Specialist Review

Threading algorithms

Jadwiga Bienkowska
Serono Reproductive Biology Institute, Rockland, MA, USA
Boston University, Boston, MA, USA

Rick Lathrop
University of California, Irvine, CA, USA

1. Background

The goal of protein structure prediction by threading is to align a protein sequence correctly to a structural model. This requires choosing both the correct structural model from a library of models and the correct alignment from the space of possible sequence-structure alignments. Once chosen, the alignment establishes a correspondence between amino acids in the sequence and spatial positions in the model. Assigning each aligned amino acid to its corresponding spatial position places the sequence into the three-dimensional (3D) protein fold represented by the model. Typically, the model represents only the spatially conserved positions of the fold, often the protein core, so producing a full-atom protein model would require further steps of loop placement and side-chain packing. Protein threading has a role in protein structure prediction that is intermediate between homology modeling (*see* Article 70, **Modeling by homology**, Volume 7) and ab initio prediction (*see* Article 66, ***Ab initio* structure prediction**, Volume 7). Like homology modeling, it uses known protein structures as templates for sequences of unknown structure. Like ab initio prediction, it seeks to optimize a potential function (an objective or score function) measuring goodness of fit of the sequence in a particular spatial configuration. Threading is the protein structure prediction method of choice when (1) the sequence has little or no primary sequence similarity to any sequence with a known structure and (2) some model from the structure library represents the true fold of the sequence.

Protein threading requires (1) a representation of the sequence, (2) a library of structural models, (3) an objective function that scores sequence-structure alignments, (4) a method of aligning the sequence to the model, and (5) a method of selecting a model from the library. Following the initial conception of the threading approach to protein structure prediction (Bowie *et al*., 1991; Jones *et al*., 1992),

there have been very many different approaches to these problems, of which this chapter can present only a few general themes.

2. Representation of the query sequence

It is widely accepted that significantly similar protein sequences also adopt a similar 3D structure. The Paracelsus Challenge demonstrated the design of a protein sequence with 50% sequence identity to a known protein but a different 3D structure (Jones *et al*., 1996), but when natural evolution produces similar protein sequences their protein structures generally are similar as well. Thus, in naturally occurring proteins, sequences that are similar to the query sequence carry useful information about its 3D structure. A multiple-sequence alignment centered on the query sequence reflects sequence variability within the protein family to which the query sequence belongs. Most modern threading algorithms exploit this fact (Jones, 1999; Fischer, 2000; Kelley *et al*., 2000; Panchenko *et al*., 2000; Rychlewski *et al*., 2000; Karplus and Hu, 2001; Skolnick *et al*., 2003).

The query sequence is often represented by a sequence profile, **P**, where the element $P_j = P(A|j)$ is a vector giving a probability distribution over the 20 amino acids at sequence position j. In this notation, a single query sequence has a profile with 1 for the original amino acids and 0 otherwise. The sequence profile is typically constructed from the search of nonredundant databases of proteins (e.g., at NCBI) and sequences are aligned using multiple-sequence alignment programs such as CLUSTAL (Higgins *et al*., 1996) or PSI-BLAST (Altschul *et al*., 1997). Some threading methods also include an independent prediction of the secondary structure (SS) (*see* Article 76, **Secondary structure prediction**, Volume 7) or other derived information as part of the sequence representation. In such cases, the query is represented as two independent vectors $P_j = \{P(A|j), P(SS|j)\}$, where SS might be helix, strand, or coil, a more detailed set of secondary structure assignments, or other information.

3. Representation of protein structure models

What is a model of protein structure? Protein structure is fully determined by the 3D coordinates of all non-hydrogen atoms. For threading, the 3D coordinates are reduced to more abstract representations of protein structure. Typically, structural core elements are defined by the secondary structure elements, α-helices and β-strands, usually with side chains removed. Among proteins with similar structures, large variations occur in the loop regions connecting the structural elements. In consequence, loop lengths, loop conformations, and loop residue interactions are rarely conserved, and often the loop residues are not represented explicitly in the structural models.

The main distinction among threading approaches is the choice of the structure model representation. Threading algorithms fall into two main categories that depend on the protein structure representation they use:

1. In the first category, a protein structure is represented as a linear model.
2. In the second category, a protein structure is represented as a higher-order model.

In a linear representation, the protein structure is modeled as a chain of residue positions that do not interact. In a second-order representation, the model also includes interacting pairs of residue positions, for example, to account for hydrophobic packing, salt bridges, or hydrogen bonding. Still higher order models have been considered to represent triples and higher multiples of interacting residue positions, but are less common.

