BIOINFORMATICS

Vol. 26 ISMB 2010, pages i294–i300
doi:10.1093/bioinformatics/btq192

# Low-homology protein threading

Jian Peng and Jinbo Xu*

Toyota Technological Institute at Chicago, IL 60637, USA

**ABSTRACT**

**Motivation:** The challenge of template-based modeling lies in the recognition of correct templates and generation of accurate sequence-template alignments. Homologous information has proved to be very powerful in detecting remote homologs, as demonstrated by the state-of-the-art profile-based method HHpred. However, HHpred does not fare well when proteins under consideration are low-homology. A protein is low-homology if we cannot obtain sufficient amount of homologous information for it from existing protein sequence databases.

**Results:** We present a profile-entropy dependent scoring function for low-homology protein threading. This method will model correlation among various protein features and determine their relative importance according to the amount of homologous information available. When proteins under consideration are low-homology, our method will rely more on structure information; otherwise, homologous information. Experimental results indicate that our threading method greatly outperforms the best profile-based method HHpred and all the top CASP8 servers on low-homology proteins. Tested on the CASP8 hard targets, our threading method is also better than all the top CASP8 servers but slightly worse than Zhang-Server. This is significant considering that Zhang-Server and other top CASP8 servers use a combination of multiple structure-prediction techniques including consensus method, multiple-template modeling, template-free modeling and model refinement while our method is a classical single-template-based threading method without any post-threading refinement.

**Contact:** jinboxu@gmail.com

## 1 INTRODUCTION

Template-based modeling (i.e. homology modeling and protein threading) is becoming more powerful and important for structure prediction along with the PDB growth and the improvement of prediction protocols. Current PDB may contain all templates for single-domain proteins according to the seminal studies in Zhang and Skolnick (2005a). This implies that the structures of many new proteins can be predicted using template-based methods.

The error of a template-based model comes from template selection and sequence-template alignment, in addition to the structure difference between the sequence and template. At higher sequence identity (>50%), template-based models can be accurate enough to be useful in virtual ligand screening (Bjelic and Aqvist, 2004; Caffrey *et al*., 2005), designing site-directed mutagenesis experiments (Skowronek *et al*., 2006; Wells *et al*., 2006), small ligand docking prediction, and function prediction (Baker and Sali, 2001; Skolnick *et al*., 2000). When sequence identity is below 30%, it is difficult to recognize the best template and generate accurate sequence-template alignments, so the resultant models have a wide range of accuracies (Chakravarty *et al*., 2008; Sanchez *et al*., 2000). Pieper *et al*. have shown that 76% of all the models in MODBASE are from alignments in which the sequence and template share <30% sequence identity (Pieper *et al*., 2006). Therefore, to greatly enlarge the pool of useful models, it is essential to improve fold recognition and alignment method for the sequence and template with <30% sequence identity. Considering that currently there are millions of proteins without experimental structures, even a slight improvement in prediction accuracy can have a significant impact on large-scale structure prediction and its applications. As reported in Melo and Sali (2007), even 1% improvement in the accuracy of fold assessment for the ∼4.2 million models in MODBASE can correctly identify ∼42 000 more models.

The alignment accuracy is determined by a scoring function used to drive sequence-template alignment. When the sequence and template are not close homologs, their alignment can be significantly improved by incorporating homologous information (i.e. sequence profile) into the scoring function. HHpred (Soding, 2005), possibly the best profile-based method, is such a representative. HHpred uses only sequence profile and predicted secondary structure for remote homolog detection. It works very well when proteins under consideration have a large amount of homologous information in the public sequence databases, but not as well when proteins under consideration are low-homology. A protein is low-homology if there is no sufficient homologous information available for it in the sequence databases (see Section 2 for quantitative definition). Many threading methods, such as MUSTER (Wu and Zhang, 2008), Phyre2 (Kelley and Sternberg, 2009) and SPARKS/SP3/SP5 (Zhang *et al*., 2004, 2008; Zhou and Zhou 2004, 2005), aim at going beyond profile-based methods by combining homologous information with a variety of structure information. However, recent CASP evaluations (Moult *et al*., 2005, 2007) demonstrate that HHpred actually is as good as if not better than these threading methods. Clearly, it is very challenging to outperform HHpred by simply adding structure information into template-based methods. In fact, Ginalski *et al*. (2005) claimed that 'presently, the advantage of including the structural information in the fitness function cannot be clearly proven in benchmarks'.

This article describes a new scoring function for protein threading. In this function, the relative importance of structure information is determined according to the amount of homologous information available. When proteins under consideration are low-homology, our method will rely more on structure information; otherwise, homologous information. This method enables us to significantly advance template-based modeling over profile-based methods such as HHpred, especially for low-homology proteins.

The capability of predicting low-homology proteins without close homologs in the PDB is particularly important because (i) a large portion of proteins in the PDB, which will be used as templates, belong to this class; and (ii) a majority number of the Pfam (Finn *et al*., 2008; Sammut *et al*., 2008) families without solved structures

*To whom correspondence should be addressed.

are low-homology (see Section 2 for exact numbers). Therefore, to predict structure for proteins in Pfam using templates, it is essential to have a method that can work well on low-homology proteins. In addition, the class of low-homology proteins may represent a substantial portion of metagenomics sequences of microbes (e.g. *Staphylococcus aureus*) generated from numerous metagenomic projects. It is very challenging to predict structure of a low-homology protein because (i) its sequence profile is not diverse enough to link it to remote homologs in the PDB; and (ii) its predicted secondary structure usually has low accuracy as secondary structure is usually predicted from homologous information.

