Protein Threading

Guillaume Collet [a] Rumen Andonov [a] Nikola Yanev [c]
Jean-François Gibrat [b]

[a] *Symbiose team, IRISA, Campus de Baulieu, 35042 Rennes, France*

[b] *INRA, Unité Mathématique, Informatique et Génome UR1077, F-78352 Jouy-en-Josas Cedex, France*

[c] *Faculty of Mathematics and Informatics, University of Sofia, 1164 Sofia, 5 James Bourchier Blvd., Bulgaria*

Key words: Integer programming, combinatorial optimization, protein threading problem, protein structure alignment

1 Introduction

The most important *in silico* methods, to exploit the amount of new genomic data, are based on the concept of homology. The principle of homology-based analysis is to identify a homology relationship between a new protein and a protein whose function is known. For remote homologs, sequence alignment methods fail. In such a case one aligns the sequence of a new protein with the 3D structures of known proteins. Such methods are called fold recognition methods or threading methods.

Lathrop & Smith [1] were the first to propose an algorithm based on a branch & bound technique providing the global alignment with the optimal score and to prove the problem to be NP-Hard. Since then, other methods have been developed that improved the efficiency of the sequence – structure global alignment algorithm ([2–4]).

This paper describes a new algorithm that expands upon algorithms proposed in previous works ([3,4]) to allow implementation of *local* sequence – structure alignments. This allows threading methods to cover the whole spectrum of alignment types needed to analyze homologous proteins.

Our definition of alignments is based on the definition of the Protein Threading Problem (PTP) given in [1].

2 Outline of the Protein Threading Problem

Query Sequence and Structure Template: A query sequence is a string of length N over the 20-letter amino acid alphabet. A structure template is an ordered set M of m blocks which correspond to the secondary structure elements (SSEs). Block k has a fixed length of L_k amino acids. Let $I \subseteq \{(k,l) \mid 1 \leq k < l \leq m\}$ be the set of blocks interactions.

Alignment: An alignment of a structure template with a query sequence corresponds to positioning blocks of the template along the sequence. A *global* alignment requires that all blocks are aligned, preserve their order, and do not overlap. This alignement has been modelized by mixed integer programming (MIP) approach in [2,3]. In this paper, we extent the model presented in [3].

3 Local alignments : towards better PTP models

Global alignment assumes that all blocks are aligned with the query sequence. However, it sometimes happens that some members of a protein family do not share exactly the same number of SSEs. An alignment which permits to omit blocks is called a *local* alignment. To solve this local alignment, we propose two models: (1) A compact model (CM) where we modify constraints to omit blocks. (2) An extended model (EM) where we add dummy positions for each block. When a dummy position is chosen, the block is omitted. These models are described very briefly below. For more details, the interested reader can refer to our research report [5].

3.1 Compact model

We define a digraph $G(V,A)$ with vertex set V and arc set A. Each vertex $(i,k) \in V$ represents block k at position i along the sequence. A block k can take $n_k = N - L_k + 1$ positions along the query sequence. A cost C_{ik} (resp. D_{ikjl}) is associated to each vertex (i,k) (resp each arc $((i,k),(j,l))$). Let y_{ik} (resp. z_{ikjl}) be binary variables associated with vertices (resp. arcs). Based on these notations, we obtain the following model:

$$\max \sum_{k=1}^{m} \sum_{i=1}^{n_k} C_{ik} y_{ik} + \sum_{((i,k),(j,l)) \in A} D_{ikjl} z_{ikjl} \qquad (1)$$

Subject to:

$$y_{ik} \in \{0,1\}, \quad k \in M,\ i \in [1,n] \tag{2}$$
$$0 \leq z_{ikjl} \leq 1, \quad ((i,k),(j,l)) \in A \tag{3}$$
$$\sum_{i=1}^{n_k} y_{ik} \leq 1, \quad k \in M \tag{4}$$
$$\sum_{j=i+L_k}^{n_l} z_{ikjl} - y_{ik} \leq 0, \quad (k,l) \in I,\ i \in [1,n_k] \tag{5}$$
$$\sum_{i=1}^{\min(j-L_k,n_k)} z_{ikjl} - y_{jl} \leq 0, \quad (k,l) \in I,\ j \in [1,n_l] \tag{6}$$
$$y_{ik} + \sum_{j=1}^{min(n_k,i+L_k-1)} y_{jl} \leq 1, \quad 1 \leq k \leq l \leq m,\ i \in [1,n_k] \tag{7}$$
$$\sum_{i=1}^{n_k} y_{ik} + \sum_{i=1}^{n_l} y_{il} - \sum_{j=L_k+1}^{n_l} \sum_{i=1}^{j-L_k} z_{ikjl} \leq 1, \quad (k,l) \in I \tag{8}$$