Approaches that represent protein structure as a linear model consider each structural position in the model independently, neglecting spatial interactions between amino acids in the sequence. This allows very fast alignment algorithms, but loses whatever structural information may be present in amino acid interactions. Approaches that use higher-order models explicitly consider spatial interactions between amino acids that are distant in the sequence but brought into close proximity in the model. This potentially allows for more realistic and informative structural models, but results in an NP-complete alignment problem (Lathrop, 1994). It is known that the information content in higher-order amino acid interactions is modest, but nonzero (Cline *et al*., 2002). What effect this has in practice, and whether the increased information content compensates for the increased complexity, is a subject of some debate within the protein threading community.

3.1. 1D models of the protein structure

A 1D model of a protein structure is a sequence of states representing the residue as if embedded in a 3D structural environment. There are two distinct types of features frequently used to characterize a state, structural features, and amino acid sequence features. The structural features include the solvent exposure of a given residue, the secondary structure of the residue, and so on. The structural features may be representations of a single specific structure or (weighted) averages of structural features from multiple structures in the same family (*see* Article 75, **Protein structure comparison**, Volume 7). The sequence features may include the original amino acids observed in the structure or a sequence profile representing the multiple alignment of sequences from the protein family of the structure's native sequence.

If we denote by s a residue position in the structure (or a position from the alignment of multiple structures), then a vector of features $\mathbf{F}(s)$ describes each position. Thus, a structure model is an ordered chain of feature vectors $\{\mathbf{F}(s)\}$. The dimensionality of the feature vector depends on the specific threading approach.

The original 1D threading papers represented the feature vector as solvent exposure states, where the solvent exposure was calculated from the exposure of amino acids present in the native structure. Since then it has been recognized that, due to variations in the amino acid's size, one must use a measure of exposure that is independent of the native amino acid size. Most recent threading methods use the polyalanine representation of a structure. Solvent exposure state is determined by the solvent exposure of an alanine placed at each residue position. Some approaches vary the radii of the solvent molecule and the β-carbon.

3.2. 2D models of the protein structure

Two-dimensional models attempt to capture the contribution of interactions between pairs of residues. They begin with a 1D representation of a protein structure, and

then overlay representations of pairs of residues that are neighbors in the folded structure. In many threading methods, the pairs are represented as a contact map, where the contact can be defined by any of several methods:

Dependent on the native amino acid side-chain orientation:

1. Residues are in physical contact in the native structure, for example, if the distance between any of their atoms is smaller than a given cutoff, say 5 Å.
2. The distance between the centroids or C_β atoms of the residue side chains is below a certain cutoff.
3. The neighbors are determined by additional geometric constraints imposed by the 3D structure, for example, the C_β atoms may have to be in line-of-sight of each other. This excludes from the neighbor set pairs that can never interact, like residues on the opposite sides of an α-helix.

Independent of the native amino acid side-chain orientation:

1. Any pair separated by a given number of residues, for example, neighbors every 1, 3, or 4 residues in an α-helix, or every 2 residues in a β-sheet.
2. Any pair that has C_α closer than a cutoff value, say 7–10 Å.

Similar to the 1D representation, a pair of residue positions s and r is represented by a feature vector $\mathbf{FF}(s,r)$. The pair associated features fall into three categories:

1. 3D distance–derived features; distance between the β-carbons, distances among all other backbone atoms, distances between the centroids of side-chain positions, and so on.
2. 1D residue separation along the amino acid sequence of the native protein.
3. Structural environments of each residue in the pair like solvent exposure or secondary structure.

Definitions of the various environmental variables differ dramatically among threading approaches. The most commonly used feature for the 2D environments is the 3D distance. Typically, the distance between two atoms is partitioned into bins that are defined by a lower and upper distance threshold. In general, a similar approach can be applied to any feature that is associated with a real or integer variable. Most feature variables require binning, such as distance, solvent exposure, and 1D sequence separation.

3.3. Higher-order structural models

Third-order and higher models attempt to capture regularities of protein structure that cannot be represented by considering amino acid pairs only. For example, adjacent pairs of cysteines may form disulfide bonds, but only one disulphide bond can form among three adjacent cysteines; Godzik *et al*. (1992) used amino acid triples to represent this and related properties. The hydrophobic contact potential of Huang *et al*. (1996) is equivalent to amino acid triples, in this case used to represent the hydrophobic core. A fourth-order representation is the Delaunay tessellation, based on the vertices of irregular tetrahedral lattice (Singh *et al*., 1996; Munson and Singh, 1997; Zheng *et al*., 1997). Higher-order models suffer from the statistician's

"curse of dimensionality"; an Nth-order model must represent 20^N N-tuples. It can be difficult to parameterize the model and the objective function (below) unless reduced amino acid alphabets are used.

