PROTEINS: Structure, Function, and Genetics 47:496–505 (2002)

Protein Sequence Threading: Averaging Over Structures

Anthony J. Russell and Andrew E. Torda*
Research School of Chemistry, Australian National University, Canberra, Australia

ABSTRACT **Multiple sequence alignments are a routine tool in protein fold recognition, but multiple structure alignments are computationally less cooperative. This work describes a method for protein sequence threading and sequence-to-structure alignments that uses multiple aligned structures, the aim being to improve models from protein threading calculations. Sequences are aligned into a field due to corresponding sites in homologous proteins. On the basis of a test set of more than 570 protein pairs, the procedure does improve alignment quality, although no more than averaging over sequences. For the force field tested, the benefit of structure averaging is smaller than that of adding sequence similarity terms or a contribution from secondary structure predictions. Although there is a significant improvement in the quality of sequence-to-structure alignments, this does not directly translate to an immediate improvement in fold recognition capability. Proteins 2002;47:496–505.**

Key words: force fields; scoring function; profile; multiple sequence alignment; protein fold recognition; structure averaging

INTRODUCTION

There may be a finite number of protein folds, but even if this is wrong it is certainly a useful rule. Often, when a new protein structure is solved it appears similar to one already in the protein data bank.[1,2] This means that, even in the absence of sequence homology, it is worth trying to find the most appropriate known structure for some sequence of interest.[3,4] This philosophy has led to protein threading becoming one of the most popular methods for protein structure prediction.[5] Implementations vary, but generally one needs a score or energy function, a method for aligning the sequence to a trial structure, and some library of representative protein structures.

In practice, the library is unlikely to contain an ideal structure for the sequence. Instead, it will have a selection of proteins that have been declared representative of their fold types. This, however, may not be ideal because we do not care about the details of any particular protein. What one really wants is a set of average structures with average properties, typical of each family of proteins. While this is simple to wish for, it is not easy to implement. The fundamental problem is that protein structures are no longer protein-like after almost any kind of averaging. Simply averaging Cartesian coordinates results in structures with nonphysical bond lengths or unusual angles. Averaging over internal angles or even in reciprocal space results in similar problems. In this work, we show how one can apply averaging over structural information so as to improve sequence-to-structure alignments. Although the method is based on structure comparison and alignment, the averaging is performed over the fields experienced by test particles. The results are given for our score or energy function, but the method is applicable to almost any score function based on pair-wise interaction functions. The testing is concerned with the quality of sequence-to-structure alignments, but this is important because better alignments mean better models and improved fold recognition.

Unlike protein structures, sequences have been routinely combined, merged, or averaged. Properties from families have been one of the driving forces for rapid multiple sequence alignments.[6–8] In the field of protein fold recognition, one of the most popular and successful methods is purely sequence based. Starting from reliable sequence homologs, a profile (with averaged properties) is gradually built up using ever more averaged sequence information.[9–11] Although couched in a different formalism, hidden Markov models are also based on the premise that a profile can be constructed that somehow averages over related sequences.[12,13] Aside from pure sequence-based approaches, multiple sequence profiles have been combined with knowledge-based potential energy schemes with an apparent improvement in detection of remote homologs.[14] Averaging over sequence properties is now so accepted that it is an integral part of applications such as secondary structure prediction.[15–20]

In contrast, there have been far fewer attempts to use multiple structures in fold recognition, even though one would expect them to be just as beneficial as multiple sequences.[21] One approach has been to ignore most of the sequence-to-structure alignment problem and simply consider ungapped alignments. This is technically easy and even seems to offer some improvement of fold recognition.[22,23] The approach, however, is of no use if one is interested in improving sequence-to-structure alignments.

To improve sequence-to-structure alignments with gaps and insertions, one should consider how the alignments are calculated. Often, this is done with an adaptation of the alignment methods commonly used in sequence alignment[24,25] and these require that each sequence residue be

*Correspondence to: Andrew Torda, Research School of Chemistry, Australian National University, Canberra ACT 0200, Australia.
E-mail: Andrew.Torda@anu.edu.au

Received 23 July 2001; Accepted 29 November 2001

given a score at each position in the template structure. It is these scores that can be readily averaged across corresponding sites in aligned protein structures. This could be seen as averaging the field experienced by a test particle in a protein structure, although the test particle would be a whole amino acid. The approach requires precalculated structural alignments, but these are tabulated in the literature.[26–32]

Because this work is concerned with protein sequence-to-structure alignments, it requires a measure of alignment quality. One could compare alignments with some structure-based superposition, but this would introduce new problems. Aside from issues of structure comparison methodology, a pair of proteins may not even have a clear, single optimal superposition. There may be a large number of different, near-optimal alignments.[33,34] The problem can be completely avoided with test data because one knows the correct answer for the sequence. Placing a sequence on a template yields a model for the sequence. This can simply be compared against the correct answer (structure) for the sequence. The better an alignment, the closer the model is to the correct answer. To compare methods, one does not look at a single pair of proteins. This work relies on a set of 572 proteins pairs chosen so as to have some structural similarity. The better a method, the better the performance, summed over the whole set. Clearly, this large set of alignments allows one to compare different methods such as with and without structure averaging. It also allows one to compare the relative effect of including extra terms. For example, it may be suspected that a particular score function will be improved by including information from secondary structure predictions or adding a term corresponding to sequence similarity between the sequence and template.

With this machinery, one can test values of parameters in the alignment calculation. For example, one could test values of gap penalties or the weight applied to some specific term. This can be taken a step further and the measure of quality can be used as the basis for some merit function in a numerical optimization procedure. The surface will not be smooth, so the work below used a mixture of grid search and simplex optimization. With this approach, various terms were tested and compared against structure averaging, but only after optimizing each term's contribution.

