observe a similar trend in **Figures 2.3(A)** and **2.3(B)** when we do a head-to-head comparison of this work with our baseline threading method. **Figures 2.2(C)** and **2.2(D)** show the bimodal distribution of TM-score of top one models predicted by our work (in red) and MUSTER (in blue). These figures illustrate the bimodality due to the diversity of Test500 data set, which has roughly a balanced combination of easy targets (both methods predict the top one model with TM-score > 0.5) and hard targets (both methods predict the top one model with TM-score <= 0.5). In **Figure 2.2(C)**, the highest peak of our work is slightly higher than that of MUSTER, which means the density of models of this work (with native contacts) in the TM-score range [0.6,0.8] is more than MUSTER. We also observe that the second highest peak of our work is lower than MUSTER in the TM-score range [0.1,0.3], which demonstrates a fewer number of models with low TM-score are predicted by our work compared to MUSTER. Moreover, in the TM-score range [0.3,0.5], the density of models of our work is more than MUSTER. **Figure 2.2(D)** shows a similar trend with the density of models of our work (with RaptorX contacts) and MUSTER being comparable for easy targets, and for a

TM-score range around [0.3,0.5], our work predicts more models than MUSTER as opposed to the TM-score range (0,0.3].

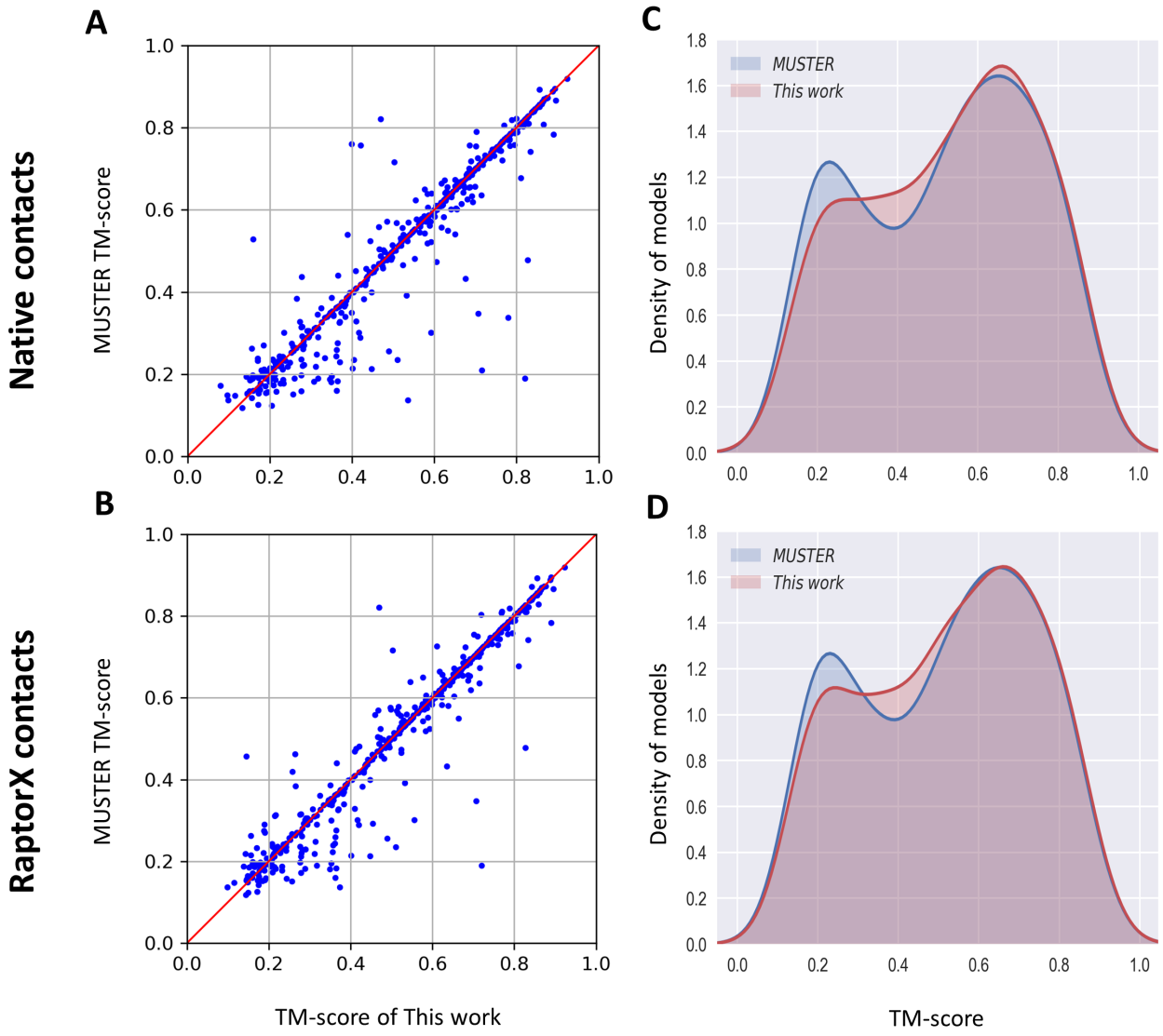


Figure 2.2 A head-to-head performance comparison of our work and MUSTER based on the accuracy of the top ranked models on Test500 dataset. (A) MUSTER versus This work (with native contact maps), (B) MUSTER versus This work (with RaptorX contact maps). Each point in (A) and (B) represents the TM-score of the top ranked models predicted by This work (x-axis) and MUSTER (y-axis), respectively. (C) Bimodal distribution of TM-score of the top ranked models predicted by MUSTER (in blue) and This work with native contact maps (in red), (D) Bimodal distribution of TM-score between MUSTER (in blue) and This work with RaptorX predicted contact maps (in red). Templates with sequence identity > 30% to the query protein are excluded.

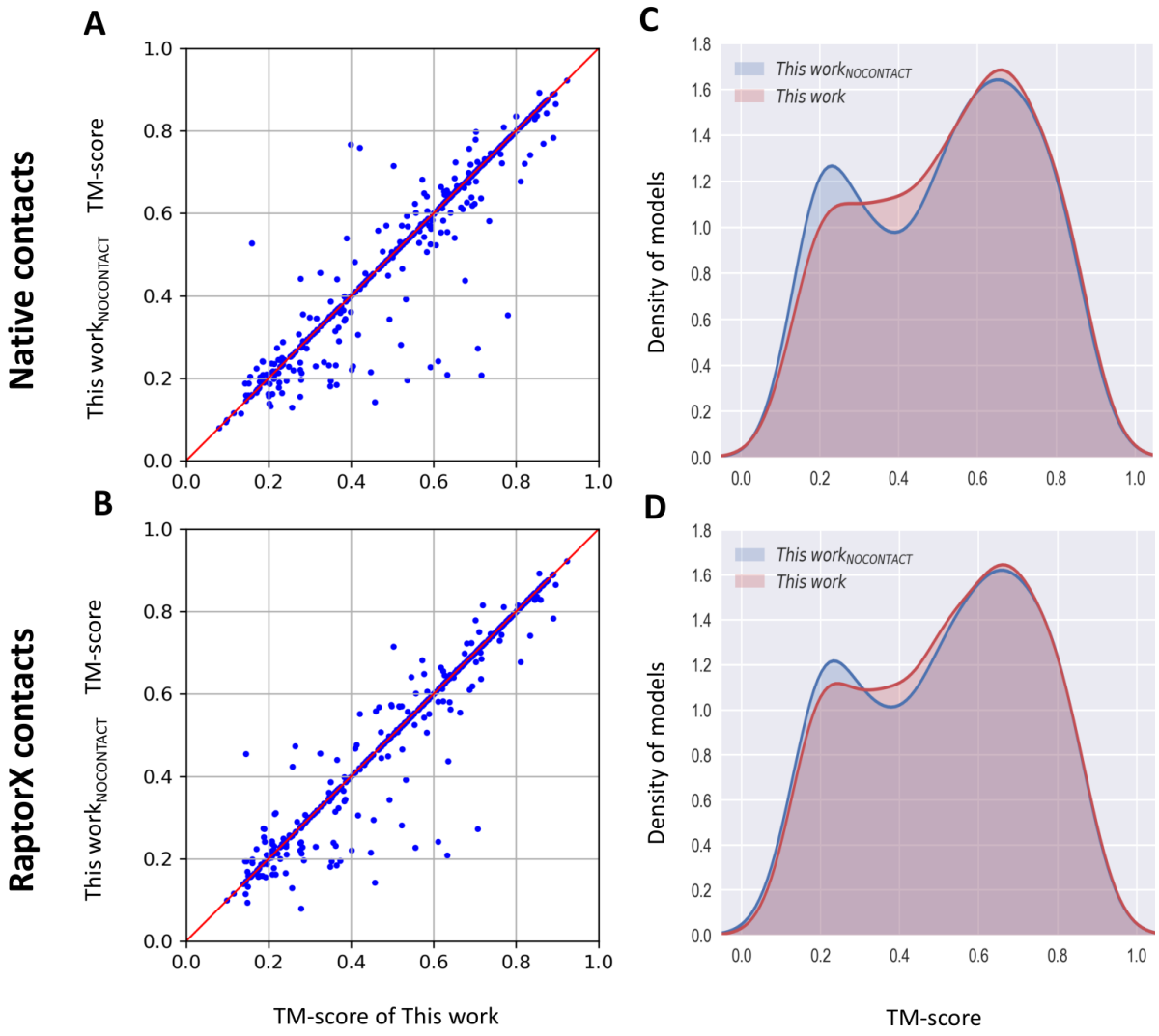


Figure 2.3 TM-score comparison between our work and our baseline threading method for the top ranked model on Test500 dataset. This work$_{NOCONTACT}$ refers to our baseline threading method, which does not use contact information; This work refers to our work. (A) This work$_{NOCONTACT}$ versus This work (with native contact maps), (B) This work$_{NOCONTACT}$ versus This work (with RaptorX contact maps). Each point in (A) and (B) represents the TM-score of the top ranked models predicted by This work (x-axis) and This work$_{NOCONTACT}$ (y-axis), respectively. (C) Bimodal distribution of TM score of the top ranked models predicted by This work$_{NOCONTACT}$ (in blue) and This work with native contact maps (in red), (D) Bimodal distribution of TM-score between This work$_{NOCONTACT}$ (in blue) and This work with RaptorX predicted contact maps (in red). Templates with sequence identity > 30% to the query protein are excluded.

A similar trend is observed in **Figures 2.3(C)** and **2.3(D)**, where we plot the TM-score distribution of the top one model predicted by both of our threading approaches – one using contact information while the other does not. For easy targets and for the TM-score range [0.3,0.5], the density of models of our work (referred to as 'This work') is more than that of our baseline threading method (referred to as 'This work$_{\text{NOCONTACT}}$'); whereas the opposite trend is shown for TM-score range (0,0.3], which indicates that our work predicts more models with a higher accuracy than that of our baseline threading method that does not include the contact information. In summary, the results demonstrate that inclusion of the residue-residue contact information boosts protein threading by shifting its performance distributions towards the higher accuracy.

2.4.2 Performance on PSICOV-150 set

Next, we compare the performance of our work with CONFOLD2, a state-of-the-art contact driven *ab initio* folding method, using the PSICOV-150 dataset after excluding four targets for which CONFOLD2 fails to report the performance. As shown in **Table 2.2**, our work consistently outperforms CONFOLD2. For targets with the predicted contact map precision ≤ 50% (122/146 cases), our work (using Cutoff-1) achieves a mean TM-score of 0.628 compared to 0.573 of CONFOLD2, whereas we achieve mean TM-score of 0.627 and 0.617 using Cutoff-2 and Cutoff-3 respectively. For the remaining targets with a high precision contact maps (24/146 cases), though there is an improvement in mean TM-score of both methods, our work (using Cutoff-1) outperforms CONFOLD2 by achieving a mean TM-score of 0.691, which is about 0.07 TM-score points better than that of CONFOLD2. The increase in TM-score reaches to 0.068 using Cutoff-2 or Cutoff-3. Considering all targets, our work (using Cutoff-1) predicts top ranked models with an average TM-score of 0.638 that is 0.058 TM-score points more than that of CONFOLD2. We achieve average TM-score of 0.637 and 0.628 using Cutoff-2 and Cutoff-3 respectively. It is also

worth mentioning that while CONFOLD2 fails to report the performance for four targets 1atzA, 1bkrA, 1c44A, and 1c52A, our work predicts the top ranked model with an average TM-score > 0.53, irrespective of different homology cutoffs.

Table 2.2 Performance comparison of our work against CONFOLD2 on PSICOV-150 dataset[a] based on the average TM-score of top ranked predicted models.

Homology cutoff	Contact Precision[b]	CONFOLD2(*p* value*)	This work
Cutoff-1[c]	<= 50%[f]	0.573 (0.004)	**0.628**
	> 50%	0.621 (0.08**)	**0.691**
	All	0.580 (0.0009)	**0.638**
Cutoff-2[d]	<= 50%	0.573 (0.007)	**0.627**
	> 50%	0.621 (0.087**)	**0.689**
	All	0.580 (0.002)	**0.637**
Cutoff-3[e]	<= 50%	0.573 (0.032)	**0.617**
	> 50%	0.621 (0.087**)	**0.689**
	All	0.580 (0.009)	**0.628**

[a] considered 146 targets as the published work of CONFOLD2 fails to report results for 4 targets namely: 1atzA, 1bkrA, 1c44A, and 1c52A whereas our work predicts the top ranked model with a mean TM-score > 0.53, irrespective of different cutoffs.

[b] calculated over all the contacts with probability of being in contact is at least 0.5 and showing here as a percentage.

[c] excluding templates with sequence identity > 0.30 to the query protein.

[d] excluding SCOP family and sequence identity > 0.30 to the query protein.

[e] excluding SCOP family and superfamily, and sequence identity > 0.30 to the query protein.

[f] calculated over 122 targets.

*one sample t-test's *p*-value of the TM-score difference of our work.

**calculated over only 24 targets, which might not be sufficiently large sample size to meaningfully evaluate statistical significance.

We also perform the t-test of the TM-score difference to examine whether the improvement attained by our work is statistically significant. As reported in **Table 2.2**, for targets with contact precision ≤ 50% and for all targets, our work is statistically significantly better than CONFOLD2 at 95% confidence level in all cutoffs. For targets with low precision contacts (≤ 50%), *p*-value is 0.032 in the most stringent cutoff (Cutoff-3) compared to 0.004 in Cutoff-1 and 0.008 in Cutoff-2. Similarly, for all targets, *p*-value is 0.009 in Cutoff-3 compared to 0.0009 in Cutoff-1 and 0.002 in Cutoff-2. For targets with high accuracy contact maps with a precision > 50%, the average TM-score of our work is slightly better than CONFOLD2, but the difference is not statistically significant at 95% confidence level. It should be noted here that there are only 24 targets with high accuracy contact maps with a precision > 50% and a sample size of only 24 may not be large enough for a meaningful statistical significance test. It is also worth mentioning that the exclusion of SCOP family, superfamily along with 30% sequence identity cutoff to the query protein is a very stringent cutoff for our work to compare it with CONFOLD2. Overall, these tests demonstrate that the inclusion of the contact information into threading yields statistically significantly better performance than the contact- assisted *ab initio* folding.

Figure 2.4(A) shows a head-to-head comparison between both methods based on the TM-score of top ranked predicted models with our work (in Cutoff-3) significantly outperforming CONFOLD2. We chose the homology Cutoff-3 for this comparison because it represents the most stringent homology cutoff. 69.2% of data points (in **Figure 2.4(A)**) lie below the diagonal line, which indicates that our work predicts a significantly better accurate top ranked model than CONFOLD2. In 28 cases, CONFOLD2 predicts the top ranked model with a TM-score < 0.5, while our work successfully predicts the correct fold (the top ranked model with a TM-score > 0.5). On the other hand, CONFOLD2 predicts the correct folds (the top ranked model with a TM-

score > 0.5) for 25 targets, but our work fails. Out of 146, our work predicts models from the top-ranked template with a TM-score >= 0.5 for 109 test proteins (marked in green quadrilateral in **Figure 2.4(A)**). Out of these 109 test proteins, our work attains better TM-score than CONFOLD2 for 97 cases. **Figure 2.4(B)** shows the TM-score distribution of top ranked models predicted by our work (in Cutoff-3) and CONFOLD2. The highest peak of the distribution of our work (in red) is larger as well as skewed towards the higher accuracy (right) side compared to CONFOLD2 (in blue), indicating that our work predicts more models with a better accuracy than the other method.

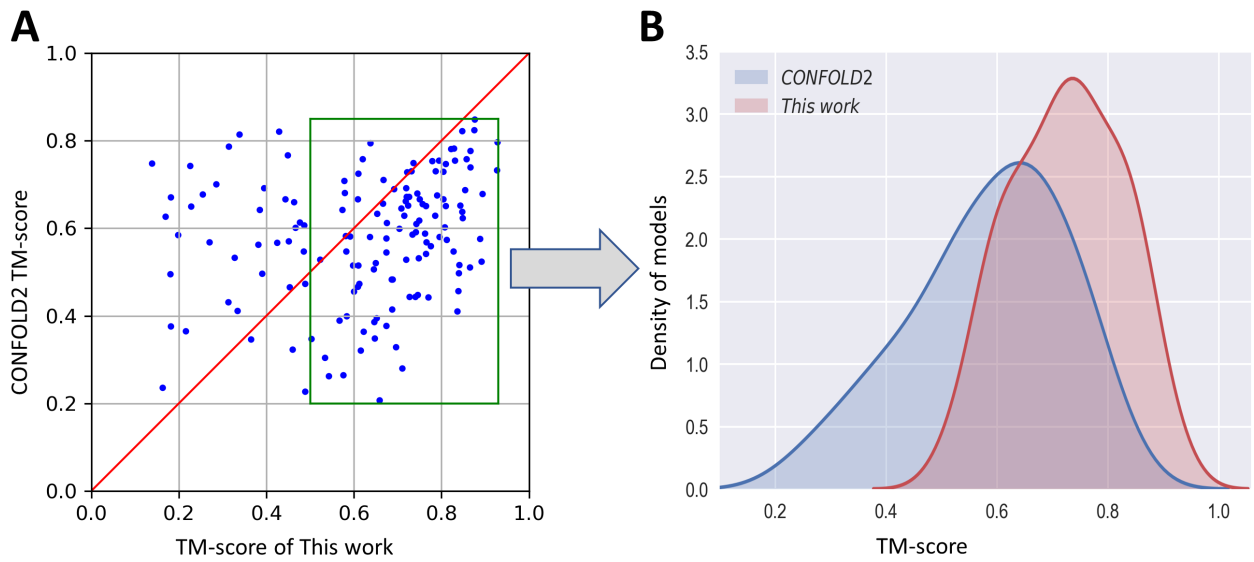


Figure 2.4 A head-to-head performance comparison between our work (using Cutoff-3) and CONFOLD2 based on the TM-score of the top ranked models on PSICOV-150 dataset. (A) Each point represents the TM-score of the top ranked models predicted by our work (x-axis) and CONFOLD2 (y-axis), respectively. For 109 test proteins, our work predicts top ranked models with TM-score >= 0.5, which is shown in green quadrilateral. (B) TM-score distribution of the top ranked models predicted by this work (in red), and CONFOLD2 (in blue) over the 109 test proteins.

2.4.3 Performance on CASP13 set

We further evaluate the performance of our work on the CASP13 dataset consisting of 20 full-length targets resulting in a total of 32 domains officially released so far. For a fair performance evaluation, the same template library and the same non-redundant (nr) sequence database are used by all competing methods and both the databases were created before CASP13

started on May 1, 2018. Furthermore, we use the same RaptorX predicted contact maps for all competing methods. We use the default settings with 65 threads to run map_align for each target. Since map_align is highly computationally expensive (Refer to **Appendix 1**) for target-by-target CPU hours needed by map_align), we only consider 11 full-length targets of length < 300 residues, resulting in a total of 14 domains. For EigenTHREADER, we use the default setting except setting values of the parameters c (the distance threshold for contact maps) as 8 Å and t (the number of eigenvectors) as seven. Since we run Al-eigen using seven eigenvectors in our work, the same number of eigenvectors is used for EigenTHREADER to make a fair performance comparison.

Table 2.3 Performance comparison over 20 full-length CASP13 targets[a] based on top ranked models by our work and two state-of-the-art contact-assisted threading methods. TM-align results are included as a reference.

Methods	Average TM-score	% time TM-score > 0.5[b]
map_align[c]	0.39	18.2
EigenTHREADER	0.43	30.0
This work	**0.45**	**40.0**
TM-align[d]	0.67	85.0

[a] officially released by CASP on December 2018.
[b] percentage of time the respective method predicts the correct fold (TM-score> 0.5).
[c] since map_align is too computationally expensive, we run it only on 11 CASP full-length targets (of length < 300 residues) out of 20 full-length targets and the results are based on those 11 targets.
[d] using native 3D structures of the query proteins officially released by CASP.

As shown in **Table 2.3**, our work (referred to as 'This work') outperforms map_align and EigenTHREADER over 20 full-length targets in terms of average TM-score of top ranked models and the percentage of time the top ranked model is predicted with a TM-score > 0.5 (i.e. with the correct fold). Our work predicts the top ranked model with an average TM-score of 0.45 compared to that of 0.43 of EigenTHREADER and 0.39 of map_align. Moreover, 40% of the time our work predicts the similar fold (the top ranked model with a TM-score > 0.5) which is about 21% and

10% better than map_align and EigenTHREADER respectively. It is worth mentioning that map_align's performance is analyzed over 11 full-length targets having length < 300 residues instead of 20 targets due to the expensive computation. To investigate whether the performance of contact-assisted threading methods is the optimal, we run TM-align (Y. Zhang and Skolnick 2005b) using the CASP13 officially released native 3D structure of the query protein. TM-align performs the structural superposition between the query protein and the template library in order to select the optimal template. The average TM-score (by TM-align) of the best template selected by TM-align is 0.67 and 85% of the time it finds the correct fold, revealing the gap between top templates found by the state-of-the-art contact-assisted threading methods and the best possible templates.

Table 2.4 Performance comparison on CASP13 dataset over 32 domains[a] based on top ranked models by our work and two state-of-the-art contact-assisted threading methods. TM-align results are included as a reference.