4. Objective function (potential or score function)

Most threading approaches do not use the physical full-atom free energy functions commonly used by macromolecular modeling software (*see* Article 74, **Molecular simulations in structure prediction**, Volume 7). Instead, most threading objective functions are determined empirically by statistical analysis of the 3D data deposited in the Protein Data Bank (PDB) (*see* Article 71, **The Protein Data Bank (PDB) and the Worldwide PDB http://www.wwpdb.org**, Volume 7). Thus, they are often referred to as *empirical potentials* or *knowledge-based potentials*. In the case of nonlinear structural models, another common name is *contact potentials*, reflecting their origin in analysis of contacts between atoms or residues in crystal structures. Many approaches augment empirical potentials with other terms thought to be important, for example, contributions from loop regions if the structural model contains only the protein core.

A great many different approaches have been explored. As examples, the hydrophobic contact potential of Huang *et al*. (1996) reflects packing in the hydrophobic core using only two residue classes, hydrophobic and polar, and is remarkable for its explanatory power given its simplicity and near absence of adjustable parameters. Maiorov and Crippen (1994) used linear programming to enforce a constraint that the native threading scores lower than others, but such approaches tend to be brittle. Bryant and Lawrence (1993) used logistic regression, based on multidimensional statistics. Boltzmann statistics is the foundation of many threading methods (Sippl, 1995). White *et al*. (1994) derived a formal probability model based on Markov Random Fields. Many other approaches have been investigated.

The most popular approach involves a negative log odds ratio between the observed and expected amino acid frequencies in a given structural environment. This yields a measure that is analogous to a physical free energy, and gives good results. Given 1D or 2D structural features F or FF defined by a specific structural library, the objective function is determined by counting amino acids with specific features in known 3D structures. The score for observing amino acid A with feature F is determined by $P(A,F)$, the probability of observing the amino acid A with the feature F in the protein structure database. Different methods apply different normalizations to the probability $P(A,F)$. The general motivation is to remove variations that do not contribute to specific sequence-structure recognition, for example, to control for the fact that some amino acids are more common than others. The score (*see* Article 67, **Score functions for structure prediction**, Volume 7) for amino acid A when found in feature F is

$$S(A, F) = -\log \frac{P(A, F)}{N(A, F)} \tag{1}$$

Here $N(A,F)$ is a normalization constant for A and F, typically derived from some assumed reference state or from assumptions of conditional independence.

Various choices of $N(A,F)$ have been explored, one of the simplest being $N(A, F) = P(A)P(F)$.

The same logic applies to pairs of residues, or pairs of atoms associated with residue positions. Some methods use only the amino acids while others use their respective backbone atoms and/or generalized side-chain atoms to define the 2D features of a pair of positions. For two amino acids A and B, where the feature FF is associated with the pair of positions, the score is given by:

$$S(A, B, FF) = -\log \frac{P(A, B, FF)}{N(A, B, FF)} \tag{2}$$

Owing to high redundancy of the PDB, typically a database of nonredundant or representative structures is used for calculation of probabilities. An objective function is often implemented by a number of 20×1 and 20×20 matrices associated with each 1D and 2D structural feature from the feature sets $\{F\}$ and $\{FF\}$ respectively.

Once the objective function is defined and its values estimated from the current database, it is fast and straightforward to calculate the score of a given sequence-structure alignment. The alignment is a placement of amino acids from a sequence $\mathbf{A} = \{A_1, \ldots, A_i, \ldots, A_L\}$ into positions $\mathbf{s} = \{q, \ldots, r, \ldots, s\}$ from the structure model, where the model is a collection of positions and pairs of positions as discussed above. A threading (alignment) is a (possibly partial) map t of sequence indexes i to model indexes, $t(i) \epsilon \mathbf{s}$. Most threading algorithms impose an ordering constraint of mapping increasing sequence indexes to increasing model indexes: if $i < j$, then $t(i) < t(j)$. In principle, relaxing such a constraint would allow threading to recognize/predict a structural topology that is not yet present in a database of known structures, but this is not usual in practice. The score S of the sequence-structure alignment $t(\mathbf{A}) = \{A_1^{t(1)}, ..., A_i^{t(i)}, ..., A_L^{t(L)}\}$ is given by:

$$\begin{aligned} S(t(\mathbf{A})) = & \sum_{F \in \{F\}} w_F \sum_i S\left(A_i^{t(i)}, F(t(i))\right) \\ & + \sum_{FF \in \{FF\}} w_{FF} \sum_{\{i,j\}, i<j} S\left(A_i^{t(i)}, A_j^{t(j)}, FF(t(i), t(j))\right) \end{aligned} \tag{3}$$