To test the process of averaging across templates, the sequence-to-structure alignments were calculated using a published, freely available code,[35] a previously described alignment methodology,[36] and a z-score optimized, pairwise score function.[37]

METHODS

Alignment Methods

Sequence-to-structure alignments were calculated using the Gotoh method[24] rather than the Needleman and Wunsch[25] method previously used.[36] Gap penalties were implemented with conventional costs for gap opening and widening rather than the more computationally expensive geometrically based penalties previously described.[36]

Protein Lists And Test Sets

The calculations involved the use of three lists of proteins or pairs of proteins, each based on structural alignments from the FSSP library.[27–30] First, alignment parameters were optimized on a set of 572 structurally similar protein pairs. Within each pair, there was 20% or less sequence identity (within structurally aligned regions) and at least 70% of the residues of the probe sequence were structurally aligned. The structural dissimilarity was bounded, requiring the root mean square (RMS) difference of coordinates within the aligned regions to be less than $0.6 \times \mathrm{RMS}_{\mathrm{cut}}$. This crudely accounted for the dependence of RMS difference on protein size and, following a published parameterization,[38] was taken as

$$rms_{\mathrm{cut}} = 4.54 + 2.36N_{\mathrm{res}}^{1/3} \quad (1)$$

where N_{res} is the number of residues in the alignment. Second, a structure library was constructed, based on structural relationships as defined by FSSP z-scores.[27–30] A simple hierarchical clustering algorithm was used and a level of dissimilarity (based on z-scores) chosen so as to give a final set of 1235 chains. Third, a test set for fold recognition was generated wherein each member was guaranteed to have a structural homolog within the library of 1235 chains. For each member of the fold library, the corresponding FSSP data was scanned and the first structural homolog chosen that possessed no more than 20% sequence identity over the structurally aligned regions, an FSSP z-score greater than or equal to 8, and at least 80% of the residues structurally aligned with the parent structure. This resulted in a test set of 181 probe sequences. All lists of proteins are available as supplementary material.[39]

Alignment scores were calculated using a neighbor-nonspecific score function/force field.[36] This has the important property that the score or quasi-energy experienced by a residue can be calculated using the coordinates of its neighbors without requiring the neighbors' identity to be known. This may be viewed as an averaging over residue types, but parameters for the interactions were optimized directly and not by any postfact averaging. For each protein template chain, the simple score profile was calculated by placing each of the 20 amino acid types at every position and storing each score.

For averaging structure, each of the 1235 members of the template library was treated as a parent structure. Related structures were used if the structural alignment covered 30% or more of the parent template's residues (with a minimum of 10) and possessed an RMS difference less than or equal to 4.5 Å over the aligned regions. These criteria resulted in the numbers of related structures varying widely over the set of 1235 parent structures. For example, 539 structures were used in the average structure profile for the immunoglobulin, 2cd0, but 107 structures had no structural homologs meeting the criteria. A full table with the number of homologs is given as supplementary material.[39] The median number of chains used for an average structure profile was 35 and the averaging

was performed in a straightforward fashion. For a given site, there are N_{hom} aligned chains and the average score (quasi-energy) $\overline{E_i(A)}$ for a residue of type A at aligned site i is simply

$$\overline{E_i(A)} = \frac{1}{N_{\text{hom}}} \sum_{j=1}^{N_{\text{hom}}} E_i^j(A), \tag{2}$$

where E_i^j(A) is indexed by both site i and protein homolog j. This was calculated using the neighbor-nonspecific force field,[36] referred to as E_{ffield} below. For calculations without averaging, $N_{\text{hom}} = 1$.

Score Functions

The main score function was constructed for native fold recognition on the basis of z-scores [37] and we refer to this as if it were a force field energy, E_{ffield}, and comes from summing over the N_{align} residues that are present in the final alignment:

$$E_{\text{ffield}} = \sum_{i=1}^{N_{\text{align}}} E_i. \tag{3}$$

Contributions (E_{ss}) were also included from secondary structure predictions, taken from the PredictProtein (PHD) server,[15] where one sums over the N_{align} aligned residues as previously described.[40]

$$E_{\text{ss}} = -\sum_{i=1}^{N_{\text{align}}} (s(p_c)(\cos(\psi_0 - \psi_i) + 1)), \tag{4}$$

where p_c is the prediction confidence and $s(p_c)$ is a switching function that returns zero unless the confidence p_c is greater than or equal to 8. In this case, the function returns 1. ψ_i is the conventional backbone dihedral angle at template site i and ψ_0 is a literature ideal value for the secondary structure taken as $\psi_0 = -47°$ for α-helices and $\psi_0 = 124°$ for β-strands.[41]

Sequence similarity contributions were implemented with a quasi-energy term, E_{seq}, defined as

$$E_{\text{seq}} = -\sum_{i=1}^{N_{\text{align}}} b_{a_i a_{i'}} \tag{5}$$

where a_i is the type of the amino acid at site i and the prime denotes the aligned residue, so $a_{i'}$ is the type of residue from the aligned template site. $b_{a_i a_{i'}}$ is the element from a BLOSUM62[42] substitution matrix, indexed by the types of amino acids a_i and $a_{i'}$.

Gaps in alignments were treated with different costs for opening and widening and based only upon the number of residues in the gap, but with different penalties used in the sequence and template. To calculate the penalties, one needs to know n_{s_gap} and n_{t_gap}, the number of gaps in the sequence and template, and l_i, the length of each gap i. Combining the main force field, gap penalties, secondary structure predictions, and sequence similarity terms yields

$$E_{\text{tot}} = E_{\text{ffield}} + k_{s\_\text{open}} n_{s\_\text{gap}} + k_{s\_\text{wdn}} \sum_{i=1}^{n_{s\_\text{gap}}} (l_i - 1) + k_{t\_\text{open}} n_{t\_\text{gap}} + k_{t\_\text{wdn}} \sum_{j=1}^{n_{t\_\text{gap}}} (l_j - 1) + k_{ss} E_{ss} + k_{\text{seq}} E_{\text{seq}}, \tag{6}$$

where each k term weights the contribution with respect to E_{ffield}. The subscripts s_open, s_wdn, t_open, and t_wdn refer to sequence gap opening and widening and template gap opening and widening, respectively.

Multiple Sequence Alignments

This work was intended to test the feasibility of averaging across different structures rather than sequences, but a crude multiple sequence term was implemented for comparison with structure averaging. Multiple sequence alignments based upon BLAST alignments were extracted from the PHD[15] server output. No attempt was made to include more remote homologs, which could be found with a more sensitive search method.[10] This conservative choice of sequence homologs also removed potential problems with the location of gaps in the multiple sequence alignment.

