PROTEINS
STRUCTURE ■ FUNCTION ■ BIOINFORMATICS

MUSTER: Improving protein sequence profile–profile alignments by using multiple sources of structure information

Sitao Wu and Yang Zhang*

Center for Bioinformatics and Department of Molecular Bioscience, University of Kansas, 2030 Becker Dr, Lawrence, Kansas 66047

ABSTRACT

We develop a new threading algorithm MUSTER by extending the previous sequence profile–profile alignment method, PPA. It combines various sequence and structure information into single-body terms which can be conveniently used in dynamic programming search: (1) sequence profiles; (2) secondary structures; (3) structure fragment profiles; (4) solvent accessibility; (5) dihedral torsion angles; (6) hydrophobic scoring matrix. The balance of the weighting parameters is optimized by a grading search based on the average TM-score of 111 training proteins which shows a better performance than using the conventional optimization methods based on the PROSUP database. The algorithm is tested on 500 nonhomologous proteins independent of the training sets. After removing the homologous templates with a sequence identity to the target >30%, in 224 cases, the first template alignment has the correct topology with a TM-score >0.5. Even with a more stringent cutoff by removing the templates with a sequence identity >20% or detectable by PSI-BLAST with an E-value <0.05, MUSTER is able to identify correct folds in 137 cases with the first model of TM-score >0.5. Dependent on the homology cutoffs, the average TM-score of the first threading alignments by MUSTER is 5.1–6.3% higher than that by PPA. This improvement is statistically significant by the Wilcoxon signed rank test with a P-value < 1.0×10^{-13}, which demonstrates the effect of additional structural information on the protein fold recognition. The MUSTER server is freely available to the academic community at http://zhang.bioinformatics.ku.edu/MUSTER.

Proteins 2008; 72:547–556.

Key words: threading; protein structure prediction; TM-score; solvent accessibility; dihedral angle prediction; hydrophobic scoring matrix.

INTRODUCTION

Template-based modeling is by far the most reliable and accurate approach to the problem of protein structure prediction.[1–3] The critical step for the template-based modeling is to identify correct template proteins from the PDB library which have similar folds as the target protein and to make correct alignment between the target sequence and the template structure. This process is called threading or fold recognition.[4–6] There have been a number of threading algorithms in literature based on various approaches, for example the sequence profile–profile alignment,[7–14] structural profile alignment,[15–17] hidden Markov models,[18–20] machine learning,[21,22] and pair-wise potentials with optimal searching.[23–26]

We recently developed a simple threading algorithm, PPA,[27] by the sequence profile alignment of target and template proteins combined with the secondary structure match. The PPA algorithm has been successfully used in CASP7 as the initial step of the I-TASSER modeling.[28] PPA is also used in the LOMETS meta-server threading where the average TM-score[29] of PPA is comparable to the best of other single programs in the LOMETS package.[30] On the other hand, the general profile–profile alignment methods[7–14] have demonstrated dominant advantages in many blind threading tests.[31–34] For example, in LiveBench-8,[31] all top four servers (BASD/MASP/MBAS, SFST/STMP, FFAS03, and ORF2/ORFS) are based on the sequence profile–profile alignment. In CAFASP[32] and the recent CASP server section,[34] several sequence-profile-based methods were ranked at the top of single threading servers. Nevertheless, many authors show that additional sequence and structure features can improve further the accuracy of the sequence-structure alignments.[13,14,25,35,36] For example, Zhou *et al.* show that a fragment-depth based structure profile can improve both the sensitivity and specificity of the sequence profile alignments.[14] Silva shows that a simplified hydrophobicity matrix based on the hydrophobic cluster analysis (HCA)[37] can detect similar folds without sharing obvious sequence similarity.[36] Skolnick *et al.* show that iterative contact predictions can significantly improve the recognition power

Grant sponsor: KU Start-up Fund; Grant number: 06194.
*Correspondence to: Yang Zhang, Center for Bioinformatics and Department of Molecular Bioscience, University of Kansas, 2030 Becker Dr, Lawrence, Kansas 66047. E-mail: yzhang@ku.edu
Received 22 October 2007; Accepted 4 December 2007
Published online 4 February 2008 in Wiley InterScience (www.interscience.wiley.com).
DOI: 10.1002/prot.21945

of the sequence-based methods especially for the remote homologous sequences.[25,26]

In this work, we try to extend the PPA threading algorithm by including various sequence and structural resources generated from many other tools, which is called MUSTER (MUlti-Source ThreadER). Because we use dynamics programming (DP)[38,39] to search the alignment space, we only consider the single-body features which can be conveniently exploited in DP. They include: (1) sequence profiles[40]; (2) secondary structures prediction[41]; (3) depth-dependent structure profiles[14]; (4) solvent accessibility[42]; (5) backbone dihedral torsion angles; (6) hydrophobic scoring matrix.[36] Since these features are not entirely independent of each other (e.g. the structure profile correlates with the solvent accessibility, and the torsion angle prediction correlates with the secondary structure), we will carefully balance the weights of the contributions from different resources based on different training methods. The goal is to systematically examine how much gain we can obtain in fold recognition when we combine the different resources of structure features with the powerful sequence profile–profile alignment methods.

METHOD

Scoring functions

The scoring function of MUSTER for aligning the ith residue on the query and the jth residue on the template is

$$\begin{aligned}\mathrm{Score}(i,j) &= \sum_{k=1}^{20}(Pc_q(i,k)+Pd_q(i,k)L_t(j,k)/2+c_1\delta(s_q(i),s_t(j)) \\ &+c_2\sum_{k=1}^{20}Ps_t(j,k)L_q(i,k)+c_3(1-2|SA_q(i)-SA_t(j)|)\ , \\ &+c_4(1-2|\varphi_q(i)-\varphi_t(j)|)+c_5(1-2|\phi_q(i)-\phi_t(j)|) \\ &+c_6M(AA_q(i),AA_t(j))+c_7 \qquad (1)\end{aligned}$$

where "q" stands for the query and "t" for the template proteins. We explain the specific terms as follows.

