PROTEINS
STRUCTURE ■ FUNCTION ■ BIOINFORMATICS

TASSER_low-zsc: An approach to improve structure prediction using low *z*-score–ranked templates

Shashi B. Pandit and Jeffrey Skolnick*

Center for the Study of Systems Biology, School of Biology, Georgia Institute of Technology, Atlanta, Georgia 30318

ABSTRACT

In a variety of threading methods, often poorly ranked (low *z*-score) templates have good alignments. Here, a new method, *TASSER_low-zsc* that identifies these low *z*-score–ranked templates to improve protein structure prediction accuracy, is described. The approach consists of clustering of threading templates by affinity propagation on the basis of structural similarity (*thread_cluster*) followed by TASSER modeling, with final models selected by using a TASSER_QA variant. To establish the generality of the approach, templates provided by two threading methods, SP^3 and SPARKS^2, are examined. The SP^3 and SPARKS^2 benchmark datasets consist of 351 and 357 medium/hard proteins (those with moderate to poor quality templates and/or alignments) of length $\leq$250 residues, respectively. For SP^3 medium and hard targets, using *thread_cluster*, the TM-scores of the best template improve by $\sim$4 and 9% over the original set (without low *z*-score templates) respectively; after TASSER modeling/refinement and ranking, the best model improves by $\sim$7 and 9% over the best model generated with the original template set. Moreover, *TASSER_low-zsc* generates 22% (43%) more foldable medium (hard) targets. Similar improvements are observed with low-ranked templates from SPARKS^2. The template clustering approach could be applied to other modeling methods that utilize multiple templates to improve structure prediction.

Proteins 2010; 78:2769–2780.

Key words: structure prediction; threading; TASSER; tertiary structure.

INTRODUCTION

Despite significant progress, the prediction of protein structure remains an unsolved problem in computational structural biology.[1–3] Historically, structure prediction methods have been divided into three general categories: comparative modeling (CM),[1,4–6] threading[7–11], and free modeling (FM).[12–15] The basic objective of CM and threading approaches is to identify a set of structurally related template proteins (with known tertiary structure) to the target sequence.[5,9] CM methods identify template proteins with a clear evolutionary relationship to the target by using sequence-based methods,[5,16] whereas threading, by including protein structural information, strives to identify template proteins that have a similar fold as the target irrespective of their evolutionary relationship.[3,8,9] Because of the convergence of threading and CM methods, both are referred to as template-based modeling (TBM).[17] In TBM, once the related template is identified, the target sequence is aligned to the template structure either indirectly by performing a sequence alignment and then transferring this alignment to the associated position in the structure or by directly incorporating structural information into the alignment procedure.[5,9–11] A full-length model is then generated by building the chain in the unaligned regions of the template. This full-length structure is then refined, with the goal of improving model quality relative to the initial TBM-provided alignment. In contrast, in template FM, one does not use any global template structural information as an input.[12,13] Thus, the possibility of assembling a novel fold exists.

In recent years, TBM has emerged as the most robust approach to protein structure prediction.[3,17] Advances in better template identification and improved alignment accuracy resulted from the use of profile–profile alignments,[18–20] inclusion of structural properties such as solvent accessibility[21] and structural profiles,[8,10,11,22–24] hidden Markov models,[25,26] machine-learning approaches,[27,28] and the employment of meta-servers.[29–31] Model refinement can be achieved by using multiple templates to generate better alignments,[32,33] iterative refinement[34,35] as well as physics-based and evolution-based potentials.[12,35,36] The ultimate success of TBM requires that similar structures to those adopted by the target be found in the Protein Data Bank (PDB).[37] Recent studies have demonstrated that the current PDB library is most likely complete hence it can provide templates for all compact, single domain proteins from which low-

Additional Supporting Information may be found in the online version of this article.
Grant sponsor: National Institutes of Health; Grant number: GM-48835.
*Correspondence to: Jeffrey Skolnick, Center for the Study of Systems Biology, School of Biology, Georgia Institute of Technology, Atlanta, Georgia 30318. E-mail: skolnick@gatech.edu.
Received 12 February 2010; Revised 12 May 2010; Accepted 29 May 2010
Published online 10 June 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/prot.22791

to-moderate resolution structures can be built.[38,39] However, the key issue is to select such templates and to generate high-quality alignments.

Among the more successful structure prediction approaches are Threading/ASSEmbly/Refinement (TASSER)[32] and its variants chunk-TASSER and pro-sp3-TASSER,[40] where in large scale benchmarking, it was shown that reasonable models could be built for both TBM and template FM for weakly/nonhomologous proteins.[32,40–43] Recent improvements resulted from incorporation of improved contact predictions as in TASSER_2.0,[44] iterative TASSER^{iter},[34] and the use of templates identified by multiple threading algorithms such as in METATASSER[43] or pro-sp3-TASSER.[40]

In threading, usually some type of knowledge-based scoring function is used to rank the particular sequence-structure alignment.[8,9,45] Furthermore, the score significance of a target aligned to a given template is evaluated in terms of its *z*-score or through use of a neural network to rank the templates.[46,47] In that regard, the *z*-score of the sequence mounted in a given structure is defined by

$$Z = (E - \langle E \rangle) \Big/ \sqrt{\left(\langle E^2 \rangle - \langle E \rangle^2\right)}$$

the quantity in $\langle\rangle$ denotes the average of the best alignment given by dynamic programming over the template library, and E is the score or energy.[9] Usually, *z*-score–based ranking identifies best-fit templates for the target sequence.[8,45,48] However, we have observed that templates with good alignments to the target native structure (TM-score $\geq$ 0.40),[49] sometimes have quite poor ranks based on their *z*-scores. Identification of such templates/alignments is important in those situations when the top ranked template/alignments are of poor quality.

In this work, our goal is to develop a methodology that includes low-ranked *z*-score but good quality templates to improve structure prediction by using the TASSER methodology.[32] We can consider a large number of templates, having low *z*-score, as an input for TASSER. However, the main issue in considering more templates is that the consensus information from templates close to the native structure is usually diminished. To circumvent this issue, we have developed the *thread_cluster* algorithm, which is used to filter and cluster structurally similar templates. Subsequently, for each cluster, we perform TASSER simulations to generate an ensemble of models, which is ranked by our in-house model ranking method described below. We refer to this protocol from template selection to model ranking as *TASSER_low-zsc*. We benchmarked this method on representative benchmark datasets by using two threading methods, SP^3[10] and SPARKS.[2,11] In benchmarking, we show that this new approach of template clustering followed by TASSER modeling/refinement generates models, which are better both in terms of structure prediction accuracy and number of foldable targets compared with those generated using the original set of selected templates.

METHODS

We compiled a benchmark set of 691 proteins (whole chains or domains) of length $\leq$250 residues with pairwise sequence identity $\leq$40% with the proteins in the PDB[37] template library. The template library consists of 17,888 proteins composed of both whole chains and domains. As for a multiple domain protein, both the whole chain and its domains are included, there is certain redundancy in the library. The benchmark proteins were released subsequently to the construction of template library. Thus, this benchmark is set up to mimic a critical assessment of structure prediction (CASP)-like[50] scenario. The list of proteins in the template library and benchmark dataset for SP^3 and SPARKS^2 is available at: http://cssb.biology.gatech.edu/skolnick/files/TASSER_low-zsc. In addition to the above dataset, we have evaluated *TASSER_low_zsc* on 108 CASP8 targets[51] of length $\leq$350 residues. Both SP^3 and SPARKS^2 were used in the meta-threading procedure in CASP8.[52] Hence, we have used the threading output obtained during CASP8.