Multiple sequence information was used by counting the amino acid types present at each position of the multiple sequence alignment and weighting the energy calculation by the relative proportion of each amino acid type present. For a sequence position i, one keeps a count, g_i^A, of the number of times residue type A is found in the multiple sequence alignment. We then say

$$E_i^j(\bar{A}) = \frac{\sum_{A=1}^{N_{\text{type}}} g_i^A E_i^j(A)}{\sum_{A=1}^{N_{\text{type}}} g_i^A}, \tag{7}$$

where $\bar{A}$ is a residue of mixed type and the summation runs over the $N_{\text{type}} = 20$ amino acid types. As above, the superscript j refers to the particular structure or structural homolog, so the expression $E_i^j(\bar{A})$ can be substituted directly into eq. 2, the expression for the energy of a residue in a possibly averaged structure. In this formulation, the counting is done over residues actually present in the multiple sequence alignment, so no advantage is taken of suggested location of gaps.

Alignment Optimization and Testing

If a sequence is aligned to a template structure, it produces a model for the sequence. If the correct structure for the sequence is known, it can be compared against the predicted model. In these calculations, the aim is to find the k values in eq. 6 that produce the best models over the entire test set of 572 protein pairs. This requires a geometric measure of similarity between the coordinates of the sequence native structure, $\vec{r}_{\text{nat}}$, and the model from the alignment, $\vec{r}_{\text{model}}$. Specifically, we measure the fraction of the distance matrix that a model has in common with

the correct structure and call this $f(\vec{r}_{\rm nat}, \vec{r}_{\rm model})$. This could be seen as similar to the Q value often used in modeling protein folding, which counts the number of native contacts present in a conformation.[43,44]

First, one calculates the difference between C^α-based distance matrices,[45,46] sometimes referred to as the distance matrix error (DME)[47,48]:

$$\mathrm{DME}_{\rm nat,model} = \left(\frac{2}{N_{\rm res}(N_{\rm res}-1)} \sum_{i<j}^{Nc^\alpha} (r_{ij}^{\rm nat} - r_{ij}^{\rm model})^2 \right)^{1/2}, \quad (8)$$

where $r_{ij}^{\rm nat}$ is the distance between $C^\alpha{}_i$ and $C^\alpha{}_j$ in the native structure and $r_{\rm ij}^{\rm model}$ is the corresponding distance in the model. Next, one defines a threshold, $\mathrm{DME}^{\rm cut} = 4.0$ Å, bearing in mind the typical C^α—C^α distance is 3.8 Å. Then one can discard elements where the two distance matrices are most different, until $\mathrm{DME}_{\rm nat,model}$ is less than or equal to $\mathrm{DME}^{\rm cut}$. The remaining fraction of the distance matrix is $f(\vec{r}_{\rm nat}, \vec{r}_{\rm model})$. In pseudocode, one can describe the process:

while ($DME_{\rm nat,model} > DME^{\rm cut}$)

{remove largest distance difference from C^α distance matrix

recalculate $DME_{\rm nat,model} f(\vec{r}_{\rm nat}, \vec{r}_{\rm model})$

= fraction of distance difference matrix remaining}

For very similar structures, $f(\vec{r}_{\rm nat}, \vec{r}_{\rm model})$ has a value close to 1. For completely dissimilar structures, it could approach zero, but rarely lies below 0.5 among compact structures. In the final merit function, M_1, the results should not be unduly influenced by completely wrong structures and thus should be weighted toward near correct structures [where $f(\vec{r}_{\rm nat}, \vec{r}_{\rm model})$ is greater than approximately 0.7]. This was achieved by using a sigmoid-like switching function that is smooth, but removes the influence of wrong, essentially random aligned fragments. The final, sigmoidal merit function for alignment optimization was then

$$M_1 = \sum_{i=1}^{N_{\rm pair}} (1 + e^{b(a - f(\vec{r}_{\rm nat_i}, \vec{r}_{\rm model_i}))})^{-1}, \quad (9)$$

where the summation runs over all $N_{\rm pair} = 572$ protein pairs. a was set to 0.7 as described above and $b = 15$ (an arbitrary decision for the shape of the sigmoid).

In summary, gap and extra parameters were optimized to produce the best alignments by using a simplex method[49] to adjust the k values of eq. 6 so as to maximize the merit of eq. 9. As there is no guarantee of a cooperative search space (it may have many local minima), the k values (eq. 6) were minimized in turn and in combination from many starting points.

Fold Recognition Parameters

This work is concerned with sequence-to-structure alignments, but some calculations were carried out to confirm that this is useful for protein fold recognition. Because we use separate force fields for alignments and ranking of models,[36] gap penalties and relative weights of terms were reoptimized again using a simplex optimization. The merit function, M_1 from eq. 9, had been designed to recognize good models from alignment. A different merit function was used for ranking generated models.

In fold recognition, the aim is to find the best structural homolog for a sequence when that homolog is hidden within a library of decoy structures. This phrasing assumes one has a library without duplication or errors. A better way to state the goal is to say that after aligning a sequence to every member of a library one wishes the best ranked models to be those most similar to the correct answer for the sequence. This is the approach used here. For each of the 572 probe sequences, 1235 models were generated, stored, and the $f(\vec{r}_{\rm nat}, \vec{r}_{\rm model})$ value for each model then calculated. A switching function was used to determine if the model was acceptable or not:

$$\begin{aligned} u(f(\vec{r}_{\rm nat}, \vec{r}_{\rm model})) &= 1 \text{ if} \quad f(\vec{r}_{\rm nat}, \vec{r}_{\rm model}) > 0.6 \\ &= 0 \text{ if} \quad f(\vec{r}_{\rm nat}, \vec{r}_{\rm model}) \le 0.6. \end{aligned} \quad (10)$$

Then, the merit M_2 associated with a set of fold recognition calculations was

$$M_2 = -\sum_{j=1}^{N_{seq}} \sum_{i=1}^{N_{\rm lib}} [(f(\vec{r}_{{\rm nat}_{ij}}, \vec{r}_{{\rm model}_{ij}}))/\mathrm{rank}_{ij}], \quad (11)$$

where $N_{\rm lib} = 1235$ structures in the protein fold library and $N_{\rm seq} = 572$, the number of probe sequences. The parameter, rank_{ij}, refers to the rank (out of 1235) of model i for sequence j. The merit function M_2 was minimized using a simplex method and adjusting k values as given by eq. 6.