Experimental results indicate that our method greatly outperforms the best profile-based method HHpred and the top CASP8 servers on low-homology proteins. Tested on the CASP8 hard targets, our method also outperforms nine of the top 10 CASP8 servers and is very close to the best Zhang-Server (Zhang, 2009). This is significant considering that the top CASP8 servers use a combination of multiple structure prediction techniques including consensus method, multiple-template modeling, template-free modeling and model refinement while our method is a classical single-template-based threading method without any post-threading refinement.

## 2 METHODS

### 2.1 NEFF: measuring the amount of homologous information

NEFF is not a new concept. It has already been used by PSI-BLAST (Altschul *et al*., 1997) to measure the amount of homologous information available for a protein. The relationship between NEFF and the modeling capability of a profile-based method has also been studied before (Casbon and Saqi, 2004; Sadreyev and Grishin, 2004). NEFF can be interpreted as the effective number of non-redundant homologs of a given protein and be calculated from the multiple-alignment of the homologs. The homologs are detected in the NCBI NR database by PSI-BLAST (five iterations and $E$-value 0.001). NEFF is calculated as the exponential of entropy averaged over all columns of the multiple-alignment, so in this sense NEFF can also be interpreted as the entropy of a sequence profile derived from the multiple-alignment. NEFF is a real value ranging from 1 to 20. A protein with small NEFF is low-homology since we cannot obtain sufficient homologous information for it from existing protein sequence databases.

The Pfam (version: 23.0) contains ∼10 000 families covering ∼75% protein sequences in UniProt (Sammut *et al*., 2008). Among the ∼6600 Pfam families without solved structures, ∼90, ∼78, ∼58 and ∼33% of them have NEFF smaller than 6, 5, 4 and 3, respectively.[1] Among the ∼18 000 HHpred templates (i.e. a set of representative structures in the PDB), ∼36% of them have NEFF <6. Later we will show that when either the sequence or template has NEFF ≤6, our method can generate much better alignments than HHpred. There are also ∼25% protein sequences in UnitProt not covered by the Pfam database. Many of these sequences are singletons (i.e. products of orphan genes) and thus, have NEFF = 1. In the foreseeable future, many of the low-homology proteins or protein families (i.e. NEFF ≤6) will not have solved structures. Therefore, to elucidate the structure of these proteins (or protein families), it is important to develop a protein threading method that can work well on low-homology proteins.

### 2.2 Method for protein alignment

Existing protein threading methods use a linear scoring function to guide sequence-template alignment. A linear function cannot accurately model correlation among protein features, although it has been observed that many features are correlated, e.g. secondary structure versus solvent accessibility. A linear function fixes the relative importance of various protein features without taking into consideration the special properties of proteins under consideration. However, the importance of structure information is not uniform across all threading instances. When the sequence and template are very similar at sequence (profile) level, using structure information may introduce noise. When the sequence and template are distantly related, structure information becomes more important.

We have recently developed a non-linear scoring function for protein threading (Peng and Xu, 2009). This scoring function measures the sequence-template similarity using a set of regression trees, which take as input protein features and output the log-likelihood of an alignment state (i.e. match or gap). A regression tree consists of many paths, each specifying a rule to calculate the probability of an alignment state. One path can be as simple as 'if (mutation score $< -50$), then the log-likelihood of a match state is $\ln 0.9$' or as complex as 'if ($-50 <$ mutation score $< -10$) and (secondary-structure score $>0.9$) and (solvent accessibility score $> 0.6$), then the log-likelihood of a match state is $\ln 0.7$'. Thus, a regression tree can model the non-linear relationship between an alignment state and protein features. Here we briefly describe this method as follows.

Let $s$ denote the target protein (i.e. sequence) and its associated features, e.g. sequence profile, predicted secondary-structure and solvent accessibility. Let $t$ denote the template and its associated information, e.g. position-specific scoring matrix, solvent accessibility and secondary structure. Let $X=\{M, I_s, I_t\}$ be a set of three possible alignment states. Meanwhile, $M$ indicates that two positions are aligned and $I_s$ and $I_t$ indicate insertion at sequence and template, respectively. Let $a=\{a_1, a_2, \ldots, a_L\}$ ($a_i \in X$) denote an alignment between $s$ and $t$ where $a_i$ represents the state at position $i$. Our threading model defines the conditional probability of $a$ given $s$ and $t$ as follows:

$$p(a|s,t)=\frac{\exp\left(\sum_i F\left(a_{i-1}\rightarrow a_i|s,t\right)\right)}{Z(s,t)}.$$

Meanwhile, $Z(s,t)$ is a normalizing factor and $F\left(a_{i-1}\rightarrow a_i|s,t\right)$ is a function that calculates the log-likelihood of the state transition from $a_{i-1}$ to $a_i$ given target and template features at position $i$. To model nonlinear relationship between an alignment state and protein features, we represent $F(a_{i-1}\rightarrow a_i|s,t)$ as a linear combination of regression trees. Each regression tree is a nonlinear function of protein features, so the final threading scoring function is non-linear. This model is much more powerful than existing methods because a state transition in this model depends on a complex function of protein features while existing methods use only a linear function. Since this method considers only state transition between two adjacent positions, the optimal alignment can still be efficiently calculated using dynamic programming.

*2.2.1 Features for a match state* In addition to the features (profile similarity, secondary-structure similarity, solvent accessibility similarity and environmental fitness score) described in our previous work (Peng and Xu, 2009), we use the following extra information to estimate the probability of one template position being aligned to one target position.

In order to determine the relative importance of homologous and structure information, the NEFF values of both the sequence and template are used as features. When NEFF is large, our threading method will count more on homologous information, otherwise on structure information.