Methods	Average TM-score	% time TM-score > 0.5[b]
map_align[c]	0.36	14.3
EigenTHREADER[d]	0.38	25.0
This work	**0.39**	**28.1**
TM-align[e]	0.70	93.75

[a] CASP officially releases 20 full-length targets in a total of 32 domains on December 2018.
[b] percentage of time the respective method predicts the correct fold (TM-score> 0.5).
[c] considering only 14 CASP13 domains of length < 300 residues and values are based on those 14 domains.
[d] considering only 28 domains because TM-score fails to calculate TM-score for the following domains: T0960-D4, T0960-D2, T0960-D1, and T0957-D2 as EigenTHREADER's predicted models do not have any common residues to the native domains.
[e] using native 3D structures of the query proteins officially released by CASP.

In **Table 2.4**, we report the results of head-to-head comparisons between all competing methods over 32 domains based on TM-score of top ranked models. For EigenTHREADER, we

exclude 4 domains namely: T0960-D4, T0960-D2, T0960-D1, and T0957-D2 because the TM-score tool fails to superimpose EigenTHREADER's predicted models with the native domains as there are no common residues. Moreover, the results of map_align are based on 14 domains of length < 300 residues. On this dataset, our work outperforms EigenTHREADER and map_align in terms of average TM-score of top ranked models and the percentage of time it finds the correct fold (the top ranked model predicted with a TM-score > 0.5). On an average, our work achieves a TM-score of 0.39 compared to 0.38 of EigenTHREADER and 0.36 of map_align. Moreover, 28.1 % of the time our work finds the correct fold (TM-score > 0.5) which is about 3% and 14% better than EigenTHREADER and map_align respectively. It is worth mentioning that our work achieves an average TM-score of 0.415 and 32% of the time it predicts the correct fold (TM-score > 0.5) by considering 28 domains like EigenTHREADER. Considering 32 domains, once again 93.75% of the time TM-align finds the correct fold (TM-score > 0.5) including the average TM-score of 0.7, indicating that there is a large room for improvement. **Figure 2.5** shows the TM-score distribution of the top ranked model predicted by our work and EigenTHREADER over 28 domains. For a low TM-score range, the density of models of our work is lower than that of EigenTHREADER as opposed to a higher TM-score range, which indicates that our work predicts more models with a better accuracy compared to EigenTHREADER.

As a representative example, we present a case study on CASP13 target T0966 with 494 residues where SPIDER3 predicts the secondary structure with the Q3 accuracy of 87.6%. It is a single domain TBM-hard target as per the CASP official domain classification. **Figure 2.6(B)** shows that TM-align detects the template (PDB ID: 2ebfX) that has a correct fold to T0966 with a TM-score (by TM-align) of 0.84 by using the native 3D structure of T0966 released by CASP. We include TM-align as a reference to see the best possible template for the target T0966 can be found

in our template library. **Figure 2.6(C)** shows that our method detects the same template as the top template and predicts the top ranked model using that template with a TM-score of 0.8 to the native structure (**Figure 2.6(D)**). However, EigenTHREADER finds a different template (PDB ID: 1phpA2), which has an incorrect fold with a TM-score (by TM-align) of 0.36 and predicts the top one model with a TM-score of 0.19 to the native structure (**Figure 2.6(E)** and **2.6(F)**, respectively).

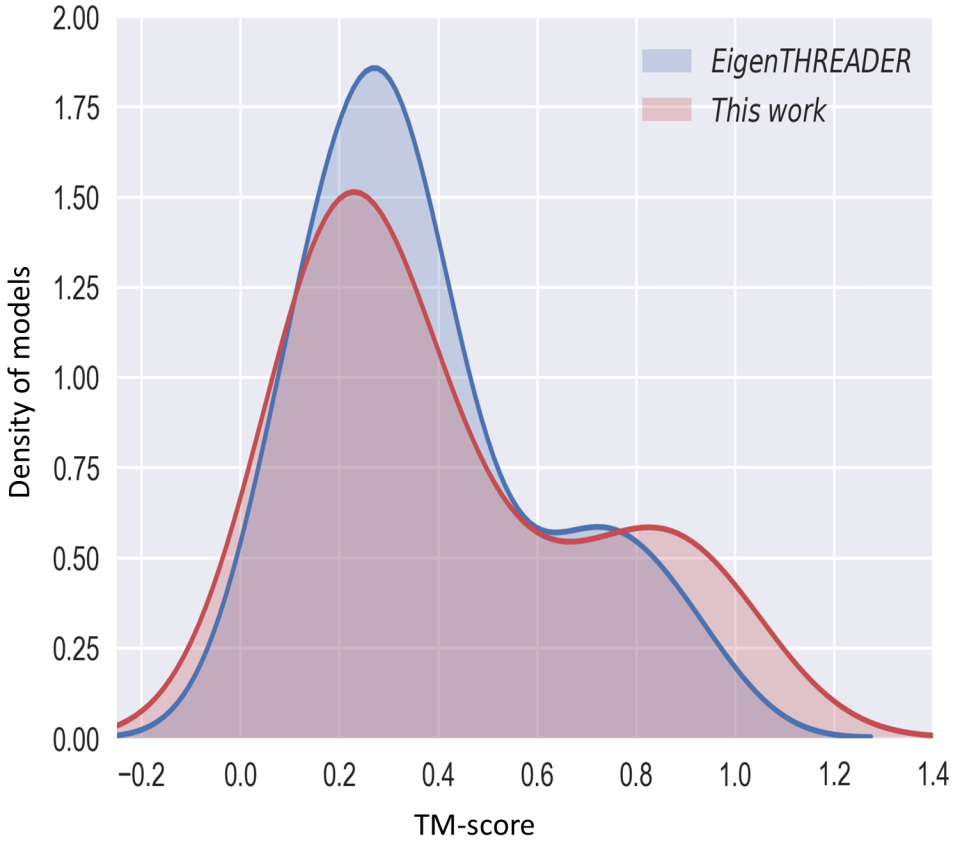


Figure 2.5 TM-score distribution of the top ranked models predicted by This work (in red) and EigenTHREADER (in blue) over 28 CASP released domains. We exclude 4 domains for which native domains do not have any residue match with EigenTHREADER's predicted models.

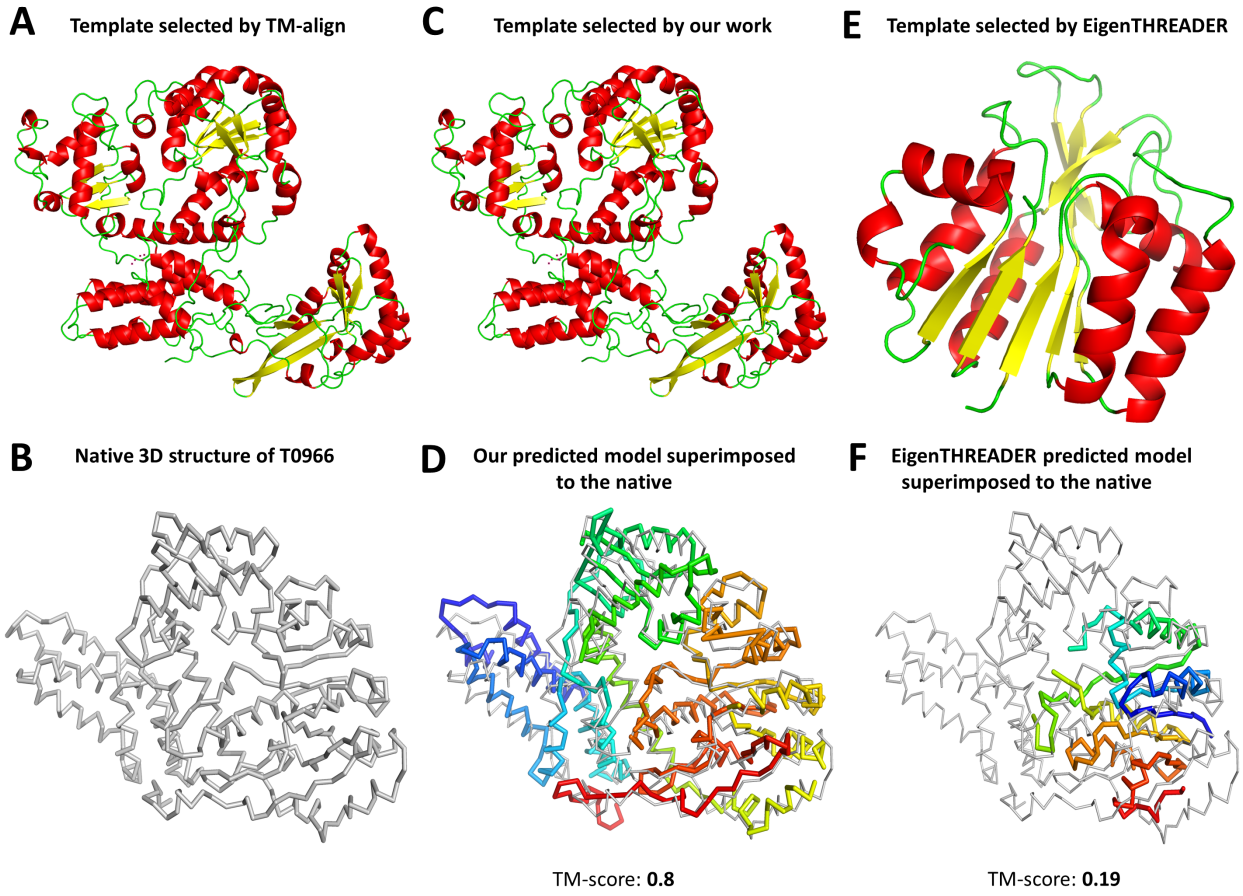


Figure 2.6 Performance of our work and EigenTHREADER on target T0966. (A) TM-align finds 2ebfX as the best template in the template library with a TM-score (by TM-align) of 0.84, (B) Experimental structure of T0966, (C) Similar to TM-align, our work finds the same template (i.e. 2ebfX) as the top ranked template, (D) Structural alignment between the top ranked model predicted by our work (in thick rainbow) with a TM-score of 0.8 and the native structure of T0966 (in thin gray). (E) EigenTHREADER finds 1phpA2 as the top ranked template with a TM-score (by TM-align) of 0.36 for the same target. (F) Structural alignment between the top ranked model predicted by EigenTHREADER (in thick rainbow) with a TM-score of 0.19 and the native structure of T0966 (in thin gray).

2.5 Conclusion

In this article, we analyze whether the inclusion of residue-residue contact information improves the performable of protein threading. We develop a new threading method by combining sequential and structural features, and subsequently incorporate the residue-residue contact information in the form of the contact map overlap (CMO) score. We benchmark our work on three different datasets comprising of a diverse set of protein targets of varied difficulties. Experimental results demonstrate that our work outperforms a popular threading method MUSTER as well as our baseline threading approach that does not utilize the contact information, the state-of-the-art contact-assisted ab initio folding method CONFOLD2, and latest contact-assisted threading methods EigenTHREADER and map_align. Collectively, our study indicates that the inclusion of the contact information improves protein threading.

Chapter 3

Evaluating the significance of contact maps in low-homology protein modeling using contact-assisted threading

3.1 Abstract

Chapter 2 discusses the utility of contact information to boost protein threading by developing a new contact-assisted threading method. However, the nature and extent to which the quality of a predicted contact map impacts the performance of contact-assisted threading remains elusive. Here, we systematically analyze and explore this interdependence by employing our newly-developed contact-assisted threading method over a large-scale benchmark dataset using predicted contact maps from four complementary methods including direct coupling analysis (mfDCA), sparse inverse covariance estimation (PSICOV), classical neural network-based meta approach (MetaPSICOV), and state-of-the-art ultra-deep learning model (RaptorX). Experimental results demonstrate that contact-assisted threading using high-quality contacts having the Matthews Correlation Coefficient (MCC) ≥ 0.5 improves the threading performance in nearly 30% of cases, while low-quality contacts with the MCC < 0.35 degrades the performance for 50% of cases. This holds true even in CASP13 dataset, where threading using high-quality contacts (MCC ≥ 0.5) significantly improves the performance of 22 instances out of 29. Collectively, our study uncovers the mutual association between the quality of predicted contacts and its possible utility in boosting the threading performance for improving low-homology protein modeling.

3.2 Introduction

In **Chapter 2**, we have discussed our newly developed contact-assisted threading method by successfully integrating the accurate residue-residue contact information for the improved protein

threading (Bhattacharya and Bhattacharya 2019a). Specifically, we have integrated residue-residue contact maps predicted by RaptorX (S. Wang et al. 2017), one of the most accurate contact prediction methods, with structural and sequential information such as profiles, secondary structures, solvent accessibility, torsion angles (psi and phi), and hydrophobicity for contact-assisted threading. Experimental results have shown that the inclusion of the contact information attains statistically significantly better performance compared to a contact-free threading method when everything else remains the same, demonstrating that the inclusion of the contact information in protein threading is a promising avenue for improving the performance of threading methods. Furthermore, in a head-to-head performance comparison utilizing the same RaptorX-derived contact maps to guarantee a fair comparison, our method has successfully outperformed state-of-the-art contact-assisted threading methods EigenTHREADER and map_align, indicating our method as one of the best contact-assisted protein threading protocols. However, it is not clear how the quality of a predicted contact map affects contact-assisted threading. Nor it is clear whether contact-assisted threading with low-quality contact maps is as advantageous over the pure threading as contact-assisted threading with high-quality contact maps such as those predicted from RaptorX. Finally, in the presence of competing contact maps of comparable qualities predicted by state-of-the-art contact predictors, is there any advantage of using one over the other in terms of the improved threading performance? While assessing the efficacy of contact maps for low-homology protein modeling requires a head-to-head comparison between the contact-assisted threading and the contact-free pure threading, neither EigenTHREADER nor map_align can perform threading in a contact-free mode. Our method, on the other hand, can be seamlessly customized to perform contact-assisted or contact-free threading modes, enabling the evaluation of the utility of contact maps for remote homology modeling.

To evaluate the significance of contact maps in low-homology protein modeling, here we systematically investigate the impact of the quality of predicted contacts on the accuracy of contact-assisted threading by employing our newly developed contact-assisted threading method over several datasets. First, we analyze predicted contact maps from RaptorX and three other complementary methods having a wide range of qualities of their predicted contacts based on different contact map evaluation criteria to objectively evaluate how to select the most informative contact map. Then, we integrate the predicted contact maps from these contact predictors one by one into our contact-assisted threading method to examine the impact of each predicted contact map on the threading performance and compare them with a baseline threading algorithm that does not utilize contact information as well as RaptorX-assisted threading. Finally, we compare the performance of our contact-assisted threading by incorporating comparable-quality contact maps predicted by the top two officially ranked contact predictors from the CASP13 experiment to further study the impact of the quality of contacts in the threading performance. Collectively, our study unravels the mutual association that exists between the quality of a contact map and the performance of contact-assisted threading.

3.3 Materials and methods

3.3.1 Scoring a query-template alignment

Our newly-developed contact-assisted threading method, described in **Chapter 2**, is an iterative query – template alignment approach where query-template alignments are performed by the Needleman-Wunsch global alignment algorithm (Needleman and Wunsch 1970) The threading scoring function consists of close and distant sequence profiles, secondary structures, solvent accessibility, structure profiles, torsion angles, and hydrophobicity match based on which the normalized alignment score or Z_{score} is calculated for ranking the templates.

A residue-residue contact map, which is a binary, square, and symmetric matrix, is a two-dimensional representation of the protein's 3D structure. A contact indicates that the spatial distance between a pair of residues is less than a given distance threshold, typically set at 8 Å, between the C_α or C_β atoms of the residue pairs. Contact Map Overlap (CMO) finds the similarity between two contact maps, where the higher CMO score indicates that a higher similarity between the two comparing contact maps. Al-Eigen (Di Lena et al. 2010), one of the state-of-the-art CMO methods, computes an overlap between two input contact maps and gives a score between [0,1] with a higher score indicating a better agreement of contact maps. We integrate the CMO score returned from Al-Eigen into our threading method for selecting the best-fit template by formulating the final score as discussed in (Bhattacharya and Bhattacharya 2019a). After identifying the best-fit template, the query-template alignment is used to copy the coordinate of the aligned residues from the template to build the final 3D model of the query protein. Please refer **Chapter 2** for further details about the method and its scoring function.

3.3.2 Template libraries, benchmark data, and predicted contact maps

We use a representative non-redundant library of templates containing 70,670 templates, collected from: https://zhanglab.ccmb.med.umich.edu/library/ (J. Yang et al. 2015).

Our first benchmark dataset is the PSICOV150 dataset (David T. Jones et al. 2012), which contains 150 single chain and single domain proteins. In order to test the impact of different types of contact maps in the performance of contact-assisted method, we choose predicted contact maps from four complementary methods having a wide range of qualities of predicted contacts including (i) a mean field direct coupling analysis (mfDCA) (Kaján et al. 2014; Morcos et al. 2011), (ii) a sparse inverse covariance estimation method (PSICOV) (David T. Jones et al. 2012), (iii) a classical neural network-based meta approach (MetaPSICOV) (David T. Jones et al. 2015), and

(iv) a state-of-the-art ultra-deep learning model (RaptorX) (S. Wang et al. 2017). Here, we give a brief introduction of each contact predictor. mfDCA, an advanced formulation of direct coupling analysis (DCA), is a statistical inference framework used to infer direct co-evolutionary couplings between pair of residues in multiple sequence alignments. Another Evolutionary Coupling Analysis (ECA) technique, PSICOV, uses sparse inverse covariance estimation for the contact prediction. Although ECA methods are useful for predicting long-range contacts in the presence of a large number of sequence homologs, their accuracy is substantially poor if the number of sequence homologs is low (Z. Wang and Xu 2013). In recent years, machine learning or deep learning-based methods boost the accuracy of contacts. One such contact predictor, MetaPSICOV, a meta predictor, which uses a two-stage neural network by combining outputs of several ECA classifiers. It was ranked as one of the best contact predictors in CASP11 and CASP 12 (Wuyun et al. 2018). Another contact predictor powered by deep learning, RaptorX, incorporates the entire protein 'image' as a context for prediction by utilizing a Residual Convolutional Neural Network, or ResNet. It was ranked as one of the best contact predictors in CASP12 and CASP13 (Schaarschmidt et al. 2018).

We use the FreeContact package (Kaján et al. 2014) to obtain contact-maps predicted by mfDCA. Since the contact likelihood scores of mfDCA predicted contact maps are not normalized in the range [0,1], we normalize the contact likelihood scores by dividing each score by the maximum likelihood score of a given predicted contact map. PSICOV and MetaPSICOV contacts are obtained directly from the MetaPSICOV benchmark dataset (David T. Jones et al. 2015). RaptorX contacts are collected by submitting jobs to the RaptorX online server (http://raptorx.uchicago.edu/ContactMap/ (S. Wang et al. 2017; 2016). Residue pairs with contact likelihood scores < 0.5 are excluded to reduce the noise in all predicted contact maps. To make a

fair performance comparison, we use the same template library for all competing methods by excluding templates with sequence identity > 30% to the query protein to remove close homologs. It should be noted that, unlike other contact predictors, RaptorX fails to predict contacts for two targets namely: 1tqhA and 1hdoA. We, therefore, consider 148 targets for the current benchmarking.

Next, we benchmark on CASP13 dataset officially released in December 2018. We consider 20 full-length targets in a total of 32 domains for which CASP organizers released experimental structures so far. We consider the top two officially ranked contact predictors in CASP13(Shrestha et al. 2019) to test the impact of using comparable-quality contact maps in the performance of our contact-assisted method. In CASP13, the contact prediction category is heavily dominated by the latest breakthroughs in deep learning technologies. For example, G498 (ranked 1) or RaptorX-Contact, developed by Xu and coworkers, has attained the top performance since CASP12. It predicts residue-residue contacts using an ultra-deep learning model. It is worth mentioning that we also use RaptorX predicted contacts for our previous study (Bhattacharya and Bhattacharya 2019a) as well as for benchmarking on the PSICOV150 dataset for this current work. The second-ranked contact predictor, G032 or TripletRes, developed by Zhang and coworkers (Y. Li et al. 2019), is implemented by a deep residual fully convolutional neural network with evolutionary coupling features from deep multiple sequence alignments.

For CASP13 benchmarking, the template library is curated before CASP13 started on May 1, 2018, which contains 69,041 template structures. For a fair comparison, we use the same template library for all competing methods. We have downloaded the predicted contact maps from the official website of CASP and subsequently exclude residue pairs with a contact probability < 0.5 from all predicted contact maps to reduce the noise. It is also worth mentioning all residue

pairs of a predicted contact map with a contact likelihood of at least 0.5 with a minimum sequence separation of 6 residues are considered for all experiments.

3.3.3 Evaluation criteria of contact maps, and the resulting contact-assisted 3D structures

We use the following evaluation measures to evaluate predicted contact maps: precision, coverage, mean false positive error, spread, and Matthews correlation coefficient (MCC) (Monastyrskyy et al. 2014; Adhikari et al. 2016; Schaarschmidt et al. 2018). Precision is the percentage of correctly predicted contacts, $Precision = \frac{TP}{TP+FP}$, where TP represents true positives or correctly predicted contacts, and FP represents false positives or incorrectly predicted contacts. Coverage is the percentage of correctly predicted contacts with respect to the number of true contacts in the native contact map (N_c), $Coverage = \frac{TP}{N_c}$. Mean false positive error is the mean of absolute deviation of all incorrectly predicted contacts and is calculated by: $Mean\ FP\ Error = \frac{1}{FP}\sum(d_{ij} - d)$, where d represents the distance threshold (usually 8 Å) and d_{ij} represents the true distance of an incorrectly predicted pair of contacts. Spread is calculated by: $Spread = \frac{1}{N_c}\sum_{i=1}^{N_c} \min\{\ dist\ (T_i - P)\}$, where N_c represents the number of true contacts, T_i is a true contact, and $\min\{\ dist\ (T_i - P)\}$ is the minimum Euclidean distance between true pair of contacts and predicted contact pairs. Matthews correlation coefficient (MCC) is calculated by: $MCC = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$, where TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative respectively.

TM-score (Y. Zhang and Skolnick 2004) is used to evaluate the quality of the predicted 3D structure of query proteins with respect to the native (experimentally determined) structures. The value of TM-score lies in the range (0,1], where a higher score indicates a better similarity. A TM-score > 0.5 suggests a highly similar structure to the native fold (Jinrui Xu and Zhang 2010).