The protein structure prediction protocol described here consists of three main steps: (a) filtering and clustering of templates via *thread_cluster*, (b) generation of an ensemble of models using TASSER, and (c) model ranking. *TASSER_low-zsc* approach is schematically described by the flowchart shown in Figure 1.

To evaluate the robustness of the *TASSER_low-zsc* methodology, we use two threading programs, viz. SP^3 and SPARKS^2. For each threading method, we classify targets into easy/medium/hard categories based on the *z*-score of the top template from the respective threading programs. In SP^3, targets whose top template has a *z*-score $\geq$ 6.0 are classified as "easy," those with a *z*-score $\leq$ 4.5 as "hard," and those having a 4.5 $<$ *z*-score $<$ 6.0 as "medium." In SPARKS^2, targets whose top template has a *z*-score $\geq$ 5.5 are classified as "easy," those with *z*-score $\leq$ 4.0 as "hard," and targets with 4.0 $<$ *z*-score $<$ 5.5 as "medium." In general, "easy" targets have templates with good threading alignments. "Medium" targets have good structure alignments to the native structure but poor threading alignments, whereas, "hard" targets generally have poor quality global structural alignments to the native state. As for "easy" targets, usually the best template (that is closest to the native structure) is among the top-ranked templates, here, we only consider medium/hard targets, as classified by their respective threading program, for the assessment of the method.

In the following, we discuss in detail each of the steps of *TASSER_low-zsc*:

Thread_cluster: filtering and clustering of templates

The top N templates as ranked by their *z*-score are clustered by using the TM-score[49] structural similarity metric, which is calculated for the common aligned

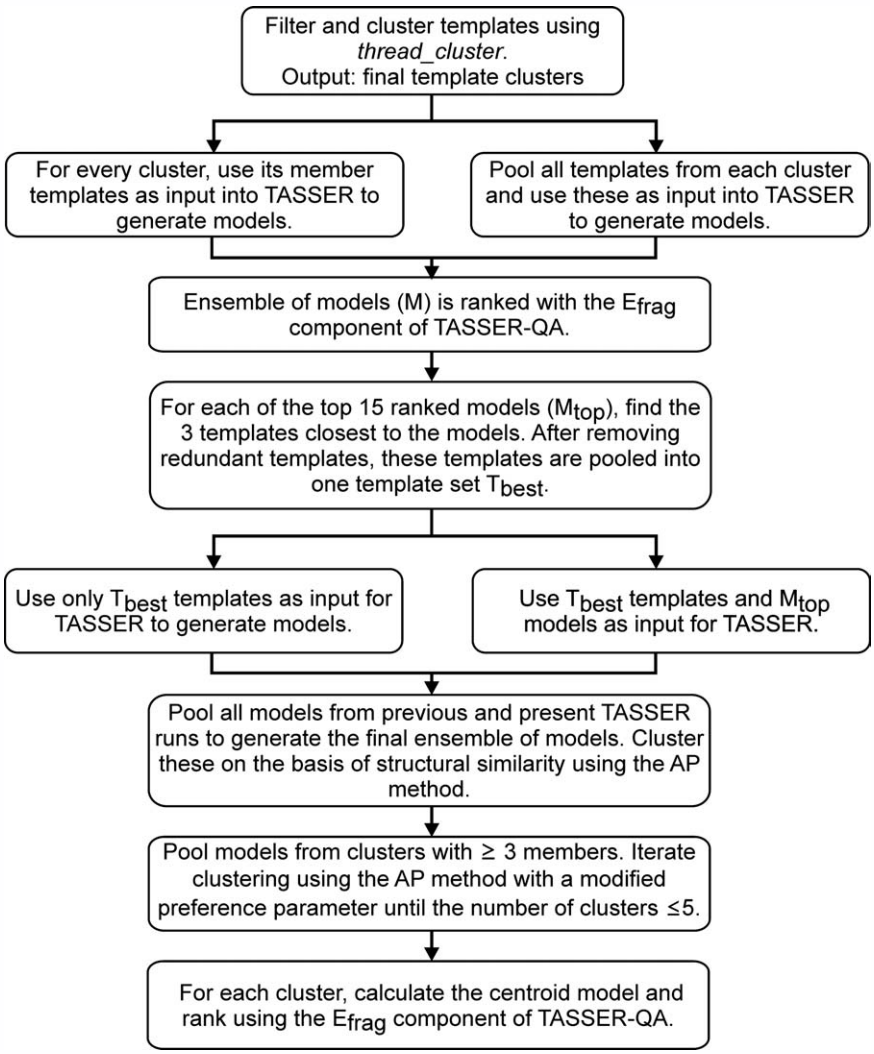

Figure 1
Flowchart of the *TASSER_low-zsc* approach.

region between templates, where the length used for the TM-score calculation is the number of common aligned residues between two templates. For different threading methods, we obtain the maximum number of templates (N) to be used for clustering. For "medium" targets, from SP[3] and SPARKS[2], we consider the top 60 and 75 z-score–ranked templates, respectively, whereas for "hard" targets, we consider the top 70 and 80 z-score–ranked templates, respectively. However, to reduce the number of templates considered, if there is significant structural similarity among the top 40 templates, (see below), we reduce N to 40.

We have used the affinity propagation (AP) method for clustering.[53] AP algorithm simultaneously considers all data points as potential "exemplars." In the clustering step, real-valued messages are exchanged between data points until a good set of exemplars and the corresponding cluster emerges. There are two kinds of messages exchanged between data points "responsibility" and "availability." In this, "responsibility" reflects the accumulated evidence of how well suited a data point is to act as an exemplar for another data point considering other possible exemplars, and "availability" is how appropriate it is for a data point to select a particular exemplar, taking into account the support from the other points for the exemplar. These real-values messages are updated during clustering. It has been shown that AP provides clusters with much lower error in comparison with other similar methods.[53]

The AP program requires the similarity between the data points and a preference parameter (P_{pref}) of a point, which is the a priori suitability of a point to serve as

exemplar. Using AP, one can obtain the desired number of clusters by changing the value of P_{pref}. A high value of P_{pref} will cause AP to find many clusters, whereas a low value of P_{pref} will lead to a small number of clusters.

In practice, the following algorithm is used for filtering and clustering the templates:

1. For each of the top 40 templates, calculate the number of templates with pairwise TM-score $\geq$0.45. If the median of this distribution is $\geq$10, then set $N = 40$, otherwise, use the top N templates as defined previously.
2. For each of the N templates, calculate the number of templates with a TM-score $\geq$0.45. If the median of this distribution is $\geq$10, set $P_{pref} = 0.45$; otherwise, set $P_{pref} = 0.35$. Next, remove any template whose best TM-score value to any other another template is less than P_{pref}. This removes templates with insignificant similarity within the list of N templates.
3. Calculate all-against-all TM-scores (for their common threading-aligned region) for the filtered set of N templates. Next, perform AP clustering (with P_{pref} set in the previous step) on the basis of their TM-score similarity among templates. After clustering, all one-member clusters are removed. In addition, from each cluster, remove those templates whose best structurally similar template, as given by TM-score, within the cluster has a TM-score $\leq P_{pref}$. Then, for each cluster, calculate the average of all-against-all TM-scores. For the next step, if the average TM-score is $\geq$0.65, only one template with the highest z-score is considered; otherwise, all the members of the cluster are considered. This constitutes the filtered N_{mod} templates. Furthermore, add the top 10 z-score–ranked templates to the N_{mod} templates and remove any redundant templates from the modified list of templates.
4. For the N_{mod} templates, calculate their all-against-all TM-scores. Perform AP clustering with P_{pref} set to the median obtained from the distribution of pairwise similarity values. In this step, the desired maximum number of clusters $N_{clus} \leq 5$. In practice, N_{clus} varies depending on the number of templates (N_{mod}). If $N_{mod} < 30$, then set $N_{clus} \leq 3$; otherwise, if $30 < N_{mod} < 40$, set $N_{clus} \leq 4$. The required number of clusters is achieved by iterating the AP clustering method with concomitant decrease in the value of P_{pref}. This step assigns every template in N_{mod} to one of the clusters. Usually, structurally similar templates are clustered together at the end of the AP clustering process.