We use the CC50 matrix developed by Kihara group (Tan *et al*., 2006) to calculate similarity between the sequence and template. This matrix is a statistical potential-based amino acid similarity matrix, originally designed for aligning distantly related protein sequences. One element CC50[$a$][$b$] in this matrix is the similarity score between two amino acids $a$ and $b$, which is computed as the correlation coefficient of the pairwise contact potentials of these two amino acids.

We also use a structure-based substitution matrix (Prlic *et al*., 2000; Tan *et al*., 2006) to improve alignment accuracy when the sequence and template

[1]The NEFF of a Pfam family and an HHpred template is directly taken from the HHpred web site. We can also compute NEFF using the HHpred package.

are distantly related. This matrix is more sensitive than BLOSUM in remote homolog detection. This scoring matrix is derived by a similar procedure as the BLOSUM matrices (Henikoff and Henikoff, 1992, 1993) are done, based upon the structure alignments of structurally similar protein pairs.

*2.2.2 Features for a gap state* The gap event is related to multiple factors. Some studies indicate that a gap event is related to its local sequence and structure context. For example, SSALN (Qiu and Elber, 2006) uses a context-specific gap penalty model, in which a gap event depends on secondary structure and solvent accessibility. Other methods, such as HHpred and Ellrott *et al*. (Ellrott *et al*., 2007) use a position-specific gap penalty model, which contains evolutionary information of a protein.

In our previous work (Peng and Xu, 2009), only context-specific gap penalty is used. In this work, we use both context-specific and position-specific gap penalty and then use NEFF to determine their relative importance. If NEFF is large, we will rely more on position-specific gap penalty (i.e. homologous information); otherwise, context-specific gap penalty (i.e. structure information). To calculate the position-specific gap penalty of a protein, we run PSI-BLAST on it (with five iterations and $E$-value 0.001) against the NCBI NR database and generate a multiple sequence alignment. Then we calculate the probability of a gap event at each residue as the ratio between the number of the gap events and the number of sequences in the multiple sequence alignment.

For context-specific gap penalty, we estimate the occurring probability of an insertion at the template using secondary-structure type, solvent accessibility, amino acid identity and hydropathy count (Do *et al*., 2006). In addition, we use a binary value to indicate if a residue is in the core region or not. A core residue is usually more conserved and shall be. Similarly, we estimate the occurring probability of an insertion at the target using predicted secondary, predicted solvent accessibility, amino acid identity and hydropathy count.

We train our threading model by maximizing the occurring probability of a set of reference alignments. See Peng and Xu (2009) for a detailed account.

*2.2.3 Geometric constraints* When the sequence and template are not close homologs, their alignment usually contains displaced gap opening or ending positions. Even a single displaced gap in an alignment may result in a big quality drop of the resultant 3D model. The template provides some geometric information that can be used to improve alignment accuracy. Suppose that two adjacent sequence positions are aligned to two template positions $j_1$ and $j_2 (j_2 > j_1 + 1)$, respectively. Since the distance between two adjacent $C_\alpha$ atoms is around 3.8 Å, the two $C_\alpha$ atoms at $j_1$ and $j_2$ should not be far apart. To tolerate some alignment errors, we use 7 Å (instead of 3.8 Å) as the distance threshold for such two $C_\alpha$ atoms. We enforce this physical constraint when generating the optimal alignment between the sequence and template. All the alignments violating this physical constraint is discarded.[2] Our experiments indicate that by applying this constraint, we can improve alignment accuracy dramatically for some threading instances.

## 2.3 Training datasets

We choose 66 protein pairs from the PDB as the training set and 50 pairs as the validation set. The NEFF (i.e. the diversity of sequence profiles) values of these 66 pairs of proteins are distributed uniformly between 1 and 11. This is very important in order to avoid structure information being dominated by homologous information. In the training set, 46 pairs are in the same fold but different superfamily level by the SCOP classification (Murzin *et al*., 1995). The other 20 pairs are in the same superfamily but different family level. Any two proteins in the training and validation set have sequence identity <30%. The proteins used for model training and validation have no high sequence identity (<30%) with the proteins in the Prosup (Lackner *et al*., 2000) and SALIGN (Marti-Renom *et al*., 2004) benchmarks and the CASP8 targets. We use TM-align (Zhang and Skolnick, 2005b) to build a reference alignment for a protein pair in SALIGN.

## 2.4 Method for template selection

After aligning a given sequence (i.e. target) to all the templates in the database, we need to pick up the best alignment, from which we can build a 3D model for the target. We use a neural network to predict the quality, measured by TM-score[3] (Zhang and Skolnick, 2007), of the 3D model built by MODELLER from a sequence-template alignment and then use the predicted quality to rank all the alignments for a given target. In this work, we predict the TM-score using the following alignment-dependent features: sequence identity, distribution of various per-position scores such as mutation score, solvent accessibility score, secondary-structure similarity score and distribution of gap sizes. In addition, we feed the NEFF values of both the sequence and template into our neural network, in order to determine the relative importance of homologous and structure information.

We trained our template selection method using the data set generated by RAPTOR for both CASP6 and CASP7 targets. Tested on these targets (using cross-validation), the absolute prediction error of TM-score is ∼0.045 on average (data not shown). The correlation coefficient between the predicted TM-score and the real one is above 0.9 on all alignments and 0.8 on low-quality ones (data not shown).

# 3 RESULTS

## 3.1 Performance on public benchmarks

We tested our method on two public benchmarks: Prosup (Lackner *et al*., 2000) and SALIGN (Marti-Renom *et al*., 2004), which contain 127 and 200 protein pairs, respectively. On average, two proteins in a pair share 20% sequence identity and 65% of structurally equivalent C$\alpha$ atoms superposed with RMSD 3.5 Å. The SALIGN set is much more challenging than Prosup, as the former includes many pairs of proteins of very different sizes.

