Structural bioinformatics

DisCovER: distance-based covariational threading for weakly homologous proteins

Sutanu Bhattacharya[1], Rahmatullah Roche[1], Debswapna Bhattacharya [1,2,*]

[1]Department of Computer Science and Software Engineering, [2]Department of Biological Sciences, Auburn University, AL 36849, USA.

*To whom correspondence should be addressed.

Abstract

Motivation: Threading a query protein sequence onto a library of weakly homologous structural templates remains challenging, even when sequence-based predicted contact or distance information is used. Contact- or distance-assisted threading methods utilize only the residue pairs predicted to be spatially close for template selection and alignment, ignoring the neighborhood effect induced by the interacting pairs.

Results: We present a new distance-based covariational threading method called DisCovER by effectively integrating information from distance maps and their topological network neighborhood. Our method first selects a subset of templates using standard profile-based threading to perform distance-based query-template alignment, and then iteratively improves the sensitivity of the alignment through double dynamic programming by incorporating topological network similarity terms induced by distance-derived inter-residue interaction network to account for the neighborhood effect. Multiple large-scale benchmarking results on query proteins classified in current literature as hard threading targets show that our method greatly outperforms not only the traditional threading methods including SparkX, HHpred, CNFpred, MUSTER, PPAS, and pGenThreader; but also the state-of-the-art contact-assisted threading approaches such as CEthreader, map_align, EigenTHREADER, and CATHER. As threading gets harder and harder by the exclusion of more and more structurally similar templates, the performance of our method gets better and better compared to the contact-assisted approaches CEthreader, map_align, and EigenTHREADER. At one of the hardest threading scenarios, the average performance of DisCovER is much better than the reported average accuracy of the only other distance-based threading method DeepThreader, which is not publicly available. Controlled experiments reveal that integration of distance information significantly improves threading template selection and alignment and is one of the key factors behind the improved performance of DisCovER.

Availability: https://github.com/Bhattacharya-Lab/DisCovER

Contact: bhattacharyad@auburn.edu

1 Introduction

Accurate prediction of the three-dimensional (3D) structure of a protein from its sequence remains challenging (Dill and MacCallum, 2012). Template-based modeling (TBM), one of the most reliable and accurate approaches for protein 3D structure prediction, uses homologous templates of analogous folds available in the Protein Data Bank (Berman *et al.*, 2000) to infer 3D structural models of unknown proteins. As such, the success of TBM intrinsically depends on the detection of homologous templates and the generation of accurate query-template alignments. However, in the absence of close sequence homology, template detection and alignment become challenging. Protein threading is a TBM approach that can address the challenge by leveraging multiple sources of information such as sequence profile, secondary structure, solvent accessibility, and torsion angles. Traditional threading methods such as HHpred (Meier and Söding, 2015), MUSTER (Wu and Zhang, 2008), SparkX (Yang *et al.*, 2011), pGenThreader (Lobley *et al.*, 2009), and CNFpred (Ma *et al.*, 2012, 2013) have shown noteworthy success by successfully modeling protein structures even in the absence of significant sequence homology. Nevertheless, the performance of traditional threading methods sharply declines when the evolutionary relationship between the query and templates becomes very low (Zheng, Wuyun, *et al.*, 2019), the so-called remote-homology threading scenario.

With significant recent progress in residue-residue contact or distance prediction mediated by sequence co-evolution and deep learning (Jones *et al.*, 2012; Kamisetty *et al.*, 2013; Jones *et al.*, 2015; Li *et al.*, 2019; Greener *et al.*, 2019; Wang *et al.*, 2017; Xu, 2019), predicted contact or distance information becomes a valuable source of additional information for remote-homology threading. Consequently, predicted contact- or distance-based threading has attracted promising attention. For instance, EigenTHREADER (Buchan and Jones, 2017) utilizes contacts

predicted by MetaPSICOV (Jones *et al.*, 2015) by performing contact map threading. map_align (Ovchinnikov *et al.*, 2017) employs iterative double dynamic programming driven by co-evolutionary contact predictor, GREMLIN (Kamisetty *et al.*, 2013). Very recently, CEthreader (Zheng, Wuyun, *et al.*, 2019) and CATHER (Du *et al.*, 2019) perform contact-assisted threading using contacts predicted by ResPre (Li *et al.*, 2019) and MapPred (Wu *et al.*, 2019), respectively, together with sequence profile-based features. DeepThreader (Zhu *et al.*, 2018) goes one step further by incorporating finer-grained distance information instead of contacts for boosting threading performance (Xu and Wang, 2019), but the method is not publicly available. While these methods exploit predicted contact or distance information during threading, often in conjunction with sequential information, they do not consider other non-linear features induced by the predicted inter-residue interactions such as the effect of the residue pairs in the neighborhood of the interacting pairs, which can be used to further improve threading accuracy. Furthermore, the prediction of finer-grained maps by discretizing inter-residue Euclidean distances in dozens of intervals (Xu, 2019; Greener *et al.*, 2019) is a very challenging problem that can suffer from high false-positive rates. A noisy distance map with high false positives can negatively affect threading performance. Although distance maps carry more information than contacts, a few well-calibrated distance intervals may be sufficient for capturing the overall fold-level information to assist threading template selection and alignment, while being noise-resilient.