where the sum over different feature categories $\{F\}$, $\{FF\}$ often includes different weights. $F(t(i))$ is the feature of the model position $t(i)$ and $FF(t(i),t(j))$ is the feature associated with a pair of model positions $t(i)$ and $t(j)$. The weights w_{F}, w_{FF}, among different feature terms are subjected to optimization in many threading approaches. One of the 1D features often included in an objective function is a gap opening and extension penalty. The gap opening and gap extension penalties are special features that can depend on the length k of the gap $w(k|F(t(i)))$ and can also depend on the features of the structural model. Many threading approaches do not allow gaps inside the core structural elements, for example, by setting the gap opening penalty to $+\infty$ for positions in the core structural elements. The gap opening and extension penalty terms imply that there are even more weights to optimize.

4.1. Methods of refining the pairwise objective function

The pairwise interaction between residues depends on both the geometric features of positions close in the 3D structure, and on the specific amino acids that are aligned to those positions. There are two methods that attempt to capture these complicated dependencies.

In the Filtered Neighbors Threading approach, the objective function is constructed specifically for each structural model (Bienkowska *et al*., 1999). Each pair of positions has its unique objective function calculated by taking a Hadamard product of two 20 × 20 matrices $S(\cdot,\cdot,FF(s,r))$ and $V(\cdot,\cdot,s,r)$. The $S(\cdot,\cdot,FF(s,r))$ matrix is the usual pairwise objective function as defined by the statistics of amino acid pairs with a given structural feature $FF(s,r)$. The $V(\cdot,\cdot,s,r)$ is a binary matrix of 0's and 1's specific to the pair of structural positions from the template. A "0" for amino acids A and B indicates that physical contact between these amino acids is not plausible when placed at respective positions in the template structure and "1" indicates that a physical contact is plausible. The plausibility of a physical contact is calculated independently of the objective function S. A number of geometrical descriptors of neighboring structural positions describe neighbor pairs. The algorithm first analyzes a set of all 3D neighbor pairs from the database of proteins and partitions the space of geometrical descriptors into regions occupied by most of the observed physical contacts and the rest of the space. Thus, any pair of neighboring positions is defined as a plausible physical contact or not depending where it is in the space of geometrical descriptors.

The approach implemented by PROSPECTOR2 iteratively constructs a pairwise objective function (Skolnick and Kihara, 2001). This algorithm uses an iterative dynamic programming approach with the pairwise objective function redefined at each iteration (see frozen approximation below). During the iteration, a sequence is threaded through the entire library of structural models. Top-scoring structures (within an empirically set score cutoff) are selected and the number of predicted contacts between sequence residues A_i and A_j is calculated as q_{ij}. If q_{ij} is greater than 3, then a "filtering" objective function is given by $V(i, j) = -\ln(q_{ij}/q^0)$, where the normalization factor is $q^0 = \sum_i \sum_j q_{ij}/L^2$. The objective function in the next iteration step is the arithmetic average of the "filtering" objective function and the original objective function $S(A_i, A_j, F(t(i), t(j)))$, if the filtering score is defined for the respective residues.

5. Aligning a sequence to a model

The goal of a threading alignment algorithm is to find an optimal match between the query sequence and a structural model among all possible sequence-structure alignments. The optimality of the match is defined by the objective function. If the objective function includes the quadratic term describing residue pairwise interactions, the general problem of finding the optimal alignment is NP-complete (Lathrop, 1994). Thus, one principal distinction among threading algorithms is determined by the objective function. The algorithms fall into three broad categories.

1. 1D algorithms that use only the information that is associated with the 1D features of the structure.
2. 1D/2D algorithms that apply the 1D search logic (objective function representation) for the optimal alignment but at various steps use the information inferred from the 2D features to redefine the objective function.
3. 2D algorithms that use the full 2D representation of the problem and deal with the higher complexity of the search space.

5.1. 1D algorithms

For one-dimensional models the sequence-structure alignment problem is analogous at an algorithmic level to sequence–sequence alignment. Given an objective function, an optimal alignment can be found using a dynamic programming alignment algorithm (Needleman and Wunsch, 1970; Smith and Waterman, 1981). This results in very fast sequence-structure alignment.

