Protein Fold Recognition Score Functions: Unusual Construction Strategies

Daniel J. Ayers,[1] **Thomas Huber,**[2] **and Andrew E. Torda**[1]*
[1]*Research School of Chemistry, Australian National University, Canberra, Australia*
[2]*ANU Supercomputer Facility, Australian National University, Canberra, Australia*

ABSTRACT **We describe two ways of optimizing score functions for protein sequence to structure threading. The first method adjusts parameters to improve sequence to structure alignment. The second adjusts parameters so as to improve a score function's ability to rank alignments calculated in the first score function. Unlike those functions known as knowledge-based force fields, the resulting parameter sets do not rely on Boltzmann statistics, have no claim to representing free energies and are purely constructions for recognizing protein folds. The methods give a small improvement, but suggest that functions can be profitably optimized for very specific aspects of protein fold recognition. Proteins 1999;36:454–461.** © **1999 Wiley-Liss, Inc.**

Key words: force field; optimization; protein structure; structure prediction; knowledge-based protein structure prediction; fold recognition; threading

INTRODUCTION

For a molecular mechanics calculation, it is common to try to use a model that directly reflects nature. Such a model should work under a range of conditions and be transferable from system to system. If, however, one is only interested in a single specific property, it may not be necessary to chase the perfect force field and parameters. It may be easier to find some score function specialized for this property. This work is based on this idea and the goal of finding functions for protein fold recognition/threading even if the functions do not reflect real energies or other molecular mechanics properties.

This goal leads to distinct differences between special purpose scoring functions and more general force fields. In a simple scoring function, it is not necessary to closely mimic the laws of physics. Instead, interactions may be represented by some set of functions which are easy to parameterize. For example, smoothed contact functions are a gross simplification of nature, but may be easy to optimize for protein sequence to structure threading.[1]

Next, there will be differences in how one chooses the parameters that characterize the interactions. In an atomistic force field, properties such as bonds lengths, angles, and so on could just be taken from experiment or some higher level calculation.[2–5] For a score function for protein sequence-structure threading, there are several approaches one could take. One could assume that protein structures follow a Boltzmann distribution and calculate a potential of mean force from known protein structures.[6–8] Another school of thought is that one should simply try to discriminate good sequence-structure pairs from unlikely ones.[9] This is usually done by optimizing a score function's ability to distinguish ideal sequence-structure pairs from some set of incorrect (misfolded) sequence-structure pairs.[1,10–14]

This work continues along these lines where one first defines a measure of score function quality and then adjusts parameters so as to maximize the quality function using some set of training proteins. This idea is extended by splitting the task of protein sequence to structure threading into two sub-problems with a special parameter set for each. It is shown how one might optimize one score function for protein sequence to structure alignment and a totally separate function for ranking the calculated alignments.

The rationale for this is that different score functions work best in different problem domains[15] and it is clear that sequence to structure alignment is a different problem to ranking of aligned structures. In the first step (alignment), there is a sequence of interest which has to be aligned to each member of a library of 10^2 to 10^3 candidate or decoy structures. In the second step (ranking), one wants to rank the 10^2 to 10^3 generated structures according to which is closest to the (unknown) native structure. During alignment calculation, the set of decoys is very large since one should allow for a gap in either the sequence or template of any length and at any position. Formally, this means that the searching problem is NP-complete.[16] Physically, this means that the set of decoys is not limited to compact, protein-like structures since it implicitly includes the astronomical set of wrong alignments and, conceptually, structures with additional or missing residues. In the second phase, one has a small set of just 10^2 to 10^3 alignments which have to be ranked. The set of alternative/decoy structures is a set of (hopefully) optimal alignments on non-optimal templates.

Score functions for both phases were parameterized by building a penalty function that operated on score function parameters. Although the score functions were continuous, the penalty functions for optimizing them were not.

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

Received 19 January 1999; Accepted 3 May 1999

Instead, they were discrete measures based on alignment quality or ranking of correct folds. This meant that a crude Monte Carlo-like scheme was used for parameter adjustment with old parameter sets as the starting point. The parameterization process also differed from earlier work in that ideal data was not restricted to native protein structures. In addition, structure-structure alignments were taken from the literature.[17–20] Regardless of debates about structural alignments,[21,22] it would still be an achievement for a score function to produce sequence to structure alignments of the quality of structure to structure alignments.

MATERIALS AND METHODS

Protein Sets for Parameterization and Testing

For parameterization with native sequence-structure pairs, a list of 370 protein databank[23,24] (PDB) structures was taken from a published collection[25,26] as previously described such that each chain had more than 100 residues, all backbone heavy atoms were present and no two protein chains had more than 25% sequence identity.

For parameterization with sequences and near-native structural homologues, sequences were chosen from the representative set of protein structures[25,26] (October 1997 release) such that no two had more than 35% sequence identity. A set of sequence-structure pairs was built by taking structural homologues from the FSSP library[17–20] of structurally aligned proteins. For each sequence, structural homologues were used where the structural alignment showed at least 70% of the sequence's residues to be aligned to the structure, all heavy backbone atoms were present in the structure and where the sequence identity (sequence to homologue's sequence) was less than 40%. This resulted in 576 sequences aligned to 1,183 template structures.