Sequence profiles

The first term in Eq. (1) is the sequence-derived profiles. $Pc_q(i,k)$ is the frequency of the kth amino acid at the ith position of the multiple sequence alignments (MSA) obtained by a PSI-BLAST search[40] of the query sequence against a nonredundant sequence database (ftp://ftp.ncbi.nih.gov/blast/db) with an E-value cutoff of 0.001. This is the frequency profile from "close" homologies. A more "distant" frequency matrix $Pd_q(i,k)$ is generated using a higher E-value cutoff of 1.0. The combination of both close and distant sequence profiles follows Skolnick's idea[25,26,43] which helps increase the MUSTER alignment sensitivity in different homology area. We tried to use different weights to the close and distance profiles and found that the equal weights give the best performance. In calculating the frequency profiles of $Pc_q(i,k)$ and $Pd_q(i,k)$, the Henikoff and Henikoff[44] weights are used to reduce the redundancy of aligned multiple sequences. Moreover, to emphasize the sequence of more significant PSI-BLAST hits, we give stronger weights to the sequences of lower E-value than those of higher E-value. That is, the sequences with E-value of $<10^{-10}$ are given a weight of 1.0. The weight is linearly decreased with the logarithm of the E-values until a weight of 0.5 is used for sequences with E-value of 1.0. $L_t(j,k)$ is the log-odds profile (Position-Specific Substitution Matrix in PSI-BLAST) of the template sequence for the kth amino acid at the jth position. The template log-odds profile is obtained by the PSI-BLAST search with an E-value of 0.001. We attempted to combine $L_t(j,k)$ with the "distant" log-odds profiles with an E-value of 1.0 for the template sequence too. But it turned out not to increase the prediction accuracy.

Secondary structure match

The second term compares the predicted secondary structure $s_q(i)$ at the ith position of the query and the real secondary structures $s_t(j)$ at the jth position of the template. $\delta(s_q(i), s_t(j))$ equals to 1 if $s_q(i) = s_t(j)$ and -1 otherwise. The secondary structure for the query is predicted by PSI-PRED[41] and that for the template is assigned by the STRIDE program,[45] both having three states of helix, strand, and coil.

Structure profiles

The third term is a depth-dependent structure-derived profile which is similar as that used by Zhou *et al.*[14] Each template structure is split into small fragments with nine residues, which, as seed fragments, are compared by gapless threading with nine-residue fragments from a set of non-homologous PDB proteins selected by PISCES.[46] The fragments similar as the seed fragment are collected from the database and used to calculate the frequency profile at each position of the template, where the similarity is defined by RMSD and the fragment depth similarity[47] between the seed fragment and the fragments in the database. Following Zhou and Zhou,[14] we collect top 25 database fragments for each seed fragment. Thus, we have 225 fragment sequences aligned at each position on the template, where $Ps_t(j,k)$ is the frequency of the kth amino acid appearing in the 225 sequences corresponding to the jth position on the template. $L_q(i,k)$ is the log-odds profile for the kth amino acid at the ith position of the query sequence from the PSI-BLAST search with a E-value cutoff of 0.001.

Solvent accessibility

The fourth term in Eq. (1) computes the match between the predicted solvent accessibility $SA_q(i)$ for the

ith residue of the query and the real SA value $SA_t(j)$ of the jth residue of the template as assigned by STRIDE.[45] To predict $SA_q(i)$, we first trained a two-state (exposure/burial) neural network machine[42] on 3365 nonredundant high-resolution protein structures on the basis of their sequence profile from PSI-BLAST.[40] The maximum SA value in an extended tripeptide (Ala-X-Ala) is taken from Ahmad *et al.*[48] Seventeen different SA cutoffs (0.05, 0.1, . . . , 0.85) are used to define the residue exposure status in the NN training. The residue exposure index is $SA_q(i) = \sum_{m=1}^{17} a_{im}/17$ where a_{im} is the two-state neural network prediction of exposure ($a_{im} = 1$) or burial ($a_{im} = 0$) with the mth SA cutoff for ith residue of the query, which has a strong correlation with the real value of SA. For an independent set of 2234 nonhomologous proteins used by Zhang and Skolnick,[49,50] the overall correlation coefficient between the predicted $SA_q(i)$ and the real exposed area assigned by STRIDE[45] is 0.71, while the same correlation for the widely-used Hopp-Woods[51] and Kyte-Doolittle[52] hydrophobicity indices are 0.42 and 0.39, respectively.

Backbone dihedral torsion angles

The fifth (and sixth) term computes the match between the predicted Psi (and Phi) angle $\varphi_q(i)$ (and $\phi_q(i)$) for the ith residue of the query and the real Psi (and Phi) angle $\varphi_t(j)$ (and $\phi_t(i)$) of the jth residue of templates. Here both $\varphi_q(i)$ (and $\phi_q(i)$) and $\varphi_t(j)$ (and $\phi_t(i)$) are normalized by 360° so that the angle values stay in $[-0.5, 0.5]$. The predicted dihedral angles for queries are obtained from our SVR-ANGLE program (Wu and Zhang, submitted), which exploits the support vector regression (SVR) technique[53] (with LIB-SVM[54] as an implementation) to train the dihedral angles on an input vector of three feature sets: sequence profile, secondary structure, and solvent accessibility. The output of SVR-ANGLE is the real value of predicted torsion angles for each residue. Based on a test of 500 nonredundant testing proteins, the correlation coefficient between the predicted Psi (Phi) angles and the experimental Psi (Phi) angles assigned by DSSP[55] is 0.71 (0.63). The average absolute error for the Psi (Phi) predictions is 0.140 (0.091).

Hydrophobic scoring matrix

The seventh term is a 20 × 20 hydrophobic scoring matrix taken from Silva,[36] which is designed to match the hydrophobic patterns of query and template. The idea was inspired by the observation of Gaboriaud *et al.*,[37] where sequences with similar distribution patterns of the hydrophobic residues (V, I, L, F, Y, W, M) are often structural homologues. The hydrophobic scoring matrix is assigned as: $M(AA_q(i), AA_t(j))$ equals 1 when both $AA_q(i)$ and $AA_t(j)$ are from the set of hydrophobic residues. The matches between all identical residue pairs (except for Pro and Gly that are scored as 1) are scored as 0.7. All other matches are assigned a null score.

Dynamic programming

The Needleman-Wunsch[38] dynamic programming algorithm is used to identify the best match between the query and the template sequences. A position-dependent gap penalty in the dynamic programming is employed, that is no gap is allowed inside the secondary structure regions (helices and strands); gap opening (g_o); and gap extension (g_e) penalties apply to other regions; ending gap-penalty is neglected. The shift constant c_7 is introduced to avoid the alignment of unrelated residues in the local regions.