Fold Recognition Measurement

The function given above is relatively smooth and suitable for optimizing parameters, but maybe not the clearest way to present results. To compare fold recognition among the different methods, we counted and plotted the number of times a sequence was able to find a correct homolog within a library at first rank, second rank, and so on. The set of 181 sequences was aligned to the library of 1235 structures as described above. Success was measured by the number of times a structural homolog was found at first, second, and successive ranks. As is common in a test of this kind, no attempt was made to account for sequences that possessed more than one desirable homolog in the library and results were only used to compare the relative performance of different terms.

RESULTS

Comparing results with or without some kind of averaging assumes one has appropriate gap penalties for each case. Similarly, comparing results with and without a sequence similarity contribution assumes that the k coefficients in eq. 6 are set to a correct value. All of the k parameters were set to, at least, local minima by a mixture of grid search and simplex optimization. One cannot claim that any of the parameters are optimal, but it is possible to

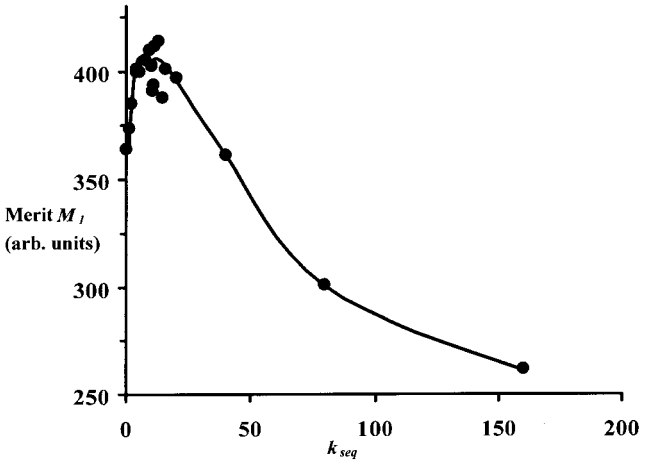

Fig. 1. Optimization of sequence similarity coefficient k_{seq}.

get an impression as to whether they are plausible. Figure 1 shows the final values for the merit function M_1 from a series of simplex optimizations intended to optimize k_{seq}, the weighting for the sequence similarity term. Some of the apparent noise near the maximum is due to other parameters being varied (the figure is a slice through a higher-dimension optimization), but some points are clear. The cost surface is not smooth and one cannot guarantee that this (or any of the parameters) is truly optimal. There is, however, a range where parameters are likely to be reasonable. This kind of behavior was found for all gap penalties and weighting coefficients examined.

Alignment Results

Given a set of reasonable parameter values, one can ask what are the best results obtainable with and without structure averaging. Figure 2 shows the best value of the merit function M_1 that was obtained with different contributions. The arbitrary units have the same scale as Figure 1. For comparison, some other results are included and discussed below.

The simplest comparison shows that averaging over structure scores offers a bit less than a 10% improvement over the bare force field [E_{ffield} from eq. (6)]. The exact number is probably not significant as it will depend on the test set and structures used for averaging. The improvement is partly due to small improvements in all models, but most significantly represents a large number of models moving from being completely wrong to substantially correct.

The merit function is a good indicator of the number of typical alignment quality, but it is not linear in the number of correct contacts or any conventional structural measure. One can, however, get some idea of the significance by comparing with other terms or improvements. The third line on the plot shows the improvement achieved by multiple sequence alignments, without structure averaging. Strictly, this would suggest that averaging over sequences is more profitable than averaging over structures. In practice, the exact size of each improvement will be dependent on the implementation. The conservative approach used for multiple sequences may have been fortuitously successful, despite no attempt being made to optimize the number or similarity of sequence homologs used.

Aside from averaging procedures, the results can be compared with the addition of extra terms to the score function. The bar in Figure 2 labeled "sec struct" shows the improvement when secondary structure predictions (E_{ss} from eq. 4) were added to the bare force fields (E_{ffield} from eq. 3) with no averaging. The improvement seems greater than that due to any averaging process. This could be a general phenomenon, but it could be a peculiarity of the score functions used here. Earlier work has suggested that these score functions are not good at recognizing correct local structure.[40] If one has a better score or energy function, one may not see such an improvement upon adding this data.

Finally, the results should be compared against the improvement seen by adding a sequence similarity matrix term, labeled "sim mat" in Figure 2. This appears to be the single most important term that can be used to improve the bare force field. The size of the improvement is surprising because the test set was selected so that no sequence had more than 20% sequence identity to its template. At the same time, the results are quite explicable. As k_{seq} is increased, the sequence similarity matrix term dominates and the method approaches pure sequence alignment. Because k_{seq} was the subject of Figure 1, we can see that at extreme values the performance is so poor that simple sequence comparison would not even be on the scale of Figure 2. It would seem that the errors due to the pair-wise, through-space terms are different and, to some extent, complementary to those due to sequence comparison. It is also clear that the careful optimization of each contribution is important to the final quality of alignments.

Given infinite patience, one could try out each possible combination of terms, but it is useful to know if results continue to improve when all the methods are combined. The second last bar of Figure 2 (labeled "single seq all") shows single sequences with multiple structures, similarity matrices, and secondary structure. The last bar, labeled "mult seq (all)," shows the same but with multiple sequence alignments. Clearly, there is useful, nonredundant information among the various terms. It also suggests that when all the factors are added together the difference between single and multiple sequence results is too small to be seen.