Fold recognition was measured using a standard set of 88 protein pairs[27] where the members of each pair are structurally similar, but supposedly not sequence homologous to each other. One member of each pair has been declared a probe sequence and the structure of the other is hidden in a library of 725 decoys. Entries in the set which had been superseded in the July 1997 PDB release were replaced by newer versions. The set contains pairs with varying degrees of structural similarity and this could be quantified. Given the length of the probe sequence, L_1, the length of the template structure, L_2 and the number of aligned residues from a structural alignment L_{ali}, one can define a ratio R_{ali}

$$R_{ali} = \frac{2 \times L_{ali}}{L_1 + L_2}. \tag{1}$$

This measure is not ideal if the length L_1 is very different from L_2, but was used to allow comparison with literature.[27]

Alignment quality was tested using a subset of sequences that had at least one homologue with $R_{ali} \geq 0.7$. This was an arbitrary threshold, but defines a goal for the parameters. Higher values correspond to similar structures, which are less of a challenge for alignment. Lower values introduce too much dissimilarity and could be seen as a noise source in the parameterization. This reduced set of 24 sequences with their associated homologues is listed in Table I.

TABLE I. Sequences Used For Alignment Testing

Probe sequence[a]	Template structures[b]					
1bct	1bct	1fosF				
1cpcA	1ash	1babA	1cpcA	1cpcB	2hbg	3sdhA
1cpt	1cpt	1oxa	2hpdA			
1dvh	1cc5	1cyj	1dvh	1ycc	451c	
1eaf	1eaf	3cla				
1frpA	1frpA	1imbA				
1fxd	1fca	1fxd				
1gfc	1aboA	1gfc				
1hryA	1hma	1hryA				
1irk	1cdkA	1csn	1irk			
1lccA	1lccA	1r69				
1mdyA	1ifi	1mdyA	2dgcA			
1pba	1ctf	1pba	2bopA			
1plq	1plq	2polA				
1pls	1btn	1dynA	1pls			
1pyp	1ino	1pyp				
1sso	1humA	1sso	3il8			
1tie	1bfg	1hce	1tie	2i1b		
1xxaA	1urnA	1xxaA				
2cyp	1arv	2cyp				
2pna	1lkkA	2pna				
3ebx	1cds	3ebx				

[a]PDB acquisition code with chain identifier appended if necessary.
[b]Proteins used as templates on to which sequence was threaded during alignment testing.

Alignment and Fold Recognition Measures

Sequence-structure alignment success was measured by calculating the average shift error. This compares the calculated alignment with some ideal[17–20] and is simply the average number of sequence positions by which each residue is incorrect. Fold recognition success was measured following Rost et al[27] using the cumulative frequency of the first successful prediction, $Q(R)$

$$Q(R) = 100 \sum_{r=1}^{R} \frac{N_{corr}(r)}{N_{prot}} \tag{2}$$

where $N_{corr}(r)$ gives the number of correct first-rank folds at rank r and N_{prot} is the number of probe sequences in the test set. A $Q(10)$ of 50 means that, considering all probe sequences, there is a 50% chance of finding the first correct homologue in the top ten guesses. The measure is used for a comparison to literature, but it may underestimate success since does not show when more than one correct homologue is detected.

Alignment and Ranking Calculations

All sequence-structure alignments, score function calculations and rankings were carried out using the sausage package.[28] Sequence to structure alignments were carried out using a function referred to as ALIGN-I or ALIGN-II depending on whether it used parameters before or after Monte Carlo optimizing. Ranking of the calculated alignments was carried out using a function referred to as RANK-I or RANK-II, again referring to parameter sets before or after Monte Carlo optimizing. The ALIGN function is based on the unusual idea of a neighbor non-specific score function.[29] This means that from each interaction pair, the coordinates of both interaction sites are used, but the identity of only one member is used. This allows a score to be calculated for a sequence residue at a template position, without knowing the alignment of the rest of the sequence. Alignments can then be calculated using an adaptation of a dynamic programming algorithm commonly used in sequence-sequence comparisons.[30] This guarantees an optimal alignment with a distinctly non-optimal score function and is an alternative approach to the alignment problem which others have treated by the frozen approximation[31–33] and double-dynamic programming methods.[8] Ranking of calculated alignments was carried out using a more conventional neighbor-specific score function.

During both alignment and ranking calculations, insertions in sequence (gaps in template) were penalized using a conventional gap opening/widening scheme so that the penalty, E_{ins} was given by

$$E_{ins} = -\begin{cases} 0 & N_{ins} = 0 \\ k_{ins}E_{open} & N_{ins} = 1 \\ k_{ins}[E_{open} + E_{wdn}(N_{ins} - 1)] & 1 < N_{ins} < N_{max} \\ k_{ins}[E_{open} + E_{wdn}(N_{max} - 1)] & N_{ins} \geq N_{max} \end{cases} \tag{3}$$

where k_{ins} was a scaling constant, N_{ins} the number of inserted residues, E_{open} was set to 0.3 and E_{wdn} (gap extension cost) was set to 0.01. For gaps in the sequence, a more sophisticated, geometric gap penalty could be used. For some alignment to a template, one can calculate the distance between sites in the sequence. The gap penalty, E_{gap}, was calculated based on the distance

$$E_{gap} = -\begin{cases} 0 & d_{C_iN_j} \leq d_0 \\ k_{gap}(d^2_{C_iN_j} - d^2_0) & d_0 < d_{C_iN_j} \leq d_{max} \\ k_{gap}\,(d^2_{\max} - d^2_0) & d_{C_iN_j} > d_{max} \end{cases} \tag{4}$$

where $d_{C_iN_j}$ is the distance between the carbonyl carbon of residue, i and the amide nitrogen of the next residue, j. Distances d_0 and d_{max} were set to 1.37 Å and 10 Å respectively.