5.2. 1D/2D algorithms

The motivation in this type of algorithm is to include 2D information for its presumed greater information content, but use 1D alignment algorithms for speed. Thus, they look for an approximately optimal score, rather than the global optimum. In addition to 1D amino acid preferences, the score includes the 2D pairwise amino acid preferences. The expectation is that the inclusion of such preferences will improve the recognition of the best structural template. The algorithm of GenThreader (Jones, 1999) uses the quadratic terms of the objective function to evaluate the alignment generated by a 1D Smith–Waterman algorithm. These 2D scores together with sequence-profile similarity scores and solvation scores are the input to a neural network that assesses the confidence of the prediction. The neural network is trained on all known pairs of sequences that are known to have similar structures. The frozen approximation (Godzik *et al*., 1992; Skolnick and Kihara, 2001) iteratively performs a 1D alignment using 2D information fixed by the previous 1D alignment step.

5.3. 2D search algorithms

All 2D threading algorithms face an alignment problem that is formally intractable, so they differ on the basis of whether they return a good approximate alignment quickly or the global optimum alignment more slowly. Gaps are typically restricted to loops or the ends of secondary structure elements in order to reduce the alignment space and because deletions and insertions are less likely in the tightly packed protein core.

The Gibbs Sampling Algorithm (Bryant, 1996) begins with a random alignment. At each step it randomly chooses a core secondary structure element C, generates all possible alternative alignments for it, calculates each new alignment score S,

chooses a new alignment with probability proportional to $\exp(-S/kT)$, and fixes C at the new location. The procedure iterates using an annealing schedule to reduce the nominal energy units kT, then picks a new random alignment and repeats. The method does not guarantee a global optimum alignment, but is very fast and gives good performance.

The divide and conquer threading algorithm (Xu *et al*., 1998) repeatedly divides the structure model into submodels, solves the alignment problem for submodels, and combines the subsolutions to find a globally optimal alignment. Dividing the model into smaller pieces means that some pairs of model positions that contribute to the pairwise score will be split between different submodels, so the cutting of a pair link is recorded and accounted for when the subsolutions are recombined.

The branch-and-bound search algorithm (Lathrop and Smith, 1996) repeatedly divides the threading search space into smaller subsets and always chooses the most promising subset to split next. Eventually, the most promising subset contains only one alignment, which is a global optimum. It relies on a lower bound on the best score achievable within each subset. An anytime version (Lathrop, 1999) returns a good approximation quickly, then iteratively improves the approximation until finally returning a global optimum alignment.

Protein threading by linear programming (Xu *et al*., 2003) formulates the threading problem as a large scale integer programming problem, relaxes it to a linear programming problem, and solves the integer program by a branch-and-bound method. It is also an optimal method.

6. Selecting a model from the library

Finding the optimal score and alignment of a sequence to a structure leaves open the question what is a likely structure of the query sequence. There are broadly two approaches to this question. The first chooses the structure on the basis of the best alignment score, usually after normalizing scores in some way so that scores from different models are comparable. The second integrates the total probability of a model across all alignments of the sequence to the model. The first approach is more popular and intuitive, though the second is better grounded in probability theory.

Unlike sequence–sequence alignment, there is no straightforward method to determine the statistical significance of the optimal score (Bryant and Altschul, 1995). The statistical significance of the score tells how likely it is to obtain a given optimal score by chance. The distribution of scores for a given query sequence and structural model depends on both the length of the sequence and the size of the model. Currently, there is no generic analytical description of the shape of the distribution of threading scores across different models and sequences, though it is well understood that the distribution of optimal scores is not normal. For gapped local alignment of two sequences, or a sequence and a sequence profile, the distribution of optimal alignment scores can be approximated by an extreme value distribution. Fitting the observed distribution of scores to the extreme value distribution function has been applied by profile threading method FFAS (Rychlewski *et al*., 2000). Some profile-based methods approximate the distribution of scores by a normal

distribution and calculate Z-scores. The Z-scores are calculated with the mean and standard deviation of the scores of a query sequence with the library of all structural models. Similarly, many threading approaches with the quadratic pairwise objective function use the optimal raw score as the primary measure of structure and sequence compatibility and estimate the statistical significance of the score assuming a normal distribution of the sequence scores threaded to a library of available models.

In the approach implemented by GenThreader (Jones, 1999), the neural network score determines the compatibility of the structure with the model. The input to the neural network is a set of values of different scores: sequence similarity scores, the solvent accessibility score, and the pairwise interaction scores. The Gibbs-sampling threading approach (Bryant, 1996) estimates the significance of the optimal score by comparison to the distribution of scores generated by threading a shuffled query sequence to the same structural model. The distribution of shuffled scores is assumed to be normal.