Fold Recognition

This work has shown that averaging over structures can improve sequence-to-structure alignments and the consequent models from threading calculations. It is, however, interesting to see if this translates to an improvement in fold recognition. This is fundamentally a much larger calculation. Rather than aligning each sequence to a single template, a sequence must be aligned to each member of a structure library (with and without averaging). For this

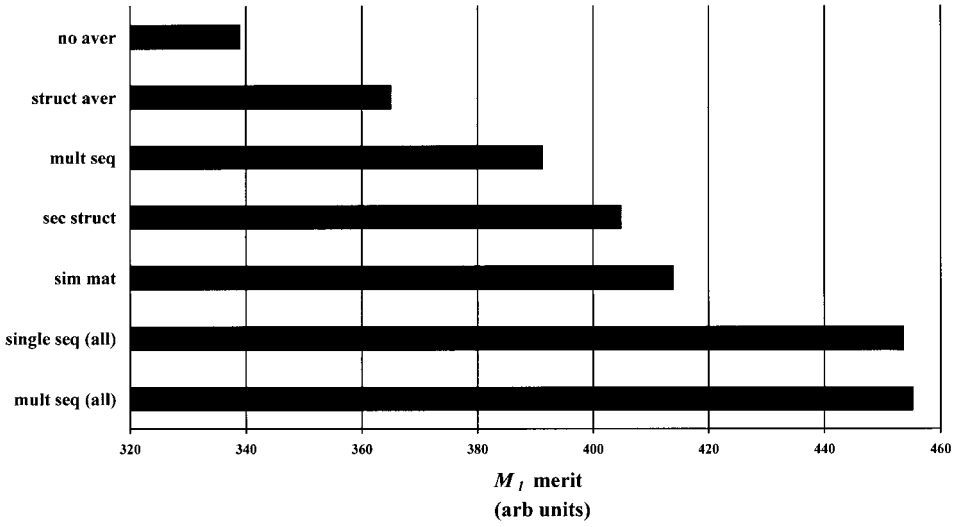

Fig. 2. Alignment performance with different contributions. For each method, the best value of the merit function M_1 from eq. 9 is shown. "No aver" refers to the bare score function, "struct aver" to the introduction of averaging across similar structures, "mult seq" to the addition of multiple sequence alignments, "sec struct" to the addition of secondary structure predictions, "sim mat" to the addition of sequence similarity information, "single seq (all)" to single sequence calculations with the terms from "struct aver," "sec struct," and "sim mat," and "mult seq (all)" is the same is "single seq (all)" but with multiple sequence alignments.

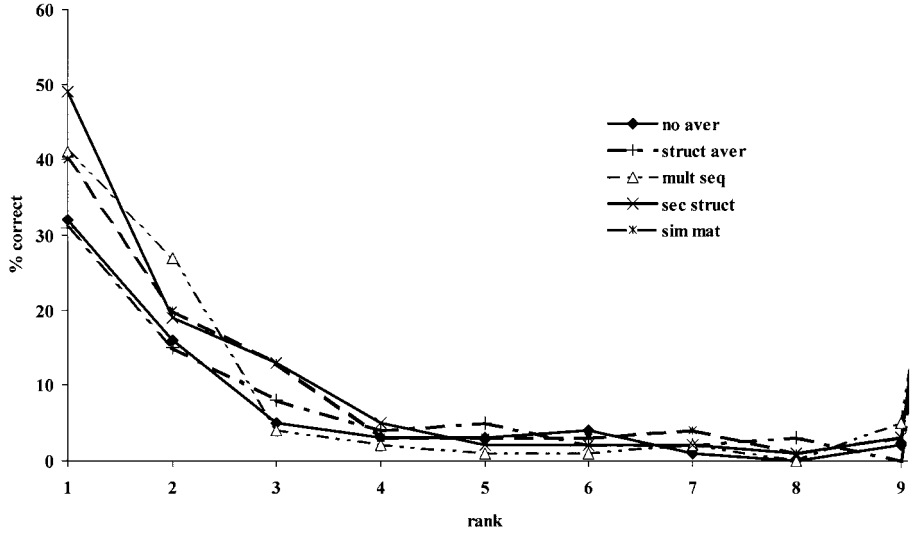

Fig. 3. Fold recognition performance with structure averaging and additional score terms. Labels are as for Figure 2.

calculation, each of 181 sequences were aligned to each member of a library of 1235 structures where the library was known to contain a structural homolog. The measure of success is how often a sequence finds its correct structure, hidden among 1234 decoys, at first rank, second rank, and so on.

Figure 3 shows this measure, with and without structure averaging, sequence averaging ,and the individual score function contributions in turn. Looking at first and second ranks, the worst performance may come from the bare, through-space, pair-wise force field, but in this data there is no detectable improvement from structure averaging. There are two likely reasons for this. First, the rescoring of models has its own noise or artefacts and this may obscure the improved scores due to the improved alignments. Second, the models for correct templates may be improved but the models for all templates, including completely inappropriate ones, are also improved, that is, the alignment procedure may be producing more reasonable guesses for sequences even from incorrect templates.

Again concentrating on first and second ranks, it does appear that there is an improvement in fold recognition from the use of multiple sequences, similarity matrices, and secondary structure predictions. In the cases of secondary structure and similarity matrices, this is readily explained. These contributions were present in the final

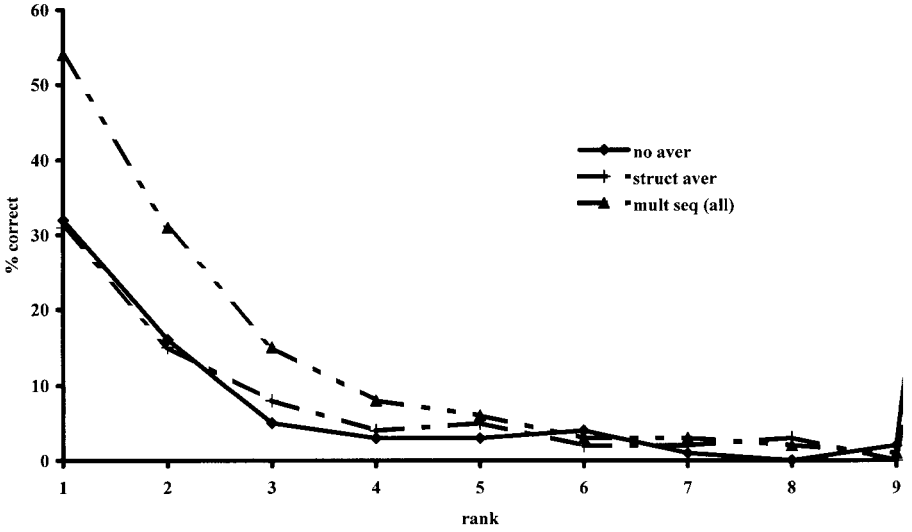

Fig. 4. Fold recognition comparison for bare force fields and with all available terms. Labels are as for Figure 2.

score and ranking calculations. In contrast, the averaging procedures were only used to produce better models.