The score functions, parameter sets, and gap and insertion penalties are summarized in Table II. Ranking force fields were used with alignments generated by more than one alignment method and this is also listed in the table.

Functional Form of Scoring Functions

Functional forms for all scoring functions were based on hyperbolic tangent functions that have been described in detail previously. Five interaction sites were used for each amino acid, located at the backbone N, C$^\alpha$, C, and O and side-chain C$^\beta$ atoms. A C$^\beta$ interaction site was calculated for glycine residues assuming ideal geometry. There were 20 types of C$^\beta$ particles corresponding the different residue types, but only one type of each backbone atom, giving a total of 24 types of interaction site.

TABLE II. Parameters Used For Alignment Testing

Score function	Purpose	Optimization[a]	Alignment parameter set	Gap penalty[b]	Insertion penalty[c]
ALIGN-I	alignment	z-score		10,000	10,000
ALIGN-II	alignment	MC		5,000	10,000
RANK-I	ranking	z-score	ALIGN-I	500	1,000
		z-score	ALIGN-II	500	100
RANK-II	ranking	MC	ALIGN-I	1,000	500
		MC	ALIGN-II	1,000	500

[a] z-score: parameters optimised by z-score optimization; MC: Monte Carlo used to adjust parameters.

[b] The gap penalty in (score units Å^{-2}).

[c] The penalty for residue insertion in (score units residue^{-1}).

Neighbor Non-Specific/ALIGN Scoring Function

The total score in the neighbor non-specific scoring function with either ALIGN-I or ALIGN-II parameters for a sequence-structure alignment over N_{res} residues is given by

$$E^{non-spec}_{tot} = \sum_{i}^{5N_{res}} \sum_{j>i}^{5N_{res}} E^{non-spec}_{pair}(i, j) + \sum_{k}^{N_{res}} E_{sol}(k) + E_{gap} + E_{ins} \tag{5}$$

where the indices i and j run over all aligned residues and the sums are performed over all $5N_{res}$ interaction centers. E_{gap} and E_{ins} are as given by Eqs. (3) and (4). $E^{non-spec}_{pair}$ is a neighbor non-specific pair score term depending on the type t_i of residue i only. At a topological distance s_{ij} and a Cartesian distance d_{ij} it is given by a sigmoidal function

$$E^{non-spec}_{pair}(i, j) = p_{pair}(s_{ij}, t_i)(1 - \tanh(w_{pair}(d_{ij} - d^0_{ij}))) \tag{6}$$

where $p_{pair}(s_{ij}, t_i)$ is a parameter determining the interaction strength, $d_{ij}{}^0$ is a reference distance determining the step position and w_{pair} determines the slope of the interaction function. Only three classes of topological distances are considered. These are $j = i + 2$, $j = i + 3$ and $j > i + 3$. Interactions between adjacent residues $j = i + 1$ are only treated by the gap penalty term, Eq. (4).

The "solvation quasi-energy" or particle environment score E_{sol} is calculated by a similar function

$$E_{sol}(i) = p_{sol}(t_i)(1 - \tanh(w_{sol}(n(i) - n_0))) \tag{7}$$

where p_{sol} is a score function parameter, w_{sol} a parameter determining the function's slope, $n(i)$ is the number of residues within a shell of 5.8 Å C$^\alpha$-C$^\alpha$ distance, but separated by more than three residues in the sequence. The neighbor count parameter n^0 is set to 3.

Neighbor Specific/RANK Scoring Function

The neighbor specific score (used in ranking calculations) is calculated from the function

$$E^{spec}_{tot} = \sum_{i}^{5N_{res}} \sum_{j>i}^{5N_{res}} E^{spec}_{pair}(i, j) + \sum_{k}^{N_{res}} E_{sol}(k) + E_{gap} + E_{ins} \quad (8)$$

where E_{ins} and E_{gap} are as given by Eqs. (3) and (4) and E_{sol} is as given by Eq. (7). Unlike the ALIGN function, the interaction parameter $p_{pair}(s_{ij},t_i,t_j)$, and the pair score E^{spec}_{pair} both depend on both residue types t_i and t_j.

$$E^{spec}_{pair}(i, j) = p_{pair}(s_{ij}, t_i, t_j)(1 - \tanh(w_{pair}(d_{ij} - d^0_{ij}))) \quad (9)$$

Gap Penalties in Score Function Optimization

The ALIGN-II parameter set optimization (described below) included the influence of gaps and insertions. For the sake of speed, gaps in sequence and structure were penalized according to a conventional gap opening/widening scheme rather than the more elaborate gap scheme of Eqs. (3) or (4). This simpler penalty, was based on N_{ins}, the number of inserted residues

$$E^{crude}_{ins} = \begin{cases} 0 & for\ N_{ins} = 0 \\ k_{opn} + k_{wdn}(N_{ins} - 1) & for\ N_{ins} \geq 1 \end{cases} \quad (10)$$

where k_{opn} is the gap opening penalty and k_{wdn} the penalty for widening an existing gap.