TASSER simulations

As shown in Figure 1, we used TASSER to generate the ensemble of models. As TASSER has been extensively described in the literature,[32,41] here, we just present a brief overview of its essential components. The TASSER force-field consist of knowledge-based statistical potentials describing short-range backbone correlations, pairwise interactions, hydrogen bonding, secondary structure propensities, and a set of predicted side-chain contact and distance restraints derived from the initial structures. Generally, the structures that provide the restraints and the starting conformations are the same. The structures generated by TASSER are then clustered by using SPICKER,[54] which provides the list of models ranked by cluster density. Here, we limit the run time of TASSER to $\sim$24 h to avoid long simulation times especially for longer proteins.

To generate a diverse set of models, the following sets of templates are considered:

1. For each of the N_{clus} clusters, templates within the cluster are used as an input for TASSER along with their consensus side-chain contacts (present in [3/4] of the templates).
2. All the templates (N_{mod}) are used as an input for TASSER.

   The ensemble of models generated in the above procedure is ranked with the E_{frag} component of TASSER-QA.[55] Next, for each of the top 15-ranked models, the three best threading templates (NR_{temp}) having TM-score $\geq$0.45 to the TASSER model are identified. If no templates could be identified with this TM-score cut-off, this value is reduced until there is at least a total of four templates identified by the top 15-ranked models. Usually for most targets, the TM-score cut-off value does not drop below 0.25. Using this set of models and templates, the following set of input structures are used in short TASSER simulations ($\sim$12 h)
5. Only the templates (NR_{temp}) are used as input structures for the simulation.
6. The templates (NR_{temp}) and those models (NR_{model}), which could identify templates with TM-score $\geq$0.45 (as described before), are used as input structures.

The main objective of performing this step of TASSER is to enrich the set of good (closer to native structures) models provided by the templates identified in the previous step. The final ensemble consists of models from the present and previous simulations.

Ranking of models

Model ranking is an important unsolved problem in the field of protein modeling.[56] In our ranking procedure, we use a combination of model clustering and ranking with the E_{frag} component of TASSER-QA.[55] The basic idea is to cluster structurally similar models and rank the representative model obtained from each cluster. In practice, our model ranking algorithm is as follows:

1. For all models, first calculate the all-against-all similarity score as measured by the TM-score. Next, clus-

ter the models by AP method with the preference parameter set as the median of the similarity score distribution. Generate the modified list of models by only considering models from clusters containing at least three members. Next, for the modified list of models, iterate AP clustering by tuning the P_{pref} parameter to result in a maximum of five clusters.
2. For each cluster, rank the models within the cluster by the average highest TM-score to all other models in the cluster. Next, superimpose all the models to the top-ranked model and generate the centroid model by averaging the coordinate positions within a root mean square deviation, RMSD, of less than 6.5 Å after superimposition. This centroid model is considered as the final model. PULCHRA[57] is then used to fix the artifacts in the centroid model due to averaging. If PULCHRA fails, the model closest to the average structure is considered as the final model.
3. Rank the final set of models using the E_{frag} component of TASSER-QA.

TASSER with original set of templates (TASSER_org)

To evaluate the improvement in structure prediction with *TASSER_low-zsc*, we generated models with TASSER using the number of templates benchmarked in previous studies.[43] For both SP[3] and SPARKS[2], we used the top 10 *z*-score–ranked templates as an input for TASSER. These are the number of templates that would be used in a standard TASSER prediction scenario.

RESULTS AND DISCUSSION

TASSER_low-zsc is evaluated on a representative benchmark dataset and on CASP8 targets by using two different threading methods, SP[3] and SPARKS[2]. In this work, we have used the TM-score[49] to assess the quality of the structure template and the predicted full-length model. The benchmarking results of the method are followed with evaluation on CASP8 targets.

Targets are classified into easy/medium/hard sets based on the top template *z*-score (see Methods). For SP[3], the number of targets classified into the easy, medium, and hard set is 340, 148, and 203, respectively. For SPARKS[2], the number of targets in the easy, medium, and hard set is 334, 155, and 202, respectively. As mentioned before, we consider only medium/hard targets for assessment. Thus, SP[3] and SPARKS[2] are evaluated on 351 and 357 targets, respectively. In what follows, we first assess the performance of the *thread_cluster* algorithm and then assess the accuracy of the structure prediction protocol, *TASSER_low-zsc*.

The main objective of the *thread_cluster* algorithm is to filter low *z*-score–ranked templates to identify a structurally consistent set. When we consider low *z*-score–ranked templates, the number of templates increases and the fraction of "good" templates (TM-score to native $\geq$0.40) could either be enriched or depleted. Hence, clustering structurally similar templates, as measured by their TM-score, attempts to retain these "good" templates. For "good" templates, the assumption here is that more than one such template is identified by the given threading procedure. First, we assess ability of *thread_cluster* to retain the best available template among all top *N* templates after clustering. For both SP[3] and SPARKS[2], Table I summarizes the comparison among the best available templates from the original template set, the top *N* set, and templates after filtering and clustering. On average, the TM-score to the native of the best template among the top *N* templates for SP[3] medium and hard targets is better than the original template set by ~8 and 11%, respectively. After applying *thread_cluster*, on average, the TM-score to native structure of the best template among available templates is better by ~4 and 9% for medium and hard targets, respectively. Similarly, for SPARKS[2], after using *thread_cluster*, on average, the TM-score to the native structure of the best template among available templates is better by ~8 and 10% for medium and hard targets, respectively. This suggests that, for most targets, *thread_cluster* can retain the best available template among the top *N* templates.

Table I
Comparison of Best Template Before and After *Thread_cluster*

Threadings	Target Type	No. of Targets	$\langle TM_{org} \rangle$	$\langle TM_{topn} \rangle$	$\langle TM_{clus} \rangle$
SP[3]	Medium	148	0.373 (61)	0.402 (71)	0.389 (69)
	Hard	203	0.349 (51)	0.388 (79)	0.380 (76)
SPARKS[2]	Medium	155	0.358 (50)	0.404 (65)	0.385 (60)
	Hard	202	0.287 (26)	0.338 (51)	0.317 (43)

TM_{org}, TM_{topn} and TM_{clus} refer to the TM-score of the best template to the native structure among original template set (top 10 templates), among the top *N* templates and among templates identified after the filtering/clustering step in *thread_cluster* respectively. The number in the parenthesis is the number of proteins with TM-score $\geq$ 0.40.