In these fold recognition results, it seems as if any improvement due to an individual term is too small to be clearly seen. In this case, one can ask if one can at least see the difference between the bare, pair-wise, through-space term, the effect of structure averaging, and the result of all terms summed. To see this, one can take the bare force field line from Figure 3 and plot corresponding lines from fold recognition calculations, but now including all available terms simultaneously with structure averaging and with sequence and structure averaging. Figure 4 shows these results and some clear trends. Although the effect of structure averaging on final fold recognition is not significant, when one combines all terms the results are distinctly improved at the first five ranks.

DISCUSSION

The results show that for some score function or energy term one can improve the quality of protein threading models by an indirect averaging across related structures. This averaging over structural properties can be done in a way that permits gaps and insertions and, in practice, there is no loss of speed in the calculations. The implementation used here relies on precalculated scores for amino acid types on the template structures, so the alignments are as fast as any profile methods in the literature.[50,51]

In the case of the score functions used here, the improvement due to structure averaging was smaller than that from multiple sequence alignments (sequence averaging). This is probably not a universal result. Not every conceivable parameter was optimized in this work. There are orders of magnitude more information available for sequence alignments than for structure comparisons. Last, the thresholds for structure alignments are not nearly as well characterized for structure comparisons as for sequence comparisons.

The most important feature of this work may not be the exact size of any improvement, but rather the transferability of the principle to other workers' force fields. The force fields used here may be "knowledge-based," but they do not rely on Boltzmann statistics. Instead, they fall into the class of force fields built by optimizing parameters for some property such as z-scores or fold recognition.[36,37,47,52–57] In principle, however, this is not important. If one has a score function and a method to score a residue type at a position on a template, then one has the ingredients to average over different structures. There is no reason this method could not be used to improve the performance of more common Boltzmann-based force fields.

It is also possible to see why the method is likely to work by considering an example. If one takes an example protein, one can look at the score felt by a typical amino acid at each position on the template and the process can be repeated in the structure-averaged case. For example, Figure 5 shows a calculation for 2cd0. The choice of protein is arbitrary, but the results are typical. The top plots show the score experienced by a hydrophobic residue (Trp) along the template while the bottom shows the results for a hydrophilic residue (Glu). Not surprisingly, the top and bottom plots often show opposite trends. A favorable spike for the hydrophobic residue often appears as a disfavored site for the Glu on the plot below. This would be expected from most low-resolution force fields.[58] It is, however, more interesting to see the effect of averaging.

On the left are the plots from the bare, pair-wise score function while on the right are plots with structure averaging. The effect is dramatic. First, the vertical scale is the same on the left and right plots, so the averaging has removed many of the large spikes. Considering a specific example, one could look at the Glu propensity without averaging (lower left) and see a large spike at residue 24. This means that the scoring function is particularly affected by some arrangement of residues in that region of