In this article, we introduce a new distance-based threading method DisCovER (Distance-based Covariational threadER) that effectively integrates information from distance maps and their topological network neighborhood into an iterative double dynamic programming framework to greatly improve threading template selection and alignment. Experimental results show the our new methods performs much better than traditional 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. DisCovER works particularly well for weakly homologous proteins and performs better and better at increasingly challenging threading scenarios compared to currently popular contact-assisted approaches. At one of the most challenging threading situations, DisCovER yields much better average performance than the reported average accuracy of DeepThreader, owing to our effective integration of inter-residue distance information and their neighborhood effect. DisCovER is freely available at https://github.com/Bhattacharya-Lab/DisCovER.

2 Materials and methods

2.1 Feature sets and inter-residue distance maps

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 deep multiple sequence alignments (MSA) (Zhang *et al.*, 2019), and predict profile-based features including secondary structure, solvent accessibility, and torsion angles using SPIDER3 (Heffernan *et al.*, 2017). We also predict inter-residue distance maps by feeding the MSAs to DMPfold (Greener *et al.*, 2019) to obtain the initial distance predictions without any iterative refinement (i.e., rawdistpred.current files) containing 20 distance bins with associated likelihoods between interacting residue pairs having minimum sequence separation of 5. The distance map is then converted to five evenly distributed distance intervals: 6Å, 8Å, 10Å, 12Å, and 14Å by summing up likelihoods for distance bins below specific distance thresholds. For the templates, we use structure-derived profiles, extract secondary structures and solvent accessibility using DSSP (Kabsch and Sander, 1983), and compute torsion angles and inter-residue distance maps directly from the structure.

2.2 Distance-based scoring of a query-template alignment

DisCovER scores a query-template alignment as follows:

$$Z_{final} = Z_{without_dist} + Z_{dist} \tag{1}$$

where Z_{final} is the normalized alignment score for selecting the top-ranked template, $Z_{without_dist}$ is the normalized alignment score based only on profile information, and Z_{dist} is the normalized alignment score using both profile information and inter-residue distance information. In the following, we describe each term in detail.

Stage 1 Scoring profile-based alignment: A profile-based query-templates alignment is scored similar to our recent work (Bhattacharya and Bhattacharya, 2019) as follows:

$$\begin{aligned} S_{without_dist}(i,j) = {} & \sum_{k=1}^{20} \frac{(f_c(i,k) + f_d(i,k))\, L_t(j,k)}{2} + \\ & c_1\delta(SS_i, SS_j) + c_2 \sum_{k=1}^{20} \left(f_s(i,k) + L_q(i,k)\right) + \\ & c_3 E(SA_i, SA_j) + c_4 E(\phi_i, \phi_j) + c_5 E(\psi_i, \psi_j) + \\ & c_6 M(AA_i, AA_j) + c_7 \end{aligned} \tag{2}$$

where the first term defines the sequence profile-profile alignment. $f_c(i,k)$ and $f_d(i,k)$ define the frequency of kth residue at the ith 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, 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 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) and the sequence profile (L_q). 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 torsion angles (ϕ, ψ), respectively. The seventh term corresponds to the hydrophobic residues of the query and the template. c is the weighting parameter adopted from (Bhattacharya and Bhattacharya, 2019). 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 $Z_{without_dist}$ based on $S_{without_dist}$ to assess the quality of each query-template alignment as follows:

$$Z_{without_dist} = \frac{S'_{without_dist} - < S'_{without_dist} >}{\sqrt{< S'^{2}_{without_dist} > - < S'_{without_dist} >^2}} \tag{3}$$

where $S'_{without_dist}$ is normalized by the larger one of the raw alignment score with full and partial alignment lengths, and $< S'_{without_dist} >$ is the average $S'_{without_dist}$ across all templates in the template library. A subset of fifty top-scoring templates is selected for the next stage.

Stage 2 Scoring distance-based alignment: Given the five evenly distributed distance maps for a query (Q) and a template (T), a similarity score is calculated as follows:

$$\begin{aligned} S_d[i',j'] = {} & (w_0 \times \mathrm{Q}_{6Å}[i][i'] \times \mathrm{T}_{6Å}[j][j']) + w_1 \times \\ & \mathrm{Q}_{8Å}[i][i'] \times \mathrm{T}_{8Å}[j][j'] + w_2 \times \mathrm{Q}_{10Å}[i][i'] \times \mathrm{T}_{10Å}[j][j'] + \\ & w_1 \times \mathrm{Q}_{12Å}[i][i'] \times \mathrm{T}_{12Å}[j][j'] + w_0 \times \mathrm{Q}_{14Å}[i][i'] \times \\ & \mathrm{T}_{14Å}[j][j']) \times w_3 \times G(0, sd, sep) \end{aligned} \tag{4}$$