During optimization of the RANK-II parameters, alignments did not have to be recalculated, so the geometric gap penalty of Eq. (4) could be used. d_{max} was set to 8 Å.

Parameter Optimization

ALIGN-I and RANK-I parameters were taken from previous work. Both had been built using a method based on z-score optimization where one maximizes the score difference between a correct sequence-structure pair and the average over incorrect (misfolded) sequence-structure pairs. The process operated on 370 sequences simultaneously with a total of more than 10^6 misfolded structures. In this section, we describe how more specialized score function goals were cast into target functions and optimized for alignments (ALIGN-II) and ranking of structures (RANK-II).

The purpose of the ALIGN-II parameters is to achieve the best possible sequence-structure alignments, so the optimization process was geared to this goal alone. Alignments (sequence-structure) were calculated at each step with a fast (but potentially non-optimal) local similarity algorithm.[34] These were compared to supposedly ideal structure-structure alignments from the FSSP library.[17–20] Alignment quality was quantified by a truncated root mean square distance difference (RMSD*) with had a threshold of (10 Å)^2 for grossly misaligned residues

$$RMSD^* = \sqrt{N^{-1} \sum_{\substack{\text{all FSSP} \\ \text{residues } i}}^{N} \begin{cases} (10\ \text{Å})^2 & \text{if residue } i \text{ is not aligned} \\ (10\ \text{Å})^2 & \text{if } (r^i_{ali} - r^i_{FSSP}) > (10\ \text{Å})^2 \\ (r^i_{ali} - r^i_{FSSP})^2 & \text{if } (r^i_{ali} - r^i_{FSSP}) < (10\ \text{Å})^2 \end{cases}} \quad (11)$$

where $(r^i_{ali} - r^i_{FSSP})$ is a measure of how far a residue is misplaced in the alignment compared to the FSSP library[17–20] and the summation runs over all residues which have been declared structurally similar in the reference alignment. Unlike the more common root mean square difference value for structural comparisons, this measure does not involve structural superpositioning. The target function used to optimize the ALIGN-II score function was the arithmetic average of the RMSD* measure of all 576 protein pairs in the training set.

$$t1 = N^{-1} \sum_{i=1}^{\substack{N=576 \\ pairs}} RMSD^* \left(\{r^i_{ali}\} \{r^i_{FSSP}\}\right) \quad (12)$$

Initial parameters were taken from the ALIGN-I score function except for gap opening and widening which were arbitrarily set. 3,400 steps of Monte Carlo at $T = 0$ with a step size of 5 (parameter units) were performed to minimize target function $t1$ with respect to 362 score function parameters. Over the course of this minimization, the value of the target function $t1$ decreased from 6.69 Å to 6.06 Å. All the calculations were performed in parallel on 12 processors of a Fujitsu AP3000 and took less than 24 hours total time.

The optimization of the fold ranking function parameters (RANK-II) was quite different since its sole purpose is to rank precalculated alignments. The function will not be confronted with all possible protein conformations, but merely a set of alignments generated by the previous alignment function. This generally excludes alignments which are unusually short or with many gaps. In order to optimize the ranking capability of the RANK-II parameters, alignments of all 576 protein sequences to all 1,183 structures in the template library were calculated, resulting in a total of 681,408 sequence to structure alignments.

Faced with optimizing rankings, one might use a measure such as Kendal's tau.[35] This is not suitable in this work as the main goal is to ensure that near-native alignments are well ranked. One is not really interested in the behavior of poor ranking alignments. This was enforced with a simple scheme wherein each native-like alignment was given a penalty according to its rank. Near-native alignments, were selected by first calculating the root mean square (RMS) difference over all aligned residues after optimal translation and rotation and accepted if the RMS difference was less than an empirical, size-dependent threshold R_{max},

$$R_{max} = 0.2(N_{ali} - 25)^{1/3} \tag{13}$$

where N_{ali} is the number of aligned residues. Up to 20 near-native alignments were permitted per sequence, but typically less than five were present.

Given this list, a ranking G was calculated at each step of parameterization by

$$G = \sum_{native\text{-}like\ j}^{N} \log_{10}(rank(j)). \tag{14}$$

This uses a logarithmic function to account for the fact that rank difference is important for well-ranked structures. That is, the difference between 1st and 5th rank is important, but between 501st and 505th is not. Although only well-ranked alignments will significantly contribute to G, the scores of the alignment for the sequence to every member of the structure library have to be calculated.

The final target function, which was optimized, is the sum of G over all proteins in the training set

$$t2 = \sum_{i=1}^{\substack{N=576\\ proteins}} G(i) \tag{15}$$

$t2$ was initially optimized with respect to five parameters (one for geometrical gap penalties, two (opening and widening) for insertions into the structure and two more to penalize unaligned sequence and structure sites. This was followed by 7,500 Monte Carlo steps at temperature $T = 0$ in which the target function $t2$ was minimized with respect to all 920 + 5 scoring function parameters. The total optimization, including generation of lists of near-native structures, was performed in parallel on a Fujitsu AP3000 and took approximately 24 hours on 12 processors.