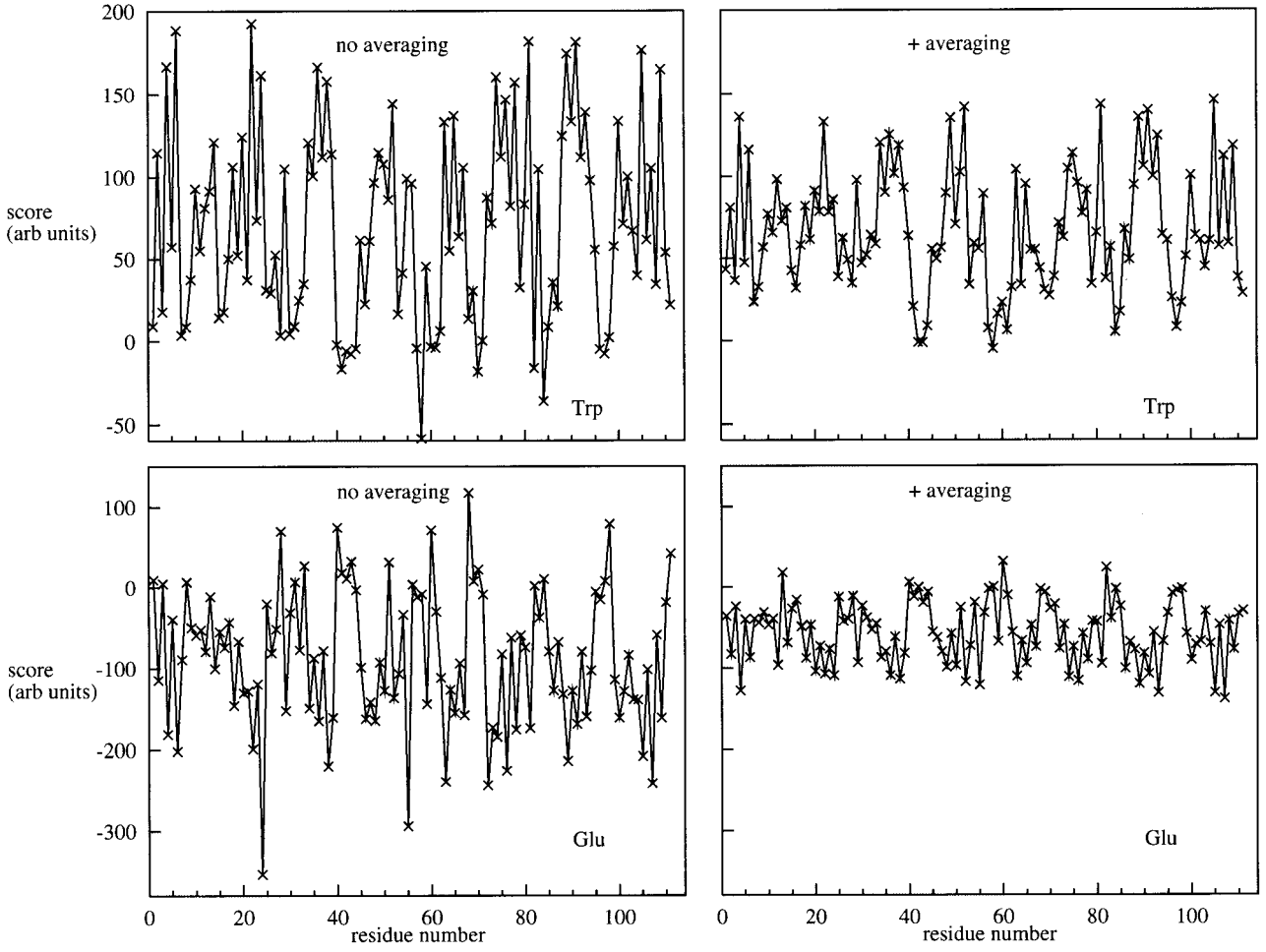

Fig. 5. Comparison of scores with and without structure averaging. For the example structure of 2cd0, the plots on the left show the score experienced by a test residue without structure averaging and the plots on the right with averaging. The top two plots use a Trp residue as the test particle and the bottom use Glu.

2cd0. Comparison with the right-hand side shows that the spike is removed, meaning the feature is really a property of 2cd0 and not one of the structure family as a whole.

The next question is whether the structure averaging was too conservative or too profligate. This was only tested in a cursory manner. The aim was simply to average over substantial pieces of structure that would provide a similar environment for a test residue. Most likely, there is no single correct amount of structure averaging. If one has a very close sequence/structure pair, averaging is not an issue. For example, aligning the sequence of 2cd0 to its own structure may well benefit from the particular spike in the score profile discussed above. When aligning more remote homologs, one wants to be less affected by the peculiarities of individual proteins. This means that the method would have to be attuned to levels of difficulty to achieve the best possible results.

The most surprising result is the lack of convincing improvement of fold recognition. This could be excused by claiming that the improvement in model quality from better alignments is not enough to be seen in the cruder fold recognition measurements. It may, however, be indicative of a weakness in the implementation here. The averaging was only performed at the sequence alignment stage.

Aside from the issue of model quality, the work has some implications for threading force fields. Other workers have noted that terms in a score or energy function should be adjusted with respect to each other and some have offered titrations of different terms such as secondary structure predictions or sequence similarity with respect to the core of the force field.[14,59] The work here showed that it is far more effective to use a full numerical optimization for the different terms, rather than simply trialing different values. Most importantly, it is necessary to treat parameters simultaneously rather than independently. For example, Figure 1 shows the variation of a merit function as one coefficient is varied. What is not shown is that other parameters such as gap penalties had to be simultaneously varied to achieve the best results.

Another clear result is that methodology in protein threading should not be judged merely on the ranking of a sequence's homologs. This will hide changes in model quality and, if alignments change across most of a struc-

ture library, it will be a reflection of many simultaneous changes.

The most disappointing aspect of the results here is the lack of improvement in fold recognition, despite improved models from alignment calculation. One could see this as a confirmation of the fact that the generation and ranking of models are different tasks with different goals. Taking this to extremes, one would be satisfied with an alignment method that produced reasonable models for appropriate templates and near random alignments for inappropriate models. One future task will be to assess the extent to which improved alignments on incorrect templates interfere with recognition of the correct fold and the extent to which small improvements in models are dwarfed by the noise within ranking calculations.

ACKNOWLEDGMENTS

The authors thank James Procter for the code used in simplex optimizations and Thomas Huber for the force fields.