Next, we compare the performance of *thread_cluster* in terms of the number of targets with TM-score $\geq$0.4 (indicative of significant structural similarity to the native structure). As is evident from Table I, by using SP[3] (SPARKS[2]) for medium targets, the number of foldable targets in the top *N* template set increases by 16% (30%) in comparison with the original template set. For hard targets, the relative number of foldable proteins increases by 55% (96%), respectively. Subsequent to the *thread_cluster* procedure, for medium targets, using SP[3] (SPARKS[2]), the improvement in terms of number of foldable targets is still 13% (20%) in comparison with the original template set, whereas for hard targets, the number increases by 49% (65%). The diminution relative to the best possible performance reflects the fact that for

Table II
Summary of Structure Prediction Results for SP^3 Medium/Hard Targets

Target Type	No. of Targets	First Model		Best of Top Five Models	
		$<TMM_{org}>$	$<TMM_{clus}>$	$<TMM_{org}>$	$<TMM_{clus}>$
Medium	148	0.345 (48)	0.363 (54)	0.380 (58)	0.407 (71)
Hard	203	0.316 (37)	0.340 (43)	0.354 (53)	0.385 (76)

TMM_{org} and TMM_{clus} refer to TM-score of the model from *TASSER_org* and *TASSER_low-zsc*, respectively. The number in the parenthesis is the number of proteins with TM-score $\geq$ 0.40.

some targets, *thread_cluster* could not recover the best template. For most of these targets, there are a few "good" templates in the list of *N* templates used for clustering. Moreover, for some targets, the "good" template does not have significant structural similarity to any other template. Hence, these templates are filtered out in the clustering process of *thread_cluster*.

Subsequent to *thread_cluster*, the ensemble of full-length models is generated and ranked as described in Methods. We present the benchmarking result of structure prediction by using templates from SP^3 and SPARKS[2] threading.

Benchmarking results using templates from SP^3

The main objective of including low *z*-score–ranking templates in TASSER modeling is to increase the accuracy of protein structure prediction. For assessment, we compared the models generated from two procedures, that is *TASSER_org*, which has models generated with TASSER using the original template set (see Methods section) and *TASSER_low-zsc*, which has the models generated with TASSER using the input templates obtained after *thread_cluster* followed by ranking of the model ensemble (see Methods). Moreover, *TASSER_org* uses the standard TASSER structure prediction scenario.[32,43] The comparison of performance between *TASSER_org* and *TASSER_low-zsc* methods is summarized in Table II. For the first model, *TASSER_low-zsc* shows on average a 5 and 7% TM-score improvement over *TASSER_org* for medium and hard targets, respectively. Similarly, on average, the best (among the top five) model TM-score improvement from *TASSER_low-zsc* over *TASSER_org* is ~7 and 9% for medium and hard targets, respectively. Moreover, for the medium (hard) set, *TASSER_low-zsc* provides 22% (43%) more foldable targets than *TASSER_org*, when the best of five models are considered.

For the detailed analysis of the improved performance of *TASSER_low-zsc*, we used the best (of top five) model (from both procedures), TM-score to the native structure, to classify the medium and hard targets into following four sets: (a) targets having best model TM-score <0.40 from both procedures, (b) targets with a best model TM-score <0.40 from *TASSER_org*, and TM-score $\geq$0.40 from *TASSER_low-zsc* method, (c) proteins with best model TM-score $\geq$0.40 from *TASSER_org* and TM-score <0.40 from *TASSER_low-zsc* method, and (d) proteins with best model TM-score $\geq$0.40 from both procedures. The average TM-score of best models from both methods for the targets classified into these four sets is summarized in Table III. The targets with a best model TM-score $\geq$0.40 to the native structure only from *TASSER_low-zsc* method [set (b) in the above classification] show an average TM-score improvement of 37% (40%) for medium (hard) targets over models from *TASSER_org*. Interestingly, even for targets whose best model TM-score to the native structure from both methods $\geq$0.40 [set (d) in above classification], *TASSER_low-zsc* on average shows an improvement of 4% in TM-score for both medium and hard targets (Table III). For very few targets, *TASSER_low-zsc* results in a model with lower accuracy in terms of its TM-score (Table III) to the native structure.

In Figure 2(A,B), for the medium and hard targets, respectively, we show the comparison of the best model from *TASSER_org* and *TASSER_low-zsc*. Overall, the best model from *TASSER_low-zsc* has a higher TM-score than *TASSER_org*. However, for certain targets, the performance of *TASSER_low-zsc* is worse. These cases are partly due to the failure of the method to select and rank models. For example, target 2zb5_A_d1 has a *TASSER_org* generated best model with a TM-score of 0.457. In contrast, *TASSER_low-zsc* results in a best model with a TM-score of 0.343. However, the best possible model among the ensemble of models from *TASSER_low-zsc* has a TM-score of 0.454. Next, we compare the distribution of the number of targets having models greater than or equal to a given TM-score threshold value. This comparison is shown in Figure 3, where the number of targets is plotted as a function of TM-score and demonstrates that

Table III
Summary of Structure Prediction Results for SP^3 Targets Classified into Foldable/Nonfoldable Categories Either From One or Both Methods

Target Classifications	Target Type	$<TMB_{org}>$	$<TMB_{zsc}>$	Target Type	$<TMB_{org}>$	$<TMB_{zsc}>$
$TMB_{org} < 0.40$ and $TMB_{zsc} < 0.40$	Medium (75)	0.261	0.273	Hard (125)	0.284	0.297
$TMB_{org} < 0.40$ and $TMB_{zsc} \geq 0.40$	Medium (15)	0.340	0.467	Hard (25)	0.345	0.484
$TMB_{org} \geq 0.40$ and $TMB_{zsc} < 0.40$	Medium (2)	0.468	0.365	Hard (2)	0.419	0.387
$TMB_{org} \geq 0.40$ and $TMB_{zsc} \geq 0.40$	Medium (56)	0.547	0.570	Hard (51)	0.529	0.550

TMB_{org} and TMB_{zsc} refer to TM-score of the best model (of the top 5) to native structure from *TASSER_org* and *TASSER_low-zsc*, respectively. The number in parenthesis is the number of proteins classified into that particular category.

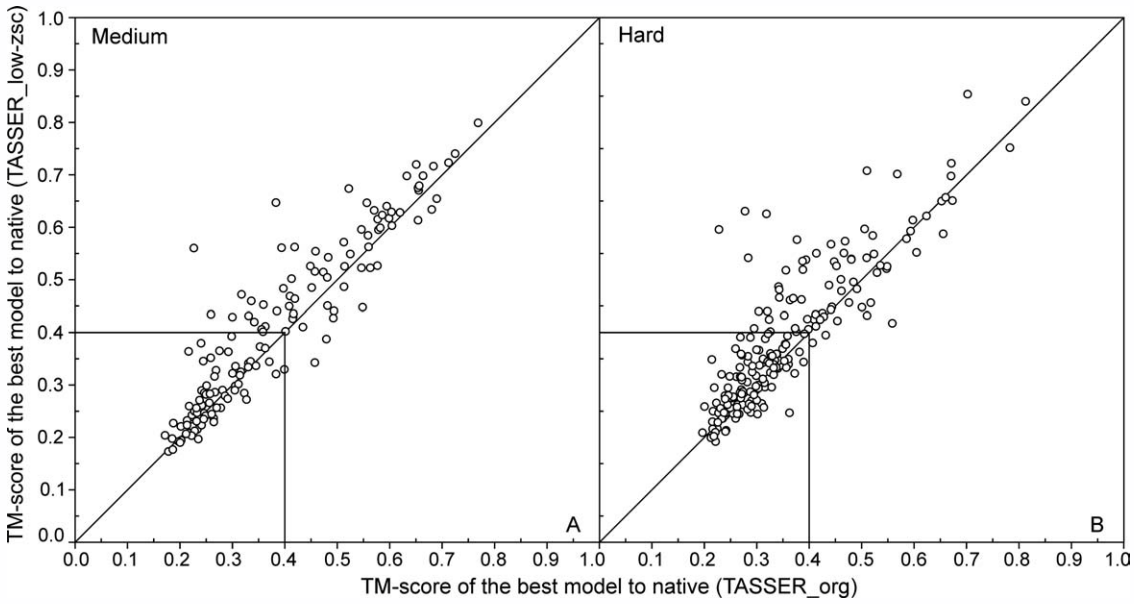

Figure 2

Using SP^3 threading, (**A**) for medium targets, scatter plot of the best model from *TASSER_org* versus the best model from *TASSER_low-zsc*. (**B**) Same as (A) but for hard targets.