Constraints (4) allow a block be aligned or not. Constraints (5) and (6) allow an arc, leaving (resp. entering) an activated vertex, be activated or not. Constraints (7) preserve the order of blocks. Finaly, constraints (8) coerce the activation of an arc if its vertices are activated.

3.2 Extended Model

Denote by $d_{ik}, i \in [1,N], k \in [1,m]$ a variable which we call dummy variables. The objective function is given by (1). This model uses constraints (2), (3), (5), (6) and (8). Additional constraints are the following:

$$d_{ik} \in \{0,1\} \quad k \in M,\ i \in [1,n_k] \tag{9}$$
$$\sum_{i=1}^{n_k} y_{ik} + \sum_{i=1}^{N} d_{ik} = 1 \quad k \in M \tag{10}$$
$$\sum_{i=1}^{j} d_{ik} + \sum_{i=1}^{min(j,n_k)} y_{ik} - \sum_{i=1}^{j} d_{ik-1} - \sum_{i=1}^{j-L_{k-1}} y_{i(k-1)} \leq 0 \quad k \in [2,m],\ j \in N \tag{11}$$

Constraints (10) state that exactly one vertex (either real or dummy), must be activated in a column. Constraints (11) preserve the order of the blocks.

4 Results

Two indicators have been used, computation time and the relative gap (RG) between the solution of the relaxed problem (LP) and the optimal solution (OPT): $RG = \frac{LP-OPT}{OPT}$. RG is a good indicator of the efficiency of the model since the smaller RG, the easier for the branch & bound algorithm to find the solution.

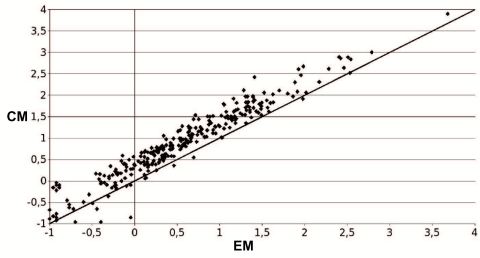

Fig. 1. Comparison of the computation times obtained by EM and CM. Each point represents an alignment. Times are expressed in seconds and are plotted using a base 10 logarithm scale.

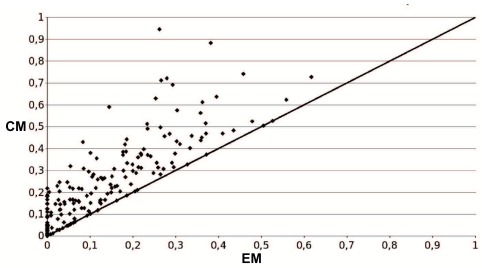

Fig. 2. Comparisons of relative gaps ($\frac{LP-OPT}{OPT}$) between models EM and CM. Each point is a sequence – structure alignment.

Figure 1 shows that EM is faster than CM for 99% of the instances. Moreover, Figure 2 shows that EM always gives a smaller RG than CM. It must be noted that LP relaxation directly gives the integer solution in 41% of the cases for the CM model and 52% of the cases for the EM model.

References

[1] R.H. Lathrop and T.F. Smith. Global optimum protein threading with gapped alignment and empirical pair potentials. *Journal of Molecular Biology* 255:641-665 (1996).

[2] J. Xu, et al. RAPTOR: optimal protein threading by linear programming. *Journal of Bioinformatics and Computational Biology* 1(1):95-118 (2003).

[3] R. Andonov, et al. Protein Threading Problem: From Mathematical Models to Parallel Implementations. *INFORMS Journal on Computing* 16(4):393:405 (2004).

[4] R. Andonov, et al. Recent Advances in Solving the Protein Threading Problem. *Grids for Bioinformatics and Computational Biology*, E-G. Talbi and A. Zomaya, editors. Chapter 14, pages 325-356. Wiley-Interscience (2007)

[5] Collet G., et al. Flexible Alignments for Protein Threading. *INRIA Research Report*, RR-6808 (2009).