Template ranking scheme

In the original PPA program,[30] the templates are ranked based on a raw alignment score (R_{score}) divided by the full alignment length (L_{full}) (including query and template ending gaps) as shown in Figure 1. In MUSTER, we use $R_{score}/L_{partial}$ as another possible ranking scheme, where $L_{partial}$ is the partial alignment length excluding query ending gap as shown in Figure 1. A combined ranking is taken as following: if the sequence identity of the first template selected by $R_{score}/L_{partial}$ to the query is higher than that selected by R_{score}/L_{full}, we use the template ranking by $R_{score}/L_{partial}$. Otherwise, we use the template ranking by R_{score}/L_{full}.

Parameter training

There are overall nine parameters in the MUSTER algorithm (i.e. c_1 to c_7, g_o and g_e), which need to be appropriately tuned. One of the often-used tuning methods is based on the PROSUP database,[56] which includes 127 nonhomologous proteins pairs with the best possible alignment obtained by the structural alignment program PROSUP. To train the threading algorithms, one can tune the parameters by maximizing the number of threading-aligned residue-pairs which are identical to that in the structural alignments.[14,30,57,58]

There are a few protein pairs in the PROSUP databases which do not have similar topology. The residue-pairs by the PROSUP structural alignments in this portion of protein pairs do not correspond to meaningful topology coincidence and may mislead the trained threading algorithms. Therefore, we remove all PROSUP protein pairs with a TM-score <0.5 in our training. Here, TM-score has been defined by Zhang and Skolnick[29] to assess the topological similarity of protein structure pairs with a score in [0,1] Statistically, a TM-score <0.17 means a randomly selected protein pair with gapless alignment taken from PDB; TM-score >0.5 corresponds to the pro-

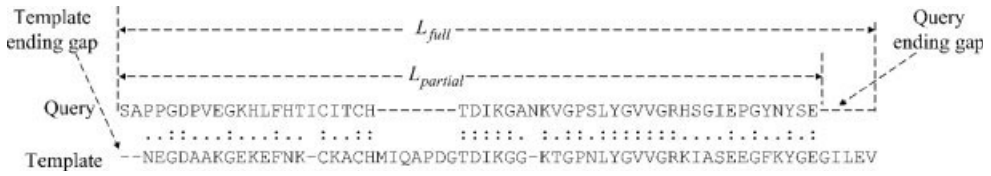

Figure 1

Illustration of the full (L_{full}) and partial ($L_{partial}$) alignment lengths used to normalize the threading alignment score (R_{score}). Symbols "-", ":" and ":" indicate an unaligned gap, an aligned nonidentical residue pair and an aligned identical residue pair, respectively. The query and template sequences are taken from 1hroA (first 53 residues) and 155c_ (first 61 residues), respectively, as an illustrative example.

tein pairs of similar fold. The statistical meaning of TM-score is independent of protein sizes.[29]

Because the number of protein pairs in PROSUP is relatively small (110 with TM-score >0.5) compared with the number of free parameters in MUSTER, we add a new set of 190 nonredundant protein structure pairs to our training set. These 190 pairs are selected with a TM-score >0.5 from 558 randomly chosen SCOP families,[59] where 120 pairs share the same "class" and "fold" but different "super-family" and 70 pairs share the same "class," "fold," and "super-family" but different "family." The protein length of the 190 protein pairs ranges from 43 to 812 with sequence identity between any pair is less than 30%. The sequence identity between any of the 190 proteins pairs and any of the 110 PROSUP protein pairs is also below 30%. A complete list of the composite 300 protein pairs are listed at our website: http://zhang.bioinformatics.ku.edu/MUSTER/data1.html with the structural alignments of the 190 protein pairs generated by TM-align[60] and that of the other 110 pairs taken from PROSUP.[56]

The second way of training the MUSTER parameters is to directly run the threading program on a small set of training proteins and optimize the parameters based on the overall TM-score of the final threading results. An advantage of this real-case training is that the target needs to scan all template proteins in the template library and the ranking system is naturally trained. For this purpose, we construct another set of 111 nonhomologous proteins, which include 39 "Easy" targets and 61 "Hard" targets taken from the PDB library (according to the PPA categorization) plus 11 new fold (NF) targets from the CASP6 experiment. This training set and the experimental structures are listed at: http://zhang.bioinformatics.ku.edu/MUSTER/data2.html.

We use a grid-search technique for both sets of training data, which split the 9-dimensional parameter space into lattices and try all the lattice points. In the training set 1, because the alignment searching on 300 protein pairs is quick, we use a finer parameter grid system and select the lattice with the highest average alignment accuracy. The program with this set of parameters is called MUSTER1. The final parameters used in MUSTER1 are $c_1 = 0.65$, $c_2 = 1.10$, $c_3 = 4.49$, $c_4 = 2.01$, $c_5 = 0.59$, $c_6 = 0.20$, $c_7 = 1.00$, $g_o = 6.99$, $g_e = 0.54$. In the training set 2, because MUSTER needs to scan all templates in the MUSTER library, the threading time is slower than that in the training set 1. We therefore use a more coarse-grained lattice system. After the initial selection, a finer tuning near the first selected lattice is performed. To avoid the homologous contamination, we exclude all templates with a sequence identity higher than 30% to the target proteins. The lattice with the highest average TM-score is selected. The program with this set of parameters is called MUSTER2. The final parameters used in MUSTER2 are $c_1 = 0.66$, $c_2 = 0.39$, $c_3 = 1.60$, $c_4 = 0.19$, $c_5 = 0.19$, $c_6 = 0.31$, $c_7 = 0.99$, $g_o = 7.01$, $g_e = 0.55$.

Z-score and target categorization

For the purpose of predicting the quality of threading alignments, we define a Z-score as

$$Z - \text{score} = \frac{(R'_{\text{score}} - \langle R'_{\text{score}} \rangle)}{\sqrt{\langle R'^2_{\text{score}} \rangle - \langle R'_{\text{score}} \rangle^2}}, \qquad (2)$$

where R'_{score} is the normalized score $R_{\text{score}}/L_{\text{full}}$ or $R_{\text{score}}/L_{\text{partial}}$ as described above; $\langle \cdots \rangle$ indicates the average over all templates. In Figure 2, we present the data of TM-score versus Z-score for all 5550 threading alignments by MUSTER2 (top 50 templates from each of the 111 train proteins). There is obviously a correlation between TM-score and Z-score, which allows us to use Z-score as an indication of the quality of the threading models. If we use Z-score = 7.5 as a cutoff of the successful alignment, the false positive and false negative rates for TM-score >0.5 are 1.2 and 5.3%, respectively. In the following, we will define the target as "Easy" ("Hard") target if the Z-score of the first alignment is higher (lower) than 7.5.