where $\mathrm{Q}_{xÅ}[i][i']$ is the likelihood score of the residue pair i and i' of the query to be within a distance threshold of xÅ; $\mathrm{T}_{xÅ}[j][j']$ is the likelihood score of the residue pair j and j' of the template to be within the distance threshold of xÅ; w_x is the corresponding weight parameter with

w_0=0.1, w_1=0.25, w_2=0.3; w_3 is the weight of the minimum of sequence separation of query residues and template residues with w_3=0.75 for separation=5 and log10(1+separation) for separation >= 6; $G(0, sd, sep)$ is a zero-mean Gaussian function, which is defined by $exp(-sep^2/(2\,sd^2))$, where sep is an absolute difference of sequence separation of query residues and sequence separation of template residues, and sd or standard deviation is a function of the smaller of sequence separation of query residues and sequence separation of template residues (Ovchinnikov *et al.*, 2017). The Smith-Waterman (Smith and Waterman, 1981) dynamic programming is then used to find the optimal alignment between query and template distance maps by maximizing the sum of the Gaussian function, resulting in optimized sum of scores (S_d^{max}).

To further improve the sensitivity of distance-based alignment, we borrow ideas from comparative network analysis. In particular, we improve the initial estimation of the similarity matrix by treating the distance maps as topological networks, where vertices are the set of interacting residues and edges are the set of inter-residue interactions weighted by associated likelihoods, and computing the topological similarity between the query-template networks. We adopt an approach similar to that was originally used in the IsoRank network alignment algorithm (Singh *et al.*, 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 of the distance maps represented by the networks. Following similar principles, we integrate two types of similarities: *connected similarity* and *structural similarity*, both attempting to estimate the topological similarity between the networks by capturing the similarity between the neighborhoods of two nodes that belong to different networks. We describe them below:

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 as follows:

$$S_c[i][j] = \frac{S_d^{max}[i-1][j-1] + S_d^{max}[i+1][j+1]}{2} \qquad (5)$$

where S_d^{max} is the maximum sum of scores.

Structural similarity (S_s) measures the topological similarity between nodes in different topological networks based on the interaction probabilities in the respective networks, such that nodes (residues) involved in conserved interactions are likely to be aligned in the network alignment. It is calculated as follows:

$$S_s[i][j] = \sum_{\substack{a \in N_{G_1}(i) \\ b \in N_{G_2}(j)}} \frac{P_{s_1}(i,a) P_{s_2}(j)(b)}{D(a) D(b)} S_d^{max}[a][b] \qquad (6)$$

where the set of residues of a distance map is denoted as N_G, $P_{s_1}(i,a)$ and $P_{s_2}(j)(b)$ are the likelihood scores of the residue pairs to be within the distance threshold for query and template distance maps, respectively. $D(a) = \sum_{p \in N_{G_1}(a)} P_{s_1}(p,a)$ and $D(b) = \sum_{q \in N_{G_2}(b)} P_{s_2}(q,b)$.

Connected similarity (S_c) and structural similarity (S_s) components are added to the optimized sum of scores (S_d^{max}) as follows:

$$\begin{aligned} S' = \; & S_d^{max} + w_4 \times S_c[i][j] + w_5 \times (w_0 \times S_{s_6Å}[i][j] \\ & + w_1 \times S_{s_8Å}[i][j] + w_2 \times S_{s_10Å}[i][j] \\ & + w_1 \times S_{s_12Å}[i][j] + w_0 \times S_{s_14Å}[i][j]) \end{aligned} \qquad (7)$$

These scores (S') are stored in a similarity matrix and are used to obtain the optimal distance alignment by using the Smith-Waterman dynamic programming algorithm. Based on a current alignment, a second step of dynamic programming is iteratively applied to further update the similarity matrix based on the distance information using a similar updating strategy discussed in (Taylor, 1999). After obtaining the optimal alignment, which maximizes the number of aligned residues and minimizes the number of gaps, the profile score and the gap-score are re-calculated to compute the alignment score (S_{dist}) between the two distance maps using a similar strategy discussed in (Ovchinnikov *et al.*, 2017). The normalized alignment score for each query-template distance map pair is normalized using Z_{dist} as follows:

$$Z_{dist} = \frac{(S'_{dist} - < S'_{dist} >)}{\sqrt{< S'^2_{dist} > - < S'_{dist} >^2}} \qquad (8)$$

where S'_{dist} is the distance map alignment score normalized by the greater one of the raw alignment score with full and partial alignment lengths, and ⟨…⟩ denotes the average of all top-fit templates.

Building full-length 3D models: After selecting the first-ranked template using Equation (1) we use MODELLER (V9.20) (Webb and Sali, 2014) to generate the full-length 3D model of a query protein using the associated query-template alignment.

