Protein Threading Based on Multiple Protein Structure Alignment

Tatsuya Akutsu takutsu@ims.u-tokyo.ac.jp
Kim Lan Sim klsim@ims.u-tokyo.ac.jp
Human Genome Center, Institute of Medical Science, University of Tokyo
4-6-1 Shirokanedai, Minato-ku, Tokyo 108-8639, Japan

Abstract

Protein threading, a method employed in protein three-dimensional (3D) structure prediction was only proposed in the early 1990's although predicting protein 3D structure from its given amino acid sequence has been around since 1970's. Here we describe a protein threading method/system that we have developed based on multiple protein structure alignment. In order to compute multiple structure alignments, we developed a similar structure search program on massive parallel computers and a program for constructing a multiple structure alignment from pairwise structure alignments, where the latter is based on the center star method for sequence alignment. A simple dynamic-programming based algorithm which uses a profile matrix obtained from the result of multiple structure alignment was also developed to compute a threading (i.e., an alignment between a target sequence and a known structure). Using this system, we participated in the threading category (category AL) of CASP3 (Third Community Wide Experiment on the Critical Assessment of Techniques for Protein Structure Prediction). The results are discussed.

1 Introduction

Protein structure prediction is one of the most important computational problems in bioinformatics. The challenge involves predicting the correct three-dimensional (3D) structure from a given amino acid sequence of a protein. This problem is of immense importance because 3D structure is a prerequisite for function, and thus a lot of methods have been proposed for protein structure prediction [14]. Most methods are based on the assumption that amino acid sequence determines the 3D structure [5].

Although the *homology modeling* method seems the most reliable, it can be applied only when 3D structure of a similar sequence is already known. In order to overcome this drawback, Bowie, Lüthy and Eisenberg proposed a new method [7]. In this method, given an amino acid sequence and a set of protein structures (or structural patterns), a structure into which the sequence is most likely to fold is computed. An *alignment* between amino acids of a sequence and spatial positions of a 3D structure is computed using a suitable *score function* in order to test whether or not a sequence is likely to fold into a structure. The process of computing an alignment between a sequence and a structure is called *protein threading*, and its alignment, a *threading*.

A lot of variants have been proposed since the work by Bowie, Lüthy and Eisenberg [8, 11, 13]. We have also proposed an approximation algorithm for protein threading [4] and a method for deriving a good score function from known 3D structure data [3]. However, the predictive accuracy of the existing methods is not satisfactory. Moreover, there is a crucial drawback in most of the existing methods: users can not know the certainty of a prediction result, although, in the case of homology modeling, the similarity between sequences can be used as a measure of the certainty of a prediction result. Therefore, we developed a protein threading method for which users can know the certainty of a prediction result to some extent. Visual inspection of a threading embedded in multiple structure alignment of known structures provides users with means to evaluate the performance of threading.

Although the developed methodology may not be new, the developed system has some useful features: in the computation of a threading, several constraints can be put on; the system includes a fast parallel program for similar structure search.

The developed system does not make prediction automatically. Instead, prediction is done *interactively*, and the results of other prediction methods such as *secondary structure prediction* can be taken into account in the form of constraints. The outline of the prediction method is as follows:

(1) Candidates of possible structures are obtained from homology search, literatures and human inspiration,

(2) Structures similar to each candidate are searched from PDB (the database of 3D structures) [6] using the parallel search program,

(3) For each candidate, a multiple structure alignment is computed from pairwise structure alignments for similar structures by using a method similar to the *center star* method [9],

(4) A protein threading (i.e., an alignment between a sequence and a structure) is computed by a simple DP algorithm in which the multiple alignment result is used as a profile.

We applied the proposed method/system to the sequences given as the common problems (targets) in CASP3. CASP3 is a very good blind test of protein structure prediction and no one knows the answers at the time of the submission of predictions. We obtained a ranking of 14th in the threading category (category AL), to which 37 teams submitted their predictions. This shows that the proposed method/system is reasonably feasible.