We evaluate our method using both reference-dependent and reference-independent alignment accuracy. The reference-dependent alignment accuracy is calculated as the percentage of correctly aligned positions judged by reference alignments, which are generated by structural alignment programs. To evaluate the reference-independent alignment accuracy, we first build a 3D model for the sequence in a protein pair using MODELLER (Sali, 1995) from its alignment to the template and then evaluate the quality of the resultant 3D model using TM-score (Zhang and Skolnick, 2005b). Since our ultimate goal is to predict 3D structure for a target, reference-independent alignment accuracy is a better measurement than reference-dependent alignment accuracy.

*3.1.1 Reference-dependent alignment accuracy* As shown in Table 1, our method shows a significant advantage over the others. The absolute improvement over our own RAPTOR threading program (Xu *et al*., 2003) is at least 24%. Our method is also better than the CASP-winning methods SP3 and SP5 by 16.5% (14.4%) and 10.7% (7.9%) on ProSup (SALIGN), respectively. The results of SPARKS/SP3/SP5 are taken from Zhang *et al*. (2008).

*3.1.2 Reference-independent alignment accuracy* The models generated by our new method in total have TM-score 66.77 and

[2]The optimal alignment satisfying the physical constraint can still be efficiently calculated using dynamic programming.

[3]TM-score evaluates the quality of a model by comparing it to the native structure and yields a number between 0 and 1. The higher the number, the better quality the model has.

**Table 1.** Reference-dependent alignment accuracy (%) on Prosup and SALIGN

| ProSup | | SALIGN | |
|---|---|---|---|
| Methods | Acc | Methods | Acc |
| PSIBLAST | 35.60 | PSIBLAST | 26.10 |
| ContraAlign | 52.79 | ContraAlign | 44.38 |
| SPARKS | 57.20 | SPARKS | 53.10 |
| SSALGN | 58.30 | SALIGN | 56.40 |
| RAPTOR | 61.30 | RAPTOR | 40.20 |
| SP3 | 65.30 | SP3 | 56.30 |
| SP5 | 68.70 | SP5 | 59.70 |
| HHpred | 69.04 | HHpred | 62.98 |
| **Our work** | **76.08** | **Our work** | **64.40** |

132.85 on Prosup and SALIGN, respectively. By contrast, HHpred achieves TM-score 56.44 and 119.83 on Prosup and SALIGN, respectively. Our method is better than HHpred by 18.3 and 10.9% on ProSup and SALIGN, respectively. A Student's $t$-test indicates that our method excels HHpred with $P$-values being $3.77E-11$ and $9.83E-13$, respectively.

To examine the performance of our method and HHpred with respect to the amount of homologous information, we divide the test protein pairs in the ProSup and SALIGN sets into 10 groups according to their NEFF values: [1,2), [2,3),…, [9,10), [10,20]. The NEFF of a protein pair is defined as the minimum NEFF of the sequence and template. Out of the 327 test protein pairs, 15, 26, 53, 72 and 114 pairs have NEFF smaller than 2, 3, 4, 5 and 6, respectively.

Then we calculate the average reference-independent alignment accuracy (measured by TM-score) of all the pairs in each group. As shown in Figure 1, when either the sequence or template has a small NEFF (<6), on average our method can generate much better 3D models than HHpred. When NEFF <2, the model quality of our method is almost 100% better than HHpred. When NEFF <3, the model quality of our method is at least 50% better than HHpred. Our method also performs as well as HHpred on high-homology targets (i.e. NEFF >7). According to Skolnick group a model with TM-score ∼0.4 can be used for functional study while a model with TM-score ∼0.2 is almost random. This implies that when NEFF <2, the HHpred models are almost random while our method can generate models useable for functional study. Since ∼90% of the Pfam families without solved structures have NEFF <6, our method can improve over HHpred on a majority number of Pfam families. This study indicates that we can significantly advance the modeling capability of low-homology proteins with NEFF≤3, which represents approximately one-third of the Pfam families without solved structures.

## 3.2 Performance on CASP8 targets

To further demonstrate the advantage of our method, we compare it with the top 14 CASP8 servers (see Table 2). Among these servers, only HHpred2, MUSTER and Phyre2 are pure threading-based methods. Other servers use a combination of multiple structure prediction techniques including consensus method, multiple-template modeling, template-free modeling and model refinement.

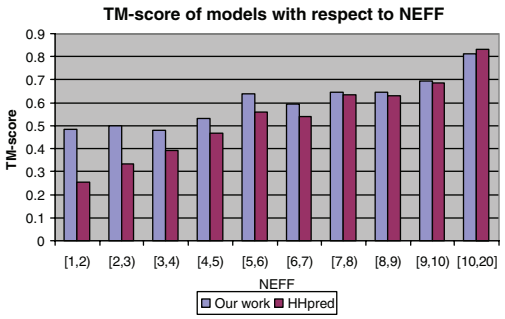


**Fig. 1.** The average TM-score of the 3D models with respect to NEFF. The models are generated by our method and HHpred on Prosup and SALIGN.