3.4 Results and discussion

3.4.1 Robust assessment of qualities of predicted contact maps

To objectively evaluate the most informative contact map, we compare the performance of each predicted contact map from different perspectives using various contact evaluation measures (Adhikari et al. 2016) over PSICOV150 dataset after excluding two targets (1tqhA and 1hdoA) for which RaptorX fails to predict contact maps.

Table 3.1 Evaluation of predicted contact maps[a] on PSICOV150 dataset[b], sorted by non-increasing order of the value of the MCC (the best performance and the best performer are listed in bold).

Contact Source	Precision	Coverage	Mean FP Error	Spread	MCC
RaptorX	72.08	**66.88**	**0.67**	**1.78**	**0.68**
MetaPSICOV	71.61	34.20	0.73	5.63	0.47
PSICOV	72.83	8.78	1.08	8.32	0.24
mfDCA	**75.22**	3.20	1.03	20.05	0.14

[a] excluding residue pairs with contact probability < 0.5.
[b] excluding two targets (1tqhA and 1hdoA) for which RaptorX could not predict contact maps.

As shown in **Table 3.1**, mfDCA attains the highest precision of 75.22% compared to 72.83% of PSICOV, 72.08% of RaptorX, and 71.61% of MetaPSICOV. From the standpoint of the precision, mfDCA seems to be the best contact predictors. However, all the other contact evaluation measures indicate that RaptorX attains the best performance. For example, RaptorX attains an MCC of 0.68 compared to 0.47 of MetaPSICOV, 0.24 of PSICOV, and 0.14 of mfDCA. RaptorX is also shown to reach the best score according to the coverage (66.88%), the mean floating point error (0.67), and the spread (1.78), whereas, MetaPSICOV, PSICOV, and mfDCA

achieve coverage of 34.2%, 8.78% and 3.2%, mean FP error of 0.73, 1.08 and 1.03, and spread of 5.63, 8.32 and 20.05 respectively. The results reveal that relying purely on one contact evaluation measure such as precision may not always be sufficient since evaluation measures focus on various aspects of the quality of predicted contacts that can sometimes be mutually contradictory. Furthermore, the fact that we only consider residue pairs with a contact probability of at least 0.5 to remove the noise, may have resulted in a very few numbers of contact pairs for mfDCA thereby artificially raising the precision. In contrast, MCC considers true and false positives and negatives, and therefore is a more balanced evaluation measure for predicted contacts.

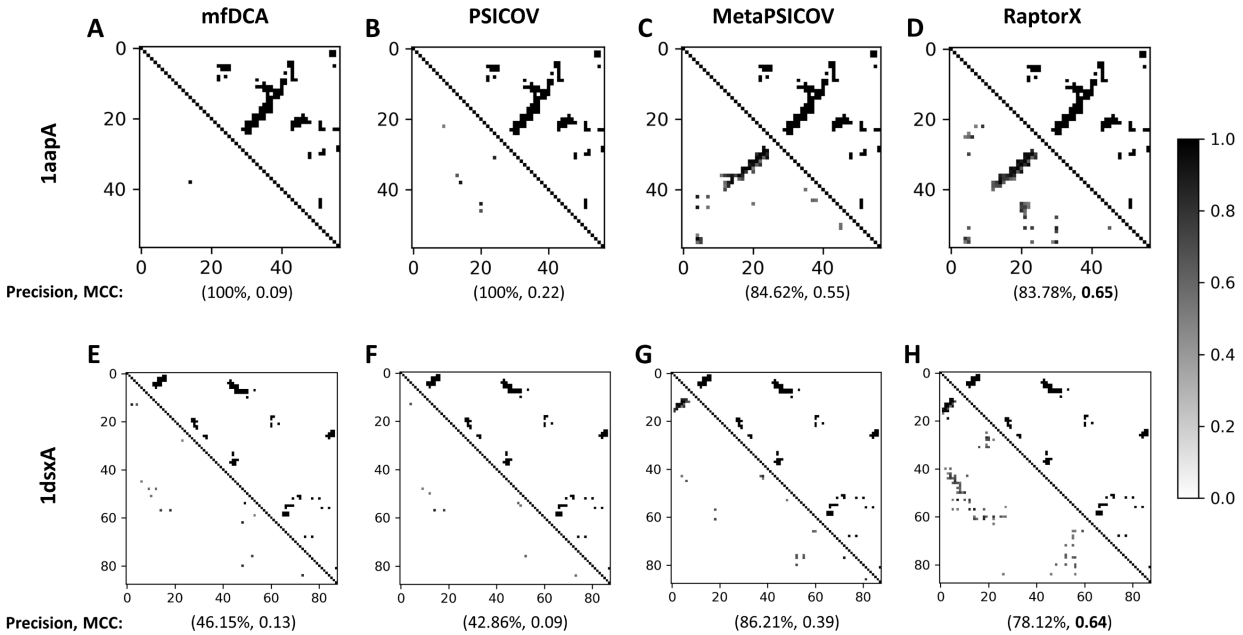


Figure 3.1 Representative examples of contact maps predicted by four complementary methods for targets 1aapA and 1dsxA. The upper triangles represent true (native) contacts of the target and the lower triangles represent predicted contacts with contact probability of at least 0.5. Numbers inside parenthesis indicate the precision (%), and the MCC respectively. For target 1aapA, (A) native contacts versus mfDCA contacts with a precision of 100% and an MCC of 0.09, (B) native contacts versus PSICOV contacts with a precision of 100% and an MCC of 0.22, (C) native contacts versus MetaPSICOV contacts with a precision of 84.62% and an MCC of 0.55, (D) native contacts versus RaptorX contacts with a precision of 83.78% and an MCC of 0.65. For target 1dsxA, (E) native contacts versus mfDCA contacts with a precision of 46.15% and an MCC of 0.13, (F) native contacts versus PSICOV contacts with a precision of 42.86% and an MCC of 0.09, (G) native contacts versus MetaPSICOV contacts with a precision of 86.21% and an MCC of 0.39, (H) native contacts versus RaptorX contacts with a precision of 78.12% and an MCC of 0.64.

As a representative example, we present two case studies on targets 1aapA (56 residues) and 1dsxA (87 residues) to illustrate mutual comparisons between the precision and the MCC, and to substantiate how the MCC is more balanced evaluation measure for predicted contacts. In **Figure 3.1**, the upper triangles represent the native contact map and the lower triangles represent the predicted contact map by different contact predictors after applying the contact likelihood score cutoff of at least 0.5. **Figures 3.1(A)** to **3.1(D)** represent native contacts of the target 1aapA versus contacts predicted by mfDCA, PSICOV, MetaPSICOV, and RaptorX respectively. Based on the precision, mfDCA and PSIOCOV appear to be the best contact predictor for the target 1aapA with a precision of 100%, as opposed to 84.62% of MetaPSICOV and 83.78% of RaptorX. However, mfDCA and PSICOV achieve high precision by predicting only a very few contact pairs correctly, but with a very low coverage. Precision of MetPSICOV and RaptorX, on the other hand, are comparatively lower due to the presence of few false positive contacts, but with a substantially higher coverage compared to mfDCA or PSICOV. MCC successfully addresses this issue with RaptorX achieving the best performance having an MCC of 0.65 compared to 0.55 of MetaPSICOV, 0.22 of PSICOV, and 0.09 of mfDCA. These results illustrate the fact that the MCC is more balanced evaluation measure and therefore better suited for predicted contact maps that are often noisy. **Figures 3.1(E)** to **3.1(H)** present a similar case study for target 1dsxA (87 residues). Once again, RaptorX predicted contact map achieves the best performance in terms of the MCC with a value of 0.64 compared to 0.39 of MetaPSICOV, 0.13 of mfDCA, and 0.09 of PSICOV; whereas MetaPSICOV contacts achieves the best performance in terms of precision (86.21%) compared to 78.12% of RaptorX, 46.15% of mfDCA, and 42.86% of PSICOV. Although this time precision offers a better balance, still it overly emphasizes prediction of true positive contacts. Overall, these examples demonstrate that MCC is more robust and consistent for noisy

contact maps compared to other contact evaluation measures. We, therefore, choose MCC as the main evaluation measure of the quality of predicted contact maps in this study.

3.4.2 Performance evaluation of contact-assisted threading with contact maps of diverse qualities

To investigate the impact of the quality of contact maps on the performance of contact-assisted threading, we benchmark our method using contact maps of diverse qualities over PSICOV150 dataset.

Table 3.2 Impact of the quality of contact maps on the performance of contact-assisted threading on PSICOV150 targets[a] based on top ranked models. The table is sorted by non-decreasing order of performance (best performance and best performer are listed in bold) with shaded row representing the performance of pure threading method.

Methods	Average TM-score (*p*-value*)	%time TM-score > 0.5[b]
rr_{mfDCA}-assisted threading[c]	0.58 (1.5e-11)	69.6
rr_{PSICOV}-assisted threading[d]	0.59 (1.3e-09)	71.6
pure threading[e]	0.63 (0.0001)	75.7
$rr_{MetaPSICOV}$-assisted threading[f]	0.64 (0.0007)	77.7
$rr_{RaptorX}$-assisted threading[g]	**0.66**	**80.4**

[a] excluding two targets (1tqhA and 1hdoA) for which RaptorX could not predict contact maps.

[b] percentage of time the respective method predicts the correct fold (TM-score > 0.5)

[c] contact-assisted threading method using mfDCA contacts

[d] contact-assisted threading method using PSICOV contacts

[e] pure threading method (without contacts)

[f] contact-assisted threading method using MetaPSICOV contacts

[g] contact-assisted threading method using RaptorX contacts

*one sample T-Test's *p*-value of the TM-score difference compared to $rr_{RaptorX}$-assisted threading.

As shown in **Table 3.2**, our contact-assisted threading method powered by the high-quality contacts from RaptorX (referred to as $rr_{RaptorX}$-assisted threading) and moderate-quality contacts from MetaPSICOV (referred to as $rr_{MetaPSICOV}$-assisted threading) outperform the contact-free pure threading method (referred to as pure threading) serving as a control in terms of the accuracy of the top ranked predicted models. Considering TM-score of top ranked models, RaptorX-assisted threading method delivers the best performance by achieving a mean TM-score of 0.66, which is 0.03 TM-score points more than that of the baseline threading method, whereas the mean TM-score improvement reaches to 0.01 for MetaPSICOV-assisted threading method compared to baseline threading method. Moreover, 80.4% and 77.7% of the time RaptorX-assisted threading method and MetaPSICOV-assisted threading method predict the correct fold (TM-score > 0.5), respectively, as opposed to 75.7% of the baseline threading method. We also perform T-Test to examine whether the performance boost attained by the contact-assisted threading work using the high-quality RaptorX contacts and the moderate-quality MetaPSICOV contacts over the baseline threading method are statistically significantly better. Compared to the baseline threading method, RaptorX-assisted threading method is statistically significantly better at 95% confidence level with a *p*-value of 0.0001. However, MetaPSICOV-assisted threading method improves the threading performance compared to the baseline threading method, but the improvement is not statistically significant at 95% confidence level with a *p*-value of 0.07 (Appendix 2). Overall, the results demonstrate that the threading method using high-quality contact maps leads to better threading performance in terms of the TM-score of top ranked models and the percentage of time finding the correct overall folds.

Table 3.2 also shows that low-quality contacts such as mfDCA and PSICOV degrade the contact-assisted threading (referred to as rr_{mfDCA}-assisted threading and rr_{PSICOV}-assisted threading

respectively) performance with respect to the pure threading method by 0.04 and 0.05 TM-score, respectively, in terms of the accuracy of top ranked predicted models. In finding the correct overall folds, the performance of mfDCA-assisted threading method and PSICOV-assisted threading method drop by around 4% and 6%, respectively, compared to the baseline threading method. The deterioration of the performance of contact-assisted threading method using low-quality contact maps mfDCA and PSICOV are also statistically significant with *p*-values of 5.8e-08 and 2.9e-05, respectively with respect to the baseline threading method. Moreover, **Table 3.2** also shows that RaptorX-assisted threading attains statistically significantly better performance compared to the other three contact-assisted threading method, mfDCA-assisted threading, PSICOV-assisted threading, and MetaPSICOV-assisted threading, with *p*-values of 1.5e-11, 1.3e-09, and 0.0007 respectively. Results presented in **Table 3.2**, therefore, demonstrate that low-quality contacts degrade the threading performance compared to the baseline threading method as opposed to high-quality contacts, which boost the threading performance.

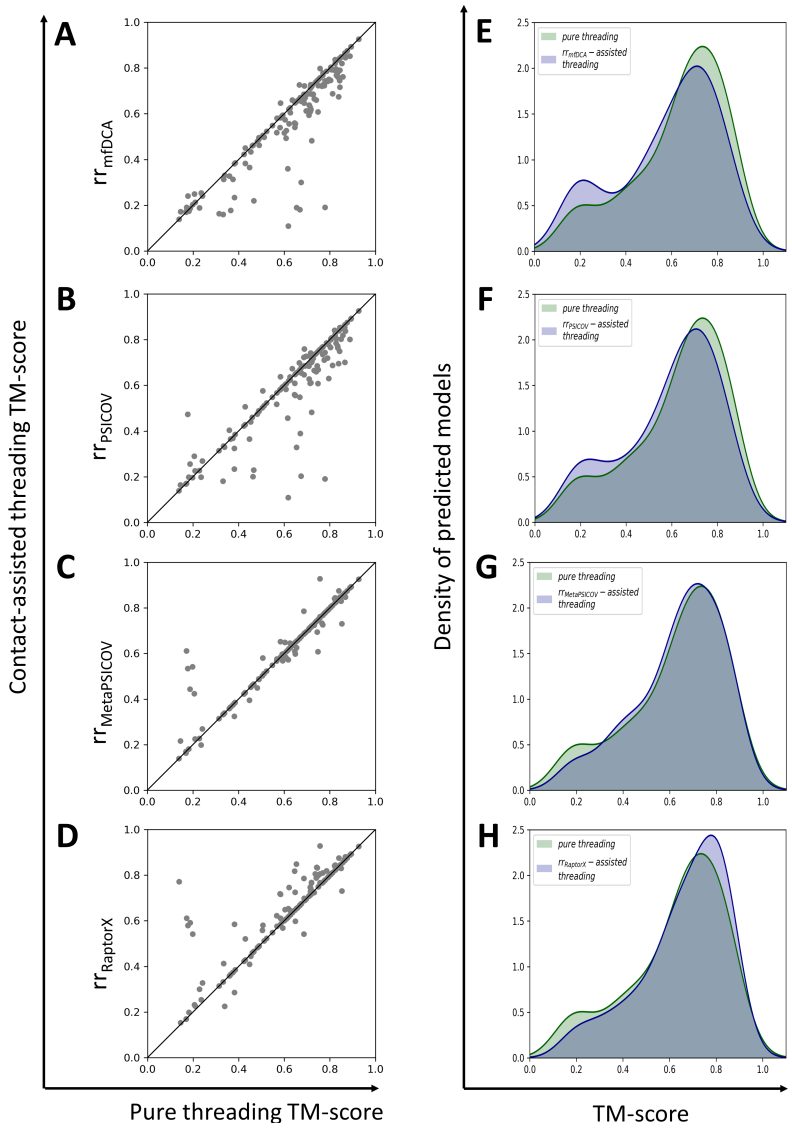


Figure 3.2 A head-to-head comparison of different contact-assisted threading methods and the baseline contact-free pure threading method on PSICOV150 dataset. (A) mfDCA-assisted threading method (referred to as rr_{mfDCA}) versus baseline threading method (referred to as Pure threading), (B) PSICOV-assisted threading method (referred to as rr_{PSICOV}) versus baseline threading method, (C) MetaPSICOV-assisted threading method (referred to as $rr_{MetaPSICOV}$) versus baseline threading method, (D) RaptorX-assisted threading method (referred to as $rr_{RaptorX}$) versus baseline threading method. Each point in each scatter plot represents joint TM-score of top ranked model predicted by baseline pure threading method and contact-assisted threading method. (E) TM-score distribution of top ranked models predicted by pure threading method versus mfDCA-assisted threading method (referred to as rr_{mfDCA}-assisted threading), (F) TM-score distribution of top ranked models predicted by pure threading method versus PSICOV-assisted threading method (referred to as rr_{PSICOV}-assisted threading), (G) TM-score distribution of top ranked models predicted by pure threading method versus MetaPSICOV-assisted threading method (referred to as $rr_{MetaPSICOV}$-assisted threading), (H) TM-score distribution of top ranked models predicted by pure threading based method versus RaptorX-assisted threading method (referred to as $rr_{RaptorX}$-assisted threading).

Figure 3.2 shows a head-to-head comparison of different contact-assisted threading methods with the baseline contact-free threading method in terms of the accuracy (TM-score) of top ranked models built from the first-ranked template. Each point in each scatter plot represents the joint TM-score of the top ranked model predicted by the pure threading and the contact-assisted threading method. In **Figures 3.2(A)** and **3.2(B)**, a majority of points are below diagonal lines, which clearly indicates that low-quality contacts (mfDCA and PSICOV) substantially degrade the threading performance compared to the baseline threading method. In contrast, we observe a slight performance improvement using the moderate-quality MetaPSICOV contacts (Figure 3.2(C)), where MetaPSICOV-assisted threading method improves the threading performance for 22 targets (out of 148) compared to the pure threading method. Moreover, **Figure 3.2(D)** shows a noticeable boost in the threading performance using the high-quality RaptorX contacts, where 35.8% of points (or 53 targets) are above the diagonal, indicating RaptorX-assisted threading method improves the TM-score of the top ranked model for 53 targets (out of 148) compared to the baseline threading method. Furthermore, we examine the TM-score distribution of the top ranked model predicted by contact-assisted threading methods and the baseline threading method in **Figures 3.2(E)** to **3.2(H)**. Specifically, in **Figures 3.2(E) and 3.2(F)**, the highest peak of the baseline threading method is larger as well as skewed towards the higher accuracy (right) side compared to mfDCA-assisted threading method and PSICOV-assisted threading method, respectively. These figures indicate that the threading method using each low-quality contact map predicts more models with low TM-score than the baseline threading method, resulting bimodality due to the second highest peak of the density of predicted models in the TM-score range [0,0.4], which deteriorates the overall threading performance. In contrast, in **Figures 3.2(G)** and **3.2(H)**, we see an opposite trend when we plot TM-score distribution of our threading approaches – one using contacts (MetaPSICOV,

and RaptorX respectively) of higher qualities while the other does not use the contact information. **Figure 3.2(G)** shows a slight performance improvement by MetaPSICOV-assisted threading method compared to the baseline threading method in that in TM-score range [0,0.3], MetaPSICOV-assisted threading method predicts fewer models as opposed to higher TM-score range, indicating using moderate contacts such as MetaPSICOV helps to improve the TM-score of a few targets compared to the purely threading based method. In **Figure 3.2(H)**, we see a significant performance boost by incorporating the high-quality RaptorX contacts in threading method. The highest peak of RaptorX-assisted threading method is larger as well as skewed towards the higher TM-score (right) side compared to the baseline threading method. In TM-score range [0.5,1.0], RaptorX-assisted threading method predicts more models as opposed to low TM-score range [0,0.5), indicating incorporating the high-quality RaptorX contacts helps to find the overall correct folds (TM-score > 0.5) for a number of targets where the purely threading based method fails. In summary, these results demonstrate that incorporating high-quality contacts in threading significantly boosts the threading performance in contrast with low-quality contacts, which degrades the performance.

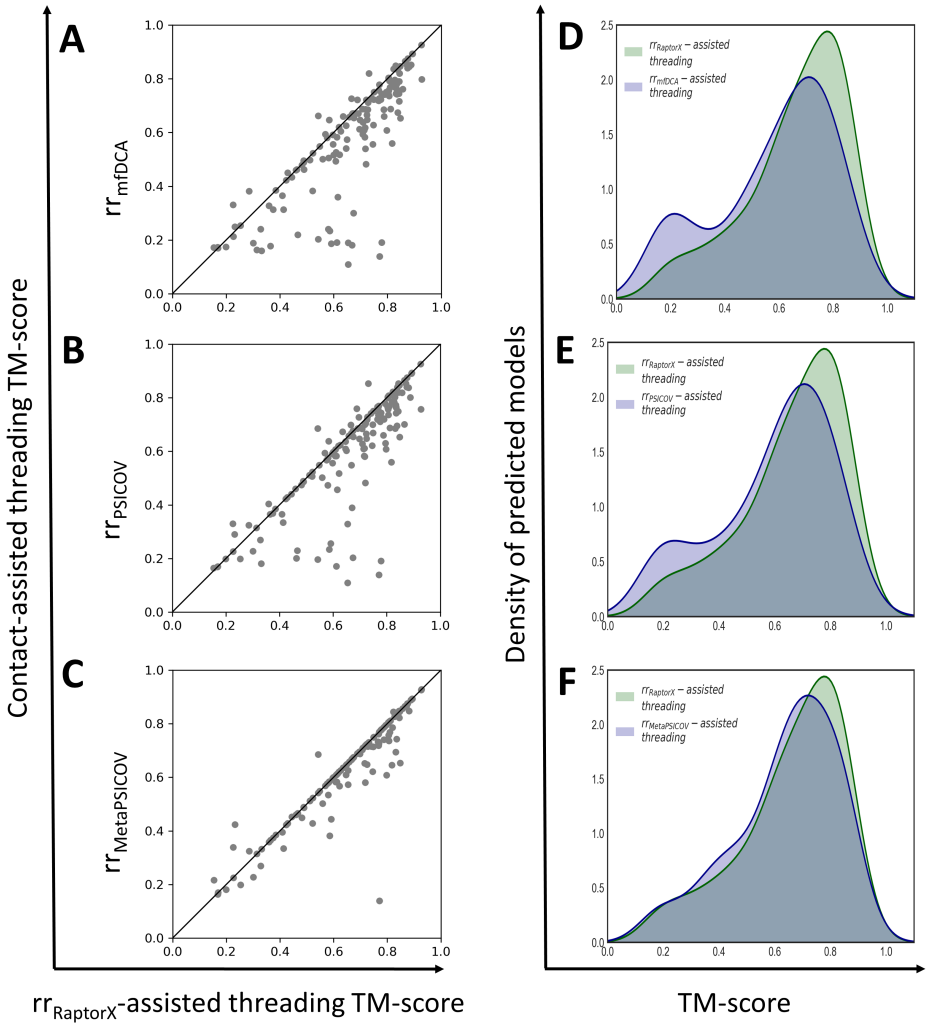


Figure 3.3 A head-to-head comparison of different contact-assisted threading methods and the baseline RaptorX-assisted threading method on PSICOV150 dataset. (A) mfDCA-assisted threading method (referred to as rr_{mfDCA}) versus baseline RaptorX-assisted threading method (referred to as $rr_{RAPTORX}$-assisted threading), (B) PSICOV-assisted threading method (referred to as rr_{PSICOV}) versus baseline RaptorX-assisted threading method, (C) MetaPSICOV-assisted threading method (referred to as $rr_{MetaPSICOV}$) versus baseline RaptorX-assisted threading method. Each point in each scatter plot represents joint TM-score of top ranked model predicted by baseline RaptorX-assisted threading and one of the other three contact-assisted threading methods respectively. (D) TM-score distribution of top ranked models predicted by RaptorX-assisted threading method versus mfDCA-assisted threading method (referred to as rr_{mfDCA}-assisted threading), (F) TM-score distribution of top ranked models predicted by RaptorX-assisted threading method versus PSICOV-assisted threading method (referred to as rr_{PSICOV}-assisted threading), (G) TM-score distribution of top ranked models predicted by RaptorX-assisted threading method versus MetaPSICOV-assisted threading method (referred to as $rr_{MetaPSICOV}$-assisted threading).