The organization of the paper is as follows. A parallel program for similar protein structure search is described in Section 2. A method for computing multiple structure alignment and a method for computing protein threading are shown in Section 3 and in Section 4 respectively. Section 5 reports our results in CASP3, and in Section 6, we propose future work.

2 Similar Structure Search on Parallel Computer

A few years ago, one of the authors developed a computer program `stralign` for computing a structure alignment between two protein structures [2] (source code written in C-language is available via `http://www.hgc.ims.u-tokyo.ac.jp/service/tooldoc/stralign`). Although `stralign` is useful for comparing two structures, it takes several tens of seconds if it is applied to large structures. In similar structure search, an input structure is compared with all structures (several thousands of structures) in PDB. It would take too long time if `stralign` were directly applied to similar structure search. Therefore, we have developed a parallel program which executes several tens of `stralign` processes on a massive parallel computer.

For that purpose, we used the following simple *master-slave* model. The master process watches the status of all slave processes. If the master process finds an idle slave process, then it sends a protein structure, which is not yet compared, to the slave process. The slave process computes a structure alignment (using `stralign`) between that structure and the input structure, and then it returns the result to the master process. Although this model is very simple, it works quite well because each comparison can be made independently.

We implemented this model using the POSIX thread library on SUN ULTRA ENTERPRISE 10000 with 64 CPU (see Fig. 1). By means of storing all 3D data in main memory, we could achieve near linear speedup ratio per slave process (up to 50 processes), where speedup ratio represents the ratio of the search time taken by a slave process to the search time taken by n slave processes. The average speedup ratio over 5 input structures is shown in Fig. 1 because the search time depends on an input structure. Since it takes a lot of time when the number of processes is small, we do not test such cases and the speedup ratio is normalized so that it becomes 10.0 in the case of 10 slave processes. Note

that comparison of an input structure with all structures (> 7000 structures) in PDB can be done in a few minutes when 50 processors are used.

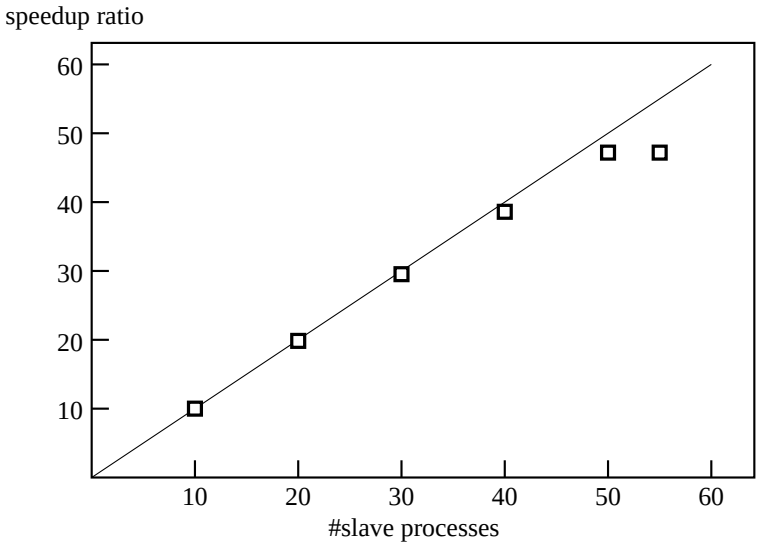

Figure 1: The speedup ratio against the number of slave processes. The linear line represents the ideal speedup ratio per slave process. The data for the above graph were generated from searching through ∼7000 structures on SUN ULTRA ENTERPRISE 10000.

3 Multiple Structure Alignment from Pairwise Structure Alignments

Although a lot of studies have been done on multiple sequence alignment, a few studies have been done on multiple structure alignment [10]. From a computational viewpoint, the problem of computing an optimal multiple structure alignment is proven to be NP-hard [1], where it is defined as a geometric problem. Moreover, even obtaining an approximate alignment is proven to be hard [1]. Therefore, development of a hueristic algorithm is a good choice.