**Table 2.** Average TM-score of our method and the CASP8 top servers on 119 CASP8 targets with respect to NEFF

| NEFF | ≤2 | ≤3 | ≤4 | ≤5 | All |
|---|---|---|---|---|---|
| #targets | 2 | 6 | 16 | 33 | 119 |
| Zhang-Server | 0.243 | 0.278 | 0.505 | 0.501 | 0.711 |
| **Our work** | **0.291** | **0.336** | **0.521** | **0.486** | **0.694** |
| pro-sp3-TASSER | 0.248 | 0.247 | 0.471 | 0.476 | 0.691 |
| RAPTOR++ | 0.264 | 0.279 | 0.491 | 0.469 | 0.683 |
| METATASSER | 0.262 | 0.275 | 0.478 | 0.457 | 0.678 |
| ROBETTA | 0.270 | 0.262 | 0.489 | 0.470 | 0.676 |
| HHpred2 | 0.265 | 0.238 | 0.480 | 0.459 | 0.675 |
| Phyre-de-novo | 0.229 | 0.267 | 0.475 | 0.455 | 0.670 |
| MUSTER | 0.207 | 0.250 | 0.477 | 0.452 | 0.670 |
| MC-REFINE | 0.255 | 0.286 | 0.485 | 0.454 | 0.668 |
| HHpred5 | 0.275 | 0.225 | 0.475 | 0.446 | 0.668 |
| MC-CLUSTER | 0.212 | 0.286 | 0.489 | 0.455 | 0.667 |
| HHpred4 | 0.264 | 0.222 | 0.475 | 0.454 | 0.667 |
| MUProt | 0.254 | 0.271 | 0.478 | 0.454 | 0.664 |
| Phyre2 | 0.258 | 0.254 | 0.473 | 0.448 | 0.653 |

For example, Zhang-Server (Zhang, 2009) first does a consensus analysis of the results generated by ∼10 individual threading programs and then refines models using distance restraints extracted from top templates. Similar to Zhang-Server, the two TASSER programs (Zhou *et al*., 2009) uses the results from two threading programs PROSPECTOR (Skolnick and Kihara, 2001) and SP3. Robetta (Raman *et al*., 2009) first generates a template-based model using HHpred and then does model refinement. Robetta also runs template-free modeling if a reliable template cannot be detected. Phyre-de-novo combines the output of both HHpred and Phyre2 and in case no good template identified, also does template-free modeling. The three MULTICOM programs (Cheng, 2008) (MUProt, MC-CLUSTER and MC-REFINE) use multiple threading programs, multiple-template techniques, model clustering and template-free modeling. Our RAPTOR++ (Xu *et al*., 2009) program uses three in-house threading programs and then employs multiple-template technique for easy targets and template-free modeling for very hard targets.

To do a fair comparison, our new threading method used the NCBI NR and a template database generated before CASP8 started (i.e. May 2008). We evaluated the model quality of the 119 CASP8 targets using both GDT-TS and TM-score. GDT-TS is similar to

**Table 3.** Average GDT-TS of our method and the CASP8 top servers on 119 CASP8 targets with respect to NEFF

| NEFF | ≤2 | ≤3 | ≤4 | ≤5 | All |
|---|---|---|---|---|---|
| #targets | 2 | 6 | 16 | 33 | 119 |
| Zhang-Server | 0.263 | 0.282 | 0.470 | 0.453 | 0.630 |
| **Our work** | **0.289** | **0.337** | **0.489** | **0.440** | **0.615** |
| RAPTOR++ | 0.272 | 0.297 | 0.468 | 0.431 | 0.608 |
| pro-sp3-TASSER | 0.260 | 0.268 | 0.446 | 0.427 | 0.607 |
| Phyre-de-novo | 0.221 | 0.269 | 0.444 | 0.412 | 0.596 |
| ROBETTA | 0.274 | 0.273 | 0.458 | 0.426 | 0.595 |
| METATASSER | 0.265 | 0.282 | 0.442 | 0.411 | 0.594 |
| HHpred2 | 0.259 | 0.256 | 0.452 | 0.417 | 0.594 |
| MC-REFINE | 0.234 | 0.287 | 0.452 | 0.407 | 0.592 |
| MC-CLUSTER | 0.217 | 0.289 | 0.456 | 0.412 | 0.591 |
| HHpred5 | 0.273 | 0.244 | 0.447 | 0.408 | 0.591 |
| MUProt | 0.235 | 0.274 | 0.448 | 0.411 | 0.589 |
| MUSTER | 0.213 | 0.260 | 0.449 | 0.408 | 0.588 |
| HHpred4 | 0.258 | 0.232 | 0.446 | 0.411 | 0.587 |
| Phyre2 | 0.254 | 0.276 | 0.446 | 0.407 | 0.571 |

and also highly correlated with TM-score.[4] The model quality of the CASP8 servers is downloaded from Zhang's CASP8 website.[5] We exclude T0498 and T0499 from evaluation because they have been discussed in Alexander *et al*. (2007) well before CASP8 started. Due to space limitation, we evaluate only the #1 models generated by one.

By comparing our method with Zhang-Server, we can see how far away our new method is from the best server in the community, although it is unfair to compare our single-template-based method with a modeling method using multiple techniques. By comparing our method with the three mainly-threading-based methods HHpred2, MUSTER and Phyre2, we can see how much we have advanced the state-of-the-art of protein threading. This is important since all the top CASP8 servers including Zhang-Server heavily depend on single-template-based threading methods.

*3.2.1 Performance on low-homology targets* As shown in Tables 2 and 3, if only the low-homology targets (NEFF ≤4) are evaluated, our method outperforms all the top CASP8 servers including Zhang-Server. In particular, when only the targets with NEFF ≤3 are considered, our method outperforms HHpred2, MUSTER and Phyre2 by 41.2, 34.4 and 32.3%, respectively. When only the targets with NEFF ≤4 are considered, our method outperforms HHpred2, MUSTER and Phyre2 by 8.5, 9.2 and 10.1%, respectively. When only the targets with NEFF ≤3 and ≤4 are evaluated, our method is better than Zhang-Server by 20.8 and 3.2%, respectively. If we exclude the five easy targets[6] (i.e. T0390, T0442, T0447, T0458 and T0471) from evaluation, then our method is better than Zhang-Server, HHpred2, MUSTER and Phyre2 by 10.5, 15.9, 14.5 and 18.0%, respectively, on the 11 hard targets with NEFF ≤4. The performance of our method on low-homology targets is significant considering that our method is a pure single-template-based threading method while Zhang-Server combines results from ∼10 threading programs and also refines models extensively. Our new method is also better than our own RAPTOR++ program on low-homology targets. In CASP8, RAPTOR++ uses three in-house threading methods, a multiple-template method for easy targets and also a template-free method for hard targets.