Template library

The protein structure templates are taken from the PDB library with a sequence identity cutoff = 70%. The theoretical models and the obsolete structures are dis-

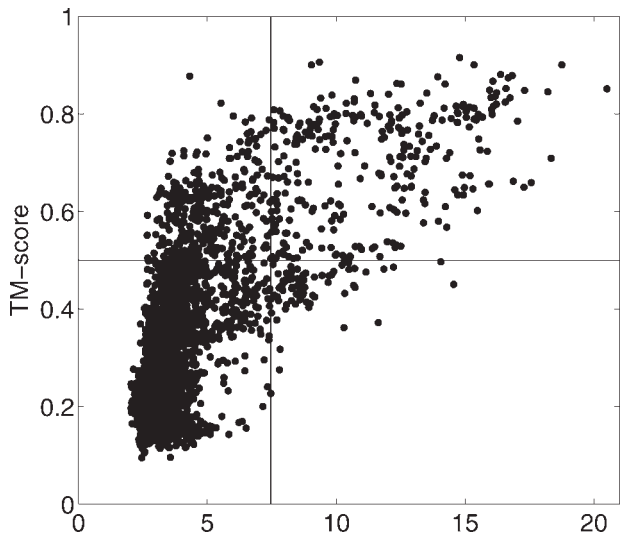

Figure 2

TM-scores of the top 50 threading alignments generated by MUSTER for each of 111 training proteins versus the Z-scores. The vertical line indicates a Z-score cutoff (= 7.5) to distinguish "Easy" and "Hard" targets and the horizontal line corresponds to TM-score = 0.5.

carded. If a template has multiple domains (e.g. Domains A and B), both the whole chain and the individual domains (e.g. Chain AB, Domain A, and Domain B) are included in the library. We found that the inclusion of individual domains increases the sensitivity of the MUSTER algorithm. As of August 1, 2007, the MUSTER template library includes 20,878 protein structures.

RESULTS

Results on training proteins

In Table I, we present the results of three programs (PPA, MUSTER1, and MUSTER2) on the two training protein sets. Column 3 is the average alignment accuracy (Acc) that is defined as fraction of the "correctly" aligned residue pairs which are the same as the golden-standard structure alignments. Column 4 is the average alignment accuracy (Acc_4) that is similar as Acc but allows a four-residue shift when compared with the structure alignments. Column 5 is the TM-score of the rank-1 models together with the RMSD and alignment coverage. Column 6 is that for the best in top-five models. As expected, the performance of the algorithms depends on whether the trained parameters are used. In the training set 1, MUSTER1 generates more accurate alignments than both PPA and MUSTER2; in the training set 2, the TM-score of the alignments by MUSTER2 is higher than that by both PPA and MUSTER1. A fair comparison has to be made based on an independent test set of proteins.

Results on testing proteins

Using PDBSELECT (2006 March),[61] we select a set of 500 nonhomologous proteins of sequence identity <25% and with length from 50 to 633, which includes 120(/53/327) $\alpha(/\beta/\alpha\beta)$ proteins. These proteins are also nonhomologous to the two sets of training proteins. A list of the 500 testing proteins and their PDB structures are available at http://zhang.bioinformatics.ku.edu/MUSTER/data3.html. For a fair comparison, the template library (including 20,878 templates) used for PPA is the same as those for MUSTER, where all the templates with sequence identity >30% to the query are excluded.

Overall performance

The performance and comparison of PPA and MUSTER on the 500 testing proteins are summarized in Table II. For the first (and the best in top-five) models, MUSTER1 identifies template alignments with an average TM-score of 0.4410 (and 0.4787), which is 2.9% (and 2.3%) higher than 0.4285 (and 0.4680) by PPA. If we use the parameters trained by threading the 111 proteins, MUSTER2 identifies even better template alignments with an average TM-score of 0.4503 (and 0.4830), which is 5.1% (and 3.2%) higher than that by PPA for the first (and the best in top-five) models. Based on the TM-

Table I

Performance of PPA and MUSTER on the Training Proteins

				TM-score (RMSD/coverage[c])	
Data	Methods	Acc[a]	Acc_4[b]	First model	Best in Top-5 models
Training Set 1	PPA	0.3686	0.6193	0.3887 (11.09/0.817)	—
(300 pairs)	MUSTER1	0.4674	0.7780	0.4549 (10.05/0.893)	—
	MUSTER2	0.4370	0.7254	0.4365 (10.37/0.873)	—
Training Set 2	PPA	—	—	0.4865 (7.95/0.884)	0.5121 (7.42/0.892)
(111 proteins)	MUSTER1	—	—	0.4957 (8.00/0.915)	0.5277 (7.20/0.920)
	MUSTER2	—	—	0.5117 (7.54/0.910)	0.5384 (7.05/0.911)

[a]Acc: Fraction of the correctly aligned residue pairs in threading that is identical to that by structural alignments.
[b]Acc_4: Fraction of the threading-aligned residue pairs with a shift $\leq$4 residues away from that by structural alignments.
[c]Coverage: The number of aligned residues/target protein length.

Table II
Performance of PPA and MUSTER on the Testing Proteins

Data	No. of targets	Homology cutoff	Methods	TM-score (RMSD/Coverage) First model	TM-score (RMSD/Coverage) Best in top-5 models
All targets	500	Cutoff-1[a]	PPA	0.4285 (10.10/0.849)	0.4680 (9.15/0.863)
			MUSTER1	0.4410 (10.33/0.904)	0.4787 (9.37/0.903)
			MUSTER2	0.4503 (9.87/0.885)	0.4830 (9.14/0.888)
		Cutoff-2[b]	PPA	0.3423 (11.95/0.824)	0.3824 (10.71/0.835)
			MUSTER1	0.3542 (12.15/0.892)	0.3996 (10.93/0.889)
			MUSTER2	0.3638 (11.53/0.865)	0.4022 (10.56/0.869)
''Easy'' targets	203	Cutoff-1	PPA	0.6430 (4.86/0.895)	0.6734 (4.52/0.902)
			MUSTER2	0.6571 (4.70/0.902)	0.6795 (4.49/0.909)
	93	Cutoff-2	PPA	0.5933 (5.26/0.868)	0.6209 (5.03/0.873)
			MUSETR2	0.6065 (5.04/0.878)	0.6267 (4.95/0.878)
''Hard'' targets	255	Cutoff-1	PPA	0.2564 (14.49/0.826)	0.3040 (12.98/0.841)
			MUSTER2	0.2842 (14.09/0.883)	0.3242 (13.09/0.885)
	365	Cutoff-2	PPA	0.2629 (14.13/0.816)	0.3075 (12.52/0.826)
			MUSTER2	0.2866 (13.65/0.867)	0.3295 (12.42/0.871)