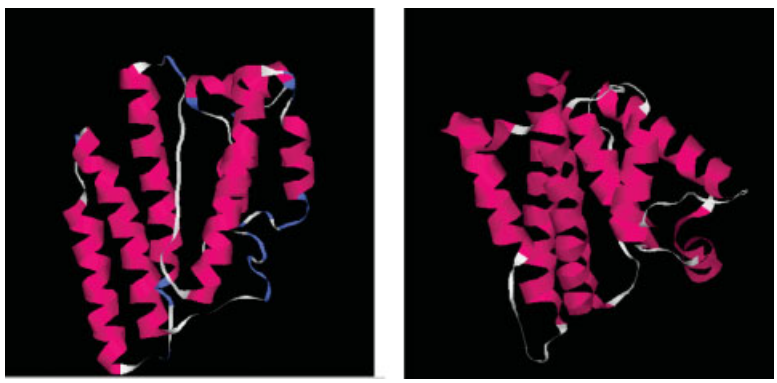
Lathrop *et al*. (1998) formulated a model-selection approach that does not rely on the optimal sequence-structure alignments. The compatibility of the sequence with a structural model is measured by the total alignment probability, where the probability is summed over all possible sequence-to-model alignments. This approach has been applied in the Bayesian fold recognition method (Bienkowska *et al*., 2000). The total probability is calculated using the filtering algorithm proposed and developed by White (1988). This method assumes that different structural folds (*see* Article 69, **Complexity in biological structures and systems**, Volume 7) are independent and equally probable structural hypotheses. The most probable model within the same fold category determines the prior probability of observing the sequence given the fold. The posterior probability of the fold given the sequence is calculated according to Bayes formula.

7. Performance of threading methods

The evaluation of different fold recognition (threading) methods takes place every two years during the Critical Assessment of Structure Prediction (CASP) (*see* Article 74, **Molecular simulations in structure prediction**, Volume 7) meeting. The prediction period is 3 months when different groups submit their predictions to a common depository. Targets for prediction are proteins that are about to have their structures solved. The CASP contest provides an opportunity to evaluate methods on the same sample set and in the context of truly "blind" predictions. This setting allows for performance comparison across methods using varied self-evaluation criteria applied by authors. Many threading approaches have been the subject of evaluation by past CASP contests. Examples of prediction methods that have been successful are: GenThreader (Jones, 1999), 3D_PSSM, (Kelley *et al*., 2000), FFAS (Rychlewski *et al*., 2000), SAM-2000 (Kelley *et al*., 2000), and PROSPECTOR (Skolnick and Kihara, 2001). Many other methods can be accessed at the CASP6 website **http://predictioncenter.llnl.gov/casp6/** (CASP, 2004) including 60 registered prediction servers. Over past CASP experiments, fold recognition methods have proved capable of recognizing distant sequence and structure similarities that are undetectable by sequence comparison methods. The evaluation of the last

CASP5 contest and description of several methods with best overall performance is described in Kinch *et al*. (2003).

Independently of the method, the accuracy of the prediction varies dramatically among proteins. Some proteins are much easier to predict than others and this can be recognized by consensus methods that automatically evaluate predictions generated by different algorithms. Consensus methods are generally more reliable than any single method alone. An example of a meta-server generating consensus predictions is Pcons **http://www.sbc.su.se/∼arne/pcons/**. Figure 1 shows an example of a partially successful prediction for protein HI0817, *Haemophilus influenzae* (target T129 in CASP5 contest). The overall topology of three C-terminal helices of the structure is correctly recognized, while the rest of the molecule is mispredicted. Even with the correctly recognized topology, the alignment of the residues in those helices is shifted by four positions. Figure 2 shows an example of a successful prediction for the single-strand binding protein (SSB), *Mycobacterium tuberculosis* H37Rv, (target T151 in CASP5 contest). All secondary structure elements are correctly aligned as well as most loops. Exceptions are one internal loop that is missing a glycine residue and the C-terminal tail. The missing residue was absent from the original target sequence submitted for predictions. This last example demonstrates that threading can be very successful in predicting protein structure. However, much effort is still required from the research community to provide a true solution to the protein structure prediction.