When all the targets are considered, our new method outperforms all the top CASP8 servers but Zhang-Server in terms of both GDT-TS and TM-score. A paired Student's *t*-test between our method and each of the top CASP8 servers indicates that the difference between our method and all the top servers excluding RAPTOR++ and TASSER is significant ($P<0.05$), as shown in Table 4.

*3.2.2 Performance on hard targets* Table 5 shows the performance of our method on the 25 hard targets. Our method is very close to Zhang-Server on the hard targets and better than all the other servers. In particular, our method outperforms our own RAPTOR++ server by ∼10% on hard targets. Our method is better than the three threading methods HHpred2, MUSTER

**Table 4.** *P*-values of our method with respect to the top CASP8 servers on all the CASP8 targets

| | GDT-TS *P*-value | TM-score *P*-value |
|---|---|---|
| Phyre2 | $2.38E-10$ | $2.05E-09$ |
| MUSTER | $9.89E-07$ | $2.62E-06$ |
| HHpred4 | $3.25E-05$ | 0.000412 |
| HHpred2 | 0.00032 | 0.00149 |
| ROBETTA | 0.000828 | 0.00266 |
| MUProt | 0.00128 | 0.000701 |
| MC-CLUSTER | 0.00347 | 0.000846 |
| HHpred5 | 0.00547 | 0.00358 |
| MC-REFINE | 0.00659 | 0.00313 |
| Phyre-de-novo | 0.0142 | 0.00139 |
| METATASSER | 0.0252 | 0.228 |
| RAPTOR++ | 0.187 | 0.0620 |
| pro-sp3-TASSER | 0.217 | 0.681 |
| Zhang-Server | −0.00198 | −0.000671 |

**Table 5.** Performance of our method and the CASP8 top servers on 25 CASP8 hard targets

| | GDT-TS | TM-score |
|---|---|---|
| Zhang-Server | 8.096 | 9.309 |
| **Our work** | **7.793** | **8.946** |
| pro-sp3-TASSER | 7.590 | 8.779 |
| ROBETTA | 7.413 | 8.407 |
| METATASSER | 7.281 | 8.404 |
| MC-CLUSTER | 7.248 | 8.250 |
| MUProt | 7.193 | 8.180 |
| MC-REFINE | 7.156 | 8.263 |
| RAPTOR++ | 7.052 | 7.805 |
| HHpred2 | 6.824 | 7.763 |
| MUSTER | 6.784 | 7.793 |
| HHpred4 | 6.749 | 7.784 |
| Phyre-de-novo | 6.614 | 7.536 |
| HHpred5 | 6.605 | 7.517 |
| Phyre2 | 6.477 | 7.268 |

[4]GDT-TS is normalized by 100 to have scale [0, 1].
[5]http://zhang.bioinformatics.ku.edu/casp8/.
[6]In this article we use the target classification by Zhang at http://zhang.bioinformatics.ku.edu/casp8/.

**Table 6.** The list of 25 CASP8 hard targets and their NEFF

| Targets | NEFF | Targets | NEFF |
|---|---|---|---|
| T0397 | 3.6 | T0474 | 4.3 |
| T0409 | 6.4 | T0476 | 1.5 |
| T0419 | 4.8 | T0478 | 4.2 |
| T0421 | 4.6 | T0480 | 3.0 |
| T0429 | 2.6 | T0482 | 2.9 |
| T0443 | 4.8 | T0484 | 2.5 |
| T0460 | 1.2 | T0489 | 5.0 |
| T0462 | 5.6 | T0495 | 3.3 |
| T0464 | 4.2 | T0496 | 4.6 |
| T0465 | 4.2 | T0504 | 3.5 |
| T0466 | 3.5 | T0510 | 7.6 |
| T0467 | 5.5 | T0514 | 7.6 |
| T0468 | 4.2 | | |

**Table 7.** Performance of our method and the CASP8 top servers on 94 CASP8 easy

| | GDT-TS | TM-score |
|---|---|---|
| Zhang-Server | 66.872 | 75.303 |
| **Our work** | **65.354** | **73.619** |
| RAPTOR++ | 65.273 | 73.485 |
| Phyre-de-novo | 64.864 | 72.950 |
| pro-sp3-TASSER | 64.710 | 73.479 |
| HHpred5 | 64.536 | 72.986 |
| METATASSER | 64.215 | 73.350 |
| MC-REFINE | 64.182 | 72.288 |
| MC-CLUSTER | 63.994 | 72.160 |
| HHpred4 | 63.959 | 72.593 |
| HHpred2 | 63.925 | 72.566 |
| MUProt | 63.771 | 71.975 |
| ROBETTA | 63.445 | 72.004 |
| MUSTER | 63.191 | 71.919 |
| Phyre2 | 61.474 | 70.430 |

and Phyre2 by 14.2, 14.9 and 20.3%, respectively. Among the 25 hard targets, our method is better than, worse than and comparable to Zhang-Server on 10, 13 and 2 of them, respectively. It is not unexpected that our method performs well on hard targets since many of them are low-homology, as shown in Table 6. If the targets with NEFF $\geq$5 are excluded, the difference between our method and Zhang-Server will further reduce.