[a]Homology Cutoff 1: excluding templates with sequence identity >30% to the query.
[b]Homology Cutoff 2: excluding templates with sequence identity >20% to the query or detectable by PSI-BLAST with an *E*-value < 0.05.

score, MUSTER2 certainly outperforms MUSTER1. Especially, the average alignment coverage of MUSTER2 is lower than MUSTER1 although the alignments of higher coverage tend to have higher TM-score.[29] This means that the higher TM-score in MUSTER2 is because of more accurate residue alignments. One reason for the better training in MUSTER2 than MUSTER1 is that the number of protein pairs in the second training set (111 × 20,878) is much larger than that of the first training set (300). Although the first training set allows a quick and much finer grids training, it could be over-trained by the small set of protein pairs especially when a number of parameters are trained. The second reason may be that the training process of the first training method does not consider the ranking scheme because there is only one template for each protein pair, while the ranking is naturally trained in the second training method. In the following, our analysis will only focus on the result of MUSTER2 (or MUSTER) unless explicitly mentioned.

The higher TM-score of MUSTER over PPA is because of both longer alignment coverage and the more accurate alignment as indicated by the smaller average RMSD. The difference between MUSTER and PPA is examined by the Wilcoxon signed rank test with a *P*-value < 1.0 ×

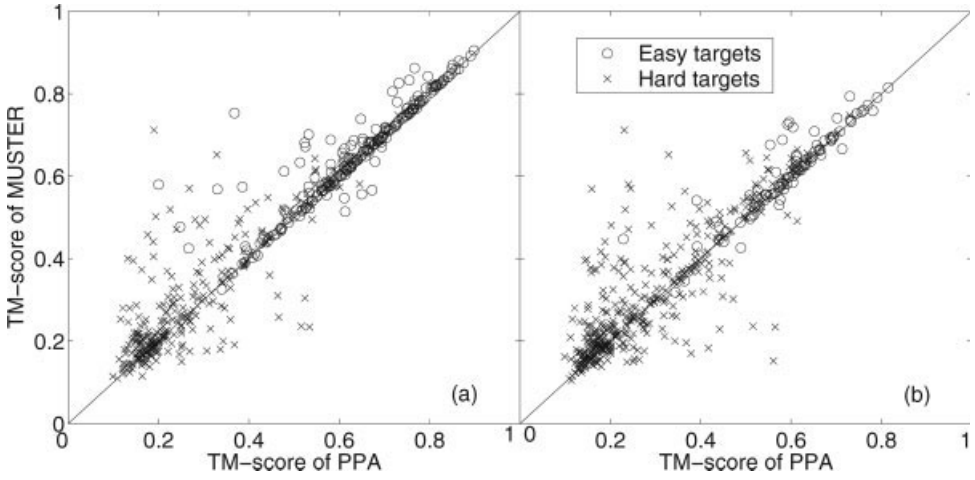

Figure 3
*TM-score comparison between PPA and MUSTER for the first threading models of 500 nonhomologous testing proteins. Circles represent the models from the "Easy" targets and crosses indicate those from the "Hard" targets. (**a**) Homology Cutoff-1 excluding templates with sequence identity to targets >30%; (**b**) homology Cutoff-2 excluding templates with sequence identity >20% or detectable by PSI-BLAST with an E-value > 0.05.*

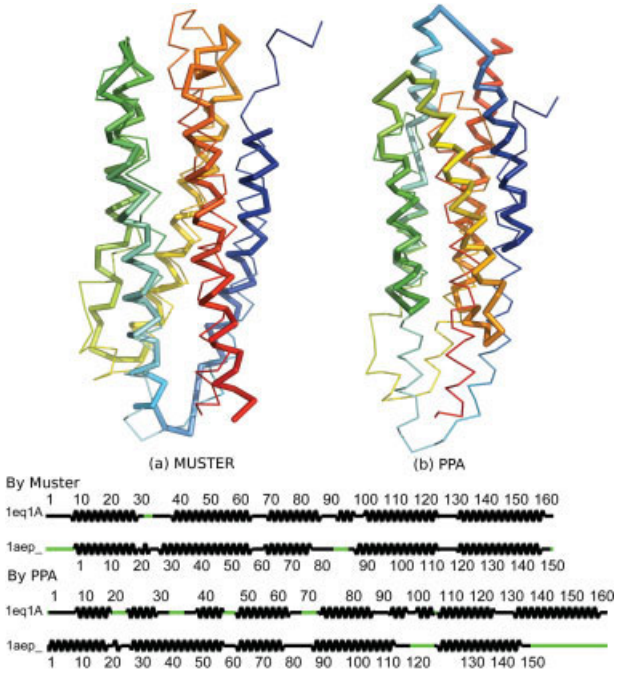

Figure 4

*The threading results for "1eq1A" by MUSTER and PPA. (**a**) The first threading model from the template "1aep_" by MUSTER; (**b**) The third threading model from the template "1aep_" by PPA. The upper part of the figure shows the superposition of 3D models to native structure. Thin line denotes C_α backbone of the native structure and thick line is that of threading models. Blue to red color runs from N- to C-terminus. The 3-D structures are plotted using PyMOL software.*[62] *The lower part of the figure shows the 1D alignment of the secondary structure elements by MUSTER and PPA, respectively. The wave lines indicate the α-helix regions and the straight lines the coil regions. The black color represents the continuous regions with residues appearing and the green color indicates the gap regions.*

10^{-16} for the first model and a *P*-value $< 1.0 \times 10^{-6}$ for the best in the top-five models. On 92 out of 500 proteins, MUSTER generates the first threading alignment with a TM-score 0.05 higher than PPA. Only on 24 targets, PPA does better than MUSTER with a TM-score difference >0.05. A head-to-head comparison of the first template alignments by PPA and MUSTER is presented in Figure 3(a) where there are obviously more points above the diagonal line which indicates that in more cases MUSTER identifies a better template alignment than PPA.