In multiple sequence alignment, the *center star* method is one of the well-known heuristics [9]. Since it is very simple, we applied the center star method to multiple structure alignment (see Fig. 2). In order to apply the center star method, the *center* structure should be determined. Since a

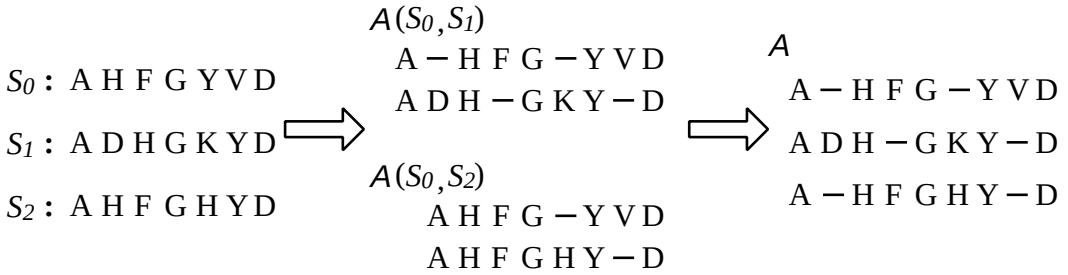

Figure 2: The center star method for multiple sequence/structure alignment. S_0 is the center in this example.

candidate structure S_0 is given in our case, we treat S_0 as the center structure. Then, using the parallel program described in Section 2, we obtain structures $S_1, S_2, \ldots, S_N$ that are structurally similar to S_0. Simultaneously, we obtain structure alignments $\mathcal{A}(S_0, S_1), \mathcal{A}(S_0, S_2), \ldots, \mathcal{A}(S_0, S_N)$, where $\mathcal{A}(S_0, S_i)$ denotes the structure alignment between S_0 and S_i. Note that structure alignment is represented in the same form as in sequence alignment [2]. After obtaining $\mathcal{A}(S_0, S_1), \ldots, \mathcal{A}(S_0, S_N)$, we apply the following procedure [9]:

1. Let I_0 be the maximum number of gap symbols placed before the first character (residue) of S_0 in any of the alignments $\mathcal{A}(S_0, S_1), \ldots, \mathcal{A}(S_0, S_N)$. Let $I_{|S_0|}$ be the maximum number of gaps placed after the last character of S_0 in any of the alignments, and let I_i be the maximum number of gaps placed between character $S_{0,i}$ and $S_{0,i+1}$, where $S_{j,i}$ denotes the i-th letter of string S_j (recall that a pairwise structure alignment is represented as a pairwise sequence alignment).
2. Create a string $\overline{S_0}$ by inserting I_0 gaps before S_0, $I_{|S_0|}$ gaps after S_0, and I_i gaps between $S_{0,i}$ and $S_{0,i+1}$.
3. For each S_j $(j > 0)$, create a pairwise alignment $\mathcal{A}(\overline{S_0}, S_j)$ between $\overline{S_0}$ and S_j by inserting gaps into S_j so that deletion of the columns consisting of gaps from $\mathcal{A}(\overline{S_0}, S_j)$ results in the same alignment as $\mathcal{A}(S_0, S_j)$.
4. Simply arrange $\mathcal{A}(\overline{S_0}, S_j)$'s into a single matrix $\mathcal{A}$ (note that all $\mathcal{A}(\overline{S_0}, S_j)$'s have the same length).

It should be note that $\mathcal{A}$ is defined so that the projection of $\mathcal{A}$ to S_0 and S_i is identical to $\mathcal{A}(S_0, S_i)$ if we delete columns consisting of gap symbols.

4 Protein Threading Based on Multiple Structure Alignment

4.1 A Simple Threading Algorithm

Protein threading is a problem of computing an (optimal) alignment between a sequence and a structure. In order to compute protein threading, *score function* is required and is important. In this paper, we use a simple profile-like score function obtained from multiple structure alignment. Note that, if a score function including interaction between two or more amino acids is considered, the protein threading problem becomes NP-hard [12] and computation of an optimal threading is very difficult [13].