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

To evaluate remote-homology threading performance, we benchmark our new method DisCovER on four recent datasets curated independently by other groups (Zheng, Wuyun, *et al.*, 2019; Du *et al.*, 2019; Wu and Zhang, 2008). Instead of considering the whole dataset containing both easy and hard test cases, we focus primarily on hard targets in which existing methods have limitations (Zhu *et al.*, 2018). Of note, we follow the target difficulty classification as done by the original curators of the respective datasets. We describe each benchmark dataset below:

The first dataset (hereafter called Dataset-I) consists of 211 hard targets (Zheng, Zhang, *et al.*, 2019) with pairwise sequence identity <30% and length ranging from 59 to 363 residues. On this dataset, DisCovER is compared against profile-based threading methods such as SparkX (Yang *et al.*, 2011), CNFpred (Ma *et al.*, 2012, 2013), MUSTER (Wu and Zhang, 2008), PPAS (Wu and Zhang, 2007), and pGenThreader (Lobley *et al.*, 2009); as well as state-of-the-art contact-assisted approaches including CEthreader (Zheng, Wuyun, *et al.*, 2019), EigenTHREADER (Buchan and Jones, 2017), and map_align (Ovchinnikov *et al.*, 2017); after excluding templates with sequence identity >30% to the query proteins.

The second dataset (hereafter called Dataset-II) comprises of 160 hard targets (Zheng, Wuyun, *et al.*, 2019) with pairwise sequence identity <95% and length ranging from 59 to 370 residues. On this dataset, we compare DisCovER exclusively against contact-assisted threading methods CEthreader, EigenTHREADER, and map_align; after excluding structurally similar templates (measured by TM-align (Zhang and Skolnick, 2005)) with the query proteins. We use four increasingly difficult template-filtering cutoffs with TM-align score of <0.7, <0.6, <0.5, and <0.4 to investigate the effectiveness of the latest threading approaches as structural homology becomes progressively weaker, thereby simulating increasingly harder threading scenarios. It should be noted here that a similar study has been reported in DeepThreader (Zhu *et al.*, 2018), the only other distance-based threading method except DisCovER, albeit on a different dataset that is not readily available. While the

unavailability of DeepThreader method prevents us to perform a head-to-head comparison between DisCovER and DeepThreader, the average performance on this dataset at one of the most challenging threading scenarios may offer some interesting insights.

The third dataset (hereafter called Dataset-III) consists of 131 hard targets (Du *et al.*, 2019) with pairwise sequence identity <25% and length ranging from 50 to 458 residues. We use this set to compare DisCovER against CATHER, a very recent contact-assisted threading method, by running DisCovER locally after excluding templates with sequence identity >30% to the query proteins, and comparing its average performance against the reported results of CATHER as well as EigenTHREADER, map_align, HHpred (Meier and Söding, 2015), SparkX, and MUSTER from (Du *et al.*, 2019).

The fourth dataset (hereafter called Dataset-IV) comprises of 500 test proteins (Wu and Zhang, 2008) with pairwise sequence identity <25% and length ranging from 50 to 633 residues. On this dataset, we compare DisCovER against two of its variants acting as controls: one that does not use distance information at all (hereafter called $\text{DisCovER}^{\text{without_dist}}$), and the other uses distance information only for template selection but not for query-template alignment generation (hereafter called $\text{DisCovER}^{\text{without_dist_aln}}$). Head-to-head comparison between DisCovER and these baseline methods can objectively evaluate the contribution of distance information in threading template identification and alignment when everything else remains the same. We use two template-exclusion cutoffs with TM-align score of <0.6 and <0.5 to study the contribution of distance information in the absence of significant structural homology.

The template libraries for DisCovER, CEthreader, EigenTHREADER, map_align, SparkX, MUSTER, PPAS, pGenThreader are generated from the same set of non-redundant protein chains downloaded from https://zhanglab.ccmb.med.umich.edu/library/ (Yang *et al.*, 2015), containing 70,670 templates. The template library for CNFpred is downloaded from http://raptorx.uchicago.edu/download/, containing 80,020 templates. Here, we evaluate threading performance by comparing the top-ranked full-length predicted 3D models (built using MODELLER) against the experimental structures of the target proteins using the TM-score metric (Zhang and Skolnick, 2004), which ranges from 0 to 1 with higher score indicating better performance and TM-score >0.5 indicating the attainment of correct overall fold (Xu and Zhang, 2010).

3 Results

3.1 Performance on Dataset-I of 211 hard targets

On 211 hard targets, our distance-based threading method DisCovER works much better than all 3 contact-assisted threading methods as well as all 5 profile-based approaches. As shown in Table 1, DisCovER attains an average TM-score of 0.546, which is substantially higher than the next best contact-assisted threading method CEthreader (0.48) and the best among profile-based threading methods SparkX (0.362). The performance improvement for DisCovER is statistically significant at 95% confidence level compared to all other methods. DisCovER also predicts much higher number of correct folds (TM-score >0.5) with a success rate of 67.3%, which is ~22% higher than the success rate of CEthreader and ~47% higher than that of SparkX. It is worth noticing that DisCovER attains more than 1½ times better TM-score and a 3-fold increase in the success rate of identifying the correct folds compared to the best profile-based approach. Even the best contact-assisted threading method CEthreader falls short of achieving an average TM-score of 0.5 and success rate of 50%, whereas DisCovER exceeds both these criteria convincingly.

Table 1. Performance of various threading methods on Dataset-I of 211 hard targets.