In Figure 4, we show a typical example from the target "1eq1A" where MUSTER does better than PPA. In this example, the first model identified by MUSTER has a TM-score = 0.71 with the template structure from "1aep_"; the first model by PPA has a TM-score = 0.19 with an incorrect template structure from "1av1A." Although the correct template "1aep_" is ranked as the 3rd model in PPA, the alignment is still not correct with a TM-score = 0.25 [Fig. 4 (b)]. Here, both the target "1eq1A" and the template "1aep_" are α-helix proteins and the secondary structure match is not the main driven force for the correct alignment. Actually, in PPA alignment, most aligned residue pairs are helix–helix matches but there is about an 11-residue shift from the correct alignment (see the lower panel of Fig. 4). From the 3D superposition of the threading model and native structure shown in Figure 4(b), the orientation of C-terminal tail is mismatched and there are big gaps between residues 27–45 and 90–107. Second, the sequence identity between "1eq1A" and "1aep_" is low (19%) and the contribution from the sequence profile alignment is small as well. When we specifically check the terms of the alignment path, the values of the seven terms for MUSTER are: 29.4 (sequence profiles), 72.6 (secondary structure match), −27.1 (structured profiles), 169.1 (solvent accessibility), 24.2 (Psi angle), 24.9 (Phi angle), and 8.9 (hydrophobic scoring matrix). The values of the two terms for PPA are 143.05 (sequence profiles) and 45.5 (secondary structure match) respectively. Obviously the main contribution to the correct alignments is from the new structure terms (solvent accessibility and dihedral angles in this example—the sum of new structural terms is much larger than the profile and secondary structure matches), which explains the reason of improvement of MUSTER over PPA by introducing additional features.

"Easy" versus "Hard" targets

In the lower part of Table II, we split the targets into "Easy" and "Hard" targets. Here we define an "Easy" target when both MUSTER and PPA categorize it as "Easy," that is *Z*-score (MUSTER) >7.5 and *Z*-score (PPA) >7.0; a "Hard" target is defined when both MUSTER and PPA categorize them as "Hard," that is *Z*-score (MUSTER) <7.5 and *Z*-score (PPA) <7.0. There are thus overall 203/255 "Easy"/"Hard" targets. The other 42 targets with nonidentical categorizations by PPA and MUSTER are not considered here.

For the "Easy" targets, the average TM-score of MUSTER is 0.6571 (0.6795) for the first (the best in top-five) models, which is 2.2% (0.9%) higher than 0.6430 (0.6734) by PPA. The percentage of the improvements is smaller than the overall testing set (5.1 and 3.2%, respectively) mainly because both the PPA and MUSTER alignments are closer to the perfect alignments and therefore there is less room for improvements. The average TM-score of the structural alignment identified by TM-align[60] for the first pair of target/template is 0.6930, which sets an upper bar where the best threading alignment can go. However, the improvement is still statistically significant as examined by the Wilcoxon signed rank test with a *P*-value $< 1.0 \times 10^{-7}$ for the first model and a *P*-value $< 1.0 \times 10^{-6}$ for the best in the top-five models.

For the "Hard" targets, the average TM-score of MUSTER is 0.2842 (0.3242) for the first (the best in top-five

models), which is 10.8% (6.6%) higher than 0.2564 (0.3040) by PPA. This improvement is higher than the overall testing set mainly because the performance of PPA in this set is much worse than the best structural alignment and there is a larger room for improvement. The TM-score by TM-align for the first target-template pairs is 0.3742. Obviously, even after the improvement, the average quality of MUSTER performance is considerably lower than the best structure alignment. This will be the major category of proteins which MUSTER has to deal with in future developments. Statistically, the improvement of MUSTER over PPA in the "Hard" targets is by Wilcoxon signed rank test with a *P*-value $< 1.0 \times 10^{-9}$ for the first model and a *P*-value $< 1.0 \times 10^{-9}$ for the best in top-five models.

Effect of using stringent cutoffs

Although we excluded the homologous templates using a sequence identity cutoff >30% to the target in the above testing as many previous studies did,[26,49,50,63] there are still some relatively "trivial" targets where homologous templates can be detected by PSI-BLAST.[40] Here, we use a second and more stringent homology cutoff where all threading templates with a sequence identity >20% to targets or can be detected by PSI-BLAST with an *E*-value < 0.05 are removed, which is called Cutoff-2 where the former one is referred to Cutoff-1.

The overall performance of PPA and MUSTER under Cutoff-2 is listed in Lines 6–8 of Table II. As expected, the average TM-scores of both PPA and MUSTER models are decreased compared with Cutoff-1. However, the average TM-score 0.3638 (0.4022) by MUSTER2 for the first (the best in top-five) model is still 6.3% (or 5.2%) higher than that of 0.3423 (0.3824) by PPA. The data is consistent with the result of Cutoff-1 in that MUSTER2 outperforms MUSTER1. The slightly larger increasing of MUSTER over PPA in Cutoff-2 is partly due to the fact that the effect of profile–profile match is somewhat reduced by the removing of the PSI-BLAST detectable templates which is the major driving force in the PPA alignment. The overall improvement from PPA to MUSTER in Cutoff-2 is examined by the Wilcoxon signed rank test with a *P*-value $< 1.0 \times 10^{-13}$ for the first model and a *P*-value $< 1.0 \times 10^{-18}$ for the best in top-five models.

In Figure 3(b), we present a head-to-head comparison of MUSTER and PPA under Cutoff-2. Again, there are more targets with a higher TM-score by MUSTER than that by PPA: In 105 out of 500 proteins, the TM-score difference is larger than 0.05 for MUSER over PPA; only in 34 cases, PPA does better than MUSTER with a TM-score difference larger than 0.05.

Under the Cutoff-2, there are respectively 93 "Easy" targets and 365 "Hard" targets. The comparison of PPA and MUSTER in both two categories is listed in Lines 11–12, and 15–16 of Table II. Both the first model and the best in top-five models of MUSTER have a higher average TM-score than that of PPA in the two sets, which demonstrates the robustness of the improvement of the MUSTER algorithm.