Since RaptorX-assisted threading method delivers the best threading performance we compare the performance of other three contact-assisted threading methods one by one against RaptorX-assisted threading acting as a control. In **Figure 3.3**, each point in each scatter plot represents the TM-score of the top ranked model predicted by RaptorX-assisted threading vs. one of the other three contact-assisted threading methods. **Figure 3.3(A)** shows a head-to-head comparison of mfDCA-assisted threading (referred to as rr_{mfDCA}) and RaptorX-assisted threading (referred to as $rr_{RaptorX}$-assisted threading method) in terms of the TM-score of the top ranked model, where a majority of the points (>70%) are below the diagonal line, RaptorX-assisted threading clearly outperforms mfDCA-assisted threading by a large margin. We see almost a similar trend in **Figure 3.3(B)** when we compare PSICOV-assisted threading (referred to as rr_{PSICOV}) with RaptorX-assisted threading method. Around 67% of points are below the diagonal line, which demonstrates the superior performance of RaptorX-assisted threading over PSICOV-assisted threading. In **Figure 3.3(C)**, we compare our contact-assisted threading approaches – one using the moderate-quality MetaPSICOV contacts (referred to as $rr_{MetaPSICOV}$) while the other using the high-quality RaptorX contacts. Around 28% more points are below the diagonal line, which illustrates the positive influence of the higher-quality contact maps (RaptorX) for the improved threading performance. In **Figures 3.3(D)** and **3.3(E)**, compared to both mfDCA- and PSICOV-assisted threading methods (referred to as rr_{mfDCA}-assisted threading and rr_{PSICOV}-assisted threading respectively), the highest peak of RaptorX-assisted threading (referred to as $rr_{RaptorX}$-assisted threading) is larger as well as skewed towards the higher TM-score side, indicating RaptorX-assisted threading finds more correct folds compared to others. Similarly, **Figure 3.3(F)** illustrates the TM-score distribution of MetaPSICOV-assisted threading (referred to as $rr_{MetaPSICOV}$-assisted threading) and RaptorX-assisted threading, where the highest peak of RaptorX-assisted threading

is still larger than MetaPSICOV-assisted threading as well as skewed towards the higher accuracy (right) side, illustrating the performance boost attained by the contact-assisted threading method using the high-quality contacts (RaptorX) over the moderate-quality contacts (MetaPSICOV). Overall, the high-quality contacts predicted from RaptorX leads to a statistically significantly better threading performance compared to that attained from the inferior-quality contacts predicted from other methods.

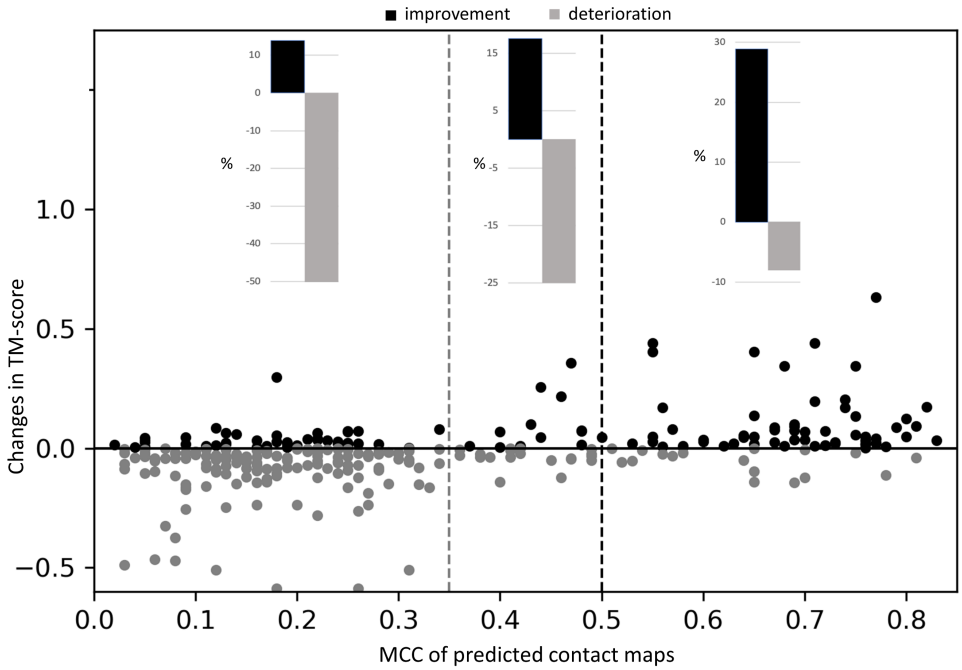


Figure 3.4 The relationship between changes in TM-score of contact-assisted threading methods compared to the pure threading method, and the MCC (Matthews correlation coefficient) of predicted contact maps, tested on PSICOV150. The dataset includes all four contact-assisted threading methods over 148 targets resulting in a total of 592 instances. Each point in the scatter plot represents the MCC of a predicted contact map and change in TM-score of a top ranked model predicted by various contact-assisted threading methods compared to pure threading. The dark points indicate improvement in TM-score (positive change in TM-score), whereas the grey points indicate performance deterioration (negative change in TM-score) compared to pure threading. The data points are separated based on the quality (measured by the MCC by considering residue pairs with contact probability of at least 0.5) of contacts: (i) 211 pairs with high quality contacts (MCC $\geq$ 0.5), (ii) 301 pairs with low-quality (MCC $<$ 0.35) contacts, and (iii) the twilight zone comprises of 80 pairs with moderate-quality contacts (0.35 $\leq$ MCC $<$ 0.5). Each bar plot represents the percentage of TM-score improvement and deterioration compared to pure threading.

Figure 3.4 shows how the quality of contact maps (measured by the MCC for residue pairs having the contact probability of at least 0.5) affects the contact-assisted threading performance as quantified by the changes in TM-score of top ranked models of contact-assisted threading methods compared to pure threading method considering all four contact-assisted threading methods over 148 targets resulting in a total of 592 instances. Each point in the scatter plot represents the MCC of a predicted contact map and the change in TM-score of a top ranked model predicted by the respective contact-assisted threading method compared to the pure threading method respectively. The data points have been separated based on the quality (MCC) of contacts: (i) 211 pairs with the high-quality contacts (MCC ≥ 0.5), (ii) 301 pairs with the low-quality (MCC < 0.35) contacts, and (iii) the twilight zone comprises of 80 pairs with the moderate-quality contacts (0.35 ≤ MCC < 0.5). The bar plot on the upper right corner of **Figure 3.4** shows that the contact-assisted threading performance is significantly improved for around 29% of the cases (out of 211), which is more than three times of the number of cases the performance is degraded, demonstrating that the high-quality contacts (MCC ≥ 0.5) boost the threading performance. In contrast, the bar plot on the upper left corner of **Figure 3.4** shows that the low-quality contacts degrade the contact-assisted threading performance for almost half of the points (out of 301) as opposed to only around 14% of the cases where the performance is improved, illustrating the adverse effect of the low-quality contacts (MCC < 0.35) on the contact-assisted threading performance. The bar plot on the upper middle section of **Figure 3.4** represents a twilight zone with the moderate-quality contacts where there is no significant difference in the number of cases the contact-assisted threading performance is improved (around 18%) or degraded (25%) out of 80 pairs. Furthermore, in **Appendix 3**, targets are grouped into three bins based on their sequence length to investigate how the quality of contacts affects the changes in TM-score of contact-assisted threading compared to the baseline pure

threading for different length bins. In **Appendix 3**, there are 34 targets of length <100 residues resulting in a total of 136 instances (Appendix 3 (A)), 47 targets of length [100,150] residues resulting in a total of 188 instances (Appendix 3(B)), and 67 targets of length >150 residues resulting in a total of 268 instances (Appendix 3(C)). For every length bin, we see a similar trend, contacts with the MCC ≥ 0.5 lead to the improved threading performance as opposed to contacts with the MCC < 0.35, which degrade the threading performance. Specifically, in the presence of high-quality contacts (MCC ≥ 0.5), Appendix 3(A) shows the highest threading performance boost of ~37% for the small proteins followed by ~29% for the proteins of length [100,150] residues (Appendix 3(B)) and ~24% for the proteins of length >150 residues (Appendix 3 (C)) compared to ~8% performance degradation in each distance bin. On the other hand, low-quality contacts (MCC < 0.35) degrade the threading performance for ~50% of the cases, irrespective of the protein length. Overall, the results show that contact maps with an MCC score of at least 0.5 lead to significantly better threading performance, whereas a score below 0.35 corresponds to a significant deterioration in the threading performance.

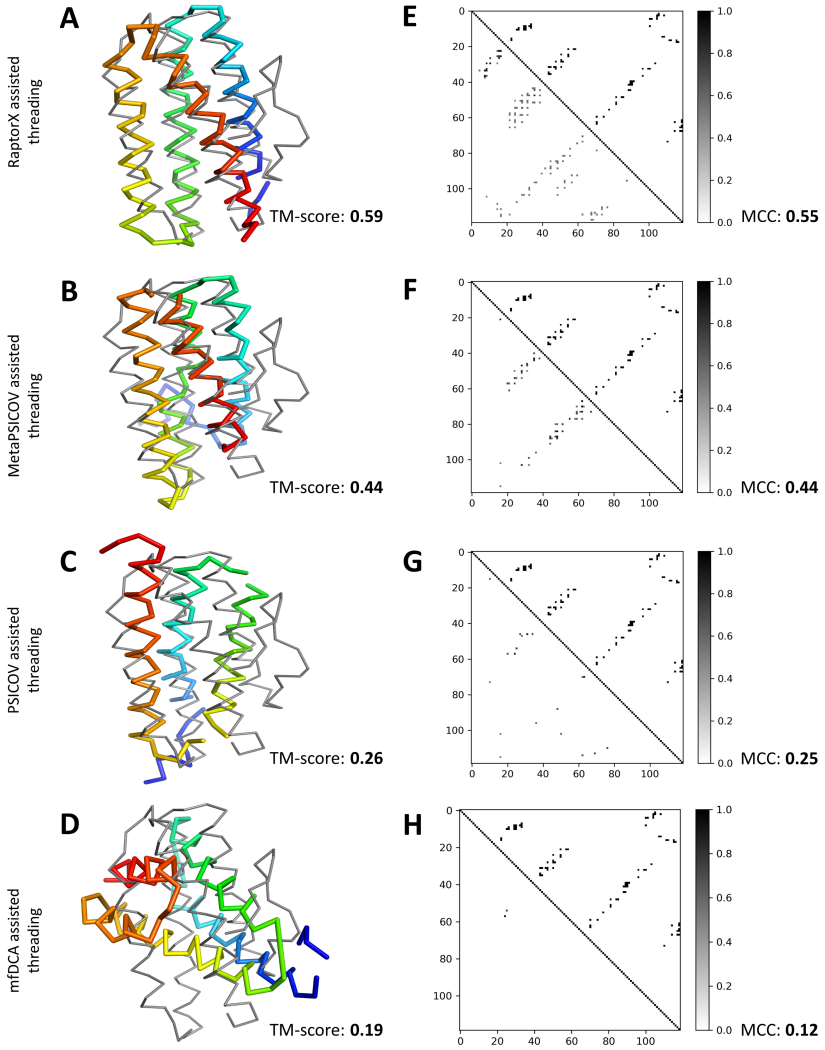


Figure 3.5 Representative example of contact-assisted threading with contact maps of diverse qualities on target 2mhrA. (A) Structural alignment between the top ranked model predicted by RaptorX-assisted threading (in thick rainbow) with a TM-score of 0.59 and the native structure of the target (in thin gray), (B) Structural alignment between top ranked model predicted by MetaPSICOV-assisted threading (in thick rainbow) with a TM-score of 0.44 and the native structure of the target (in thin gray), (C) Structural alignment between top ranked model predicted by PSICOV-assisted threading (in thick rainbow) with a TM-score of 0.26 and the native structure of the target (in thin gray), (D) Structural alignment between top ranked model predicted by mfDCA-assisted threading (in thick rainbow) with a TM-score of 0.19 and the native structure of the target (in thin gray). (E) native contact map (upper triangle) versus predicted contact map by RaptorX (lower triangle) with an MCC of 0.55. (F) native contact map (upper triangle) versus predicted contact map by MetaPSICOV (lower triangle) with an MCC of 0.44. (G) native contact map (upper triangle) versus predicted contact map by PSICOV (lower triangle) with an MCC of 0.25. (H) native contact map (upper triangle) versus predicted contact map by mfDCA (lower triangle) with an MCC of 0.12.

A representative example sheds some light on the impact of diverse quality of contacts on the threading performance, as shown in **Figure 3.5** for target 2mhrA from the PSICOV150 dataset that is a Hemerythrin HHE cation binding domain (David T. Jones et al. 2012) of 118 residues. **Figure 3.5(A)** shows RaptorX-assisted threading predicts the correct fold (the top ranked model predicted with a TM-score > 0.5) with a TM-score of 0.59 (and a root-mean-square deviation or RMSD of 4.8 Å) by using RaptorX predicted contacts with an MCC of 0.55 (**Figure 3.5(E)**). In contrast, **Figures 3.5(B)**, **3.5(C)** and **3.5(D)** reveal the inability of the other three contact-assisted threading methods in finding the correct fold due to the inferior-quality of contacts. In particular, threading using the moderate-quality MetaPSICOV contacts (MCC of 0.44, **Figure 3.5(F)**) predicts the 3D structure of the target with a TM-score of 0.44 (and **a** RMSD of 12.15 Å, Figure 3.5(B)) while as shown in **Figures 3.5(C)** and **3.5(D)**, TM-scores (and RMSD) are 0.26 (and 11.85 Å) and 0.19 (and 13.48 Å) for method using PSICOV contacts (an MCC of 0.25, Figure 3.5(G)) and mfDCA contacts (an MCC of 0.12, Figure3.5(H)) respectively.

3.4.3 Performance evaluation of contact-assisted threading with contact maps from top CASP13 groups

To further study the effect of the quality of contacts in the threading performance over challenging CASP13 targets, we employ contact-assisted threading using the top two officially ranked contact maps on CASP13 dataset, consisting of 20 full-length targets (and 32 domains) officially released so far with native structures, the same template library and the same nr sequence database, curated before CASP13 started on May 1, 2018, are used by all competing methods. For each target, we make the prediction for the full sequence without utilizing any domain information. After the prediction phase, the threading performance at the domain level is evaluated using the domain definitions provided by the official CASP13 assessors.

Table 3.3 Impact of high-quality contacts on the performance of contact-assisted threading on CASP13 dataset[a] based on average TM-score of top ranked models.

Target type	TripletRes- assisted threading (*p*-value*)[b]	RaptorX-Contact-assisted threading (*p*-value*)[c]	pure threading [d]
Full-length	0.457 (0.001)	0.449 (0.006)	0.403
Domain level	0.392 (0.0002)	0.387 (0.0008)	0.340

[a] CASP officially released 20 full-length targets in a total of 32 domains on December 2018.
[b] Zhang and coworkers participated in CASP13 with TripletRes (group number G032).
[c] Xu and coworkers participated in CASP13 with RaptorX-Contact (group number G498).
[d] pure threading method (without contacts).
*one sample T-Test's *p*-value of the TM-score difference compared to pure threading.

Table 3.3 shows incorporating the high-quality contacts statistically significantly outperforms the baseline pure threading method both for full-length targets and domain level targets. Over 20 full-length targets (and 32 domains), the mean TM-score of threading methods using TripletRes contacts (referred to as TripletRes-assisted threading) and RaptorX-Contact (referred to as RaptorX-Contact-assisted threading) are 0.457 (and 0.392) and 0.449 (and 0.387), respectively, as opposed to 0.403 (and 0.34) of the baseline pure threading method. Moreover, the performance improvement of TripletRes and RaptorX-Contact are also statistically significant with *p*-values of 0.001 (and 0.0002) and 0.006 (and 0.0008), respectively, for full-length (and domain level) targets. Additionally, **Appendix 4** shows how the threading performance is affected by the quality of contacts over 20 full-length targets. The set contains 40 instances, out of which, there are 29 instances with the high-quality contacts (MCC ≥ 0.5) as opposed to only one instance (TripletRes contact map for T1008) with an MCC < 0.35. The figure demonstrates how the high-

quality contacts with an MCC ≥ 0.5 lead to a significant threading performance boost (22 out of 29), illustrating contact maps with an MCC score of at least 0.5 lead to a significantly better threading performance. A case study shown in **Appendix 5** for CASP13 target T0954 of length 350 residues demonstrates the impact of the high-quality contacts on the threading performance. The baseline pure threading method attains a TM-score of 0.301 for the target, whereas the contact-assisted threading using the high-quality contact maps (MCC ≥ 0.5) from RaptorX-contact and TripletRes successfully predict the correct fold with TM-score ≥ 0.56, illustrating how the high-quality contacts with an MCC ≥ 0.5 boost the threading performance.

3.5 Conclusion

Chapter 2 demonstrates contact-assisted threading as a promising avenue for remote-homology protein modeling. However, the nature of the interdependence between the quality of contact maps and the contact-assisted threading performance remains elusive. Here, we present a large-scale analysis to study their mutual association by employing contact-assisted threading using contact maps of diverse qualities predicted from various contact predictors ranging from pure co-evolutionary methods (mfDCA and PSICOV) to hybrid approaches that combine sequence co-evolution and machine learning such as the classical neural network (MetaPSICOV) and the ultra-deep learning model (RaptorX). Experimental results demonstrate that contact-assisted threading method using the high-quality RaptorX contacts and the moderate-quality MetaPSICOV contacts outperform the baseline contact-free threading, whereas, the low-quality contacts predicted from mfDCA and PSICOV deteriorate the threading performance compared to the baseline pure threading method. Contact-assisted threading with the best-quality contacts (RaptorX) delivers the best threading performance that is statistically significantly better compared to the contact-free threading, demonstrating that the accurate (MCC ≥ 0.5) residue-residue contact

information is highly effective in boosting the threading performance as opposed to the low-quality (MCC < 0.35) contact information. This holds true even on the CASP13 dataset, where contacts with an MCC ≥ 0.5 lead to an improved threading performance. Collectively, our study shows that contact-assisted threading is effective in the presence of the high-quality (MCC ≥ 0.5) contact maps – indicating an evolving new direction for the improved protein threading that is likely to mature further with future advancements in contact prediction methods.

Chapter 4

DisCovER: distance- and orientation-based covariational threading for weakly homologous proteins

4.1 Abstract

Threading a query protein sequence onto a library of weakly homologous structural templates remains challenging, even when the sequence-based predicted contact or distance information is used. Contact- or distance-assisted threading methods utilize only the spatial proximity of the interacting residue pairs for the template selection and the alignment, ignoring their orientation. Moreover, existing threading methods fail to consider the neighborhood effect induced by the query-template alignment. We present a new distance- and orientation-based covariational threading method called DisCovER by effectively integrating information from inter-residue distances and orientations along with the topological network neighborhood of a query-template alignment. Our method first selects a subset of templates using the standard profile-based threading coupled with topological network similarity terms to account for the neighborhood effect, and subsequently performs distance- and orientation-based query-template alignment using an iterative double dynamic programming framework. Multiple large-scale benchmarking results on query proteins classified as weakly homologous from the Continuous Automated Model Evaluation (CAMEO) experiment and from the current literature show that our method outperforms several existing state-of-the-art threading approaches. The study also shows that the integration of the neighborhood effect with the inter-residue distances and orientations information synergistically contributes to the improved performance of DisCovER. Availability: https://github.com/Bhattacharya-Lab/DisCovER.

4.2 Introduction

We have so far discussed the utility of the predicted contact information as a valuable source of additional information for remote-homology threading, as well as the development of state-of-the-art contact-assisted threading methods such as EigenTHREADER and map_align. Very recently, CEthreader (Zheng, Wuyun, et al. 2019) and CATHER (Du et al. 2020) perform contact-assisted threading using contacts predicted by ResPre (Y. Li et al. 2019) and MapPred (Q. Wu et al. 2020), respectively, together with sequence profile-based features. DeepThreader (Zhu et al. 2018) goes one step further by incorporating the finer-grained distance information instead of contacts for boosting the threading performance (Jinbo Xu and Wang 2019).

While these methods exploit the predicted contact or distance information during threading often in conjunction with the sequential information, they do not consider two key factors that can further improve the threading accuracy. First, the recent extension of deep residual network architecture has resulted in accurate inter-residue orientations predicted from coevolution (J. Yang et al. 2020) in addition to distances, but none of the threading methods incorporate the orientation information. Second, most of the threading approaches do not include the effect of the residue pairs in the neighborhood of an aligned query-template residue pair. That is, they ignore the neighborhood effect induced by the query-template alignment.

In this chapter, we introduce a new distance- and orientation-based threading method DisCovER (Distance- and orientation-based Covariational threadER) that effectively integrates information from inter-residue distances and orientations along with the topological network neighborhood of a query-template alignment using an iterative double dynamic programming framework to greatly improve the threading template selection and the alignment. Experimental results show that our new method performs better than profile-based threading methods SparkX,

HHpred, CNFpred, MUSTER, PPAS, and pGenThreader; as well as state-of-the-art contact-assisted approaches CEthreader, map_align, EigenTHREADER, and CATHER, especially on weakly homologous proteins. At one of the most challenging threading situations, DisCovER yields better performance than the RaptorX server (Källberg et al. 2012; Zhu et al. 2018) participating in the Continuous Automated Model Evaluation (CAMEO) experiment (Haas et al. 2019) and employing the distance-based threading method DeepThreader (Zhu et al. 2018). DisCovER is freely available at https://github.com/Bhattacharya-Lab/DisCovER.