Methods	TM-score	*p*-value	#Correct folds	Success rate
DisCovER	**0.546**	-	**142**	**67.3%**
CEthreader	0.480	6.54E-13	94	44.5%
map_align	0.473	1.46E-14	93	44.1%
EigenTHREADER	0.469	2.47E-19	89	42.2%
SparkX	0.362	9.90E-44	42	19.9%
CNFpred	0.333	3.58E-51	32	15.2%
MUSTER	0.332	2.04E-51	34	16.1%
PPAS	0.312	1.05E-55	29	13.7%
pGenThreader	0.293	9.45E-59	21	10.0%

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 predicted models with TM-score >0.5. Values in bold represent the best performance.

Figure 1 shows the head-to-head comparison between DisCovER and the next best contact-assisted threading method CEthreader and the best among profile-based threading methods SparkX. DisCovER attains higher TM-score for 155 and 184 targets compared to CEthreader and SparkX, respectively. Promisingly, DisCovER predicts correct folds (TM-score >0.5) for 117 targets whereas CEthreader fails to correctly fold any of these targets. Furthermore, the TM-score distribution for DisCovER is skewed towards higher TM-score regions compared to CEthreader and SparkX, underscoring the overall superior performance of DisCovER. In summary, the advantage of DisCovER in threading remote-homology proteins over the others is significant.

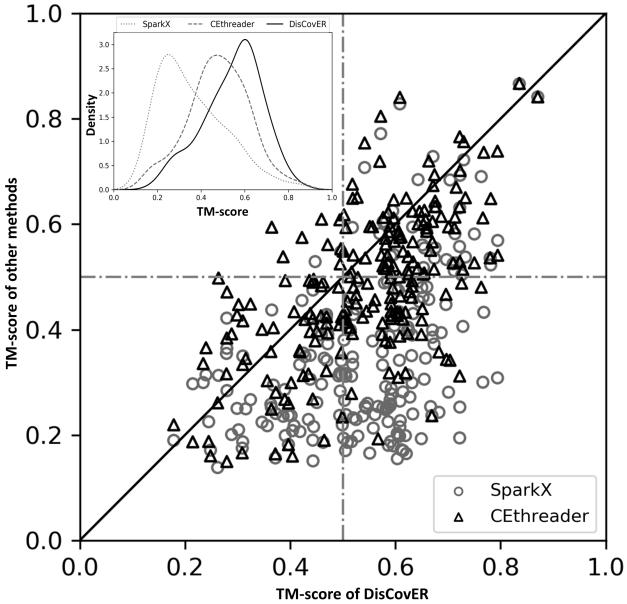

Fig. 1. Head-to-head performance comparison between DisCovER (x-axis) vs. CEthreader and SparkX (y-axis) on Dataset-I of 211 hard targets. The inset shows the TM-score distributions of DisCovER, CEthreader, and SparkX.

3.2 Performance on Dataset-II of 160 hard targets

We further evaluate the performance of DisCovER on 160 hard targets after excluding templates with high structural similarity (measured by TM-align) to conduct a rigorous performance assessment at increasingly difficult threading scenarios. In Table 2, 'TM-align <x' means templates

Table 2. Benchmark results on Dataset-II of 160 hard targets.

	TM- align <0.7				TM- align <0.6			
	TM-score	*p*-value	#Correct folds	Success rate	TM-score	*p*-value	#Correct folds	Success rate
DisCovER	**0.548**	-	**109**	**68.1%**	**0.516**	-	**95**	**59.4%**
CEthreader	0.476	1.27E-11	72	45.0%	0.440	9.25E-13	52	32.5%
EigenTHREADER	0.467	2.98E-16	65	40.6%	0.445	1.12E-12	51	31.9%
map_align	0.469	8.26E-13	75	46.9%	0.444	7.92E-10	65	40.6%
	TM- align <0.5				TM- align <0.4			
	TM-score	*p*-value	#Correct folds	Success rate	TM-score	*p*-value	#Correct folds	Success rate
DisCovER	**0.472**	-	**74**	**46.3%**	**0.409**	-	**35**	**21.9%**
CEthreader	0.389	2.73E-16	9	5.6%	0.330	2.55E-17	1	0.6%
EigenTHREADER	0.396	1.93E-14	13	8.1%	0.330	1.64E-13	1	0.6%
map_align	0.388	1.76E-14	27	16.9%	0.345	2.19E-08	14	8.8%

'TM-align <x' illustrates templates with TM-align score less than x with the query protein are only considered, and the rest are excluded. The column "*p*-value" represents one sample t-test's *p*-value of the TM-score difference compared to DisCovER. The column "#Correct folds" represents the number of predicted models with TM-score >0.5. Values in bold represent the best performance.