CONCLUSIONS

We develop a new threading program MUSTER by extending the secondary structure enhanced sequence profile-profile alignment algorithm (PPA[27]). To improve PPA, we add four additional structure-derived features to enhance the power of fold recognitions: (1) depth-dependent structure profiles[14]; (2) neural-network-based solvent accessibility predictions[42]; (3) SVR-based backbone torsion angle predictions; (4) hydrophobic scoring matrix.[36] These single-body features can be easily implemented in the dynamic programming procedure. We use several techniques to increase sensitivity of the threading alignments by combining both close and distant profiles, and weighting the PSI-BLAST sequences with the *E*-values. The final templates are ranked by the combination of two normalizing schemes.

We test two schemes to optimize the MUSTER parameters. The first is using 300 nonhomologous protein structure pairs where the goal is to maximize the percentage of aligned residue pairs that is identical to that in structural alignments.[56,60] The second is running MUSTER on 111 nonhomologous training targets and maximizing the TM-score of the final threading models. Although the first scheme allows a finer grid search of the parameters, the final result on the 500 independent proteins shows that the second scheme works better. The reason may be due to the fact that the number of the scanned alignment pairs in the second scheme is larger than the first one, which helps avoid the over-training of the parameters. The ranking of templates is also trained in the second scheme.

We test MUSTER on 500 nonhomologous proteins with two levels of homology cutoffs. In the first cutoff, we exclude all templates with sequence identity >30%. The average TM-score of the first rank models identified by MUSTER is about 5.1% higher than that by PPA. In 92 cases, the TM-score improvement by MUSTER over PPA is larger than 0.05 while only in 24 cases PPA does so over MUSTER. The improvement mainly occurs for the "Hard" targets where the average TM-score increase by 10.8%. For the "Easy" targets, the TM-score increase is 2.2%. This is partly because the alignments of "Hard" targets in PPA are less close to the perfect alignment and therefore there is a larger space to improve. Second, for most of the "Hard" targets the alignments are harder to be detected by the simple sequence profile–profile comparison and additional structure information as incorporated in MUSTER helps increase the alignment quality.

In the second more stringent cutoff, we exclude templates with a sequence identity >20% to the targets or detectable by PSI-BLAST with an *E*-value <0.05. The average alignment quality of the threading is worse than the first cutoff. But the average TM-score improvement of MUSTER over PPA is 6.3%, which demonstrates the robustness of the improvement. The total CPU time of running MUSTER is similar as PPA, that is, about 30 min to scan a medium size protein (200 residues) through a library of 20,878 templates.

Here, we note that we do not intend to compare MUSTER with many other threading algorithms rather than benchmark it strictly with PPA. One reason is that the threading performance is usually sensitive to the template library which varies considerably when using different pair-wise sequence cutoff and at different updating time. The interpretation of threading results of the algorithms from different groups may be misleading because of the different template libraries they are based on. By comparing MUSTER with PPA, we are using exactly the same template library and based on the same homology cutoffs. Therefore, the improvement should correspond purely to the progress of the algorithm. Second, the secondary structure bounded sequence profile–profile alignment algorithm, as used in PPA, represents a large set of popular threading algorithms[7–14] in the literature. The performance of this type of algorithms has been well benchmarked in previous blind and meta-server experiments.[31,32,34,64] In a recent local meta-server development, we show that the performance of PPA is comparable to the best single threading algorithms used in LOMETS[30] which includes FUGUE,[15] HHSEARCH,[18] PROSPECT2,[23] SAM-T02,[65] SPARKS2,[13] SP3,[14] PAINT,[30] although the template libraries used in these method are not the same as PPA. We also obtained the newest version of the HHpred-1.5.0 from Soding,[18] which is one of the best-performed single server in CASP7 server section. The alignment results of PPA and HHpred are quite comparable with an average TM-score/RMSD/Coverage of 0.4941/7.34/88% and 0.4890/7.66/86%, respectively (J. Soding, private communication). These data may give an approximate and indirect comparison of the MUSTER algorithm with other methods.

The on-line MUSTER server is freely available to academic users at our website: http://zhang.bioinformatics.ku.edu/MUSTER.

REFERENCES

1. 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.
2. Wang G, Jin Y, Dunbrack RL, Jr. Assessment of fold recognition predictions in CASP6. Proteins 2005;61(Suppl 7):46–66.
3. 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.
4. 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.
5. Jones DT, Taylor WR, Thornton JM. A new approach to protein fold recognition. Nature 1992;358:86–89.
6. Godzik A, Kolinski A, Skolnick J. Topology fingerprint approach to the inverse protein folding problem. J Mol Biol 1992;227:227–238.
7. Jaroszewski L, Rychlewski L, Li Z, Li W, Godzik A. FFAS03: a server for profile–profile sequence alignments. Nucleic Acids Res 2005;33: W284–W288.
8. Rychlewski L, Jaroszewski L, Li W, Godzik A. Comparison of sequence profiles. Strategies for structural predictions using sequence information. Protein Sci 2000;9:232–241.
9. 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.
10. Pei J, Sadreyev R, Grishin NV. PCMA: fast and accurate multiple sequence alignment based on profile consistency. Bioinformatics 2003;19:427–428.
11. Marti-Renom MA, Madhusudhan MS, Sali A. Alignment of protein sequences by their profiles. Protein Sci 2004;13:1071–1087.
12. Ginalski K, Pas J, Wyrwicz LS, von Grotthuss M, Bujnicki JM, Rychlewski L. ORFeus: detection of distant homology using sequence profiles and predicted secondary structure. Nucleic Acids Res 2003;31:3804–3807.
13. 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.
14. 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.
15. Shi J, Blundell TL, Mizuguchi K. FUGUE: sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. J Mol Biol 2001;310: 243–257.
16. Elofsson A, Fischer D, Rice DW, Le Grand SM, Eisenberg D. A study of combined structure/sequence profiles. Fold Des 1996;1: 451–461.
17. Kelley LA, MacCallum RM, Sternberg MJ. Enhanced genome annotation using structural profiles in the program 3D-PSSM. J Mol Biol 2000;299:499–520.
18. Soding J. Protein homology detection by HMM-HMM comparison. Bioinformatics 2005;21:951–960.
19. Karplus K, Barrett C, Hughey R. Hidden Markov models for detecting remote protein homologies. Bioinformatics 1998;14:846–856.
20. Eddy SR. Profile hidden Markov models. Bioinformatics 1998;14: 755–763.
21. Jones DT. GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. J Mol Biol 1999;287: 797–815.
22. Cheng J, Baldi P. A machine learning information retrieval approach to protein fold recognition. Bioinformatics 2006;22:1456–1463.
23. Xu Y, Xu D. Protein threading using PROSPECT: design and evaluation. Proteins 2000;40:343–354.
24. Xu J, Li M, Kim D, Xu Y. RAPTOR: optimal protein threading by linear programming. J Bioinform Comput Biol 2003;1:95–117.
25. Skolnick J, Kihara D. Defrosting the frozen approximation: PROSPECTOR—a new approach to threading. Proteins 2001;42:319–331.
26. Skolnick J, Kihara D, Zhang Y. Development and large scale benchmark testing of the PROSPECTOR 3.0 threading algorithm. Protein 2004;56:502–518.
27. Wu S, Skolnick J, Zhang Y. Ab initio modeling of small proteins by iterative TASSER simulations. BMC Biol 2007;5:17.
28. Zhang Y. Template-based modeling and free modeling by I-TASSER in CASP7. Proteins 2007;69(Suppl 8):108–117.