4.3 Materials and methods

4.3.1 Feature sets and inter-residue geometries

Our feature set includes both sequential and pairwise features for the query protein and templates. For a query protein, we generate sequence profiles based on multiple sequence alignments (MSA) (C. Zhang et al. 2020), and predict profile-based features including secondary structures, solvent accessibility, and backbone dihedral angles using SPIDER3 (Heffernan et al. 2017). We also predict inter-residue geometries including distances and orientations by feeding the MSAs into trRosetta (J. Yang et al. 2020). The predicted distance map is then binned into 9 segments at 1Å interval: <6Å, <7Å, …, and <14Å, by summing up probabilities for distance bins below specific distance thresholds. The predicted inter-residue orientations include two dihedral angles (ω and θ), both binned into 24 evenly spaced segments with a bin width of 15° each, and one planar angle (φ), binned into 12 evenly spaced segments with a bin width of 15° each. All distance orientation bins have associated likelihood values for the query protein predicted by trRosetta. For the templates, we use structure-derived profiles, extract secondary structures and solvent accessibility using DSSP (Wolfgang Kabsch and Sander 1983), and compute backbone

dihedral angles, inter-residue distance maps, and the orientation information including ω, θ dihedrals and φ angle from the structure.

4.3.2 Geometry-based scoring of a query-template alignment

DisCovER scores a query-template alignment as follows:

$$Z_{final} = Z_{without\_geometry} + Z_{with\_geometry} \quad (1)$$

where Z_{final} is the normalized alignment score for selecting the top-ranked template, $Z_{without\_geometry}$ is the normalized alignment score based only on the profile information with the neighborhood effect, and $Z_{with\_geometry}$ is the normalized alignment score using the profile information, the neighborhood, and the inter-residue geometries including distances and orientations. In the following, we describe each term in detail.

Stage 1 Scoring profile-based alignment with neighborhood effect: A profile-based query-template alignment is scored for aligning the ith residue of the query and the jth residue of the template similar to our recent work (Bhattacharya and Bhattacharya 2019a) as follows:

$$S_{without\_geometry}(i,j) = \sum_{k=1}^{20} \frac{\left(f_c(i,k) + f_d(i,k)\right) L_t(j,k)}{2} + c_1\delta\left(SS_i, SS_j\right) + c_2 \sum_{k=1}^{20} \left(f_s(j,k) + L_q(i,k)\right) + c_3 E\left(SA_i, SA_j\right) + c_4 E\left(\phi_i, \phi_j\right) + c_5 E\left(\psi_i, \psi_j\right) + c_6 M\left(AA_i, AA_j\right) + c_7 \quad (2)$$

where the first term defines the sequence profile-profile alignment. $f_c(i,k)$ and $f_d(i,k)$ define the frequency of the kth residue at the ith query position of the MSA for "close" and "distant" homologous sequences, respectively. The frequency is determined using the Henikoff weighting scheme (Henikoff and Henikoff 1994). L_t is the log-odds profile of the template for the kth residue at the jth position, which is obtained by PSI-BLAST (Altschul et al. 1997) with an E-value of

0.001. The next term measures the consistency between the predicted and the observed three-state secondary structures, such that the function δ returns 1 if two variables are matched and -1 otherwise. The next term is the agreement between the structure-derived profiles (f_s) of the kth residue at the jth position of the template structure and the sequence profile (L_q) of the kth residue at the ith position of the query sequence. The function E in the next three terms is defined by: $E(x_i, x_j) = (1 - 2|x_i - x_j|)$, where the variables are the predicted and the observed values of relative solvent accessibility (SA) and backbone dihedral angles (ϕ and ψ) of the ith position of the query and the jth position of the template, respectively. The seventh term corresponds to the match between the hydrophobic residues of the query and the template. c is the weighting parameter adopted from (Bhattacharya and Bhattacharya 2019a).

To further improve the sensitivity of profile-based alignments, we borrow ideas from comparative network analysis. We adopt an approach similar to that was originally used in the IsoRank network alignment algorithm (Singh, Xu, and Berger 2008) and very recently adopted in network-based structural alignment of RNA sequences (Chen et al. 2019), in which two nodes in different networks are more likely to be aligned to each other if their neighbors are also aligned well to one another. This results in a similarity diffusion scheme to compute the agreement between the networks, leading to an improved alignment. Following similar principles, we integrate the *connected similarity*, attempting to estimate the topological agreement between the query and the template by capturing the similarity between the neighborhoods of two residues.

Connected similarity (S_c) is based on the principle that one query-template residue pair is likely to be aligned if their neighboring residues are also aligned. It is calculated for the residue pair (i, j) as follows:

$$S_c(i,j) = \frac{1}{2}\left(S_{without_{geometry}}(i-1, j-1) + S_{without\_geometry}(i+1, j+1)\right) \quad (3)$$

Connected similarity (S_c) is then added to the profile-based alignment score as follows:

$$S_{without\_geometry}(i,j) += S_c(i,j) \tag{4}$$

We use the Needleman-Wunsch (Needleman and Wunsch 1970) global dynamic programming to score every query-template alignment. To select the top-fit templates, we compute the $Z_{without\_geometry}$ based on the raw alignment score $S_{without\_geometry}$ to assess the quality of each query-template alignment as follows:

$$Z_{without\_geometry} = \frac{(S'_{without\_geometry} - < S'_{without\_geometry} >)}{\sqrt{< S'^{2}_{without\_geometry} > - < S'_{without\_geometry} >^2}} \tag{5}$$

where $S'_{without\_geometry}$ is the larger one of the raw alignment score $S_{without\_geometry}$ divided by the full alignment length (including the query and the template ending gaps) and the partial alignment length (excluding the query ending gaps). $< S'_{without\_geometry} >$ is the average $S'_{without\_geometry}$ across all the templates in the template library. A subset of fifty top-scoring templates is selected for the next stage.

Stage 2 Scoring distance- and orientation-based alignment: A similarity score is calculated for each row of the query (Q) and the template (T) distance maps as follows:

$$S^{row-row}_{with\_geometry}[(i,i'),(j,j')] = \sum_{k\in\{<6Å,<7Å,\ldots,<14Å\}} w_k \times \mathrm{Q}^{d}_{\mathrm{k}}[i][i'] \times \mathrm{T}^{d}_{\mathrm{k}}[j][j'] \times w_{sep}(s) \times G(0, sd, sep) + \sum_{k\in\{15°,30°,\ldots,360°\}} \mathrm{Q}^{\omega}_{\mathrm{k}}[i][i'] \times \mathrm{T}^{\omega}_{\mathrm{k}}[j][j'] \times w_{sep}(s) + \sum_{k\in\{15°,30°,\ldots,360°\}} \mathrm{Q}^{\theta}_{\mathrm{k}}[i][i'] \times \mathrm{T}^{\theta}_{\mathrm{k}}[j][j'] \times w_{sep}(s) + \sum_{k\in\{15°,30°,\ldots,180°\}} \mathrm{Q}^{\varphi}_{\mathrm{k}}[i][i'] \times \mathrm{T}^{\varphi}_{\mathrm{k}}[j][j'] \times w_{sep}(s) \tag{6}$$

where the first term calculates the similarity between the predicted distance map of the query and the true distance map of the template at a given distance threshold of k, where $k \in$ {<6Å, <7Å, …, <14Å}. $\mathrm{Q}^{d}_{\mathrm{k}}[i][i']$ is the predicted likelihood value of the residue pair i and i' of the query to be

within a distance threshold of kÅ; $T_k^d[j][j']$ is a Heaviside step function that has a value 1 if the residue pair j and j' of the template is within the distance threshold of kÅ and 0 otherwise; w_k is the corresponding weight parameter adapted from the literature (Du et al. 2020) with $w_k = \frac{1}{1+exp^{|k-10|}}$; $w_{sep}(s)$ is the weight of the minimum of sequence separation of query residues and template residues defined as $w_{sep}(s)$ = 0.75 for s = 5 and $\log_{10}(1+s)$ for s ≥ 6, similar to other studies (Ovchinnikov et al. 2017), and $s = \min(|i-i'|, |j-j'|)$; $G(0, sd, sep)$ is a zero-mean Gaussian function, which is also adopted from the literature (Ovchinnikov et al. 2017) and defined as $exp(-sep^2/(2\,sd^2))$, where sep is an absolute difference of the sequence separation of query residues and the sequence separation of template residues, and sd or standard deviation is a function of the smaller of the sequence separation of query residues and the sequence separation of template residues. The next three terms calculate dihedral (ω, θ) and planar (φ) angles similarities between the query and template residue pair. We treat the orientation information similar to distances and compute the similarity between the predicted ω, θ dihedrals or φ angle of the query and the corresponding true angles of the template at a specific angle bin of k, where k ∈{15°,30°,…,360°} for the ω and θ dihedrals, and k ∈{15°,30°,…,180°} for the φ angle. Analogous to distances, $Q_k^{\omega}[i][i']$, $Q_k^{\theta}[i][i']$, and $Q_k^{\varphi}[i][i']$ are the predicted likelihood values of the residue pair i and i' of the query to be within an angle bin of k° for the angles ω, θ, and φ, respectively. Similarly, $T_k^{\omega}[j][j']$, $T_k^{\theta}[j][j']$, and $T_k^{\varphi}[j][j']$ are the box functions corresponding to the angles ω, θ, and φ, respectively, having values 1 if a residue pair j and j' of the template is within an angle bin of k° and 0 otherwise. $w_{sep}(s)$ is the weight term described before.

Our double dynamic programming framework for computing the optimal alignment score between the query and each of the fifty top-scoring templates selected from the previous stage

comprises of two dynamic programming steps. In the first step, we perform row-by-row comparisons between the query and the template. Dynamic programming is used to find the alignment for the two rows being matched which maximizes the composite distance- and orientation-based alignment score described in Equation 6. These scores are stored in a similarity matrix and are used to obtain the optimal alignment by using the Smith-Waterman (Smith and Waterman 1981) algorithm. At this step, however, the scores for individual row-row comparisons are overestimated since the alignments for each pair are independently computed in the first step. We subsequently update the similarity matrix using a second step based on the current alignment by employing a second dynamic programming. Such an iterative updating strategy is originally proposed in (Taylor 1999) and later adopted in (Ovchinnikov et al. 2017) although our score is quite different. After obtaining the optimal alignment from the similarity matrix, the profile score and the gap-score are re-calculated to compute the raw alignment score ($S_{with\_geometry}$). The similarity score for each query-template pair is normalized using $Z_{with\_geometry}$ as follows:

$$Z_{with\_geometry} = \frac{\left(S'_{with\_geometry} - < S'_{with\_geometry} >\right)}{\sqrt{< S'^{2}_{with\_geometry} > - < S'_{with\_geometry} >^{2}}} \tag{7}$$

where $S'_{with\_geometry}$ is the larger one of the raw alignment score $S_{with\_geometry}$ divided by the full alignment length (including the query and the template ending gaps) and the partial alignment length (excluding the query ending gaps). $\langle \ldots \rangle$ denotes the average $S'_{with\_geometry}$ of all the top-scoring templates.

Building full-length 3D models: After selecting the first-ranked template using Equation (1), we use MODELLER (V9.22) (Webb and Sali 2014) to generate the full-length 3D model of a query protein using the associated query-template alignment. In addition to employing the standard automodel() class of MODELLER for model building purely through the satisfaction of spatial

restraints from the query-template alignment, we also experiment with model building with additional restraints from predicted distances, orientations, and secondary structures by redefining the automodel.special_restraints() class. Specifically, we feed bounded harmonic restraints for the predicted distance thresholds corresponding to the 9 distance bins used in query-template alignments with a minimum likelihood cutoff of 0.85 using the physical.xy_distance() function, bounded harmonic restraints for the predicted orientation information derived from the highest likelihood bins having the minimum likelihood cutoff of 0.85 with the φ angle using physical.angle() function and ω, θ dihedrals using physical.dihedral() function, and secondary structure restraints for realizing the predicted secondary structure using the secondary_structure() module. Of note, all of these additional restraints are integrated to the list of spatial restraints derived from the query-template alignment to instruct MODELLER to satisfy them as best as it can.

4.3.3 Benchmark datasets, methods to compare, template libraries used, and threading performance evaluation

To evaluate the remote-homology threading performance, we benchmark our new method DisCovER using targets from the Continuous Automated Model Evaluation (CAMEO) experiments consisting of 117 proteins classified as "hard" (Haas et al. 2019), released between 8 December 2018 and 1 June 2019 having length between 50 and 500 residues (range is 51 to 487). On this dataset, DisCovER is compared against profile-based threading methods such as SparkX (Y. Yang et al. 2011), CNFpred (Ma et al. 2012; 2013), MUSTER (S. Wu and Zhang 2008), PPAS (S. Wu and Zhang 2007), and pGenThreader (Lobley, Sadowski, and Jones 2009); as well as state-of-the-art contact-assisted methods including CEthreader (Zheng, Wuyun, et al. 2019) utilizing ResPRE-predicted contact maps (Y. Li et al. 2019) and EigenTHREADER (Buchan and Jones

2017) utilizing DMPfold-predicted maps (Greener, Kandathil, and Jones 2019). The template libraries for DisCovER, CEthreader, EigenTHREADER, SparkX, MUSTER, PPAS, and pGenThreader are generated from the same set of 70,670 templates downloaded from https://zhanglab.ccmb.med.umich.edu/library/ (J. Yang et al. 2015), curated before the release of the CAMEO targets. The template library for CNFpred is downloaded from http://raptorx.uchicago.edu/download/. For all methods, we evaluate the threading performance by comparing the top-ranked full-length predicted 3D models, built using the standard automodel() class of MODELLER from the query-template alignment, against the experimental structures of the target proteins using the TM-score metric (Y. Zhang and Skolnick 2004), which ranges from 0 to 1 with a higher score indicating a better performance and the TM-score >0.5 indicating the attainment of the correct overall fold (Jinrui Xu and Zhang 2010). Of note, DisCovER utilizes distances and orientations predicted from trRosetta (J. Yang et al. 2020), which uses a training set collected from a snapshot as of 1 May 2018, older than the CAMEO test set used here. DisCovER also relies on the secondary structure predictor SPIDER3 (Heffernan et al. 2017) which uses a much older training dataset and thus independent of the CAMEO test set. We collect publicly available multiple sequence alignments (MSAs) independently generated using non-overlapping protein sequence databases from https://yanglab.nankai.edu.cn/trRosetta/benchmark/ to feed into trRosetta and SPIDER3. Furthermore, the template library used in DisCovER excludes any templates released after starting of the CAMEO experiments (8 December 2018), free from any overlap. Finally, the "hard" target difficulty classification of the CAMEO test set defined by CAMEO (Haas et al. 2019) warrants their non-overlapping and weakly homologous nature, thereby enabling us to focus on difficult targets in which existing methods have limitations (Zhu et al. 2018).

The definition of "hard" can be made even more stringent by requiring that the TM-score of the HHpredB server (Söding 2005) participating in the CAMEO to be less than 0.5. This reduces the number of targets to 60. This harder set simulates one of the most challenging threading situations while enabling a comparison between DisCovER and the distance-based threading method DeepThreader (Zhu et al. 2018). DeepThreader method is not publicly available, but the RaptorX server (Källberg et al. 2012; Zhu et al. 2018) participates in the CAMEO and employs the DeepThreader method according to the CAMEO assessment paper (Haas et al. 2019). While RaptorX uses PDB90 as the template database and builds 3D models using Rosetta (Jinbo Xu and Wang 2019) as opposed to MODELLER, we compare the performance of DisCovER on this common set of 60 very hard targets to that of DeepThreader after downloading the predictions submitted by the RaptorX server from the official website of CAMEO (https://cameo3d.org/) and computing their TM-scores.

We are unable to directly compare DisCovER with two other state-of-the-art contact-assisted methods map_align (Ovchinnikov et al. 2017) and CATHER (Du et al. 2020) on the CAMEO test set, because map_align is too computationally expensive to run locally given our limited computational resources and CATHER is only available as a webserver and thus not suitable for a large-scale benchmarking. However, the published work of CATHER reports the mean TM-scores of 3D models predicted using various threading methods including CATHER, map_align, EigenTHREADER, HHpred (Söding 2005), SparkX, and MUSTER over a dataset of 480 targets including 304 easy, 45 medium, and 131 hard targets with a pairwise sequence identity <25% and the length ranging from 50 to 500 residues (Du et al. 2020). We use this set to compare DisCovER against CATHER and map_align by running DisCovER locally after excluding templates with the sequence identity >30% to the query proteins, and comparing its average

performance against the reported results of CATHER and map_align, in addition to the other threading methods presented.

4.4 Results and discussion

4.4.1 Performance on 117 hard targets from CAMEO

Table 4.1 Benchmark results of various threading methods on 117 hard targets from CAMEO. Values in bold represent the best performance.

Methods	TM-score	*p*-value*	#Correct folds†
pGenThreader	0.423	1.1E-08	46
PPAS	0.456	9.3E-05	54
MUSTER	0.459	0.0006	54
SparkX	0.461	0.0003	57
CNFpred	0.464	0.004	56
EigenTHREADER	0.461	1.9E-05	58
CEthreader	0.483	0.029	61
DisCovER	**0.505**	-	**67**

*Column "*p*-value" represents one sample t-test's p-value of the TM-score difference compared to DisCovER.
†Column "#Correct folds" represents the number of models with TM-score >0.5

Over the 117 hard targets from the CAMEO, our distance- and orientation-based threading method DisCovER performs better than the two contact-assisted threading methods as well as all five profile-based approaches. As shown in **Table 4.1**, DisCovER attains a mean TM-score of 0.505, which is higher than the next best contact-assisted threading method CEthreader (0.483) and the best among profile-based threading methods CNFpred (0.464). The performance improvement for DisCovER is statistically significant at 95% confidence level compared to all other methods. DisCovER also predicts the highest number of correct folds (TM-score >0.5) with

a success rate of 57.3%, which is ~5% higher than the success rate of CEthreader and ~9% higher than that of CNFpred. Of note, the best contact-assisted threading method CEthreader falls short of achieving a mean TM-score of 0.5, whereas DisCovER exceeds this criterion.

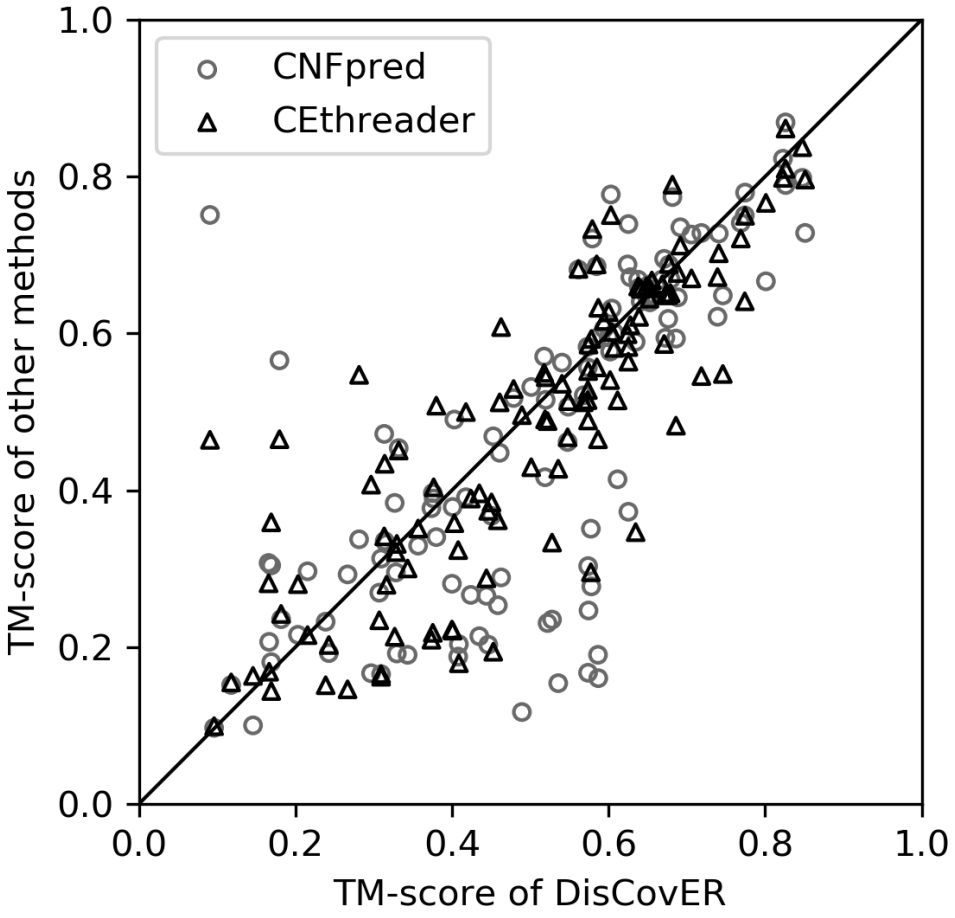


Figure 4.1 Head-to-head performance comparison between DisCovER (x-axis) vs. CEthreader and CNFpred (y-axis) on 117 hard targets from CAMEO.

Figure 4.1 shows the head-to-head comparison between DisCovER and the best contact-assisted threading method CEthreader and the best profile-based threading method, CNFpred. DisCovER attains higher TM-score for 74 and 66 targets compared to CEthreader and CNFpred, respectively. In summary, the advantage of DisCovER in threading remote-homology proteins over the others is significant.

4.4.2 Contribution of individual components

Table 4.2 Contribution of individual features to DisCovER performance.

Methods	Mean TM-score		
	CAMEO dataset		MUSTER dataset
	117 hard targets	60 very hard targets	86 easy and hard targets
DisCovER$^{\text{No neighborhood, No geometry}\dagger}$	0.470	0.329	0.404
DisCovER$^{\text{No geometry}\dagger}$	0.472	0.335	0.405
DisCovER$^{\text{No distance}}$	0.488	0.349	0.420
DisCovER$^{\text{No orientation}}$	0.503	0.372	0.416
DisCovER	0.505	0.376	0.432
TM-align‡	0.636	0.542	0.562

†geometry includes distance and orientation information.
‡TM-align results are included as a reference.