with TM-align score less than x to the query proteins are only considered, and the rest of the templates are excluded. As reported in Table 2, our distance-based threading method DisCovER statistically significantly outperforms all 3 state-of-the-art contact-assisted threading methods across all template-filtering cutoffs <0.7, <0.6, <0.5, and <0.4. Considering templates with TM-align <0.7 (and <0.6) to the query proteins, DisCovER attains a much better average TM-score of 0.548 (and 0.516) compared to 0.476 (and 0.44) for CEthreader, 0.467 (and 0.445) for EigenTHREADER, and 0.469 (and 0.444) for map_align. Moreover, DisCovER successfully predicts correct folds with a success rate of ~68% (and ~59%) for TM-align <0.7 (and <0.6) with query proteins, which is ~1.5 (and ~1.8), ~1.7 (and ~1.9), and ~1.4 (and ~1.45) times better than the success rates of CEthreader, EigenTHREADER, and map_align, respectively. The performance advantage of DisCovER becomes even more pronounced at stringent structural similarity cutoffs. In particular, when TM-align cutoff is <0.5, DisCovER predicts correct folds with a success rate of ~46%, which is ~8.3, ~5.7, and ~2.7 times higher than that of CEthreader, EigenTHREADER, and map_align, respectively. The trend continues when TM-align cutoff <0.4, in which DisCovER successfully predicts correct folds with a success rate of ~22%, which is orders of magnitude higher compared to the others. The performance advantage of DisCovER is also statistically significant at 95% confidence level for all other methods across all template-filtering cutoffs. Figure 2 shows head-to-head performance comparisons at various structural similarity cutoffs. For TM-align cutoffs of <0.7, <0.6, <0.5, <0.4; DisCovER attains higher TM-scores for 117, 122, 121, 131 targets than CEthreader; 125, 121, 118, 116 targets than EigenTHREADER; and 118, 115, 119, 110 targets than map_align; respectively. The results highlight the significant advantage of DisCovER.

One may argue that our distance-based threading method DisCovER has better performance than contact-assisted approaches simply because distance maps contain more fine-grained information compared to contact maps. This is not true when we look at the average TM-score reported by another distance-based threading method DeepThreader (Zhu *et al*., 2018), which uses a much finer-grained distance map discretization than ours (12 bins for DeepThreader vs. 5 for DisCovER). DeepThreader has reported an average TM-score of 0.39 for a stringent template-filtering cutoff of 0.5, whereas DisCovER attains a much better average TM-score of 0.472 for a template-filtering cutoff of 0.5. We note here that we are unable to perform a head-to-head comparison between DisCovER and DeepThreader due to the unavailability of DeepThreader method and the lack of ready access to its underlying test set. Thus, the performance of DisCovER and DeepThreader is not easily comparable due to the differences in the underlying test dataset, predicted distance maps, and possibly template library. Nevertheless, a noticeably higher average TM-score attained by DisCovER compared to DeepThreader at one of the most stringent template-filtering cutoffs suggests that the calibration of distance intervals adopted in DisCovER coupled with its induced topological network neighborhood effect may be advantageous for threading, particularly for very remotely homologous proteins.

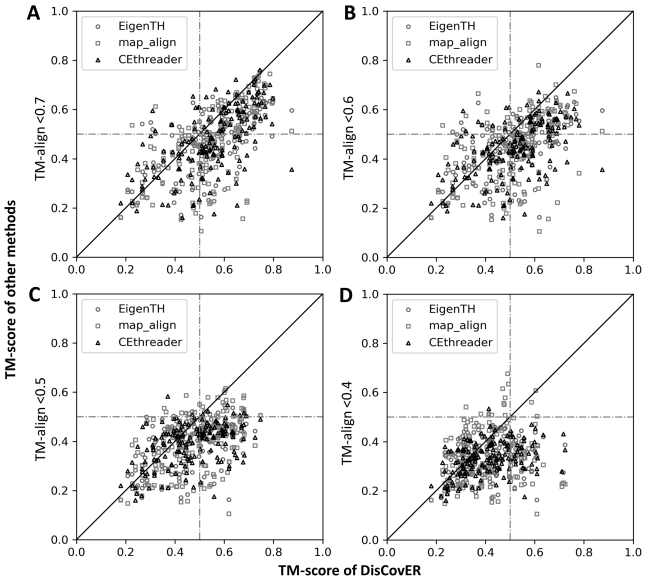

Fig. 2. Head-to-head performance compactions between DisCovER (x-axis) vs. other tested contact-assisted threading methods (y-axis): CEthreader, EigenTHREADER (EigenTH), and map_align on Dataset-II of 160 hard targets, at various template-filtering cutoffs.

Table 3. Benchmark results on Dataset-IV of 500 targets.

	TM- align <0.6				TM- align <0.5			
	TM-score	*p*-value	#Correct folds	Success rate	TM-score	*p*-value	#Correct folds	Success rate
DisCovER	**0.522**	**-**	**309**	**61.8 %**	**0.465**	**-**	**213**	**42.6 %**
DisCovER^{without_dist_aln}	0.491	3.09E-14	252	50.4 %	0.427	7.56E-15	157	31.4 %
DisCovER^{without_dist}	0.464	7.38E-32	220	44.0 %	0.386	1.23E-45	95	19.0 %

3.3 Performance on Dataset-III of 131 hard targets

The performance of DisCovER is further benchmarked against a recent contact-assisted threading method CATHER (Du *et al.*, 2019) by running DisCovER over 131 hard targets used in CATHER and directly comparing the average TM-score with the performance reported in the published work of CATHER (Du *et al.*, 2019) over the same set. Consistent with our results on other datasets, DisCovER significantly outperforms CATHER by attaining an average TM-score of 0.5369 as opposed to 0.4561 of CATHER (Supplementary Table S1). DisCovER also greatly outperforms the reported results of other profile-based methods including HHpred (0.3268), SparkX (0.3485), and MUSTER (0.3593) as well as other contact-assisted approaches including EigenTHREADER (0.3861) and map_align (0.383). The results continue to demonstrate the competitive advantage of DisCovER over current threading methods.