29. Zhang Y, Skolnick J. Scoring function for automated assessment of protein structure template quality. Proteins 2004;57:702–710.
30. Wu S, Zhang Y. LOMETS: a local meta-threading-server for protein structure prediction. Nucleic Acids Res 2007;35:3375–3382.
31. Rychlewski L, Fischer D. LiveBench-8: the large-scale, continuous assessment of automated protein structure prediction. Protein Sci 2005;14:240–245.
32. Fischer D, Rychlewski L, Dunbrack RL, Jr, Ortiz AR, Elofsson A. CAFASP3: the third critical assessment of fully automated structure prediction methods. Proteins 2003;53 (Suppl 6):503–516.
33. Moult J, Fidelis K, Rost B, Hubbard T, Tramontano A. Critical assessment of methods of protein structure prediction (CASP)–round 6. Proteins 2005;61 (Suppl 7):3–7.
34. Battey J, Kopp J, Bordoli L, Read R, Clarke N, Schwede T. Automated server predictions in CASP7. Proteins 2007;69(Suppl 8):68–82.
35. Teodorescu O, Galor T, Pillardy J, Elber R. Enriching the sequence substitution matrix by structural information. Proteins 2004;54:41–48.
36. Silva PJ. Assessing the reliability of sequence similarities detected through hydrophobic cluster analysis. Proteins, in press.
37. Gaboriaud C, Bissery V, Benchetrit T, Mornon JP. Hydrophobic cluster analysis: an efficient new way to compare and analyse amino acid sequences. FEBS Lett 1987;224:149–155.
38. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 1970;48:443–453.
39. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1981;147:195–197.
40. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25:3389–3402.
41. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 1999;292:195–202.
42. Chen H, Zhou HX. Prediction of solvent accessibility and sites of deleterious mutations from protein sequence. Nucleic Acids Res 2005;33:3193–3199.
43. Zhou H, Skolnick J. Ab initio protein structure prediction using chunk-TASSER. Biophys J 2007;93:1510–1518.
44. Henikoff S, Henikoff JG. Position-based sequence weights. J Mol Biol 1994;243:574–578.
45. Frishman D, Argos P. Knowledge-based protein secondary structure assignment. Proteins 1995;23:566–579.
46. Wang G, Dunbrack RL, Jr. PISCES: a protein sequence culling server. Bioinformatics (Oxford England) 2003;19:1589–1591.
47. Chakravarty S, Varadarajan R. Residue depth: a novel parameter for the analysis of protein structure and stability. Structure 1999;7:723–732.
48. Ahmad S, Gromiha MM, Sarai A. Real value prediction of solvent accessibility from amino acid sequence. Proteins 2003;50:629–635.
49. Zhang Y, Skolnick J. Automated structure prediction of weakly homologous proteins on a genomic scale. Proc Natl Acad Sci USA 2004;101:7594–7599.
50. Zhang Y, Skolnick J. Tertiary structure predictions on a comprehensive benchmark of medium to large size proteins. Biophys J 2004;87:2647–2655.
51. Hopp TP, Woods KR. Prediction of protein antigenic determinants from amino acid sequences. Proc Natl Acad Sci USA 1981;78:3824–3828.
52. Kyte J, Doolittle RF. A simple method for displaying the hydropathic character of a protein. J Mol Biol 1982;157:105–132.
53. Vapnik V. The nature of statistical learning theory. Berlin: Springer; 1995.
54. Chang C-C, Lin C-J.LIBSVM: a library for support vector machines. (http://www.csie.ntu.edu.tw/~cjlin/libsvm).
55. Kabsch W, Sander C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 1983;22:2577–2637.
56. Domingues FS, Lackner P, Andreeva A, Sippl MJ. Structure-based evaluation of sequence comparison and fold recognition alignment accuracy. J Mol Biol 2000;297:1003–1013.
57. Cristobal S, Zemla A, Fischer D, Rychlewski L, Elofsson A. A study of quality measures for protein threading models. BMC Bioinformatics 2001;2:5.
58. Xu Y, Xu D, Crawford OH, Einstein, Larimer F, Uberbacher E, Unseren MA, Zhang G. Protein threading by PROSPECT: a prediction experiment in CASP3. Protein Eng 1999;12:899–907.
59. Murzin AG, Brenner SE, Hubbard T, Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247:536–540.
60. Zhang Y, Skolnick J. TM-align: a protein structure alignment algorithm based on the TM-score. Nucleic Acids Res 2005;33:2302–2309.
61. Hobohm U, Sander C. Enlarged representative set of protein structures. Protein Sci 1994;3:522–524.
62. Delano WL. The PyMOL molecular graphics system (http://pymol.sourceforge.net/).
63. Zhang Y, Kolinski A, Skolnick J. TOUCHSTONE II: a new approach to ab initio protein structure prediction. Biophys J 2003;85:1145–1164.
64. Ginalski K, Elofsson A, Fischer D, Rychlewski L. 3D-Jury: a simple approach to improve protein structure predictions. Bioinformatics 2003;19:1015–1018.
65. Karplus K, Karchin R, Draper J, Casper J, Mandel-Gutfreund Y, Diekhans M, Hughey R. Combining local-structure, fold-recognition, and new fold methods for protein structure prediction. Proteins 2003;53(Suppl 6):491–496.