In addition to profile-based alignments, DisCovER incorporates a) the neighborhood effect, b) the distance, and c) the orientation. To investigate the contributions of each of these components to the DisCovER performance, we perform ablation studies on the CAMEO test set as well as another independent dataset from the MUSTER paper (S. Wu and Zhang 2008) by gradually removing each component from the full-fledged DisCovER method one at a time and evaluate the performance. The same template library, sequence databases, and the same set of features are used in all cases to generate the query-template alignments that are then fed into the standard automodel() class of MODELLER to generate the 3D structures. While the study on the CAMEO test set exclusively focuses on hard and very hard targets, the study on the MUSTER dataset comprises of targets from easy to hard categories. Specifically, the original MUSTER test set contains 500 targets, including 203 easy and 255 hard targets as defined in the published work of MUSTER, with a pairwise sequence identity <25% and the length ranging from 50 to 633 residues.

For the purpose of our study, targets with a sequence identity >40% to the training set of trRosetta are excluded following the literature (Zheng, Wuyun, et al. 2019) in order to make the test set free from any overlap, thus reducing the number of targets in the MUSTER test set to 86. Moreover, we exclude templates with a sequence identity >20% to the query proteins or detectable by BLAST (Altschul et al. 1997) with an E-value <0.05 to remove the homologous templates, a practice adopted from the literature (S. Wu and Zhang 2008).

As reported in **Table 4.2**, the full-fledged DisCovER attains the best performance (mean TM-score of 0.505) than any of its ablated variants over the 117 hard targets from the CAMEO. Without orientations, the mean TM-score decreases to 0.503, which is further decreased to 0.488 without the distance information. The performance of the ablated variant of DisCovER that only incorporates distances but no orientations outperform the variant that only incorporates orientations but no distances. It is interesting to note that the average performance of either variant is still better than state-of-the-art contact-assisted methods CEthreader and EigenTHREADER, indicating that the incorporation of either distances or orientations information in DisCovER is sufficient to outperform top contact-assisted threading methods such as CEthreader. When both the distance and the orientation terms are excluded, the mean TM-score drops to 0.472, but it is still better than the top profile-based threading method CNFpred. Upon further exclusion of the neighborhood effect, the mean TM-score slightly reduces to 0.470. The results demonstrate that all components contribute to the improved performance of DisCovER, with distances and/or orientations information having significant contribution. We note that DisCovER incorporates distances and orientations information only in stage 2 for computing the optimal alignment score between the query and each of the fifty top-scoring templates selected from stage 1 based on profile-based threading in combination with the neighborhood effect. That is, the distance- and

orientation-based alignment further improves the alignment accuracy over the top template recognition performance achieved by profile-based threading with the topological network neighborhood. The trend remains very similar considering the subset of 60 very hard CAMEO targets, although the exclusion of the neighborhood effect results in a noticeable performance decline in this case (a mean TM-score drop from 0.335 to 0.329) that shows the effectiveness of incorporating the topological network neighborhood for challenging threading scenarios.

The ablation study on the MUSTER dataset amplifies the importance of incorporating the orientation information and the complementarity of distances and orientations. As shown in **Table 4.2** (last column), the ablated variant of DisCovER that only incorporates orientations but no distances (mean TM-score 0.420) outperform the variant that only incorporates distances but no orientations (mean TM-score 0.416). That is, the incorporation of the orientation information, which is a major contribution of this work, significantly contributes to the threading performance. It is also interesting to note that the full-fledged DisCovER that integrates both distances and orientations attains a noticeably higher mean TM-score of 0.432 than its ablated variants using distances or orientations alone, demonstrating the complementarity of distances and orientations information. We notice that in the MUSTER dataset, which contains a mix of easy and hard targets, there is a positive but minor contribution of the neighborhood effect. This is consistent with our earlier observation in the CAMEO dataset that the topological network neighborhood contributes the most for very hard targets. Overall, all components contribute to the improved threading performance of DisCovER with the orientation and the distance information playing significant and complementary roles.

Table 4.2 also reports a reference oracle method that uses TM-align(Y. Zhang and Skolnick 2005b) to structurally align the experimental structure of the query protein with each of the

templates in the template library to select the structurally closest template and the resulting optimal query-template alignment is then fed into the standard automodel() class of MODELLER to generate the 3D structures. Not surprisingly, the TM-align-based oracle achieves much better performance with a mean TM-score > 0.5 even for the very hard CAMEO targets (**Table 4.2**, last row), indicating that there is still a large room for improvement.

4.4.3 3D model building using MODELLER from query-template alignment with additional restraints

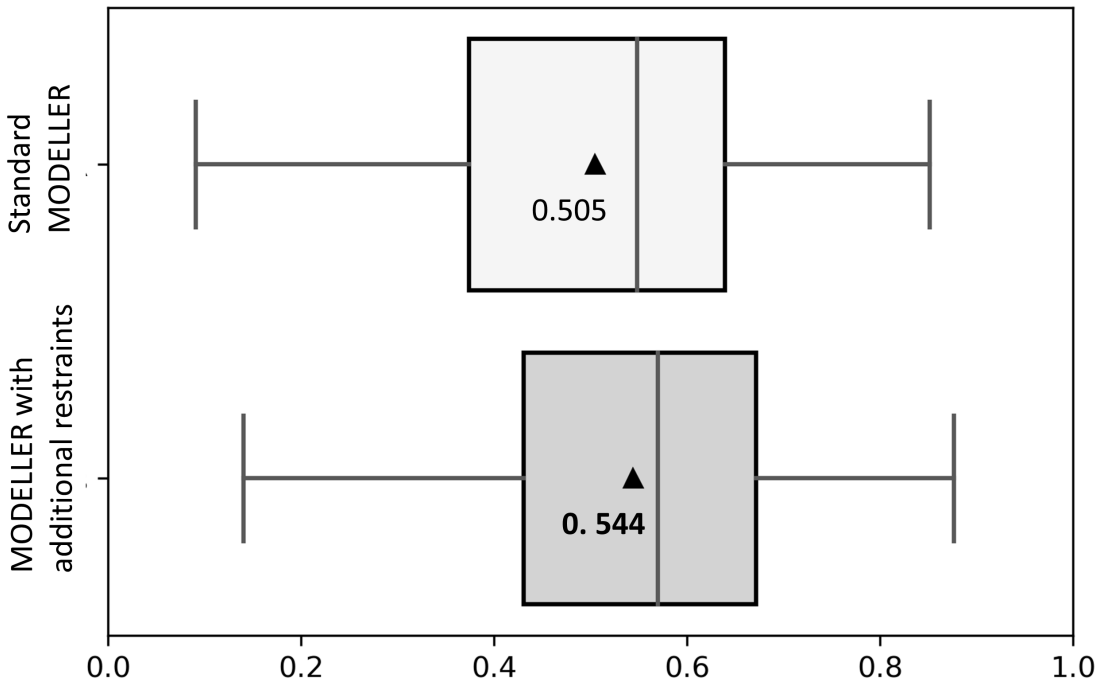


Figure 4.2 3D model building performance using standard MODELLER and MODELLER with additional restraints for 117 hard targets from CAMEO. The x-axis represents TM-score. The mean TM-scores are represented by triangles.

To examine whether the additional information used in DisCovER for threading template selection and alignment can further improve the full-length 3D model building, we compare the standard automodel() class of MODELLER that builds 3D models using spatial restraints collected from query-template alignments to another approach using MODELLER that integrates additional

restraints from predicted distances, orientations, and secondary structures. As shown in **Figure 4.2**, the mean TM-score attained by MODELLER with additional restraints is 0.544 over the 117 hard targets from the CAMEO, better than that of the standard MODELLER. MODELLER with additional restraints also attains 75 correct folds while shifting the TM-score distribution towards the higher accuracy. That is, 3D model building using MODELLER from query-template alignments with additional restraints can be an effective use of the additional information used in DisCovER. We follow this model building approach henceforth.

4.4.4 Performance comparison with CAMEO server RaptorX employing DeepThreader

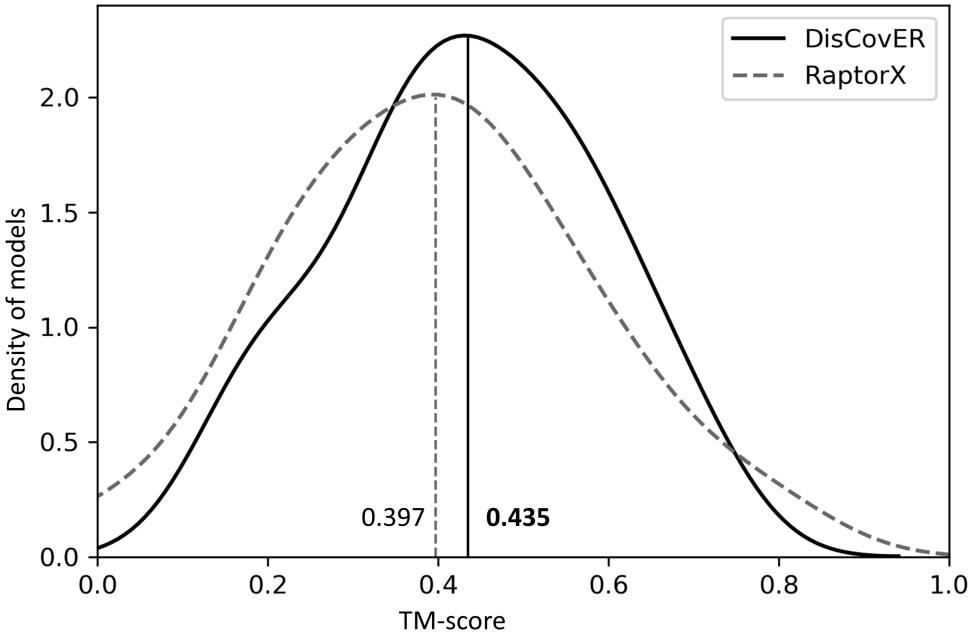


Figure 4.3 TM-score distribution of DisCovER and RaptorX on 60 very hard targets from CAMEO.

We compare the performance of DisCovER to that of DeepThreader (Zhu et al. 2018) on the 60 very hard targets from the CAMEO after downloading the predictions submitted by the CAMEO server RaptorX (Källberg et al. 2012; Zhu et al. 2018), which, according to the CAMEO assessment paper (Haas et al. 2019), employs the DeepThreader method, otherwise not publicly

available to run. As shown in Figure 4.3, DisCovER achieves a mean TM-score of 0.435, outperforming RaptorX (mean TM-score of 0.397), while skewing the overall TM-score distribution towards the higher accuracy. DisCovER attains higher TM-scores for 36 targets (60%) compared to RaptorX. In summary, DisCovER attains better overall performance over the 60 very hard targets from the CAMEO for which HHpredB has TM-score less than 0.5. That is, DisCovER outperforms DeepThreader at one of the most challenging threading situations.

4.4.5 Performance on 480 targets from CATHER

The performance of DisCovER is further benchmarked against recent contact-assisted threading methods: CATHER (Du et al. 2020) and map_align (Ovchinnikov et al. 2017) by running DisCovER over 480 targets containing 304 easy, 45 medium, and 131 hard targets used in CATHER and directly comparing the mean TM-score with the results reported in the published work of CATHER (Du et al. 2020) over the same set.

Table 4.3 Benchmark results on 480 targets from CATHER. Mean TM-score are reported.

Type	HHpred	SparkX	MUSTER	map_align	EigenTHREADER	CATHER	DisCovER
Easy	0.691	0.692	0.728	0.678	0.682	0.747	**0.750**
Medium	0.376	0.456	0.500	0.418	0.442	0.543	**0.615**
Hard	0.327	0.349	0.359	0.383	0.386	0.456	**0.551**
All	0.562	0.576	0.606	0.573	0.579	0.649	**0.683**

Note: Except DisCovER, the results of other methods are taken from the reported results of CATHER. Values in bold represent the best performance.

As shown in **Table 4.3**, DisCovER outperforms all the competing methods by attaining a mean TM-score of 0.683 over all targets in the dataset, which is about ~0.03 TM-score better than the next best method CATHER. The performance gap between DisCovER and CATHER and the other competing methods is more pronounced with the increasing target difficulty. Specifically,

while DisCovER outperforms the next-best method CATHER by 0.003 TM-score points for easy targets, the performance gap between DisCovER and CATHER steadily increases to ~0.07 and ~0.1 TM-score points for medium and hard targets, respectively, underscoring the advantage of DisCovER for weakly homologous protein targets. In particular, DisCovER significantly outperforms CATHER and map_align by attaining a mean TM-score of 0.551 as opposed to 0.456 of CATHER and 0.383 of map_align over 131 hard targets. DisCovER also greatly outperforms the reported mean TM-scores of other profile-based methods including HHpred (0.327), SparkX (0.349), and MUSTER (0.359) as well as the other contact-assisted approach EigenTHREADER (0.386) over the hard targets. DisCovER also delivers noticeably better performance than the other methods including contact-assisted approaches CATHER, map_align, and EigenTHREADER as well as the profile-based methods HHpred, SparkX, and MUSTER for the medium difficulty targets. We note that we are unable to perform a target-by-target analysis since the CATHER paper reports only the average performance. Nonetheless, the better average performance of DisCovER across all target categories, especially for the medium and the hard targets, continues to demonstrate its competitive advantage over current threading methods.

4.4.6 Effect of homologous information

The performance of DisCovER is weakly correlated with the number of effective sequence homologs, as quantified by Nf (C. Zhang et al. 2020). As shown in **Appendix 6**, the Spearman correlation between the TM-score attained by DisCovER and Nf are 0.23 over the 117 hard targets from the CAMEO. We have similar observation (maximum Spearman correlation of 0.25) across various target categories from the CATHER dataset. In summary, there is a weak correlation between the performance of DisCovER and Nf.

4.4.7 Running time

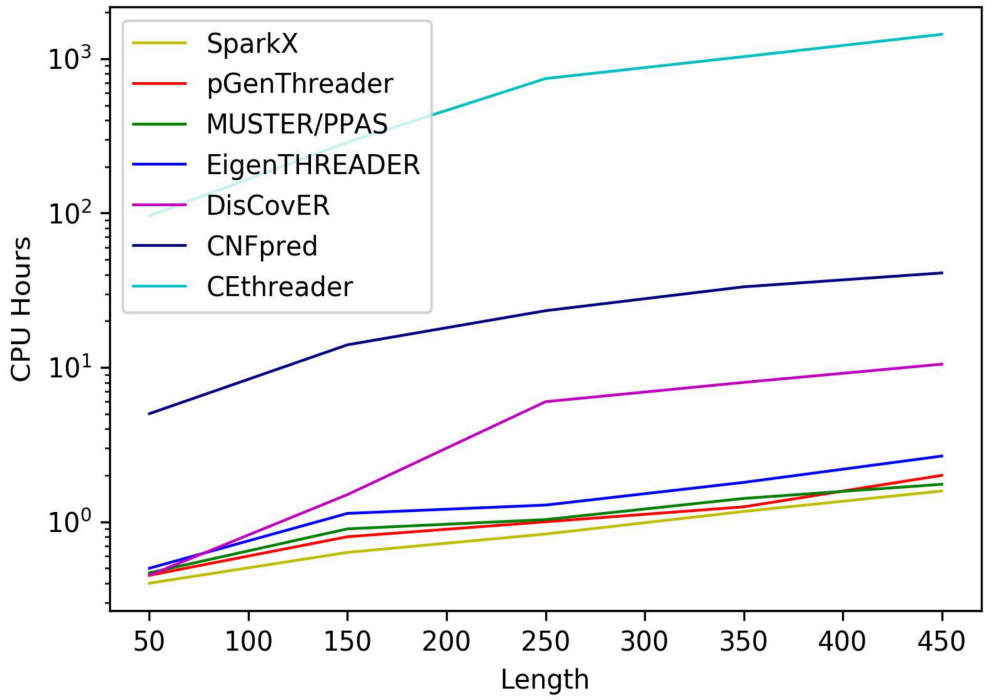


Figure 4.4 The running time of eight methods building alignments for 117 hard targets from CAMEO.

Figure 4.4 shows the running time of various threading methods with respect to the target length. All methods are run on the same Linux machine with 128 GB RAM and using a single CPU thread of Intel Xeon Processor (2.20 GHz). While it is expected that DisCovER is slower than most profile-based threading methods, but the running time of DisCovER is considerably faster than the top profile-based method CNFpred and orders of magnitude faster than the top contact-assisted approach CEthreader. Overall, DisCovER is reasonably efficient in terms of the running time.

4.5 Conclusion

This chapter introduces DisCovER, a new protein threading method that effectively integrates the covariational signal encoded in inter-residue distances and orientations information

along with the topological network neighborhood to significantly improve the threading template selection and the alignment for weakly homologous proteins. Experimental results show that our method yields better accuracy than existing threading methods, including profile-based methods and latest contact-assisted approaches such as CEthreader, EigenTHREADER, map_align, and CATHER. Controlled experiments reveal that the distance and the orientation information contributes significantly to the superior performance of DisCovER, complemented by the neighborhood effect particularly for weakly homologous proteins. At one of the most challenging threading situations, DisCovER outperforms the CAMEO server RaptorX employing the distance-based threading method DeepThreader.

It is important to note that the performance of our method is weakly correlated with the number of sequence homologs available for the query protein. This suggests that our distance- and orientation-based coevolutionary threading method DisCovER is well-suited for remotely homologous targets. Being reasonably efficient in terms of its running time, our study opens the possibility of successfully extending threading for many more protein sequences that were previously not amenable to template-based modeling.

Chapter 5

Conclusion

The development of computational approaches for accurately predicting the protein three-dimensional (3D) structure directly from the sequence information is of central importance in structural biology. Protein threading, the most widely used distant-homology modeling technique, aims to address the challenge by leveraging multiple sources of information by mining the evolutionary profile of the query and templates to reveal the potential distant homology and perform distant-homology modeling to predict the 3D structure of the query protein. However, remote homology detection via threading remains challenging, in part, due to the limitations of the threading scoring function for selecting optimal structural templates (Bhattacharya and Bhattacharya 2019b).

In Chapter 2, in light of the recent advancements in residue-residue contact prediction technologies powered by sequence co-evolution and deep learning, we have discussed a new threading method by integrating the residue-residue contact information with various sequential and structural features to improve the threading scoring function for better template selection (Bhattacharya and Bhattacharya 2019a). Our contact-assisted threading attains a statistically significant boost in the threading performance with the incorporation of the true contacts as well as the predicted contacts compared to a baseline contact-free threading acting as a control. Furthermore, our contact-assisted threading greatly outperforms the cutting-edge contact-assisted *ab initio* folding method, CONFOLD2, that utilizes the same predicted contacts - indicating that contact-assisted threading can be advantageous over contact-driven *ab initio* folding.

Next, in Chapter 3, we systematically explore the mutual association between the quality of predicted contacts (Bhattacharya and Bhattacharya 2020), and the resulting threading performance. The study reveals that contact-driven threading with the low-quality contacts predicted from pure co-evolutionary analysis is not as useful as incorporating the high-quality contacts from hybrid approaches that combine sequence co-evolution and machine learning, in that the high-quality contacts lead to an improved threading performance while the low-quality contacts deteriorate it. On the Critical Assessment of protein Structure Prediction (CASP13) targets, our method outperforms contact-assisted threading methods EigenThreader and map_align based on the average TM-score of the top ranked models as well as the success rate of identifying the correct overall folds, thereby attaining the state-of-the-art performance.

In Chapter 4, instead of relying on binary contacts, we move one step further by developing a new distance- and orientation-based covariational threading method DisCovER, which is capable of effectively integrate information from inter-residue distances and orientations along with the topological network neighborhood of a query-template alignment to greatly improve the threading template selection and the alignment. The major contributions of DisCovER are: (a) successfully incorporating inter-residue orientations into threading, and (b) considering the network neighborhood effect induced by the query-template alignment, i.e., the effect of the residue pairs in the neighborhood of an aligned query-template residue pair. DisCovER outperforms several existing state-of-the-art threading approaches on multiple large-scale benchmarking datasets. Controlled experiments reveal that the integration of the neighborhood effect with the inter-residue distances and orientations information synergistically contributes to the improved performance of DisCovER. The open-source DisCovER software package, licensed under the GNU General Public License v3, is freely available at https://github.com/Bhattacharya-Lab/DisCovER.

Very recently, NDthreader (F. Wu and Xu 2021) and ProALIGN (Kong et al. 2021) utilizes deep learning to optimally predict query-template alignments using the distance potential and demonstrate the state-of-the-art threading performance. Therefore, our future work can be further extended by the integration of deep learning into distant-homology protein modeling via interaction map threading. Moreover, recent CASP experiments have witnessed dramatic advances by DeepMind's AlphaFold series (Senior et al. 2020; 2019) in *ab initio* protein structure prediction, significantly outperforming the other groups. The success of AlphaFold series is primarily attributed to the successful application of deep neural networks for accurately predicting inter-residue spatial proximity information coupled with end-to-end training, significantly improving the accuracy of protein structure prediction (Pearce and Zhang 2021). The integration of deep learning into various stages of protein modeling represents an exciting future direction that shall have a transformative impact on distant-homology protein modeling via interaction map threading, complementing and supplementing *ab initio* protein structure prediction methods developed by DeepMind.

References

Adhikari, Badri. 2020. "A Fully Open-Source Framework for Deep Learning Protein Real-Valued Distances." *Scientific Reports* 10 (1): 13374. https://doi.org/10.1038/s41598-020-70181-0.

Adhikari, Badri, Debswapna Bhattacharya, Renzhi Cao, and Jianlin Cheng. 2015. "CONFOLD: Residue-Residue Contact-Guided Ab Initio Protein Folding." *Proteins: Structure, Function, and Bioinformatics* 83 (8): 1436–49. https://doi.org/10.1002/prot.24829.

Adhikari, Badri, and Jianlin Cheng. 2018. "CONFOLD2: Improved Contact-Driven Ab Initio Protein Structure Modeling." *BMC Bioinformatics* 19 (1): 22. https://doi.org/10.1186/s12859-018-2032-6.

Adhikari, Badri, Jie Hou, and Jianlin Cheng. 2018. "DNCON2: Improved Protein Contact Prediction Using Two-Level Deep Convolutional Neural Networks." *Bioinformatics* 34 (9): 1466–72. https://doi.org/10.1093/bioinformatics/btx781.