Here we define a threading formally. Let $X = x_1 \ldots x_n$ be an input amino acid sequence (i.e., X is a string over alphabet Σ such that $|\Sigma| = 20$), which is a target sequence for 3D structure prediction. Let $\mathcal{A}$ be a structure alignment computed by the method described in Section 3 from a candidate protein structure. A threading $\mathcal{T}$ between X and $\mathcal{A}$ is obtained by inserting *gap symbols* (denoted by '$-$') into or at either end of X and $\mathcal{A}$ such that the resulting sequences are of the same length l, where columns of $\mathcal{A}$ must be preserved (see Fig. 3).

An optimal threading is computed in the following way (see Fig. 3). Let c_i be the i-th column of $\mathcal{A}$. Let $c_1^1, c_i^2, \ldots, c_i^{N+1}$ be amino acids in column c_i. Then, we define the score between x_j and c_i by

$$score(x_j, c_i) = \sum_{k=1}^{N+1} s(x_j, c_i^k),$$

where $s(x, y)$ is a usual score function for sequence alignment (e.g., Dayhoff matrix, PAM matrix). Currently, we use PAM250 matrix for $s(x, y)$. Then, an optimal threading is computed by a simple DP (dynamic programming) algorithm as in usual sequence alignment [16]. Note that the time complexity is $O(nmN)$ in this case because $N + 1$ scores are taken into account for each pair of x_j and c_i.

In some cases, we use the sum of K best scores of $s(x_j, c_i^k)$ in place of $\sum_{h=1}^{N+1} s(x_j, c_i^k)$, where K is some constant. Note that it is just a heuristic and there is no special reason.

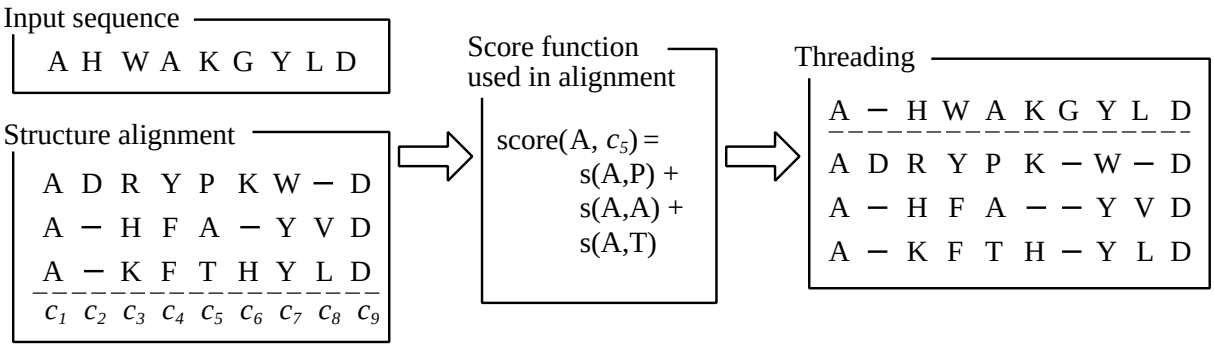

Figure 3: Threading based on structure alignment.

4.2 Protein Threading with Constraints

Although we implemented and examined the above algorithm, its performance was not good because the number of proteins having strong structural similarities but having weak sequence similarities was small. Thus, we computed appropriate threadings interactively using additional information. In particular, we used the results of secondary structure prediction by PHD [15].

Assume that the user convince that part of an input sequence $x_i \dots x_{i+k}$ must corresponds to part of a structure alignment $c_j \dots c_{j+k}$ from additional information such as PHD results. Then, we can put this constraint by letting the score function as follows:

$$score(x, c_{j+h}) = \begin{cases} +\infty, & \text{if } x \text{ is the same amino acid as } x_{i+h}, \\ 0, & \text{otherwise,} \end{cases}$$

where an appropriate constant (i.e., a very large constant) is used instead of $+\infty$ in practice. This modification is easy and does not increase the order of the time complexity of the DP procedure. Although it is a simple modification, it is very useful when we can use additional information.