As shown in Table 7, our method is better than HHpred2, MUSTER and Phyre2 by 2.2, 3.4 and 6.3%, respectively, on the 94 easy targets. Zhang-Server is better than our method, partially because Zhang-Server uses multiple templates to model an easy target. By using multiple templates for a single target, it is possible to generate a model with accuracy higher than any single-template-based models.

## 4 DISCUSSION

Homologous information is very effective in detecting remote homologs, as evidenced by the profile-based method HHpred, which performed better than or as well as several top threading methods in recent CASP events. This paper proposes a new threading method and shows that homologous information is not sufficient for low-homology protein threading. In particular, when NEFF $\leq$6 we can improve alignment accuracy over profile-based methods by using more structure information. Our experimental result indicates that our method outperforms all the top CASP8 servers on low-homology targets (NEFF $\leq$4). Our method also performs well on both the CASP8 hard and easy targets and is slightly worse than the best CASP8 server. This result is encouraging considering that the top CASP8 servers use a combination of multiple techniques to do structure prediction while our method is only a classical single-template-based threading method. Our method is clearly better than several threading-based methods such as MUSTER, HHpred and Phyre2 on both low-homology and hard CASP8 targets.

The capability of predicting structures for low-homology proteins is very important. The Pfam database contains $\sim$6600 families without solved structures. To predict structures for these families, we have to rely on templates remotely similar to these families. A simple statistics shows that $\sim$33, 58 and 90% of the 6600 Pfam families have NEFF <3, 4 and 6, respectively. In our current template database, approximately one-third of the templates have NEFF <6. Therefore, if we align the 6600 Pfam families to our templates one-by-one, around 93% of the threading pairs will contain at least one protein with NEFF <6. This is surprising given that the NCBI NR sequence database currently contains millions of protein sequences. Along with the NCBI NR growth, the NEFF values of the Pfam families are also likely to increase. It will be interesting to study how fast the NEFF values will increase.

The percentage of low-homology proteins in CASP8 is much smaller than that in Pfam. Only 16 (13.4%) and 43 (36.1%) of the 119 CASP8 targets have NEFF <4 and 6, respectively. That is, the CASP8 targets are biased towards high-homology proteins. This is not unexpected since the CASP organizers obtain most of the targets from the worldwide structure genomics centers. These centers tend to solve structures for the targets in a large Pfam family to maximize the number of sequences within (homology) modeling distance of the structures in the PDB. A large Pfam family contains many proteins and thus, is more likely to have a large NEFF.

Our data also show that most CASP8 hard targets are low-homology. This is reasonable since it is very challenging to predict structures for a low-homology target. However, not all low-homology targets are hard. We can easily predict structures for low-homology targets (i.e. T0471, T0458, T0447, T0442 and T0390) as long as they have good templates. The major challenge we are facing now is to identify the best template for a given target. As long as the best template can be identified, we can generate a reasonable alignment as shown in Figure 1.

Since our new threading method demonstrates its superiority over other similar methods such as HHpred, MUSTER, Phyre2 and SP3/SP5, in particular on low-homology targets without close homologs in the PDB. A natural extension of this work is to incorporate our new method into Zhang-Server to see how much we can advance the state-of-the-art of protein modeling.

*Funding*: TTI-C research funding and National Institutes of Health grants R01GM081642 and R01GM081871A1.

*Conflict of Interest*: none declared.

## REFERENCES


Alexander,P.A. *et al*. (2007) The design and characterization of two proteins with 88% sequence identity but different structure and function. *Proc. Natl Acad. Sci. USA*, **104**, 11963–11968.

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.

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

Bjelic,S. and Aqvist,J. (2004) Computational prediction of structure, substrate binding mode, mechanism, and rate for a malaria protease with a novel type of active site. *Biochemistry*, **43**, 14521–14528.

Caffrey,C.R. *et al*. (2005) Homology modeling and SAR analysis of Schistosoma japonicum cathepsin D (SjCD) with statin inhibitors identify a unique active site steric barrier with potential for the design of specific inhibitors. *Biol. Chem.*, **386**, 339–349.

Casbon,J.A. and Saqi,M.A.S. (2004) Analysis of superfamily specific profile-profile recognition accuracy. *BMC Bioinformatics*, **5**.

Chakravarty,S. *et al*. (2008) Systematic analysis of the effect of multiple templates on the accuracy of comparative models of protein structure. *BMC Struct. Biol.*, **8**.

Cheng,J.L. (2008) A multi-template combination algorithm for protein comparative modeling. *BMC Struct Biol.*, **8**.

Do,C.B. *et al*. (2006) CONTRAlign: discriminative training for protein sequence alignment. In Apostolico,A. *et al.* (eds) *Proceedings of the 10th Annual International Conference on Research in Computational Molecular Biology*. Springer, Venice, Italy, pp. 160–174.

Ellrott,K. *et al*. (2007) Improvement in protein sequence-structure alignment using insertion/deletion frequency arrays. Computational systems bioinformatics/Life Sciences Society. *Comp. Syst. Bioinformatics Conf.*, **6**, 335–342.

Finn,R.D. *et al*. (2008) The Pfam protein families database. *Nucleic Acids Res.*, **36**, D281–D288.

Ginalski,K. *et al*. (2005) Practical lessons from protein structure prediction. *Nucleic Acids Res.*, **33**, 1874–1891.

Henikoff,S. and Henikoff,J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl Acad. Sci. USA*, **89**, 10915–10919.

Henikoff,S. and Henikoff,J.G. (1993) Performance evaluation of amino acid substitution matrices. *Proteins*, **17**, 49–61.

Kelley,L.A. and Sternberg,M.J.E. (2009) Protein structure prediction on the web: a case study using the Phyre server. *Nat. Protocols*, **4**, 363–371.