TASSER_low-zsc is better for almost all TM-score threshold cut-off values.

In the above sections, we analyzed and discussed the results of *TASSER_low-zsc* models after selection and ranking from an ensemble of models generated after various TASSER simulations. In addition, from this ensemble, we can find the best model and determine the maximum possible improvement for *TASSER_low-zsc* over *TASSER_org*. For medium and hard targets, the best possible model from *TASSER_low-zsc* has an average TM-score to native of 0.442 and 0.425, respectively. The number of foldable proteins is 81 (100) for medium (hard) targets. Hence, given a perfect algorithm to select and rank models, we could achieve a TM-score improvement of ~16 and 20% for medium and hard targets, respectively, over models from *TASSER_org* method. However, in *TASSER_low-zsc*, we are able to achieve a ~7–9% TM-score improvement on average with respect to *TASSER_org*. This demands that we further improve the model selection and ranking.

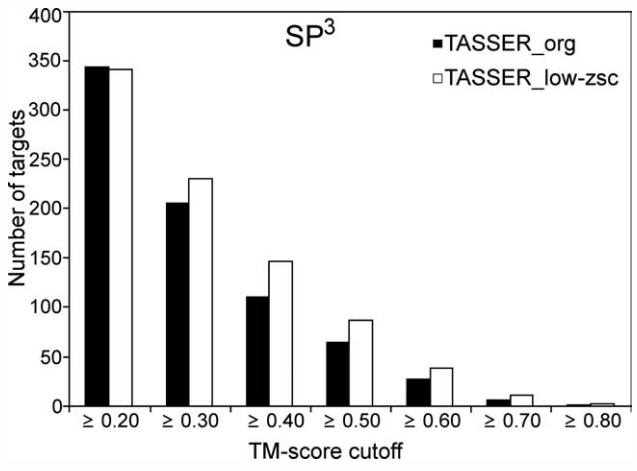

Figure 3

Histogram comparison between *TASSER_org* and *TASSER_low-zsc* on the 351 target SP^3 benchmark dataset. TM-scores are from the best of top five models.

The improvement in structure prediction accuracy from *TASSER_low-zsc* with respect to *TASSER_org* is due to a combination of different factors: the identification of more "good" templates, the fact that structurally similar templates are clustered together enriches the consensus information with concomitant improvement in contact prediction accuracy, and the ability of TASSER to improve over the initial template structure. Finally, improved model selection and ranking also contributes to the success of the method, which requires a set of similar models close to the native structure. Because of the convoluted effects of these factors, it is difficult to ascertain their relative contributions.

Below, we discuss examples, where a combination of these factors contributes to the improvement in the model from *TASSER_low-zsc* when compared with *TASSER_org*. As shown in Figure 4, target 2qwv_A has a *TASSER_org* best model with a TM-score of 0.383. In the

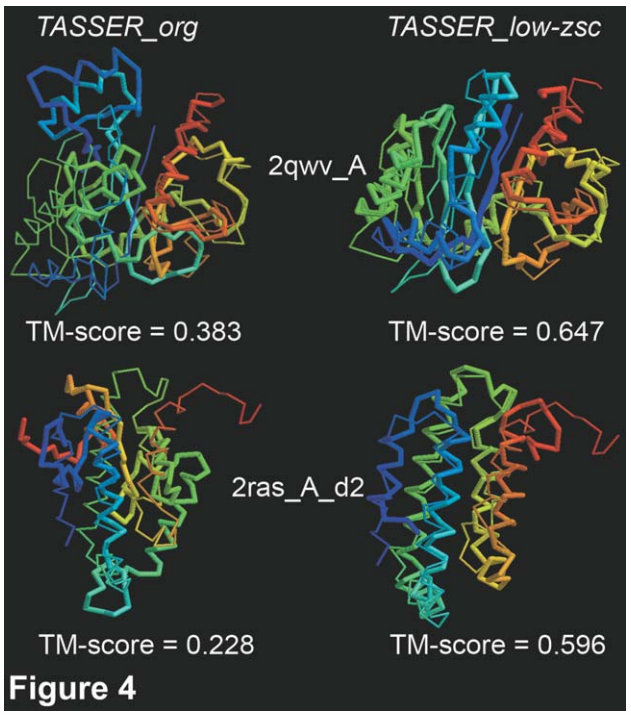

Figure 4

Examples showing an improved predicted model from *TASSER_low-zsc* in comparison with *TASSER_org* for two targets from the SP[3] benchmark dataset. For each target, the superposition of the native structure (thin backbone) and best model (thick backbone) from *TASSER_org* (on the left) and from *TASSER_low-zsc* (on the right) are shown. Blue to red goes from the N- terminus to the C-terminus.

top 10 templates (the set used in *TASSER_org*), there is only one good template, which has a TM-score of 0.572 to native structure and ranked seventh. There is another template with a TM-score of 0.434 to native structure; however, it is ranked 41 in *z*-score–based ranking. Interestingly, *thread_cluster* could cluster these two good templates into a single cluster along with other templates having a TM-score $\geq$ 0.35 to the native structure. This probably contributed to the successful prediction from *TASSER_low-zsc*, which has the best of top five models with a TM-score of 0.647 to the native structure.

Another interesting example is target 2ras_A_d2 shown in Figure 4. For this protein, the best template has a TM-score of 0.539 (ranked 10th) among the top 10 templates. However, the remaining nine templates have a TM-score to native <0.20. The best model from *TASSER_org* has a TM-score of 0.228, even though the best template is included in TASSER_*org* modeling. In the low *z*-score–ranked templates, there are several templates with TM-score $>$ 0.40; these are ranked at positions >12. The best of these has a TM-score to native of 0.605 and is ranked 27th. Interestingly, *thread_cluster* clusters all these templates having TM-score $>$ 0.40 to native into one cluster. Furthermore, the selection and ranking procedure resulted in the best model (among top five models) having a TM-score of 0.596 (see Fig. 4).

This suggests that *TASSER_low-zsc* can improve the structure prediction accuracy over the standard *TASSER_org* models. Apart from providing better models for targets, which have templates with TM-score $<$ 0.40 in the top 10 templates, *TASSER_low-zsc* also provides better models for other targets whose TM-score >0.4.