RESULTS

The aim of this work is to test the feasibility of optimizing separate functions for sequence to structure alignments and ranking of calculated alignments. The first question is whether the score functions can be improved given a good starting point. Figure 1 shows the value of the target function $t1$, Eq. (12) over the course of the optimization which was used to move from the ALIGN-I to ALIGN-II parameter sets. One might expect this to be a rugged energy surface since each step involves readjustment of the entire set of alignments. Despite this, the plot clearly shows that the ALIGN-II parameter set is driven to some minimum on the energy surface.

Figure 2 shows the progress of the optimization of the ranking parameter set beginning with RANK-I, moving to RANK-II. In this case, each function evaluation requires calculating the score of all alignments followed by reranking. The target function does improve although not as smoothly as in the previous plot. It also appears that some small improvement may have been achieved by even more optimization steps. For this optimization, one can look at a quantity other than the target function used to measure quality. At each step, the percentage of correct structural homologues at each rank was stored and is displayed in Figure 3. For example, less than 25% of correct homologs are found in the first five places with the initial parameters. This rises to more than 30% by the end of the calculation.

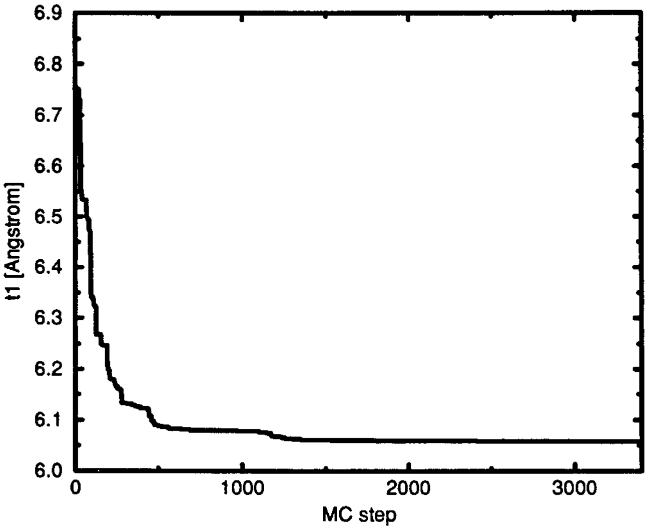

Fig. 1. Optimization of ALIGN-II parameter set. Function $t1$, Eq. (12), measures the quality of the score function at each step in Å.

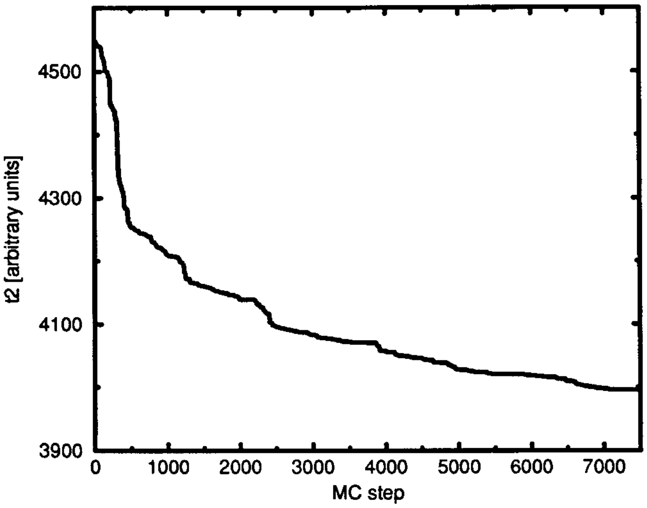

Fig. 2. Optimization of RANK-II parameter set. Function $t2$, Eq. (15), measures the quality of the score function at each step in arbitrary units.

With apparently improved parameters on training data, the next step was to measure the score functions' performance on test protein sets.[27] Firstly, the alignment quality was measured since this underpins the subsequent fold

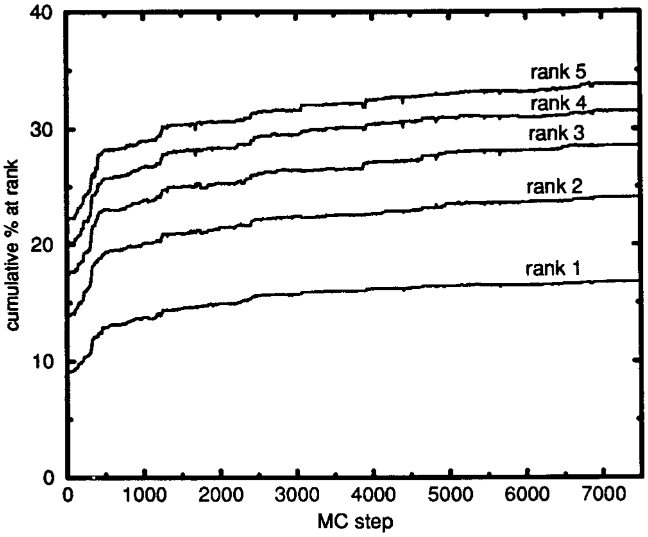

Fig. 3. Progress of RANK-II parameter set optimization in top five ranks. Each line shows the average cumulative percentage correct native like structures at that rank.