Lackner,P. *et al*. (2000) ProSup: a refined tool for protein structure alignment. *Protein Eng.*, **13**, 745–752.

Marti-Renom,M.A. *et al*. (2004) Alignment of protein sequences by their profiles. *Protein Sci.*, **13**, 1071–1087.

Melo,F. and Sali,A. (2007) Fold assessment for comparative protein structure modeling. *Protein Sci.*, **16**, 2412–2426.

Moult,J. *et al*. (2005) Critical assessment of methods of protein structure prediction (CASP) – round 6. *Proteins Struct. Funct. Bioinformatics*, **61**, 3–7.

Moult,J. *et al*. (2007) Critical assessment of methods of protein structure prediction – round VII. *Proteins Struct. Funct. Bioinformatics*, **69**, 3–9.

Murzin,A.G. *et al*. (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.

Peng,J. and Xu,J. (2009) Boosting protein threading accuracy. In Batzoglou,S. (ed.) *Proceedings of the 13th Annual International Conference on Research in Computational Molecular Biology (RECOMB)*. Springer, Tucson, AZ, pp. 31–45.

Pieper,U. *et al*. (2006) MODBASE: a database of annotated comparative protein structure models and associated resources. *Nucleic Acids Res.*, **34**, D291–D295.

Prlic,A. *et al*. (2000) Structure-derived substitution matrices for alignment of distantly related sequences. *Protein Eng.*, **13**, 545–550.

Qiu,J. and Elber,R. (2006) SSALN: An alignment algorithm using structure-dependent substitution matrices and gap penalties learned from structurally aligned protein pairs. *Proteins Struct. Funct. Bioinformatics*, **62**, 881–891.

Raman,S. *et al*. (2009) Structure prediction for CASP8 with all-atom refinement using Rosetta. *Proteins Struct. Funct. Bioinformatics*, **77** (Suppl 9), 89–99.

Sadreyev,R.I. and Grishin,N.V. (2004) Quality of alignment comparison by COMPASS improves with inclusion of diverse confident homologs. *Bioinformatics*, **20**, 818–828.

Sali,A. (1995) Comparative protein modeling by satisfaction of spatial restraints. *Mol. Med. Today*, **1**, 270–277.

Sammut,S.J. *et al*. (2008) Pfam 10 years on: 10 000 families and still growing. *Brief. Bioinformatics*, **9**, 210–219.

Sanchez,R. *et al*. (2000) Protein structure modeling for structural genomics. *Nat. Struct. Biol.*, **7**, 986–990.

Skolnick,J. and Kihara,D. (2001) Defrosting the frozen approximation: PROSPECTOR – a new approach to threading. *Proteins Struct., Funct. Genetics*, **42**, 319–331.

Skolnick,J. *et al*. (2000) Structural genomics and its importance for gene function analysis. *Nat. Biotechnol.*, **18**, 283–287.

Skowronek,K.J. *et al*. (2006) Theoretical model of restriction endonuclease HpaI in complex with DNA, predicted by fold recognition and validated by site-directed mutagenesis. *Proteins Struct. Funct. Bioinformatics*, **63**, 1059–1068.

Soding,J. (2005) Protein homology detection by HMM-HMM comparison. *Bioinformatics*, **21**, 951–960.

Tan,Y.H. *et al*. (2006) Statistical potential-based amino acid similarity matrices for aligning distantly related protein sequences. *Proteins Struct. Funct. Bioinformatics*, **64**, 587–600.

Wells,G.A. *et al*. (2006) Novel properties of malarial S-adenosylmethionine decarboxylase as revealed by structural modelling. *J. Mol. Graphics Model.*, **24**, 307–318.

Wu,S.T. and Zhang,Y. (2008) MUSTER: improving protein sequence profile-profile alignments by using multiple sources of structure information. *Proteins Struct. Funct. Bioinformatics*, **72**, 547–556.

Xu,J. *et al*. (2003) RAPTOR: optimal protein threading by linear programming. *J. Bioinformatics Comput. Biol.*, **1**, 95–117.

Xu,J. *et al*. (2009) Template-based and free modeling by RAPTOR++ in CASP8. *Proteins Struct. Funct. Bioinformatics*, **77** (Suppl 9), 133–137.

Zhang,C. *et al*. (2004) An accurate, residue-level, pair potential of mean force for folding and binding based on the distance-scaled, ideal-gas reference state. *Protein Sci.*, **13**, 400–411.

Zhang,W. *et al*. (2008) SP5: improving protein fold recognition by using torsion angle profiles and profile-based gap penalty model. *PLoS ONE*, **3**.

Zhang,Y. (2009) I-TASSER: Fully automated protein structure prediction in CASP8. *Proteins: Struct. Funct. Bioinformatics*, **77**, 100–113.

Zhang,Y. and Skolnick,J. (2005a) The protein structure prediction problem could be solved using the current PDB library. *Proc. Natl Acad. Sci. USA*, **102**,1029–1034.

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

Zhang,Y. and Skolnick,J. (2007) Scoring function for automated assessment of protein structure template quality. *Proteins Struct. Funct. Bioinformatics*, **68**, 1020.

Zhou,H. *et al*. (2009) Performance of the Pro-sp3-TASSER server in CASP8. *Proteins Struct. Funct. Bioinformatics*, **77** (Suppl 9), 123–127.

Zhou,H.Y. and Zhou,Y.Q. (2004) Single-body residue-level knowledge-based energy score combined with sequence-profile and secondary structure information for fold recognition. *Proteins Struct. Funct. Bioinformatics*, **55**, 1005–1013.

Zhou,H.Y. and Zhou,Y.Q. (2005) Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. *Proteins Struct. Funct. Bioinformatics*, **58**, 321–328.