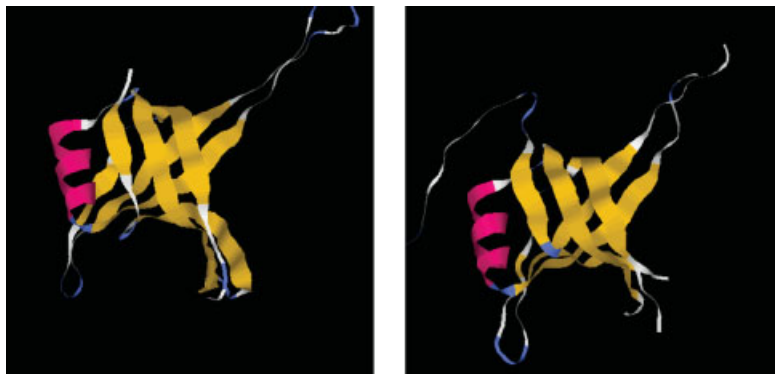
(a) (b)

```
T0129_str     93/92     NVFTQADSLSDWANQFLLGIGLAQPELAKEKGEIGEAVDDLQDICQLGYDEDDNEELAE
T0129_pred    95/96     QADSLSDWANQFLLGIGLAQPELAKEKGEIGEAVDDLQDICQLG--YDED------DNEE

T0129_str     153/152   ALEEIIEYVRTIAXLFYSHFN
T0129_pred    147/148   ELAEALEEIIEYVRTIAMLFY
```

(c)

Figure 1 Example of a partially correct structure prediction. (a) shows the predicted structure of the HI0817 protein. (b) shows the solved structure of that protein PDB code 1izmA. The left side of the picture corresponds to the C-terminal portion of the molecule. (c) shows the CE algorithm structural alignment of the C-terminal regions of the 1izmA and predicted structure of T0129. The topology of the three C-terminal helices is predicted correctly but the structural alignment shifts residues by four positions. Such shifts are typical misprediction of threading algorithms due to the periodicity of the helix structure. The predicted coordinates are the best structure prediction submitted to the CASP5 contest and are available from the CASP5 website



(a) (b)

```
T0151_str    5/6      TTITIVGNLTADPELRFTPSGAAVANFTVASTPRIYDRQTGEWKDGEALFLRCNIWREAA
T0151_pred   5/6      TTITIVGNLTADPELRFTPSGAAVANFTVAS-TPRIYDRQTGEWKDEALFLRCNIWREAA

T0151_str    65/66    ENVAESLTRGARVIVSGRLKQRSFETREGEKRTVIEVEV---DEIGPS
T0151_pred   64/66    ENVAESLTRGARVIVSGRLKQRSFETREGEKRTVIEVEVDEIGPSLRY
```

(c)

Figure 2 Example of a correct structure prediction. (a) shows the predicted structure of the HI0817 protein. (b) shows the solved structure of that protein PDB code 1izmA. (c) shows the CE algorithm structural alignment of the 1ue6A structure and the predicted structure of T0151. The structural elements are shown in the same colors as in (a) and (b), β-strands in yellow and α-helices in magenta. The structural alignment correctly aligns most residues between the model and the structure. The only misalignment is introduced by the omission of the GLY residue (indicated in red) from the predicted structure and the C-terminal tail. The predicted coordinates are the best structure prediction submitted to the CASP5 contest and are available from the CASP5 website

References

Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W and Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, **25**(17), 3389–3402.

Bienkowska JR, Rogers RG Jr and Smith TF (1999) Filtered neighbors threading. *Proteins*, **37**(3), 346–359.

Bienkowska JR, Yu L, Zarakhovich S, Rogers RG Jr and Smith TF (2000) Protein fold recognition by total alignment probability. *Proteins*, **40**(3), 451–462.

Bowie JU, Luthy R and Eisenberg D (1991) A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, **253**(5016), 164–170.

Bryant SH (1996) Evaluation of threading specificity and accuracy. *Proteins*, **26**(2), 172–185.

Bryant SH and Altschul SF (1995) Statistics of sequence-structure threading. *Current Opinion in Structural Biology*, **5**(2), 236–244.

Bryant SH and Lawrence CE (1993) An empirical energy function for threading protein sequence through the folding motif. *Proteins*, **16**(1), 92–112.

CASP (2004) CASP6, **http://predictioncenter.llnl.gov/casp6/**.

Cline MS, Karplus K, Lathrop RH, Smith TF, Rogers RG Jr and Haussler D (2002) Information-theoretic dissection of pairwise contact potentials. *Proteins*, **49**(1), 7–14.

Fischer D (2000) Hybrid fold recognition: combining sequence derived properties with evolutionary information. *Pacific Symposium on Biocomputing*, Hawaii, USA, 119–130.

Godzik A, Kolinski A and Skolnick J (1992) Topology fingerprint approach to the inverse protein folding problem. *Journal of Molecular Biology*, **227**(1), 227–238.

Higgins DG, Thompson JD and Gibson TJ (1996) Using CLUSTAL for multiple sequence alignments. *Methods in Enzymology*, **266**, 383–402.

Huang ES, Subbiah S, Tsai J and Levitt M (1996) Using a hydrophobic contact potential to evaluate native and near-native folds generated by molecular dynamics simulations. *Journal of Molecular Biology*, **257**(3), 716–725.