TABLE III. Average Shift Errors For Alignment Score Functions

Alignment score function[a]	Shift error[b]
ALIGN-I	5.1
ALIGN-II	4.6

[a] Acronyms as in Table II.

[b] Average residue shift error with respect to structural alignment.

recognition/ranking step. To test this directly, Table III gives the average shift error for the initial (ALIGN-I) and optimized (ALIGN-II) parameter sets. Not surprisingly, the ALIGN-II set performs better. The optimization procedure was primitive, but was designed so that the parameter set would improve for its specific task.

The next measure was to compare the parameter sets (RANK-I and RANK-II) and the overall fold recognition capability. This was done using the entire set of sequences and homologs and the $Q(R)$ measure (Eq. (2)). The first test of ranking ability was done using alignments calculated with the new alignment parameters. Figure 4 shows that the new RANK-II parameter set performs better (except at one rank) although the difference is small. Next a different property of the ranking functions was tested. Earlier work has shown a weakness of the score functions is the inability to tolerate structural errors. A structural perturbation that seems small to a human may give a huge change in score or energy. For this reason, the ranking parameter sets were compared, but using the poorer alignments generated by ALIGN-I and the results are shown in Figure 5. With these weaker alignments, the performance of the RANK-II function is clearly superior and suggests that the newer RANK-II score function is more tolerant of alignment shift errors. This desirable property is not surprising since a tolerance of structural error was built into the parameterization data. There is further evidence for the improved tolerance of structural errors. From the complete test set, one can extract the more similar sequence structure pairs based on R_{ali}, Eq. (1). If one considers the more similar pairs with, for example, $R_{ali} \geq 0.7$, then the advantage of the RANK-II parameters disappears (data not shown).

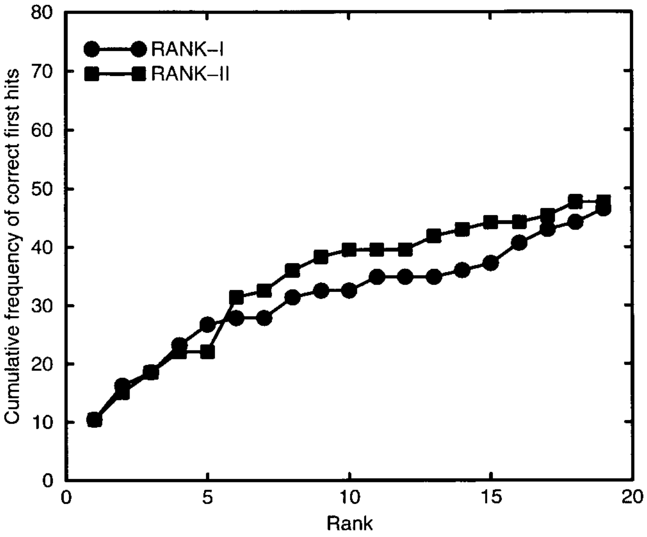

Fig. 4. $Q(R)$ of RANK-II and RANK-I ranking score functions for alignments calculated using ALIGN-II.

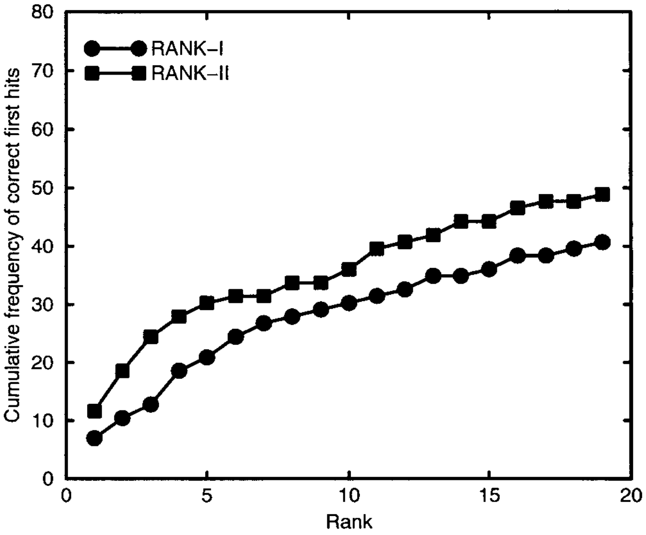

Fig. 5. $Q(R)$ of RANK-II and RANK-I ranking score functions for alignments calculated using ALIGN-I.

DISCUSSION

One problem encountered in the construction of knowledge-based force fields is whether all parameters are adequately defined by the experimental data. Some amino acid pairs are relatively rare and the associated parameters ill-determined. This has prompted some debate as to whether a sparse data correction is[7,36] or is not[37] necessary. This work attempted to avoid this question by increasing the parameterization data and treating precalculated structural alignments as ideal data. Obviously, there are cases where the structural alignments are ambiguous[21,22] but these exceptions are not an issue given the more serious lack of reliability of alignment methods. The use of slightly imperfect data in the parameterization has a more interesting implication. We know that a practical weakness of some score functions is that they are low-resolution representations, but can be too sensitive to structural details. For example, a Glu to Asp mutation may be very common in nature and should not be penalized in a score function. The use of natural near-native structures in the parameterization tends to build this lack of specificity into the score functions and may be partly responsible for the tolerance of weaker alignments demonstrated by Figure 5. The alignments in that calculation are not optimal (shown by Table III), but performance with the RANK-II force field, based on near-native structures is clearly better than the RANK-I parameter set based on only native structures.