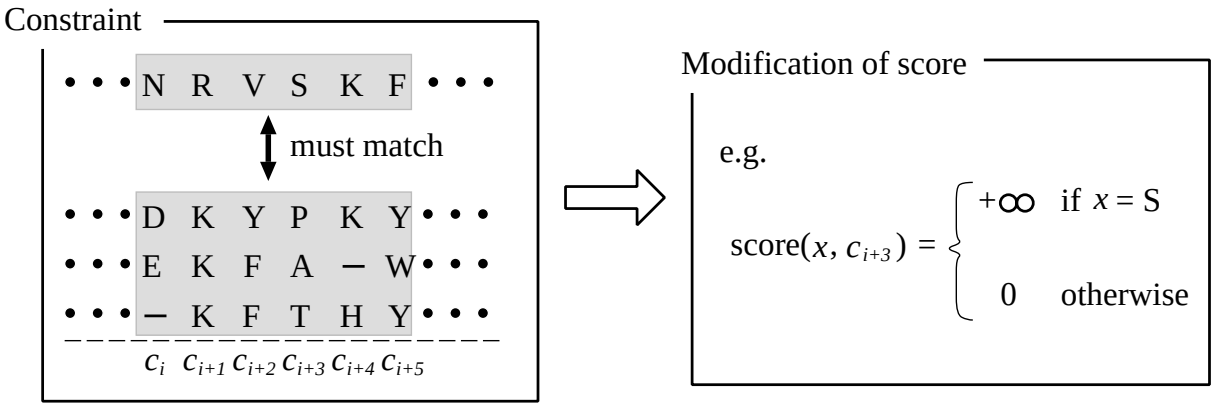

Figure 4: Modification of a score function in order to make use of constraints.

5 Our Results of Predictions in CASP3

Since the proposed prediction method is interactive, statistical test (using known 3D structures as a test set) is meaningless. Therefore, we made prediction on 35 proteins given as targets (problems)

in CASP3 (`http://predictioncenter.llnl.gov/`), among which 17 predictions are taken into the evaluation in the threading (AL) category (22 targets among 43 targets were taken into consideration in the AL category). CASP is a very fair competition for protein structure prediction because no one knows the correct answers (3D structures) at the time of submission of predictions [14]. After the submission of predictions, the organizers of CASP3 compare and evaluate the submitted predictions.

Three among 17 predictions (predictions for three targets T0043, T0053, T0081) were evaluated as similar to the correct folds. Even among the predictions by the best team, only eight were evaluated as similar to the correct folds. T0043 (PDB id: 1HKA) is 7,8-dihydro-6-hydroxymethylpterin-pyrophosphokinase from *Escherichia coli*, T0053 is CbiK protein from *Salmonella typhimurium*, and T0081 (PDB id: 1B93) is methylglyoxal synthase from *E. coli*. It is noteworthy to state that for target T0043, we were the only team who made a near correct prediction.

Biological knowledge of the second author (background of the first author is computer science, and background of the second author is applied biology) played an important role for making predictions, especially for generating candidates of template structures. However, even for the 3 targets for which we made good predictions, the obtained alignments were not good. In general, to select the appropriate fold from many possible candidates would otherwise have been difficult without the useful information generated from the developed system. Moreover, without the system, we could not have generated any threadings.

6 Concluding Remarks

We developed a practical method/system for computing protein threadings based on multiple structure alignment. As reported in Section 5, the performance of the system was not satisfactory. The followings are considered as the reasons.

(1) Since we did not use PSI-BLAST and we did not classify the known structures, we obtained candidates of template structures from homology search (FASTA and BLAST), literatures and human inspiration. Therefore, in some cases, we failed to generate appropriate candidates.