Adhikari, Badri, Jackson Nowotny, Debswapna Bhattacharya, Jie Hou, and Jianlin Cheng. 2016. "ConEVA: A Toolbox for Comprehensive Assessment of Protein Contacts." *BMC Bioinformatics* 17 (1): 517. https://doi.org/10.1186/s12859-016-1404-z.

Akutsu, Tatsuya, and Satoru Miyano. 1999. "On the Approximation of Protein Threading." *Theoretical Computer Science* 210 (2): 261–75. https://doi.org/10.1016/S0304-3975(98)00089-9.

Altschul, Stephen F., Thomas L. Madden, Alejandro A. Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman. 1997. "Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs." *Nucleic Acids Research* 25 (17): 3389–3402. https://doi.org/10.1093/nar/25.17.3389.

Baker, David, and Andrej Sali. 2001. "Protein Structure Prediction and Structural Genomics." *Science* 294 (5540): 93–96. https://doi.org/10.1126/science.1065659.

Berman, Helen M., John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. 2000. "The Protein Data Bank." *Nucleic Acids Research* 28 (1): 235–42. https://doi.org/10.1093/nar/28.1.235.

Bhattacharya, Sutanu, and Debswapna Bhattacharya. 2019a. "Does Inclusion of Residue-Residue Contact Information Boost Protein Threading?" *Proteins: Structure, Function, and Bioinformatics* 87 (7): 596–606. https://doi.org/10.1002/prot.25684.

———. 2019b. "How Effective Is Contact-Assisted Protein Threading?" In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 553. BCB '19. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3307339.3342624.

———. 2020. "Evaluating the Significance of Contact Maps in Low-Homology Protein Modeling Using Contact-Assisted Threading." *Scientific Reports* 10 (1): 2908. https://doi.org/10.1038/s41598-020-59834-2.

Bhattacharya, Sutanu, Rahmatullah Roche, and Debswapna Bhattacharya. 2020. "DisCovER: Distance- and Orientation-Based Covariational Threading for Weakly Homologous Proteins." *BioRxiv*, December, 2020.01.31.923409. https://doi.org/10.1101/2020.01.31.923409.

Bhattacharya, Sutanu, Rahmatullah Roche, Md Hossain Shuvo, and Debswapna Bhattacharya. 2021. "Recent Advances in Protein Homology Detection Propelled by Inter-Residue Interaction Map Threading." *Frontiers in Molecular Biosciences* 8. https://doi.org/10.3389/fmolb.2021.643752.

Bienkowska, Jadwiga, and Rick Lathrop. 2005. "Threading Algorithms." In *Encyclopedia of Genetics, Genomics, Proteomics and Bioinformatics*. American Cancer Society. https://doi.org/10.1002/047001153X.g409202.

Bowie, J. U., R. Luthy, and D. Eisenberg. 1991. "A Method to Identify Protein Sequences That Fold into a Known Three-Dimensional Structure." *Science* 253 (5016): 164–70. https://doi.org/10.1126/science.1853201.

Brylinski, Michal, and Jeffrey Skolnick. 2010. "Comparison of Structure-Based and Threading-Based Approaches to Protein Functional Annotation." *Proteins: Structure, Function, and Bioinformatics* 78 (1): 18–134. https://doi.org/10.1002/prot.22566.

Buchan, Daniel W. A., and David T. Jones. 2017. "EigenTHREADER: Analogous Protein Fold Recognition by Efficient Contact Map Threading." *Bioinformatics* 33 (17): 2684–90. https://doi.org/10.1093/bioinformatics/btx217.

Chen, Chun-Chi, Hyundoo Jeong, Xiaoning Qian, and Byung-Jun Yoon. 2019. "TOPAS: Network-Based Structural Alignment of RNA Sequences." *Bioinformatics* 35 (17): 2941–48. https://doi.org/10.1093/bioinformatics/btz001.

Cheng, Jianlin, and Pierre Baldi. 2006. "A Machine Learning Information Retrieval Approach to Protein Fold Recognition." *Bioinformatics* 22 (12): 1456–63. https://doi.org/10.1093/bioinformatics/btl102.

Di Lena, Pietro, Piero Fariselli, Luciano Margara, Marco Vassura, and Rita Casadio. 2010. "Fast Overlapping of Protein Contact Maps by Alignment of Eigenvectors." *Bioinformatics* 26 (18): 2250–58. https://doi.org/10.1093/bioinformatics/btq402.

Dill, Ken A., and Justin L. MacCallum. 2012. "The Protein-Folding Problem, 50 Years On." *Science* 338 (6110): 1042–46. https://doi.org/10.1126/science.1219021.

Ding, Wenze, and Haipeng Gong. 2020. "Predicting the Real-Valued Inter-Residue Distances for Proteins." *Advanced Science* 7 (19): 2001314. https://doi.org/10.1002/advs.202001314.

Du, Zongyang, Shuo Pan, Qi Wu, Zhenling Peng, and Jianyi Yang. 2020. "CATHER: A Novel Threading Algorithm with Predicted Contacts." *Bioinformatics* 36 (7): 2119–25. https://doi.org/10.1093/bioinformatics/btz876.

Ginalski, Krzysztof, Jakub Pas, Lucjan S. Wyrwicz, Marcin von Grotthuss, Janusz M. Bujnicki, and Leszek Rychlewski. 2003. "ORFeus: Detection of Distant Homology Using Sequence Profiles and Predicted Secondary Structure." *Nucleic Acids Research* 31 (13): 3804–7. https://doi.org/10.1093/nar/gkg504.

Gniewek, Pawel, Andrzej Kolinski, Andrzej Kloczkowski, and Dominik Gront. 2014. "BioShell-Threading: Versatile Monte Carlo Package for Protein 3D Threading." *BMC Bioinformatics* 15 (1): 22. https://doi.org/10.1186/1471-2105-15-22.

Greener, Joe G., Shaun M. Kandathil, and David T. Jones. 2019. "Deep Learning Extends de Novo Protein Modelling Coverage of Genomes Using Iteratively Predicted Structural Constraints." *Nature Communications* 10 (1): 1–13. https://doi.org/10.1038/s41467-019-11994-0.

Haas, Juergen, Rafal Gumienny, Alessandro Barbato, Flavio Ackermann, Gerardo Tauriello, Martino Bertoni, Gabriel Studer, Anna Smolinski, and Torsten Schwede. 2019. "Introducing 'Best Single Template' Models as Reference Baseline for the Continuous Automated Model Evaluation (CAMEO)." *Proteins: Structure, Function, and Bioinformatics* 87 (12): 1378–87. https://doi.org/10.1002/prot.25815.

Hanson, Jack, Kuldip Paliwal, Thomas Litfin, Yuedong Yang, and Yaoqi Zhou. 2018. "Accurate Prediction of Protein Contact Maps by Coupling Residual Two-Dimensional Bidirectional Long Short-Term Memory with Convolutional Neural Networks." *Bioinformatics* 34 (23): 4039–45. https://doi.org/10.1093/bioinformatics/bty481.

He, Baoji, S. M. Mortuza, Yanting Wang, Hong-Bin Shen, and Yang Zhang. 2017. "NeBcon: Protein Contact Map Prediction Using Neural Network Training Coupled with Naïve Bayes Classifiers." *Bioinformatics* 33 (15): 2296–2306. https://doi.org/10.1093/bioinformatics/btx164.

Heffernan, Rhys, Yuedong Yang, Kuldip Paliwal, and Yaoqi Zhou. 2017. "Capturing Non-Local Interactions by Long Short-Term Memory Bidirectional Recurrent Neural Networks for Improving Prediction of Protein Secondary Structure, Backbone Angles, Contact Numbers and Solvent Accessibility." *Bioinformatics* 33 (18): 2842–49. https://doi.org/10.1093/bioinformatics/btx218.

Henikoff, Steven, and Jorja G. Henikoff. 1994. "Position-Based Sequence Weights." *Journal of Molecular Biology* 243 (4): 574–78. https://doi.org/10.1016/0022-2836(94)90032-9.

Hubbard, Tim J. P., Alexey G. Murzin, Steven E. Brenner, and Cyrus Chothia. 1997. "SCOP: A Structural Classification of Proteins Database." *Nucleic Acids Research* 25 (1): 236–39. https://doi.org/10.1093/nar/25.1.236.

Jaroszewski, Lukasz, Leszek Rychlewski, Zhanwen Li, Weizhong Li, and Adam Godzik. 2005. "FFAS03: A Server for Profile–Profile Sequence Alignments." *Nucleic Acids Research* 33 (suppl_2): W284–88. https://doi.org/10.1093/nar/gki418.

Jones, D. T., W. R. Taylort, and J. M. Thornton. 1992. "A New Approach to Protein Fold Recognition." *Nature* 358 (6381): 86–89. https://doi.org/10.1038/358086a0.

Jones, David T. 1999. "GenTHREADER: An Efficient and Reliable Protein Fold Recognition Method for Genomic Sequences11Edited by B. Honig." *Journal of Molecular Biology* 287 (4): 797–815. https://doi.org/10.1006/jmbi.1999.2583.

Jones, David T., Daniel W. A. Buchan, Domenico Cozzetto, and Massimiliano Pontil. 2012. "PSICOV: Precise Structural Contact Prediction Using Sparse Inverse Covariance Estimation on Large Multiple Sequence Alignments." *Bioinformatics* 28 (2): 184–90. https://doi.org/10.1093/bioinformatics/btr638.

Jones, David T., Tanya Singh, Tomasz Kosciolek, and Stuart Tetchner. 2015. "MetaPSICOV: Combining Coevolution Methods for Accurate Prediction of Contacts and Long Range Hydrogen Bonding in Proteins." *Bioinformatics* 31 (7): 999–1006. https://doi.org/10.1093/bioinformatics/btu791.

Kabsch, W. 1976. "A Solution for the Best Rotation to Relate Two Sets of Vectors." *Acta Crystallographica Section A* 32 (5): 922–23. https://doi.org/10.1107/S0567739476001873.

Kabsch, Wolfgang, and Christian Sander. 1983. "Dictionary of Protein Secondary Structure: Pattern Recognition of Hydrogen-Bonded and Geometrical Features." *Biopolymers* 22 (12): 2577–2637. https://doi.org/10.1002/bip.360221211.

Kaján, László, Thomas A. Hopf, Matúš Kalaš, Debora S. Marks, and Burkhard Rost. 2014. "FreeContact: Fast and Free Software for Protein Contact Prediction from Residue Co-Evolution." *BMC Bioinformatics* 15 (1): 85. https://doi.org/10.1186/1471-2105-15-85.

Källberg, Morten, Haipeng Wang, Sheng Wang, Jian Peng, Zhiyong Wang, Hui Lu, and Jinbo Xu. 2012. "Template-Based Protein Structure Modeling Using the RaptorX Web Server." *Nature Protocols* 7 (8): 1511–22. https://doi.org/10.1038/nprot.2012.085.

Kamisetty, Hetunandan, Sergey Ovchinnikov, and David Baker. 2013. "Assessing the Utility of Coevolution-Based Residue–Residue Contact Predictions in a Sequence- and Structure-Rich Era." *Proceedings of the National Academy of Sciences* 110 (39): 15674–79. https://doi.org/10.1073/pnas.1314045110.

Kandathil, Shaun M., Joe G. Greener, and David T. Jones. 2019. "Prediction of Interresidue Contacts with DeepMetaPSICOV in CASP13." *Proteins: Structure, Function, and Bioinformatics* 87 (12): 1092–99. https://doi.org/10.1002/prot.25779.

Kinch, Lisa N, and Nick V Grishin. 2002. "Evolution of Protein Structures and Functions." *Current Opinion in Structural Biology* 12 (3): 400–408. https://doi.org/10.1016/S0959-440X(02)00338-X.

Kong, Lupeng, Fusong Ju, Wei-Mou Zheng, Jianwei Zhu, Shiwei Sun, Jinbo Xu, and Dongbo Bu. 2021. "ProALIGN: Directly Learning Alignments for Protein Structure Prediction via Exploiting Context-Specific Alignment Motifs." *BioRxiv*, January, 2020.12.28.424539. https://doi.org/10.1101/2020.12.28.424539.

Lee, Seung Yup, and Jeffrey Skolnick. 2010. "TASSER_WT: A Protein Structure Prediction Algorithm with Accurate Predicted Contact Restraints for Difficult Protein Targets." *Biophysical Journal* 99 (9): 3066–75. https://doi.org/10.1016/j.bpj.2010.09.007.

Li, Jin, and Jinbo Xu. 2020. "Study of Real-Valued Distance Prediction For Protein Structure Prediction with Deep Learning." *BioRxiv*, November, 2020.11.26.400523. https://doi.org/10.1101/2020.11.26.400523.

Li, Yang, Jun Hu, Chengxin Zhang, Dong-Jun Yu, and Yang Zhang. 2019. "ResPRE: High-Accuracy Protein Contact Prediction by Coupling Precision Matrix with Deep Residual Neural Networks." *Bioinformatics* 35 (22): 4647–55. https://doi.org/10.1093/bioinformatics/btz291.

Lobley, Anna, Michael I. Sadowski, and David T. Jones. 2009. "PGenTHREADER and PDomTHREADER: New Methods for Improved Protein Fold Recognition and Superfamily Discrimination." *Bioinformatics* 25 (14): 1761–67. https://doi.org/10.1093/bioinformatics/btp302.

Ma, Jianzhu, Jian Peng, Sheng Wang, and Jinbo Xu. 2012. "A Conditional Neural Fields Model for Protein Threading." *Bioinformatics* 28 (12): i59–66. https://doi.org/10.1093/bioinformatics/bts213.

Ma, Jianzhu, Sheng Wang, Zhiyong Wang, and Jinbo Xu. 2014. "MRFalign: Protein Homology Detection through Alignment of Markov Random Fields." *PLOS Computational Biology* 10 (3): e1003500. https://doi.org/10.1371/journal.pcbi.1003500.

Ma, Jianzhu, Sheng Wang, Feng Zhao, and Jinbo Xu. 2013. "Protein Threading Using Context-Specific Alignment Potential." *Bioinformatics* 29 (13): i257–65. https://doi.org/10.1093/bioinformatics/btt210.

Malod-Dognin, Noël, and Nataša Pržulj. 2014. "GR-Align: Fast and Flexible Alignment of Protein 3D Structures Using Graphlet Degree Similarity." *Bioinformatics* 30 (9): 1259–65. https://doi.org/10.1093/bioinformatics/btu020.

Mariani, Valerio, Marco Biasini, Alessandro Barbato, and Torsten Schwede. 2013. "LDDT: A Local Superposition-Free Score for Comparing Protein Structures and Models Using Distance Difference Tests." *Bioinformatics* 29 (21): 2722–28. https://doi.org/10.1093/bioinformatics/btt473.

Markowitz, Victor M., I-Min A. Chen, Ken Chu, Ernest Szeto, Krishna Palaniappan, Manoj Pillay, Anna Ratner, et al. 2014. "IMG/M 4 Version of the Integrated Metagenome Comparative Analysis System." *Nucleic Acids Research* 42 (D1): D568–73. https://doi.org/10.1093/nar/gkt919.

Marti-Renom, Marc A., M. S. Madhusudhan, and Andrej Sali. 2004. "Alignment of Protein Sequences by Their Profiles." *Protein Science* 13 (4): 1071–87. https://doi.org/10.1110/ps.03379804.

Mirdita, Milot, Lars von den Driesch, Clovis Galiez, Maria J. Martin, Johannes Söding, and Martin Steinegger. 2017. "Uniclust Databases of Clustered and Deeply Annotated Protein Sequences and Alignments." *Nucleic Acids Research* 45 (D1): D170–76. https://doi.org/10.1093/nar/gkw1081.

Mitchell, Alex L, Alexandre Almeida, Martin Beracochea, Miguel Boland, Josephine Burgin, Guy Cochrane, Michael R Crusoe, et al. 2020. "MGnify: The Microbiome Analysis Resource in 2020." *Nucleic Acids Research* 48 (D1): D570–78. https://doi.org/10.1093/nar/gkz1035.

Mitchell, Alex L, Maxim Scheremetjew, Hubert Denise, Simon Potter, Aleksandra Tarkowska, Matloob Qureshi, Gustavo A Salazar, et al. 2018. "EBI Metagenomics in 2017: Enriching the Analysis of Microbial Communities, from Sequence Reads to Assemblies." *Nucleic Acids Research* 46 (D1): D726–35. https://doi.org/10.1093/nar/gkx967.

Monastyrskyy, Bohdan, Daniel D'Andrea, Krzysztof Fidelis, Anna Tramontano, and Andriy Kryshtafovych. 2014. "Evaluation of Residue–Residue Contact Prediction in CASP10." *Proteins: Structure, Function, and Bioinformatics* 82 (S2): 138–53. https://doi.org/10.1002/prot.24340.

Morcos, Faruck, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S. Marks, Chris Sander, Riccardo Zecchina, José N. Onuchic, Terence Hwa, and Martin Weigt. 2011. "Direct-Coupling Analysis of Residue Coevolution Captures Native Contacts across Many Protein Families." *Proceedings of the National Academy of Sciences* 108 (49): E1293–1301. https://doi.org/10.1073/pnas.1111471108.

Moult, John. 1996. “The Current State of the Art in Protein Structure Prediction.” *Current Opinion in Biotechnology* 7 (4): 422–27. https://doi.org/10.1016/S0958-1669(96)80118-2.

Moult, John, Krzysztof Fidelis, Andriy Kryshtafovych, Torsten Schwede, and Anna Tramontano. 2014. “Critical Assessment of Methods of Protein Structure Prediction (CASP) — Round x.” *Proteins: Structure, Function, and Bioinformatics* 82 (S2): 1–6. https://doi.org/10.1002/prot.24452.

Needleman, Saul B., and Christian D. Wunsch. 1970. “A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins.” *Journal of Molecular Biology* 48 (3): 443–53. https://doi.org/10.1016/0022-2836(70)90057-4.

Olmea, Osvaldo, Burkhard Rost, and Alfonso Valencia. 1999. “Effective Use of Sequence Correlation and Conservation in Fold Recognition11Edited by J. M. Thornton.” *Journal of Molecular Biology* 293 (5): 1221–39. https://doi.org/10.1006/jmbi.1999.3208.

Ovchinnikov, Sergey, Hahnbeom Park, Neha Varghese, Po-Ssu Huang, Georgios A. Pavlopoulos, David E. Kim, Hetunandan Kamisetty, Nikos C. Kyrpides, and David Baker. 2017. “Protein Structure Determination Using Metagenome Sequence Data.” *Science* 355 (6322): 294–98. https://doi.org/10.1126/science.aah4043.

Pearce, Robin, and Yang Zhang. 2021. “Deep Learning Techniques Have Significantly Impacted Protein Structure Prediction and Protein Design.” *Current Opinion in Structural Biology* 68 (June): 194–207. https://doi.org/10.1016/j.sbi.2021.01.007.

Peng, Jian, and Jinbo Xu. 2009. “Boosting Protein Threading Accuracy.” In *Research in Computational Molecular Biology*, edited by Serafim Batzoglou, 31–45. Lecture Notes in Computer Science. Springer Berlin Heidelberg.

———. 2010. “Low-Homology Protein Threading.” *Bioinformatics* 26 (12): i294–300. https://doi.org/10.1093/bioinformatics/btq192.

———. 2011. “A Multiple-Template Approach to Protein Threading.” *Proteins: Structure, Function, and Bioinformatics* 79 (6): 1930–39. https://doi.org/10.1002/prot.23016.

Petrey, Donald, and Barry Honig. 2005. “Protein Structure Prediction: Inroads to Biology.” *Molecular Cell* 20 (6): 811–19. https://doi.org/10.1016/j.molcel.2005.12.005.

Roche, Rahmatullah, Sutanu Bhattacharya, and Debswapna Bhattacharya. 2021. “Hybridized Distance- and Contact-Based Hierarchical Structure Modeling for Folding Soluble and Membrane Proteins.” *PLOS Computational Biology* 17 (2): e1008753. https://doi.org/10.1371/journal.pcbi.1008753.

Rost, Burkhard, Reinhard Schneider, and Chris Sander. 1997. “Protein Fold Recognition by Prediction-Based Threading11Edited by F. E. Cohen.” *Journal of Molecular Biology* 270 (3): 471–80. https://doi.org/10.1006/jmbi.1997.1101.

Rychlewski, Leszek, Weizhong Li, Lukasz Jaroszewski, and Adam Godzik. 2000. “Comparison of Sequence Profiles. Strategies for Structural Predictions Using Sequence Information.” *Protein Science* 9 (2): 232–41. https://doi.org/10.1110/ps.9.2.232.

Schaarschmidt, Joerg, Bohdan Monastyrskyy, Andriy Kryshtafovych, and Alexandre M. J. J. Bonvin. 2018. “Assessment of Contact Predictions in CASP12: Co-Evolution and Deep Learning Coming of Age.” *Proteins: Structure, Function, and Bioinformatics* 86 (S1): 51–66. https://doi.org/10.1002/prot.25407.

Seemayer, Stefan, Markus Gruber, and Johannes Söding. 2014. “CCMpred—Fast and Precise Prediction of Protein Residue–Residue Contacts from Correlated Mutations.” *Bioinformatics* 30 (21): 3128–30. https://doi.org/10.1093/bioinformatics/btu500.

Senior, Andrew W., Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, et al. 2019. “Protein Structure Prediction Using Multiple Deep Neural Networks in the 13th Critical Assessment of Protein Structure Prediction (CASP13).” *Proteins: Structure, Function, and Bioinformatics* 87 (12): 1141–48. https://doi.org/10.1002/prot.25834.

———. 2020. “Improved Protein Structure Prediction Using Potentials from Deep Learning.” *Nature* 577 (7792): 706–10. https://doi.org/10.1038/s41586-019-1923-7.

Shibberu, Y., and A. Holder. 2011. “A Spectral Approach to Protein Structure Alignment.” *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 8 (4): 867–75. https://doi.org/10.1109/TCBB.2011.24.

Shibberu, Yosi, Allen Holder, and Kyla Lutz. 2010. “Fast Protein Structure Alignment.” In *Bioinformatics Research and Applications*, edited by Mark Borodovsky, Johann Peter Gogarten, Teresa M. Przytycka, and Sanguthevar Rajasekaran, 152–65. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer. https://doi.org/10.1007/978-3-642-13078-6_18.

Shrestha, Rojan, Eduardo Fajardo, Nelson Gil, Krzysztof Fidelis, Andriy Kryshtafovych, Bohdan Monastyrskyy, and Andras Fiser. 2019. “Assessing the Accuracy of Contact Predictions in CASP13.” *Proteins: Structure, Function, and Bioinformatics* 87 (12): 1058–68. https://doi.org/10.1002/prot.25819.