Jones DT (1999) GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. *Journal of Molecular Biology*, **287**(4), 797–815.

Jones DT, Moody CM, Uppenbrink J, Viles JH, Doyle PM and Harris CJ (1996) Towards meeting the Paracelsus challenge: the design, synthesis, and characterization of paracelsin-43, an alpha-helical protein with over 50% sequence identity to an all-beta protein. *Proteins*, **24**(4), 502–513.

Jones DT, Taylor WR and Thornton JM (1992) A new approach to protein fold recognition. *Nature*, **358**(6381), 86–89.

Karplus K and Hu B (2001) Evaluation of protein multiple alignments by SAM-T99 using the BAliBASE multiple alignment test set. *Bioinformatics*, **17**(8), 713–720.

Kelley LA, MacCallum RM and Sternderg MJ (2000) Enhanced genome annotation using structural profiles in the program 3D-PSSM. *Journal of Molecular Biology*, **299**(2), 499–520.

Kinch LN, Wrabl JO, Krishna SS, Majumdar I, Sadreyev RI, Qi Y, Pei J, Cheng H and Grishia NV (2003) CASP5 assessment of fold recognition target predictions. *Proteins*, **53**(Suppl 6), 395–409.

Lathrop RH (1994) The protein threading problem with sequence amino acid interaction preferences is NP-complete. *Protein Engineering*, **7**(9), 1059–1068.

Lathrop RH (1999) An anytime local-to-global optimization algorithm for protein threading in theta (m2n2) space. *Journal of Computational Biology*, **6**(3-4), 405–418.

Lathrop RH, Rogers RG Jr, Smith TF and White JV (1998). A Bayes-optimal sequence-structure theory that unifies protein sequence-structure recognition and alignment. *Bulletin of Mathematical Biology*, **60**(6), 1039–1071.

Lathrop RH and Smith TF (1996) Global optimum protein threading with gapped alignment and empirical pair score functions. *Journal of Molecular Biology*, **255**(4), 641–665.

Maiorov VN and Crippen GM (1994) Learning about protein folding via potential functions. *Proteins*, **20**(2), 167–173.

Munson PJ and Singh RK (1997) Statistical significance of hierarchical multi-body potentials based on Delaunay tessellation and their application in sequence-structure alignment. *Protein Science*, **6**(7), 1467–1481.

Needleman SB and Wunsch CD (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, **48**(3), 443–453.

Panchenko AR, Marchler-Bauer A and Bryant SH (2000) Combination of threading potentials and sequence profiles improves fold recognition. *Journal of Molecular Biology*, **296**(5), 1319–1331.

Rychlewski L, Jaroszewski L, Li W and Godzik A (2000) Comparison of sequence profiles. Strategies for structural predictions using sequence information. *Protein Science*, **9**(2), 232–241.

Singh RK, Tropsha A and Vaisman II (1996) Delaunay tessellation of proteins: four body nearest-neighbor propensities of amino acid residues. *Journal of Computational Biology*, **3**(2), 213–221.

Sippl MJ (1995) Knowledge-based potentials for proteins. *Current Opinion in Structural Biology*, **5**(2), 229–235.

Skolnick J and Kihara D (2001) Defrosting the frozen approximation: PROSPECTOR–a new approach to threading. *Proteins*, **42**(3), 319–331.

Skolnick J, Zhang Y, Arakaki AK, Kolinski A, Boniecki M, Szilagyi A and Kihara D (2003) TOUCHSTONE: a unified approach to protein structure prediction. *Proteins*, **53**(Suppl 6), 469–479.

Smith TF and Waterman MS (1981) Identification of common molecular subsequences. *Journal of Molecular Biology*, **147**(1), 195–197.

White JV (1988) Modelling and filtering for discretely valued time series. In *Bayesian Analysis of Time Series and Dynamic Models*, Spall JC (Ed.), Marcel Dekker: New York, pp. 255–283.

White JV, Muchnik I and Smith TF (1994) Modeling protein cores with Markov random fields. *Mathematical Biosciences*, **124**(2), 149–179.

Xu J, Li M, Lin G, Kim D and Xu Y (2003) Protein threading by linear programming. *Pacific Symposium on Biocomputing*, Hawaii, USA 264–275.

Xu Y, Xu D and Uberbacher EC (1998) An efficient computational method for globally optimal threading. *Journal of Computational Biology*, **5**(3), 597–614.

Zheng W, Cho SJ, Vaisman II and Tropsha A (1997) A new approach to protein fold recognition based on Delaunay tessellation of protein structure. *Pacific Symposium on Biocomputing*, Hawaii, USA 486–497.