(2) For many candidates, the number of proteins having strong structural similarities but having weak sequence similarities was small. The probability of getting a good profile for protein threading decreases proportionally to the sample size.

(3) We did not have enough time for tuning and improving the system. We began to develop the system only a few months before the submission deadlines of CASP3.

But we believe that the methodology we used is not bad, because the idea of using the results of multiple structure alignment is very natural (although it may not be new). In particular, it can be assumed that, as the PDB becomes larger, the number of similar structures is expected to increase concomitantly. Thus, we foresee that the performance of our system will improve.

Finally, we briefly mention about future work.

- Currently, our method/system is far from automatic since candidates of protein structures and constraints should be given by users. Therefore, making the system much more automatic is important.
- The developed program of similar structure search is not fast enough for interactive use, and massive parallel computers are expensive. Moreover, the number of known protein structures is rapidly increasing. Therefore, much faster algorithms for similar structure search should be developed.
- Only sequence information was used after multiple structure alignment was obtained. Information such as polarity, area buried, and detailed physico-chemical properties of the amino acid residues should also be taken into account.

Acknowledgments

This work is supported in part by a Grant-in-Aid "Genome Science" for Scientific Research on Priority Areas from the Ministry of Education, Science, Sports and Culture in Japan.

References

[1] Akutsu, T. and Halldórsson, M.M., On the approximation of largest common subtrees and largest common point sets, *Lecture Notes in Computer Science*, 834:405–413, 1994.

[2] Akutsu, T., Protein structure alignment using dynamic programming and iterative improvement, *IEICE Trans. on Information and Systems*, E79-D:1629–1636, 1996.

[3] Akutsu, T. and Tashimo, H., Linear programming based approach to the derivation of a contact potential for protein threading, *Proc. Pacific Symposium on Biocomputing 1998*, World Scientific, 413–424, 1998.

[4] Akutsu, T. and Miyano, S., On the approximation of protein threading, *Theoretical Computer Science*, 210:261–275, 1999.

[5] Anfinsen, C.B., Principles that govern the folding of protein chains, *Science*, 181:223–230, 1973.

[6] Bernstein, F.C. *et. al.*, The Protein Data Bank: A computer-based archival file for macromolecular structures, *J. Molecular Biology*, 112:535–542, 1976.

[7] Bowie, J.U., Lüthy, R., and Eisenberg, D., A method to identify protein sequences that fold into a known three-dimensional structures, *Science*, 253:164–170, 1991.

[8] Godzik, A. and Skolnick, J., Sequence-structure matching in globular proteins: application to supersecondary and tertiary structure determination, *Proc. National Academy of Science USA*, 89:12098–12102, 1992.

[9] Gusfield, D., Efficient methods for multiple sequence alignment with guaranteed error bounds, *Bulletin of Mathematical Biology*, 55:141–154, 1993.

[10] Holm, L. and Sander, C., The FSSP database of structurally aligned protein fold families, *Nucleic Acids Research*, 22:3600–3609, 1994.

[11] Jones, D.T., Taylor, W.R., and Thornton, J.M., A new approach to protein fold recognition, *Nature*, 358:86–89, 1992.

[12] Lathrop, R.H., The protein threading problem with sequence amino acid interaction preferences is NP-complete, *Protein Engineering*, 7:1059–1068, 1994.

[13] Lathrop, R.H. and Smith, T.F., Global optimum protein threading with gapped alignment and empirical pair score functions, *J. Molecular Biology*, 255:641–665, 1996.

[14] Moult, J., Hubbard, T., Bryant, S.H., Fidelis, K., and Pedersen, J.T., Critical assessment of methods of protein structure prediction (CASP): Round II, *PROTEINS: Structure, Function, and Genetics*, Suppl. 1:2–6, 1997.

[15] Rost, B. and Sander, C., Prediction of protein structure at better than 70% accuracy, *J. Molecular Biology*, 232:584–599, 1993.

[16] Waterman, M.S., *Introduction to Computational Biology*, Capman & Hall, London, 1995.