Benchmarking results using templates from SPARKS[2]

In the previous section, we have demonstrated that using low *z*-score templates from SP[3], *TASSER_low-zsc* improves structure prediction, as assessed by TM-score, with respect to the model generated with the original template set (*TASSER_org*). We next assess its applicability to templates generated from a different threading program, SPARKS[2]. In Table I, we have already shown that for SPARKS[2] *thread_cluster* could recover the best template for most medium/hard targets. Here, we compare the prediction success of *TASSER_low-zsc* and *TASSER_org*. The structure prediction procedure is the same as that used for SP[3] (see Methods). The performance comparison between *TASSER_low-zsc* and *TASSER_org* is presented in Table IV. We note that the average TM-score to native of the best possible model among the ensemble of models is 0.428 and 0.365 for medium and hard targets, respectively. Hence, given the ability to choose the best model from the ensemble, we could achieve an improvement of ~16% (~21%) for medium (hard) targets. In practice, for the first model, *TASSER_low-zsc* shows an average TM-score improvement of ~5% (~7%) over *TASSER_org* for medium (hard) targets. For the best of five models, a similar TM-score improvement of ~6% and 7% is observed for medium and hard targets, respectively. Furthermore, the number of foldable targets using *TASSER_low-zsc* increases by 11% for medium proteins. For hard proteins, *TASSER_low-zsc* has almost twice the number of foldable targets as *TASSER_org*.

Next, for a detailed analysis of the improvement observed with *TASSER_low-zsc*, we classified the medium/hard targets into four sets as defined above for the evaluation of SP[3] (see "Benchmarking results using templates from SP[3]"). The results are presented in Table S1

Table IV

Summary of Structure Prediction Results for SPARKS[2] Medium/Hard Targets

Target Type	No. of Targets	First Model		Best of Top Five Models	
		$\langle TMM_{org} \rangle$	$\langle TMM_{clus} \rangle$	$\langle TMM_{org} \rangle$	$\langle TMM_{clus} \rangle$
Medium	155	0.335 (43)	0.351 (44)	0.369 (57)	0.395 (63)
Hard	202	0.267 (16)	0.283 (22)	0.302 (20)	0.324 (38)

TMM_{org} and TMM_{clus} refer to the TM-score of the model from the *TASSER_org* and *TASSER_low-zsc*, respectively. The number in the parenthesis is the number of proteins with TM-score $\geq$ 0.40.

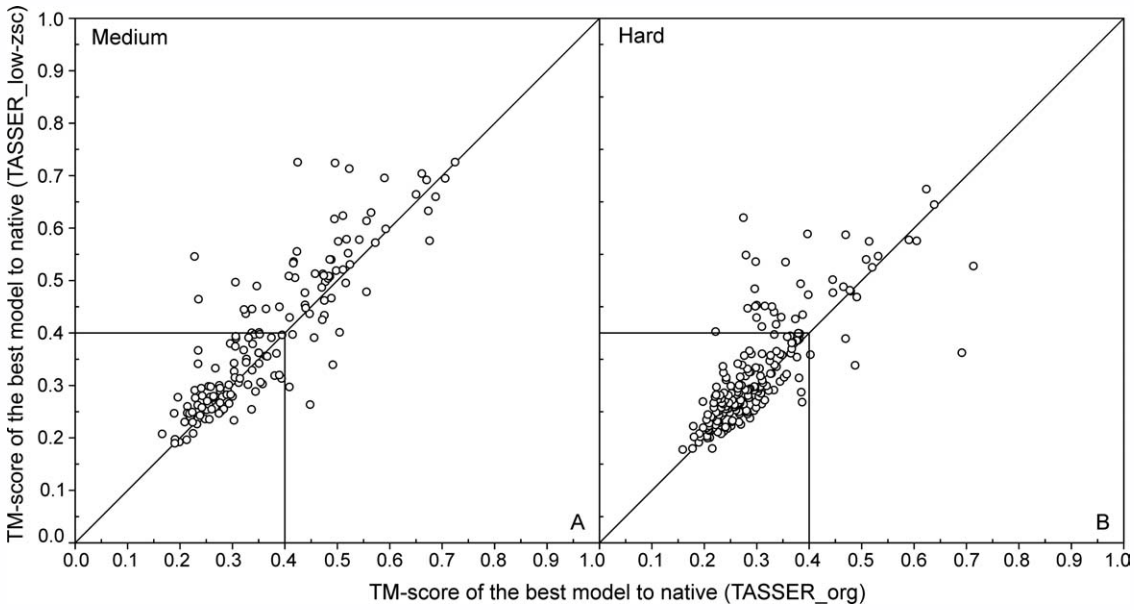

Figure 5

Using SPARKS[2] threading, (**A**) for medium targets, scatter plot of the best model from *TASSER_org* approach versus the best model from *TASSER_low-zsc*. (**B**) Same as (A) but for hard targets.

(see in Supporting Information). The improvement in targets with best (of top five) model TM-score to native structure $\geq$0.40 only from *TASSER_low-zsc* is ~42% (~43%) for medium (hard) proteins over *TASSER_org*. Similarly, targets with best model TM-score $\geq$0.40 to the native structure from both methods shows on average a TM-score improvement of ~7% (~2%). However, there are a few targets having a TM-score $\geq$0.40 from *TASSER_org*, for which *TASSER_low-zsc* gives poor quality models (TM-score $<$ 0.40). This is partly due to issues in model ranking. Thus, the behavior of *TASSER_low-zsc* when SPARKS[2] is used is very similar to that of SP[3].

In Figure 5(A,B), we show the comparison of the best model from *TASSER_org* and *TASSER_low-zsc* for medium and hard targets, respectively. On average, *TASSER_low-zsc* performs better than *TASSER_org* as was the case when templates from SP[3] are used. Similarly, as shown in Figure S1 (see in Supporting Information), for TM-scores $\geq$0.3, the improvement of *TASSER_low-zsc* is seen over all values of the TM-score. For some targets, the predicted model from *TASSER_low-zsc* has a lower TM-score in comparison with the *TASSER_org* model. As discussed before, this is partly due to issues in model ranking.

Prediction results for CASP8 targets

We have used *TASSER_low-zsc* with the templates derived from SP[3] and SPARKS[2] for the assessment of CASP8 targets.

The targets are classified into easy/medium/hard categories based on *z*-score of the first template. For SP[3], the number of targets classified into the easy, medium, and hard set is 87, 13, and 8, respectively. Similarly, in the case of SPARKS[2], the number of easy, medium, and hard targets is 91, 3, and 14, respectively. Because little if any improvement from *TASSER_low-zsc* is expected for easy targets, we are limited to a very small subset of CASP8 proteins on which to perform the analysis, and the results are not likely to be statistically significant. Nevertheless, it is of interest to see how *TASSER_low-zsc* would have performed in CASP8. In the analysis, we have used 21 and 17 medium/hard targets from SP[3] and SPARKS[2], respectively (see Table S2 in Supporting Information). In the following section, we present the assessment of structure prediction using templates selected by SP[3] and then from SPARKS[2].