One aspect that is not dealt with properly is the degree of tolerance which is acceptable or desirable, and clearly the decisions are quite arbitrary. One set of alignments were used with associated thresholds and decisions on structural similarity.[17–20] A different set of aligned structures with different thresholds[38] would produce different parameters. This could be used to tune functions for special purposes. For example, one can probably improve performance on very weak homologs by using corresponding parameterization data.

The parameterization methods used here only demonstrate the feasibility of the approach, rather than produce final score functions, but this is enough to make several points. There are properties which are desirable in fold recognition methods and which should be built into a parameterization strategy. Unfortunately, such optimization methods will always be difficult since one is working with functions that depend on a large number of native and perhaps misfolded protein structures. The approach here has been to use a very expensive target function such as Eq. (12) or (15), but to make the method tractable by using good starting parameters and assuming that only a few optimization steps are necessary. Even given a good starting point, there is no evidence that the global optimum in parameter space has been found and this has suggested two future directions. Firstly, it will be useful to postpone the search for the globally optimum parameters and instead use a better local minimizer to find locally optimum parameters. Next, it is still interesting to explore parameter space more thoroughly. With the present results, one other observation can be made. The ranking score function parameters (RANK-II) converge more slowly than the alignment parameters (ALIGN-II). Figure 2 shows $t2$ still improving after more than 7,000 steps whereas $t1$ stops improving in less than 2,000 steps (Fig. 1). The most likely explanation is very simple and does not involve speculation about the shapes of the energy (cost) surfaces. The alignment score function used in $t1$ is neighbor non-specific, using the identity of only one member of each interaction pair. Consequently it has less than half the number of adjustable parameters of $t2$. The search space is correspondingly smaller and faster to search.

Another question is whether the functions optimized are ideal. Function $t1$ (Eq. (12)) is based on the degree to which alignments correspond to the best structure that could be made given an appropriate template. Its' use does involve arbitrary decisions for treating unaligned residues and functional forms. More seriously, the implementation relies on alignments from a potentially non-optimal similarity algorithm[34] rather than the full dynamic programming algorithm we have generally used for alignments.[30]

The function $t2$, (Eq. (15)) used to optimize ranking has a more unusual property. Optimization should move correct alignments to a better rank within some library, but this depends on the members of the library. This may not give parameters that are physically meaningful, but results in ones that are tuned to the whole fold recognition machinery. It has been noted that different score functions perform best for different sets of decoys.[15] The optimization scheme used here explicitly generates parameters for some set of decoys.

The results hint at some problems with either optimization or testing. The plots of score function optimization (Figs. 1 and 2) show an apparently significant performance improvement, but the fold recognition tests (Figs. 4 and 5) show much smaller differences. There are two likely reasons. Firstly, there are some limits of the framework of the current score functions with the low-resolution pairwise interactions, few topological classes, and neglect of sequence similarity scores. There are probably more subtle problems to do with measures of success. For example, sequence shift error is a popular measure, but it is not sensitive to differences in local structure. It is also bounded by the size of the proteins and would be more meaningful in the context of errors corresponding to random alignments.

The issue of different functions for different tasks will be pursued further. If one accepts that different functions will be best for calculating alignments and ranking them, it is quite possible that different functional forms will be best for each task. Certainly, there is no reason to believe that the same interaction sites should be used in both cases. For example, the alignment score function should favor the creation of protein-like structures with reasonable hydrogen bond networks. This suggests that interaction sites at the backbone hydrogen bonding atoms are important. In the next step, most of the candidate alignments will be compact, regular, and protein-like. To discriminate amongst these, the contribution from hydrogen bonding sites will be less useful.

Considering all the questions raised by score-function optimization, it would seem that there is not yet a clear ideal method, but one should be able to improve on the simple idea of discriminating native from non-native structures.[1,10–14] It is also clear that there will be trade-offs among speed, accuracy, specificity, and tolerance of structural changes. The formulations used here illustrate one set of trade-offs. Most importantly, the optimization machinery is robust, able to deal with large parameterization data sets and seems well suited to the development of new forms of score function.

ACKNOWLEDGMENTS

We thank Fujitsu Corporation (Japan) for support of Thomas Huber and generous computational assistance.