3.4 Contribution of distance information

To evaluate the contribution of distance information, we use dataset IV to perform head-to-head performance comparison between DisCovER and its two variants acting as controls: DisCovER^{without_dist} and DisCovER^{without_dist_aln}, where both variants share the same methodology but the former does not use distance information at all and the later uses distance information only for template selection but not for query-template alignment generation. To make a fair comparison, the same template library, sequence database, and the same set of features are used for all variants. We choose two template-filtering cutoffs with TM-align <0.6 and <0.5 to study the contribution of distance information on threading performance for challenging protein targets. As shown in Table 3, DisCovER delivers much better performance than both variants DisCovER^{without_dist} and DisCovER^{without_dist_aln} in terms of average TM-score and prediction of the number of correct folds (i.e., success rate). The improvement in TM-score attained by DisCovER is statistically significant at 95% confidence level irrespective of template-filtering cutoffs with the majority of targets attaining higher TM-score compared to the variants (Supplementary Figure S1). The success rate of predicting the correct fold for DisCovER is >15% and >10% higher than DisCovER^{without_dist} and DisCovER^{without_dist_aln}, respectively. DisCovER is able to correctly fold 74 additional targets compared to the two variants (Supplementary Figure S2). Of note, DisCovER^{without_dist_aln} consistently attains better performance compared to DisCovER^{without_dist}. The performance difference is particularly noticeable at a more stringent template-filtering cutoff (TM-align < 0.5) with DisCovER^{without_dist_aln} resulting in more than 1½-fold increase in the success rate of identifying correct folds compared to DisCovER^{without_dist}. That is, distance information not only significantly improves template selection for weakly homologous targets, but also leads to much better alignment of the selected top template. The accumulative improvement in better template identification and superior alignment quality ultimately results in improved threading performance for our distance-based method DisCovER.

3.5 Effect of homologous information

The performance of DisCovER is weakly correlated with the number of effective sequence homologs, as quantified by Nf (Zhang *et al.*, 2019). As shown in Supplementary Figure S3, the Pearson correlation between TM-score attained by DisCovER and Nf is 0.145 over 211 hard targets of Dataset-I. In this set, DisCovER predicts models with an average TM-score >0.5 for 101 targets having Nf <50. Similarly, we observe weak correlation (maximum Pearson correlation of 0.277) between modeling accuracy of DisCovER and Nf for other benchmark datasets, indicating DisCovER can predict quality models with few available sequences.

4 Conclusion

This article presents DisCovER, a new protein threading method that effectively utilizes the covariational signal encoded in predicted distance maps and their topological network neighborhood to significantly improve threading template selection and alignment for weakly homologous proteins. Experimental results show that our method yields much better accuracy than existing threading methods, including traditional profile-based methods and latest contact-assisted approaches. DisCovER works particularly well in challenging threading scenarios compared to state-of-the-art contact-assisted approaches such as CEthreader, map_align, and EigenTHREADER. As we raise the bar for low-homology threading by applying increasingly stringent template-filtering cutoffs to exclude structurally similar templates, performance improvement of DisCovER gets more and more pronounced compared to currently popular contact-assisted approaches. At one of the most stringent template-filtering cutoffs, DisCovER attains much better average performance than the reported average accuracy of DeepThreader, the only other distance-based threading method existing in the current literature.

Our effective integration of inter-residue distance information and their neighborhood effect captured through the similarity of the topological networks induced by the distance map is useful for both protein alignment and template selection, cooperatively contributing to the overall improved performance of DisCovER. 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 across all the benchmark datasets. This suggests that our distance-based coevolutionary threading method DisCovER is well-suited for weakly homologous targets. These results demonstrate that it is now feasible to successfully extend threading for many more protein sequences that were previously not amenable to template-based modeling.

Acknowledgements

This work was made possible in part by a grant of high performance computing resources and technical support from the Alabama Supercomputer Authority.

Funding

This work has been partially supported by an Auburn University new faculty start-up grant to DB.

Conflict of Interest: none declared.

References

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*, **25**, 3389–3402.

Berman,H.M. *et al.* (2000) The Protein Data Bank. *Nucleic Acids Res*, **28**, 235–242.

Bhattacharya,S. and Bhattacharya,D. (2019) Does inclusion of residue-residue contact information boost protein threading? *Proteins: Structure, Function, and Bioinformatics*, **87**, 596–606.

Buchan,D.W.A. and Jones,D.T. (2017) EigenTHREADER: analogous protein fold recognition by efficient contact map threading. *Bioinformatics*, **33**, 2684–2690.