The *thread_cluster* procedure on templates from SP[3] shows an average improvement of ~10 and 5% for medium and hard targets, respectively (Table S3 in Supporting Information). For both medium and hard categories, only one additional protein has a TM-score $\geq$0.40 after *thread_cluster*. We used the modeling procedure as described in the Methods section. In Figure 6, for medium/hard targets, we show the comparison of the best model from *TASSER_org* and *TASSER_low-zsc*. For medium targets, the first (best) model using *TASSER_low-zsc* shows an average TM-score improvement of ~4% (~6%) over *TASSER_org* (Table S4 in Supporting Information). However, for hard targets, the average TM-score

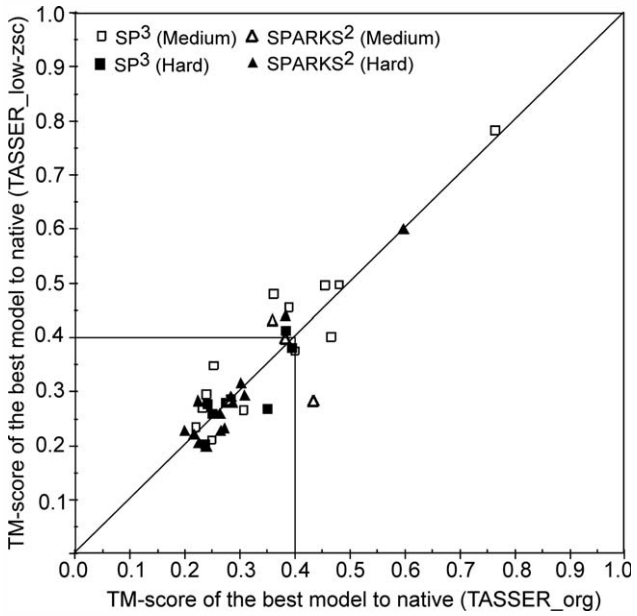

Figure 6

For CASP8 targets, scatter plot of the best model from *TASSER_org* versus the best model from *TASSER_low-zsc* by using templates from SP^3 or $SPARKS^2$.

of the best model from *TASSER_low_zsc* becomes slightly worse in comparison with *TASSER_org* (Table S4 in Supporting Information). Further, detailed analysis showed that, for most of hard targets, good templates (TM-score $\geq$ 0.40) are not in top *N* set of templates. Thus, their best model has a TM-score $<$ 0.40 for both modeling procedures (Table S5A in Supporting Information). An interesting example is target T0480, classified in the medium category. The best model from *TASSER_org* has a TM-score to native of 0.390. The clustering method (*thread_cluster*) could cluster low-rank templates (at positions 15, 28, 31, 33, and 40) having TM-scores $>$0.40 into one cluster. This resulted in a best model from *TASSER_low_zsc* with a TM-score to native of 0.456.

Using templates from $SPARKS^2$, *thread_cluster* is not able to recover the best possible template after clustering for one of the medium targets, T0514. For this target, there is only one template with TM-score $>$0.40, and this could not cluster with any other templates. A similar issue exists for hard target T0399. Subsequent to modeling, the best models from *TASSER_low_zsc* on average do not show significant improvement with respect to *TASSER_org* (Table S4 in Supporting Information). Then, we classified the various medium/hard targets as to whether their TM-scores are above or below 0.4 for the *TASSER_org* and *TASSER_low_zsc* in Table S5B (see in Supporting Information). In case of medium targets, because of T0514, *TASSER_org* method uses a template with TM-score $>$0.4 but *TASSER_low_zsc* does not identify this template; this reduced the TM-score from 0.434 to 0.282, but these results are anecdotal, as just one target is considered.

Figure 6 shows the comparison of the best model from both modeling procedures for medium/hard targets. For most hard targets, good templates (TM-score $\geq$0.40) were not present in top *N* set of templates. Hence, an improvement in structure prediction accuracy using *TASSER_low_zsc* similar to the benchmark studies could not be observed with CASP8 targets. An interesting case is T0482, which is classified as a hard target by $SPARKS^2$. The best model from *TASSER_org* has a TM-score to native of 0.383, after *TASSER_low_zsc*, two templates at position 7 and 8 with TM-score $>$ 0.38 clustered together with other similar templates. This then provides the best model from *TASSER_low_zsc* with a TM-score to native of 0.441. But, again, caution should be taken in interpreting the generality of these results as there are very few targets in the medium/hard regime.

These results suggest that *TASSER_low-zsc* could be extended to other threading procedures as well. In addition, *thread_cluster* could be combined with other modeling procedures, which use information from multiple templates as an input to improve structure prediction.

CONCLUSIONS

We have developed the *TASSER_low-zsc* approach to improve protein structure prediction by using low *z*-score ranked but good quality templates. Template clustering is performed by using the *thread_cluster* algorithm, which attempts to retain only structurally similar templates as measured by their TM-score. *TASSER_low-zsc* was benchmarked for medium/hard targets from the SP^3 and $SPARKS^2$ threading methods. The best model from *TASSER_low-zsc* shows an average TM-score improvement of $\sim$6–9% with respect to the best model generated from *TASSER_org*. Furthermore, the number of foldable targets is significantly improved in *TASSER_low-zsc*.

A key unresolved issue is why there are good quality alignments in poorly ranked templates? Some unpublished work examining the set of structural alignments of the top-ranked templates to native indicates that the majority have a good structural alignment to the native fold over a least portion of their structure. If one requires that a set of three templates generate mutually consistent structural alignments, then $\sim$14% of the residues have $>$90% probability of being part of the best structural alignment to native. In other words, there is a residual core of aligned residues that are commonly identified by threading (with the implication that the set of proteins are evolutionary related), and that this is the signal that is detected in the poorly ranked templates. In future work, we plan on exploring this issue in much greater detail.

At this juncture, what is encouraging is that *TASSER_low-zsc* significantly increases the fraction of medium/hard targets that are foldable. For example, if the best of top five models are considered, for medium/hard targets from SP^3, the percentage of foldable proteins increases from 39/26% using *TASSER_org* to 48/37%; this is precisely the regime of target difficulty, where progress has been quite slow. Despite this success, additional extensions are required. *Thread_cluster* will be used with other protein structure prediction methods, which exploit information from multiple templates as well as meta-approaches. Work along this direction is in progress.