REFERENCES

1. Huber T, Torda AE. Proteins fold recognition without Boltzman statistics or explicit physical basis. Protein Sci.1998;7:142–149.
2. Halgren TA. Merck molecular force field. 1. Basis, form, scope, parameterization, and performance or MMFF94. J Comput Chem 1996;17:490–519.
3. MacKerell AD, Bashford D, Bellott M, et al. All-atom empirical potential for molecular modeling and dynamics studies of proteins. J Phys Chem B 1998;102:3586–3616.
4. Cornell WD, Cieplak P, Bayly CI, et al. A second generation force field for the simulation of proteins, nucleic acids, and organic molecules. J Am Chem Soc 1995;117:5179–5197.
5. van Gunsteren WF, Billeter SR, Eising AA, et al. Biomolecular simulation: the GROMOS96 manual and user guide. Zurich and Groningen: vdf Hochschulverlag AG an der ETH Zurich and BIOMOS b.v.; 1996.
6. Hendlich M, Lackner P, Weitckus S, et al. 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.
7. Sippl MJ. Calculation of conformational ensembles from potentials of mean force. An approach to the knowledge-based prediction of local structures in globular proteins. J Mol Biol 1990;213: 859–883.
8. Jones DT, Taylor WR, Thornton JM. A new approach to protein fold recognition. Nature 1992;358:86–89.
9. Maiorov VN, Crippen GM. Contact potential that recognizes the correct folding of globular proteins. J Mol Biol 1992;227:876–888.
10. Hao MH, Scheraga HA. How optimization of potential functions affects protein folding. Proc Natl Acad Sci USA 1996;93:4984–4989.
11. Koretke KK, Luthey-Schulten Z, Wolynes PG. Self-consistently optimized statistical mechanical energy functions for sequence structure alignment. Protein Sci 1996;5:1043–1059.
12. 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.
13. Ulrich P, Scott W, van Gunsteren WF, Torda AE. Protein structure prediction force fields—parametrization with quasi-Newtonian dynamics. Proteins 1997;27:367–384.
14. Koretke KK, Luthey-Schulten Z, Wolynes PG. Self-consistently optimized energy functions for protein structure prediction by molecular dynamics. Proc Natl Acad Sci USA 1998;95:2932–2937.
15. Park BH, Huang ES, Levitt M. Factors affecting the ability of energy functions to discriminate correct from incorrect folds. J Mol Biol 1997;266:831–846.
16. Lathrop RH. The protein threading problem with sequence amino acid interaction preferences is NP-complete. Protein Eng 1994;7: 1059–1068.
17. Holm L, Sander C. Touring protein fold space with Dali/FSSP. Nucleic Acids Res 1998;26:316–319.
18. Holm L, Sander C. Dali/FSSP classification of three-dimensional protein folds. Nucleic Acids Res 1997;25:231–234.
19. Holm L, Sander C. The FSSP database—fold classification based on structure-structure alignment of proteins. Nucleic Acids Res 1996;24:206–209.
20. Holm L, Sander C. The FSSP database of structurally aligned protein fold families. Nucleic Acids Res 1994;22:3600–3609.
21. Godzik A. The structural alignment between two proteins: is there a unique answer? Protein Sci 1996;5:1325–1338.
22. Feng ZK, Sippl MJ. Optimum superimposition of protein structures: ambiguities and implications. Fold Des 1996;1:123–132.
23. Abola EE, Bernstein FC, Bryant SH, Koetzle TF, Weng J. Protein data bank. In: Allen FH, Bergerhoff G, Sievers R. Crystallographic databases-information content, software systems, scientific applications. Bonn, Cambridge, Chester: Data Commission of the International Union of Crystallography; 1987. p 107–132.
24. Abola EE, Sussman JL, Prilusky J, Manning NO. Protein data bank archives of three-dimensional macromolecular structures. Methods Enzymol 1997;277:556–571.
25. Hobohm U, Scharf M, Schneider R, Sander C. Selection of a representative set of structures from the Brookhaven protein data bank. Protein Sci 1992;1:409–417.
26. Hobohm U, Sander C. Enlarged representative set of protein structures. Protein Sci 1994;3:522–524.
27. Rost B, Schneider R, Sander C. Protein fold recognition by prediction-based threading. J Mol Biol 1997;270:471–480.
28. Huber T, Ayers D, Torda AE, Russell A. Sausage: sequence-structure alignment using a statistical approach guided by experiment: http://www.rsc.anu.edu.au/∼torda/sausage.html and ftp://ftp.rsc.anu.edu/pub/torda/sausage/README
29. Huber T, Torda AE. Protein sequence threading, the alignment problem and a two step strategy. J Comput Chem 1999; in press.
30. 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.
31. Godzik A, Kolinski A, Skolnick J. Topology fingerprint approach to the inverse protein folding problem. J. Mol. Biol. 1992;227:227–238.
32. Sippl MJ. Boltzmann's principle, knowledge-based mean fields and protein folding. An approach to the computational determination of protein structures. J Comput Aided Mol Des 1993;7:473–501.
33. Wilmanns M, Eisenberg D. Inverse protein folding by the residue pair preference profile method: estimating the correctness of alignments of structurally compatible sequences. Protein Eng 1995;8:627–639.
34. Huang X, Miller W. A time-efficient, linear-space local similarity algorithm. Adv Appl Math 1991;12:337–357.
35. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. In: Numerical recipes in Fortran 77: the art of scientific computing. New York: Press Syndicate of the University of Cambridge; 1997. p 637–639.
36. Kocher JPA, Rooman MJ, Wodak SJ. Factors influencing the ability of knowledge-based potentials to identify native sequence-structure matches. J Mol Biol 1994;235:1598–1613.
37. Samudrala R, Moult J. An all-atom distance-dependent conditional probability discriminatory function for protein structure prediction. J Mol Biol 1998;275:895–916.
38. Gibrat J-F, Madej T, Bryant SH. Surprising similarities in structure comparison. Curr Opin Struct Biol 1996;6:377–385.