Chen,C.-C. *et al.* (2019) TOPAS: network-based structural alignment of RNA sequences. *Bioinformatics*, **35**, 2941–2948.

Dill,K.A. and MacCallum,J.L. (2012) The Protein-Folding Problem, 50 Years On. *Science*, **338**, 1042–1046.

Du,Z. et al. (2019) CATHER: a novel threading algorithm with predicted contacts. *Bioinformatics*. doi: 10.1093/bioinformatics/btz876.

Greener,J.G. *et al.* (2019) Deep learning extends de novo protein modelling coverage of genomes using iteratively predicted structural constraints. *Nat Commun*, **10**, 1–13.

Heffernan,R. *et al.* (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**, 2842–2849.

Henikoff,S. and Henikoff,J.G. (1994) Position-based sequence weights. *Journal of Molecular Biology*, **243**, 574–578.

Jones,D.T. *et al.* (2015) MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. *Bioinformatics*, **31**, 999–1006.

Jones,D.T. *et al.* (2012) PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, **28**, 184–190.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Kamisetty,H. *et al.* (2013) Assessing the utility of coevolution-based residue–residue contact predictions in a sequence- and structure-rich era. *PNAS*, **110**, 15674–15679.

Li,Y. *et al.* (2019) ResPRE: high-accuracy protein contact prediction by coupling precision matrix with deep residual neural networks. *Bioinformatics*, **35**, 4647–4655.

Lobley,A. *et al.* (2009) pGenTHREADER and pDomTHREADER: new methods for improved protein fold recognition and superfamily discrimination. *Bioinformatics*, **25**, 1761–1767.

Ma,J. *et al.* (2012) A conditional neural fields model for protein threading. *Bioinformatics*, **28**, i59–i66.

Ma,J. *et al.* (2013) Protein threading using context-specific alignment potential. *Bioinformatics*, **29**, i257–i265.

Meier,A. and Söding,J. (2015) Automatic Prediction of Protein 3D Structures by Probabilistic Multi-template Homology Modeling. *PLOS Computational Biology*, **11**, e1004343.

Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, **48**, 443–453.

Ovchinnikov,S. *et al.* (2017) Protein structure determination using metagenome sequence data. *Science*, **355**, 294–298.

Singh,R. *et al.* (2008) Global alignment of multiple protein interaction networks with application to functional orthology detection. *PNAS*, **105**, 12763–12768.

Smith,T.F. and Waterman,M.S. (1981) Identification of common molecular subsequences. *Journal of Molecular Biology*, **147**, 195–197.

Taylor,W.R. (1999) Protein structure comparison using iterated double dynamic programming. *Protein Science*, **8**, 654–665.

Wang,S. *et al.* (2017) Accurate De Novo Prediction of Protein Contact Map by Ultra-Deep Learning Model. *PLOS Computational Biology*, **13**, e1005324.

Webb,B. and Sali,A. (2014) Protein Structure Modeling with MODELLER. In, Kihara,D. (ed), *Protein Structure Prediction*, Methods in Molecular Biology. Springer, New York, NY, pp. 1–15.

Wu,Q. *et al.* (2020) Protein contact prediction using metagenome sequence data and residual neural networks. *Bioinformatics*, **36,** 41-48.

Wu,S. and Zhang,Y. (2007) LOMETS: A local meta-threading-server for protein structure prediction. *Nucleic Acids Res*, **35**, 3375–3382.

Wu,S. and Zhang,Y. (2008) MUSTER: Improving protein sequence profile–profile alignments by using multiple sources of structure information. *Proteins: Structure, Function, and Bioinformatics*, **72**, 547–556.

Xu,J. (2019) Distance-based protein folding powered by deep learning. *PNAS*, **116**, 16856–16865.

Xu,J. and Wang,S. (2019) Analysis of distance-based protein structure prediction by deep learning in CASP13. *Proteins: Structure, Function, and Bioinformatics*, **87**, 1069–1081.

Xu,J. and Zhang,Y. (2010) How significant is a protein structure similarity with TM-score = 0.5? *Bioinformatics*, **26**, 889–895.

Yang,J. *et al.* (2015) The I-TASSER Suite: protein structure and function prediction. *Nature Methods*, **12**, 7–8.

Yang,Y. *et al.* (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**, 2076–2082.

Zhang,C. *et al.* (2019) DeepMSA: constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins. *Bioinformatics*. doi: 10.1093/bioinformatics/btz863.

Zhang,Y. and Skolnick,J. (2004) Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, **57**, 702–710.

Zhang,Y. and Skolnick,J. (2005) TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res*, **33**, 2302–2309.

Zheng,W., Wuyun,Q., *et al.* (2019) Detecting distant-homology protein structures by aligning deep neural-network based contact maps. *PLOS Computational Biology*, **15**, e1007411.

Zheng,W., Zhang,C., *et al.* (2019) LOMETS2: improved meta-threading server for fold-recognition and structure-based function annotation for distant-homology proteins. *Nucleic Acids Res*, **47**, W429–W436.

Zhu,J. *et al.* (2018) Protein threading using residue co-variation and deep learning. *Bioinformatics*, **34**, i263–i273.