REFERENCES

1. Baker D, Sali A. Protein structure prediction and structural genomics. Science 2001;294:93–96.
2. Skolnick J, Fetrow J, Kolinski A. Structural genomics and its importance for gene function analysis. Nat Biotechnol 2000;18:283–287.
3. Zhang Y. Protein structure prediction: when is it useful? Curr Opin Struct Biol 2009;19:145–155.
4. Tress M, Ezkurdia I, Grana O, Lopez G, Valencia A. Assessment of predictions submitted for the CASP6 comparative modeling category. Proteins 2005;61(Suppl 7):27–45.
5. Marti-Renom MA, Stuart AC, Fiser A, Sanchez R, Melo F, Sali A. Comparative protein structure modeling of genes and genomes. Annu Rev Biophys Biomol Struct 2000;29:291–325.
6. Schwede T, Kopp J, Guex N, Peitsch MC. SWISS-MODEL: an automated protein homology-modeling server. Nucleic Acids Res 2003; 31:3381–3385.
7. Bowie JU, Luthy R, Eisenberg D. A method to identify protein sequences that fold into a known three-dimensional structure. Science 1991;253:164–170.
8. Skolnick J, Kihara D, Zhang Y. Development and large scale benchmark testing of the PROSPECTOR 3.0 threading algorithm. Proteins 2004;56:502–518.
9. Skolnick J, Kihara D. Defrosting the frozen approximation: PROSPECTOR—a new approach to threading. Proteins 2001;42:319–331.
10. Zhou H, Zhou Y. Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. Proteins 2005;58:321–328.
11. Zhou H, Zhou Y. Single-body residue-level knowledge-based energy score combined with sequence-profile and secondary structure information for fold recognition. Proteins 2004;55:1005–1013.
12. Simons KT, Strauss C, Baker D. Prospects for ab initio protein structural genomics. J Mol Biol 2001;306:1191–1199.
13. Pillardy J, Czaplewski C, Liwo A, Lee J, Ripoll DR, Kazmierkiewicz R, Oldziej S, Wedemeyer WJ, Gibson KD, Arnautova YA, Saunders J, Ye YJ, Scheraga HA. Recent improvements in prediction of protein structure by global optimization of a potential energy function. Proc Natl Acad Sci USA 2001;98:2329–2333.
14. Dominy BN, Books CL. Identifying native-like protein structures using physics-based potentials. J Comput Chem 2002;23:147–160.
15. Liwo A, Khalili M, Scheraga HA. Ab initio simulations of protein-folding pathways by molecular dynamics with the united-residue model of polypeptide chains. Proc Natl Acad Sci USA 2005;102: 2362–2367.
16. Altschul SF, Koonin EV. Iterated profile searches with PSI-BLAST–a tool for discovery in protein databases. Trends Biochem Sci 1998; 23:444–447.
17. Kopp J, Bordoli L, Battey JN, Kiefer F, Schwede T. Assessment of CASP7 predictions for template-based modeling targets. Proteins 2007;69(Suppl 8):38–56.
18. Jaroszewski L, Rychlewski L, Li W, Godzik A. Comparison of sequence profiles. Strategies for structural predictions using sequence information. Protein Sci 2000;9:232–241.
19. Yona G, Levitt M. Within the twilight zone: a sensitive profile-profile comparison tool based on information theory. J Mol Biol 2002; 315:1257–1275.
20. Marti-Renom MA, Madhusudhan MS, Sali A. Alignment of protein sequences by their profiles. Protein Sci 2004;13:1071–1087.
21. Liu S, Zhang C, Liang S, Zhou Y. Fold recognition by concurrent use of solvent accessibility and residue depth. Proteins 2007;68:636–645.
22. Fischer D. Hybrid fold recognition combining sequence derived properties with evolutionary information. In: Altman RB, Dunker AK, Hunter L, Lauderdale K, Klein TE, editors. Pacific Symposium on Biocomputing 2000, Hawaii: World Scientific; 2000; pp 119–130.
23. Kelley LA, MacCallum RM, Sternberg MJE. Enhanced genome annotation using structural profiles in the program 3D-PSSM. J Mol Biol 2000;299:499–520.
24. Chivian D, Baker D. Homology modeling using parametric alignment ensemble generation with consensus and energy-based model selection. Nucleic Acids Res 2006;34:e112.
25. Karplus K, Barrett C, Hughey R. Hidden Markov models for detecting remote protein homologies. Bioinformatics 1998;14:846–856.
26. Soding J. Protein homology detection by HMM-HMM comparison. Bioinformatics 2005;21:951–960.
27. Rangwala H, Karypis G. Profile-based direct kernels for remote homology detection and fold recognition. Bioinformatics 2005;21: 4239–4247.
28. Cheng J, Baldi P. A machine learning information retrieval approach to protein fold recognition. Bioinformatics 2006;22:1456–1463.
29. Ginalski K, Elofsson A, Fischer D, Rychlewski L. 3D-Jury: a simple approach to improve protein structure predictions. Bioinformatics 2003;19:1015–1018.
30. Fischer D. 3D-SHOTGUN: a novel, cooperative, fold-recognition meta-predictor. Proteins 2003;51:434–441.
31. Wu S, Zhang Y. LOMETS: a local meta-threading-server for protein structure prediction. Nucleic Acids Res 2007;35:3375–3382.
32. Zhang Y, Skolnick J. Automated structure prediction of weakly homologous proteins on a genomic scale. Proc Natl Acad Sci USA 2004;101:7594–7599.
33. Cheng J. A multi-template combination algorithm for protein comparative modeling. BMC Struct Biol 2008;8:18.
34. Lee SY, Skolnick J. Development and benchmarking of TASSER (iter) for the iterative improvement of protein structure predictions. Proteins 2007;68:39–47.
35. Wu S, Skolnick J, Zhang Y. Ab initio modeling of small proteins by iterative TASSER simulations. BMC Biology 2007;5:17.
36. Pillardy J, Czaplewski C, Liwo A, Wedemeyer WJ, Lee J, Ripoll DR, Arlukowicz P, Oldziej S, Arnautova YA, Scheraga HA. Development of physics-based energy functions that predict medium-resolution structures for proteins of the alpha, beta and alpha/beta structural classes. J Phys Chem B 2001;105:7299–7311.
37. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE. The Protein Data Bank. Nucleic Acids Res 2000;28:235–242.
38. Kihara D, Skolnick J. The PDB is a covering set of small protein structures. J Mol Biol 2003;334:793–802.
39. Zhang Y, Skolnick J. The protein structure prediction problem could be solved using the current PDB library. Proc Natl Acad Sci USA 2005;102:1029–1034.
40. Zhou H, Skolnick J. Protein structure prediction by pro-sp3-TASSER. Biophys J 2009;96:2119–2127.
41. Zhang Y, Skolnick J. Tertiary structure predictions on a comprehensive benchmark of medium to large size proteins. Biophys J 2004; 87:2647–2655.
42. Zhou H, Skolnick J. Ab initio protein structure prediction using chunk-TASSER. Biophys J 2007;93:1510–1518.

43. Zhou H, Pandit SB, Lee SY, Borreguero J, Chen H, Wroblewska L, Skolnick J. Analysis of TASSER-based CASP7 protein structure prediction results. Proteins 2007;69(Suppl 8):90–97.
44. Lee SY, Skolnick J. Benchmarking of TASSER_2.0: an improved protein structure prediction algorithm with more accurate predicted contact restraints. Biophys J 2008;95:1956–1964.
45. Mirny LA, Finkelstein AV, Shakhnovich EI. Statistical significance of protein structure prediction by threading. Proc Natl Acad Sci USA 2000;97:9978–9983.
46. Jones DT. GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. J Mol Biol 1999;287: 797–815.
47. Xu Y, Xu D, Olman V. A practical method for interpretation of threading scores: an application of neural networks. Stat Sin Spec Issue Bioinformatics 2002;12:159–177.
48. Xu J. Fold recognition by predicted alignment accuracy. IEEE/ACM Trans Comput Biol Bioinform 2005;2:157–165.
49. Zhang Y, Skolnick J. Scoring function for automated assessment of protein structure template quality. Proteins 2004;57:702–710.
50. Moult J. A decade of CASP: progress, bottlenecks and prognosis in protein structure prediction. Curr Opin Struct Biol 2005;15:285–289.
51. Kryshtafovych A, Krysko O, Daniluk P, Dmytriv Z, Fidelis K. Protein structure prediction center in CASP8. Proteins 2009;77(Suppl 9):5–9.
52. Zhou H, Pandit SB, Skolnick J. Performance of the Pro-sp3-TASSER server in CASP8. Proteins 2009;77(Suppl. 9):123–127.
53. Frey BJ, Dueck D. Clustering by passing messages between data points. Science 2007;315:972–976.
54. Zhang Y, Skolnick J. SPICKER: a clustering approach to identify near-native protein folds. J Comput Chem 2004;25:865–871.
55. Zhou H, Skolnick J. Protein model quality assessment prediction by combining fragment comparisons and a consensus C(alpha) contact potential. Proteins 2008;71:1211–1218.
56. Kryshtafovych A, Fidelis K. Protein structure prediction and model quality assessment. Drug Discov Today 2009;14:386–393.
57. Rotkiewicz P, Skolnick J. Fast procedure for reconstruction of full-atom protein models from reduced representations. J Comput Chem 2008;29:1460–1465.