REFERENCES

1. Abola EE, Bernstein FC, Bryant SH, Koetzle TF, Weng RM. Protein data bank. In: Allen FH, Bergerhoff G, Sievers R, editors. Crystallographic databases-information content, software systems, scientific applications. Bonn, Cambridge, Chester: Data Commission of the International Union of Crystallography; 1987. p. 107–132.
2. 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.
3. Jones DT, Taylor WR, Thornton JM. A new approach to protein fold recognition. Nature 1992;358:86–89.
4. Hendlich M, Lackner P, Weitckus S, Floeckner H, Froschauer R, Gottsbacher K, Casari G, Sippl MJ. Identification of native protein folds amongst a large number of incorrect models. The calculation of low energy conformations from potentials of mean force. J Mol Biol 1990;216:167–180.
5. Murzin AG. Structure classification-based assessment of CASP3 predictions for the fold recognition targets. Proteins 1999;S3:88–103.
6. Barton GJ. Protein multiple sequence alignment and flexible pattern matching. Method Enzymol 1990;183:403–428.
7. Taylor WR. Identification of protein sequence homology by consensus template alignment. J Mol Biol 1986;188:233–258.
8. Gotoh O. Multiple sequence alignment: algorithms and applications. Adv Biophys 1999;36:159–206.
9. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol 1990;215:403–410.
10. 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.
11. Schaffer AA, Wolf YI, Ponting CP, Koonin EV, Aravind L, Altschul SF. IMPALA: matching a protein sequence against a collection of PSI-BLAST-constructed position-specific score matrices. Bioinformatics 1999;15:1000–1011.
12. Krogh A, Brown M, Mian IS, Sjölander K, Haussler D. Hidden Markov models in computational biology: Applications to protein modeling. J Mol Biol 1994;235:1501–1531.
13. Eddy SR., Hidden Markov models. Curr Opin Struct Biol 1996;6: 361–365.
14. Panchenko AR, Marchler-Bauer A, Bryant SH. Combination of threading potentials and sequence profiles improves fold recognition. J Mol Biol 2000;296:1319–1331.
15. Rost B, Sander C, Schneider R. PHD — an automatic mail server for protein secondary structure prediction. Comput Appl Biosci 1994;10:53–60.
16. Rost B, Sander C. Combining evolutionary information and neural networks to predict protein secondary structure. Proteins 1994;19: 55–72.
17. Rost B, Sander C. Progress of 1D protein-structure prediction at last. Proteins 1995;23:295–300.
18. Rost B. PHD: Predicting one-dimensional protein structure by profile-based neural networks. Meth Enzymol 1996;266:525–539.
19. Cuff JA, Barton GJ. Application of multiple sequence alignment profiles to improve protein secondary structure prediction. Proteins 2000;40:502–511.
20. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 1999;292:195–202.
21. Finkelstein AV. 3D protein folds: homologs against errors — a simple estimate based on the random energy model. Phys Rev Lett 1998;80:4823–4825.
22. Reva BA, Skolnick J, Finkelstein AV. Averaging interaction energies over homologs improves protein fold recognition in gapless threading. Proteins 1999;35:353–359.
23. Rykunov D, Lobanov MY, Finkelstein AV. Search for the most stable folds of protein chains: III. Improvement in fold recognition by averaging over homologous sequences and 3D structures. Proteins 2000;40:494–501.
24. Gotoh O. An improved algorithm for matching biological sequences. J Mol Biol 1982;162:705–708.
25. 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.
26. Wang Y, Addess KJ, Geer L, Madej T, Marchler-Bauer A, Zimmerman D, Bryant SH. MMDB: 3D structure data in Entrez. Nucleic Acids Res 2000;28:243–245.
27. Holm L, Sander C. The FSSP database of structurally aligned protein fold families. Nucleic Acids Res 1994;22:3600–3609.
28. Holm L, Sander C. The FSSP database: fold classification based on structure–structure alignment of proteins. Nucleic Acids Res 1996;24:206–209.
29. Holm L, Sander C. Dali/FSSP classification of three-dimensional protein folds. Nucleic Acids Res 1997;25:231–234.
30. Holm L, Sander C. Touring protein fold space with dali/FSSP. Nucleic Acids Res 1998;26:316–319.
31. Shindyalov IN, Bourne PE. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Eng 1998;11:739–747.
32. Shindyalov IN, Bourne PE. A database and tools for 3-D protein structure comparison and alignment using the combinatorial extension (CE) algorithm. Nucleic Acids Res 2001;29:228–229.
33. Feng ZK, Sippl MJ. Optimum superimposition of protein structures: ambiguities and implications. Fold Des 1996;1:123–32.
34. Godzik A. The structural alignment between two proteins: is there a unique answer? Protein Sci 1996;5:1325–1338.
35. Huber T, Russell AJ, Ayers D, Torda AE. SAUSAGE: Protein threading with flexible force fields. Bioinformatics 1999;15:1064–1065.
36. Huber T, Torda AE. Protein sequence threading, the alignment problem and a two step strategy. J Comput Chem 1999;20:1455–1467.
37. Huber T, Torda AE. Protein fold recognition without boltzmann statistics or explicit physical basis. Protein Sci 1998;7:142–149.
38. Maiorov VN, Crippen GM. Significance of root-mean-square deviation in comparing three-dimensional structures of globular proteins. J Mol Biol 1994;235:625–634.
39. Torda AE. http://www.rsc.anu.edu.au/~torda/mult_strct/; 2001.
40. Ayers DJ, Gooley PR, Widmer-Cooper A, Torda AE. Enhanced protein fold recognition using secondary structure information from NMR. Protein Sci 1999;8:1127–1133.
41. IUPAC-IUB Commission on Biochemical Nomenclature. Abbreviations and symbols for the description of the conformation of polypeptide chains. Tentative rules. Biochemistry 1970;9:3471–3479.
42. Henikoff S, Henikoff JG. Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci USA 1992;89:10915–10919.
43. Onuchic JN, Luthey-Schulten Z, Wolynes PG. Theory of protein folding: The energy landscape perspective. Annu Rev Phys Chem 1997;48:545–600.
44. Goldstein RA, Luthey-Schulten ZA, Wolynes PG. Protein tertiary structure recognition using optimized Hamiltonians with local interactions. Proc Natl Acad Sci USA 1992;89:9029–9033.
45. Rooman MJ, Rodriguez J, Wodak SJ. Automatic definition of recurrent local structure motifs in proteins. J Mol Biol 1990;213: 327–336.
46. Levitt M. Molecular dynamics of native protein. II. Analysis and nature of motion. J Mol Biol 1983;168:621–657.

47. Crippen GM. Easily searched protein folding potentials. J Mol Biol 1996;260:467–475.
48. Havel TF. The sampling properties of some distance geometry algorithms applied to unconstrained polypeptide chains: a study of 1830 independently computed conformations. Biopolymers 1990; 29:1565–1585.
49. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. Numerical recipes in C. Cambridge, UK: Cambridge University Press; 1992.
50. Rice DW, Eisenberg D. A 3D–1D substitution matrix for protein fold recognition that includes predicted secondary structure of the sequence. J Mol Biol 1997;267:1026–1038.
51. Ouzounis C, Sander C, Scharf M, Schneider R. Prediction of protein structure by evaluation of sequence-structure fitness. Aligning sequences to contact profiles derived from three dimensional structures. J Mol Biol 1993;232:805–825.
52. Xia Y, Levitt M. Extracting knowledge-based energy functions from protein structures by error rate minimization: Comparison of methods using lattice model. J Chem Phys 2000;113:9318–9330.
53. Ulrich P, Scott W, van Gunsteren WF, Torda AE. Protein structure prediction force fields — parametrization with quasi-Newtonian dynamics. Proteins 1997;27:367–384.
54. Koretke KK, Luthey-Schulten Z, Wolynes PG. Self-consistently optimized statistical mechanical energy functions for sequence structure alignment. Protein Sci 1996;5:1043–1059.
55. Mirny LA, Shakhnovich EI. How to derive a protein folding potential — a new approach to an old problem. J Mol Biol 1996;264:1164–1179.
56. Hao MH, Scheraga HA. How optimization of potential functions affects protein folding. Proc Natl Acad Sci USA 1996;93:4984–4989.
57. Chiu TL, Goldstein RA. Optimizing energy potentials for success in protein tertiary structure prediction. Fold Des 1998;3:223–228.
58. Dosztányi Z, Torda AE. Amino acid substitution matrices based on force fields. Bioinformatics 2001;17:686–699.
59. Rost B, Schneider R, Sander C. Protein fold recognition by prediction-based threading. J Mol Biol 1997;270:471–480.