Singh, Rohit, Jinbo Xu, and Bonnie Berger. 2008. “Global Alignment of Multiple Protein Interaction Networks with Application to Functional Orthology Detection.” *Proceedings of the National Academy of Sciences* 105 (35): 12763–68. https://doi.org/10.1073/pnas.0806627105.

Skolnick, Jeffrey, and Daisuke Kihara. 2001. “Defrosting the Frozen Approximation: PROSPECTOR— A New Approach to Threading.” *Proteins: Structure, Function, and Bioinformatics* 42 (3): 319–31. https://doi.org/10.1002/1097-0134(20010215)42:3<319::AID-PROT30>3.0.CO;2-A.

Skolnick, Jeffrey, Daisuke Kihara, and Yang Zhang. 2004. “Development and Large Scale Benchmark Testing of the PROSPECTOR_3 Threading Algorithm.” *Proteins: Structure, Function, and Bioinformatics* 56 (3): 502–18. https://doi.org/10.1002/prot.20106.

Skolnick, Jeffrey, and Hongyi Zhou. 2017. “Why Is There a Glass Ceiling for Threading Based Protein Structure Prediction Methods?” *The Journal of Physical Chemistry B* 121 (15): 3546–54. https://doi.org/10.1021/acs.jpcb.6b09517.

Smith, T. F., and M. S. Waterman. 1981. “Identification of Common Molecular Subsequences.” *Journal of Molecular Biology* 147 (1): 195–97. https://doi.org/10.1016/0022-2836(81)90087-5.

Söding, Johannes. 2005. “Protein Homology Detection by HMM–HMM Comparison.” *Bioinformatics* 21 (7): 951–60. https://doi.org/10.1093/bioinformatics/bti125.

———. 2017. “Big-Data Approaches to Protein Structure Prediction.” *Science* 355 (6322): 248–49. https://doi.org/10.1126/science.aal4512.

Steinegger, Martin, Milot Mirdita, and Johannes Söding. 2019. “Protein-Level Assembly Increases Protein Sequence Recovery from Metagenomic Samples Manyfold.” *Nature Methods* 16 (7): 603–6. https://doi.org/10.1038/s41592-019-0437-4.

Steinegger, Martin, and Johannes Söding. 2018. “Clustering Huge Protein Sequence Sets in Linear Time.” *Nature Communications* 9 (1): 2542. https://doi.org/10.1038/s41467-018-04964-5.

Suzek, Baris E., Yuqi Wang, Hongzhan Huang, Peter B. McGarvey, Cathy H. Wu, and the UniProt Consortium. 2015. “UniRef Clusters: A Comprehensive and Scalable Alternative for Improving Sequence Similarity Searches.” *Bioinformatics* 31 (6): 926–32. https://doi.org/10.1093/bioinformatics/btu739.

Taylor, William R. 1999. “Protein Structure Comparison Using Iterated Double Dynamic Programming.” *Protein Science* 8 (3): 654–65. https://doi.org/10.1110/ps.8.3.654.

Teichert, Florian, Ugo Bastolla, and Markus Porto. 2007. “SABERTOOTH: Protein Structural Alignment Based on a Vectorial Structure Representation.” *BMC Bioinformatics* 8 (1): 425. https://doi.org/10.1186/1471-2105-8-425.

Teichert, Florian, Jonas Minning, Ugo Bastolla, and Markus Porto. 2010. “High Quality Protein Sequence Alignment by Combining Structural Profile Prediction and Profile Alignment Using SABERTOOTH.” *BMC Bioinformatics* 11 (1): 251. https://doi.org/10.1186/1471-2105-11-251.

The UniProt Consortium. 2019. “UniProt: A Worldwide Hub of Protein Knowledge.” *Nucleic Acids Research* 47 (D1): D506–15. https://doi.org/10.1093/nar/gky1049.

Venclovas, Česlovas. 2003. “Comparative Modeling in CASP5: Progress Is Evident, but Alignment Errors Remain a Significant Hindrance.” *Proteins: Structure, Function, and Bioinformatics* 53 (S6): 380–88. https://doi.org/10.1002/prot.10591.

Wang, Sheng, Wei Li, Renyu Zhang, Shiwang Liu, and Jinbo Xu. 2016. “CoinFold: A Web Server for Protein Contact Prediction and Contact-Assisted Protein Folding.” *Nucleic Acids Research* 44 (W1): W361–66. https://doi.org/10.1093/nar/gkw307.

Wang, Sheng, Siqi Sun, Zhen Li, Renyu Zhang, and Jinbo Xu. 2017. “Accurate De Novo Prediction of Protein Contact Map by Ultra-Deep Learning Model.” *PLOS Computational Biology* 13 (1): e1005324. https://doi.org/10.1371/journal.pcbi.1005324.

Wang, Yan, Qiang Shi, Pengshuo Yang, Chengxin Zhang, S. M. Mortuza, Zhidong Xue, Kang Ning, and Yang Zhang. 2019. “Fueling Ab Initio Folding with Marine Metagenomics Enables Structure and Function Predictions of New Protein Families.” *Genome Biology* 20 (1): 229. https://doi.org/10.1186/s13059-019-1823-z.

Wang, Zhiyong, and Jinbo Xu. 2013. “Predicting Protein Contact Map Using Evolutionary and Physical Constraints by Integer Programming.” *Bioinformatics* 29 (13): i266–73. https://doi.org/10.1093/bioinformatics/btt211.

Webb, Benjamin, and Andrej Sali. 2014. “Protein Structure Modeling with MODELLER.” In *Protein Structure Prediction*, edited by Daisuke Kihara, 1–15. Methods in Molecular Biology. New York, NY: Springer. https://doi.org/10.1007/978-1-4939-0366-5_1.

Wu, Fandi, and Jinbo Xu. 2021. “Deep Template-Based Protein Structure Prediction.” *PLOS Computational Biology* 17 (5): e1008954. https://doi.org/10.1371/journal.pcbi.1008954.

Wu, Qi, Zhenling Peng, Ivan Anishchenko, Qian Cong, David Baker, and Jianyi Yang. 2020. “Protein Contact Prediction Using Metagenome Sequence Data and Residual Neural Networks.” *Bioinformatics* 36 (1): 41–48. https://doi.org/10.1093/bioinformatics/btz477.

Wu, Sitao, and Yang Zhang. 2007. “LOMETS: A Local Meta-Threading-Server for Protein Structure Prediction.” *Nucleic Acids Research* 35 (10): 3375–82. https://doi.org/10.1093/nar/gkm251.

———. 2008. “MUSTER: Improving Protein Sequence Profile–Profile Alignments by Using Multiple Sources of Structure Information.” *Proteins: Structure, Function, and Bioinformatics* 72 (2): 547–56. https://doi.org/10.1002/prot.21945.

———. 2010. “Recognizing Protein Substructure Similarity Using Segmental Threading.” *Structure* 18 (7): 858–67. https://doi.org/10.1016/j.str.2010.04.007.

Wu, Tianqi, Zhiye Guo, Jie Hou, and Jianlin Cheng. 2021. “DeepDist: Real-Value Inter-Residue Distance Prediction with Deep Residual Convolutional Network.” *BMC Bioinformatics* 22 (1): 30. https://doi.org/10.1186/s12859-021-03960-9.

Wuyun, Qiqige, Wei Zheng, Zhenling Peng, and Jianyi Yang. 2018. “A Large-Scale Comparative Assessment of Methods for Residue–Residue Contact Prediction.” *Briefings in Bioinformatics* 19 (2): 219–30. https://doi.org/10.1093/bib/bbw106.

Xu, Jinbo. 2019. “Distance-Based Protein Folding Powered by Deep Learning.” *Proceedings of the National Academy of Sciences* 116 (34): 16856–65. https://doi.org/10.1073/pnas.1821309116.

Xu, Jinbo, Ming Li, Dongsup Kim, and Ying Xu. 2003. “Raptor: Optimal Protein Threading by Linear Programming.” *Journal of Bioinformatics and Computational Biology* 01 (01): 95–117. https://doi.org/10.1142/S0219720003000186.

Xu, Jinbo, and Sheng Wang. 2019. “Analysis of Distance-Based Protein Structure Prediction by Deep Learning in CASP13.” *Proteins: Structure, Function, and Bioinformatics* 87 (12): 1069–81. https://doi.org/10.1002/prot.25810.

Xu, Jinrui, and Yang Zhang. 2010. "How Significant Is a Protein Structure Similarity with TM-Score = 0.5?" *Bioinformatics* 26 (7): 889–95. https://doi.org/10.1093/bioinformatics/btq066.

Xu, Ying, and Dong Xu. 2000. "Protein Threading Using PROSPECT: Design and Evaluation." *Proteins: Structure, Function, and Bioinformatics* 40 (3): 343–54. https://doi.org/10.1002/1097-0134(20000815)40:3<343::AID-PROT10>3.0.CO;2-S.

Xu, Ying, Dong Xu, and Edward C. Uberbacher. 1998. "An Efficient Computational Method for Globally Optimal Threading1." *Journal of Computational Biology* 5 (3): 597–614. https://doi.org/10.1089/cmb.1998.5.597.

Yan, Renxiang, Dong Xu, Jianyi Yang, Sara Walker, and Yang Zhang. 2013. "A Comparative Assessment and Analysis of 20 Representative Sequence Alignment Methods for Protein Structure Prediction." *Scientific Reports* 3 (September): 2619.

Yang, Jianyi, Ivan Anishchenko, Hahnbeom Park, Zhenling Peng, Sergey Ovchinnikov, and David Baker. 2020. "Improved Protein Structure Prediction Using Predicted Interresidue Orientations." *Proceedings of the National Academy of Sciences* 117 (3): 1496–1503. https://doi.org/10.1073/pnas.1914677117.

Yang, Jianyi, Renxiang Yan, Ambrish Roy, Dong Xu, Jonathan Poisson, and Yang Zhang. 2015. "The I-TASSER Suite: Protein Structure and Function Prediction." *Nature Methods* 12 (1): 7–8. https://doi.org/10.1038/nmeth.3213.

Yang, Yuedong, Eshel Faraggi, Huiying Zhao, and Yaoqi Zhou. 2011. "Improving Protein Fold Recognition and Template-Based Modeling by Employing Probabilistic-Based Matching between Predicted One-Dimensional Structural Properties of Query and Corresponding Native Properties of Templates." *Bioinformatics* 27 (15): 2076–82. https://doi.org/10.1093/bioinformatics/btr350.

Zemla, Adam. 2003. "LGA: A Method for Finding 3D Similarities in Protein Structures." *Nucleic Acids Research* 31 (13): 3370–74. https://doi.org/10.1093/nar/gkg571.

Zhang, Chengxin, Wei Zheng, S. M. Mortuza, Yang Li, and Yang Zhang. 2020. "DeepMSA: Constructing Deep Multiple Sequence Alignment to Improve Contact Prediction and Fold-Recognition for Distant-Homology Proteins." *Bioinformatics* 36 (7): 2105–12. https://doi.org/10.1093/bioinformatics/btz863.

Zhang, Haicang, and Yufeng Shen. 2020. "Template-Based Prediction of Protein Structure with Deep Learning." *BMC Genomics* 21 (11): 878. https://doi.org/10.1186/s12864-020-07249-8.

Zhang, Yang, and Jeffrey Skolnick. 2004. "Scoring Function for Automated Assessment of Protein Structure Template Quality." *Proteins: Structure, Function, and Bioinformatics* 57 (4): 702–10. https://doi.org/10.1002/prot.20264.

———. 2005a. "The Protein Structure Prediction Problem Could Be Solved Using the Current PDB Library." *Proceedings of the National Academy of Sciences* 102 (4): 1029–34. https://doi.org/10.1073/pnas.0407152101.

———. 2005b. "TM-Align: A Protein Structure Alignment Algorithm Based on the TM-Score." *Nucleic Acids Research* 33 (7): 2302–9. https://doi.org/10.1093/nar/gki524.

Zheng, Wei, Qiqige Wuyun, Yang Li, S. M. Mortuza, Chengxin Zhang, Robin Pearce, Jishou Ruan, and Yang Zhang. 2019. "Detecting Distant-Homology Protein Structures by Aligning Deep Neural-Network Based Contact Maps." *PLOS Computational Biology* 15 (10): e1007411. https://doi.org/10.1371/journal.pcbi.1007411.

Zheng, Wei, Chengxin Zhang, Qiqige Wuyun, Robin Pearce, Yang Li, and Yang Zhang. 2019. "LOMETS2: Improved Meta-Threading Server for Fold-Recognition and Structure-Based Function Annotation for Distant-Homology Proteins." *Nucleic Acids Research* 47 (W1): W429–36. https://doi.org/10.1093/nar/gkz384.

Zhou, Hongyi, and Yaoqi Zhou. 2005. "Fold Recognition by Combining Sequence Profiles Derived from Evolution and from Depth-Dependent Structural Alignment of Fragments." *Proteins: Structure, Function, and Bioinformatics* 58 (2): 321–28. https://doi.org/10.1002/prot.20308.

Zhu, Jianwei, Sheng Wang, Dongbo Bu, and Jinbo Xu. 2018. "Protein Threading Using Residue Co-Variation and Deep Learning." *Bioinformatics* 34 (13): i263–73. https://doi.org/10.1093/bioinformatics/bty278.

Appendix 1

Target by target CPU hours needed by map_align over 11 CASP13 full-length targets of length < 300 residues.

Target	Length	CPU hours
T0951	276	2450
T0953s1	72	162.5
T0953s2	249	1950
T0955	41	65
T0957s1	163	840
T0957s2	164	930
T0958	96	292.5
T0968s1	126	510
T0968s2	116	450
T1008	80	225
T1016	203	1300

Appendix 2

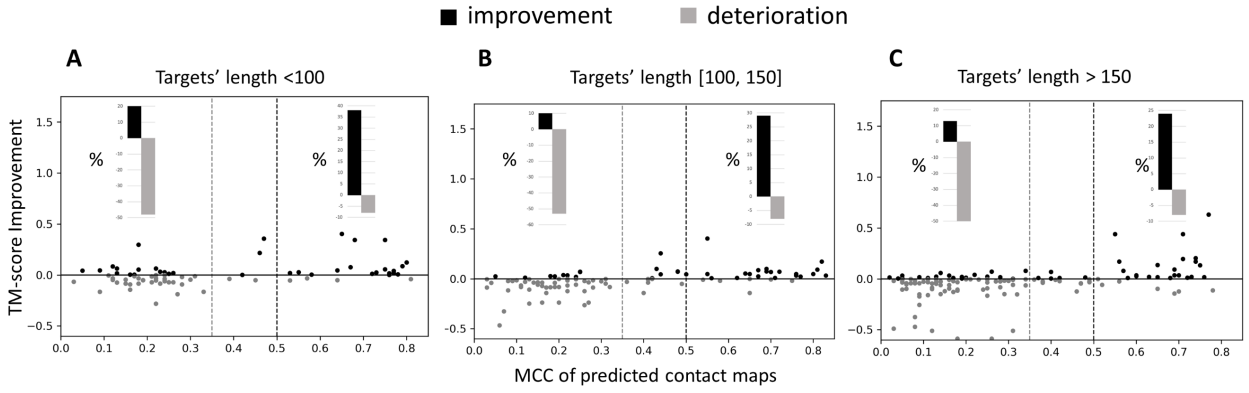
***p*-value of different contact-assisted threading methods on PSICOV150 dataset compared to the baseline pure threading method.** (*p*-values < 0.05 listed in bold)*

	mfDCA-assisted threading	PSICOV-assisted threading	MetaPSICOV-assisted threading
pure threading	**5.80E-08**	**2.90E-05**	0.07

*excluding two targets (1tqhA and 1hdoA) for which RaptorX could not predict contact maps.

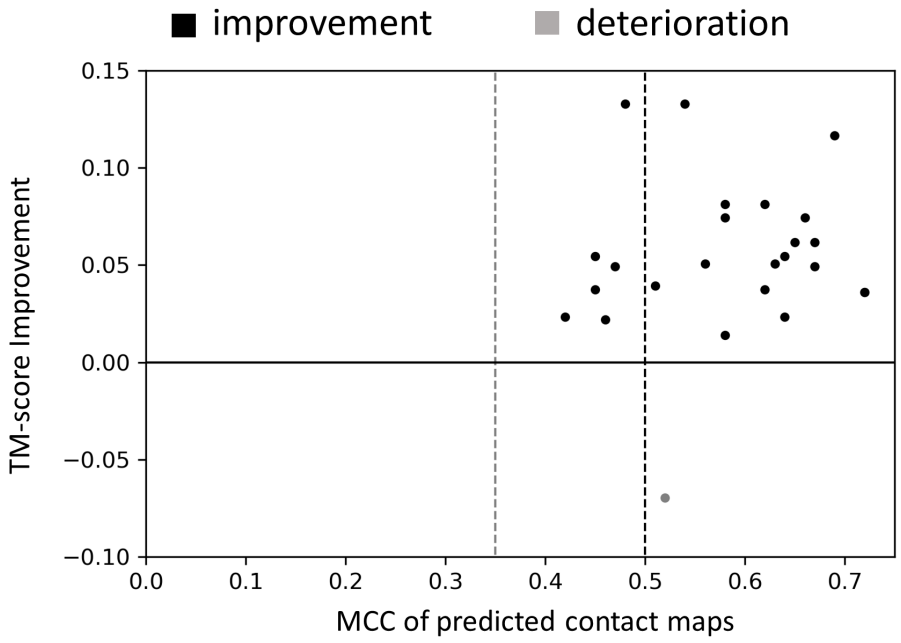
Appendix 3

The relationship between the changes in TM-score of contact-assisted threading methods compared to the baseline pure threading method, and the MCC of predicted contact maps, tested on PSICOV150. The targets are grouped into three bins based on their sequence length. (A) 34 targets of sequence length < 100 residues are considered. This set includes 136 instances, considering all four contact-assisted threading methods. (B) 47 targets of sequence length [100,150] residues are considered, resulting in a total of 188 instances by considering all four methods. (C) 67 targets of sequence length >150 residues are considered, resulting in a total of 268 instances by considering all four contact-assisted threading methods.



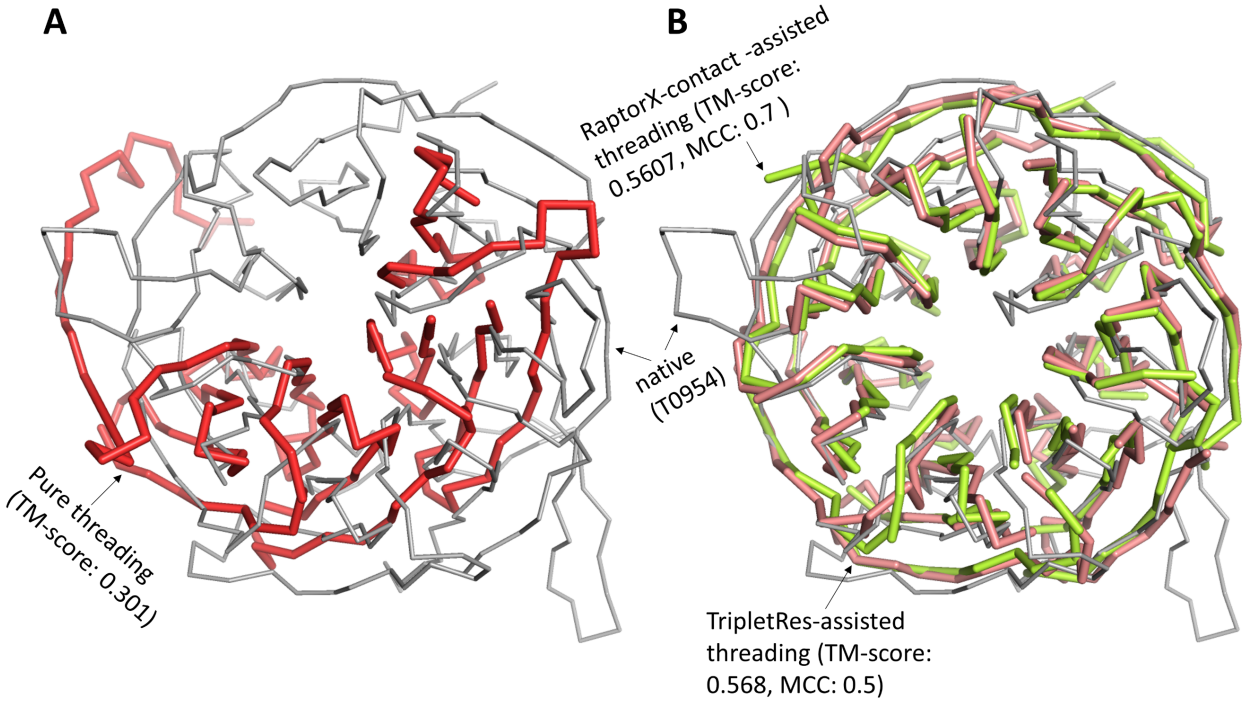
Appendix 4

The relationship between the changes in TM-score of contact-assisted threading methods compared to the baseline pure threading method, and the MCC of predicted contact maps, tested on the officially released 20 full-length targets of CASP13. This set includes a total of 40 instances, considering two contact-assisted threading methods. Out of these, there are 29 instances with high-quality contacts (MCC ≥ 0.5) as opposed to only one instance (TripletRes contact map for T1008) with an MCC < 0.35. An instance for which there is a change in the TM-score (either positive or negative) compared to the baseline pure threading is only plotted.



Appendix 5

A representative example of contact-assisted threading with the top2 officially ranked contact predictors of CASP13 on target T0954. (A) Structural alignment between the top ranked model predicted by the pure threading method (in thick red) with a TM-score of 0.301 and the native structure of the target (in thin gray). (B) Structural alignment between the top ranked model predicted by threading methods using the high-quality (MCC ≥ 0.5) contacts, TripletRes (in thick salmon red) and RaptorX-contact (in thick limon yellow), with TM-scores of > 0.56 and the native structure of the target (in thin gray).



Appendix 6

TM-score of DisCovER *vs* Nf (the number of sequence homologs) on (A) 117 hard targets from CAMEO having a Spearman correlation of 0.23, (B) 304 easy targets from CATHER having a Spearman correlation of 0.19, (C) 45 medium targets from CATHER having a Spearman correlation of 0.16, and (D) 131 hard targets from CATHER having a Spearman correlation of 0.25.