2008－BIO－13 (14)
2008／6／27

タンパク質ベータシート予測
—動的計画法と形式文法によるアプローチ—

加藤 有己 [†]　　阿久津 達也 [†]　　関 浩之 [‡]

[†] 京都大学 化学研究所 バイオインフォマティクスセンター
[‡] 奈良先端科学技術大学院大学 情報科学研究科

概要 タンパク質２次構造予測はバイオインフォマティクスにおける主要な課題の１つである．特に，β シートはアミノ酸配列において幾つかの領域にまたがって現れるため，その領域を予測することは容易ではない．本稿では，連続する逆平行 β シートを予測するための動的計画法に基づくアルゴリズムを提案する．このアルゴリズムはより一般的な β シートのクラスを扱えるように拡張することが可能である．計算機実験では，提案アルゴリズムは予測精度において良い性能をあげている．また，提案アルゴリズムと形式文法に基づく手法の関連性について述べる．さらに，平面的な β シートを予測する問題は NP 困難であることを示す．

Prediction of Protein Beta-Sheets: Dynamic Programming versus Grammatical Approach

Yuki Kato[†], Tatsuya Akutsu[†] and Hiroyuki Seki[‡]

[†]Bioinformatics Center, Institute for Chemical Research, Kyoto University
[‡]Graduate School of Information Science, Nara Institute of Science and Technology

Abstract Protein secondary structure prediction is one major task in bioinformatics. In particular, it is a challenge to predict β-sheet structures since they range over several discontinuous regions in an amino acid sequence. In this paper, we propose a dynamic programming algorithm for some kind of antiparallel β-sheet, where the proposed approach can be extended for more general classes of β-sheets. Experimental results for real data show that our prediction algorithm has good performance in accuracy. We also show a relation between the proposed algorithm and a grammar-based method. Furthermore, we prove that prediction of planar β-sheet structures is NP-hard.

1 Introduction

Protein structure prediction is one of the central problems in bioinformatics and computational biology, and various approaches have so far been proposed. Secondary structure prediction is one of the major approaches. It asks which type of secondary structure (α-helix, β-strand, or others) each residue belongs to. Since it is a kind of classification problem, various machine learning and pattern recognition techniques have been applied, including hidden Markov models [2, 11], logic programming [14], neural networks [15], stochastic tree grammars [1] and support vector machines [8]. Although the overall prediction accuracy of existing methods is around 75% [12], it is recognized that β-strand regions are more difficult to predict than α-helix regions. This discrepancy may come from the fact that β-sheet structures typically range over several discontinuous regions, whereas α-helices are continuous and thus depend more on local sequence patterns.

Recently, Chiang et al. [5] proposed some grammar-based methods for protein secondary structure prediction. In particular, they proposed use of *range concatenation grammar* (RCG) [3] for β-sheet modeling. They suggested that linearly ordered β-sheets can be modeled by using a simple RCG and can be predicted in $O(n^5)$ time, where n is the number of residues in a given protein sequence. They also suggested that β-barrels and more complex β-sheet structures can be modeled by using RCG, while the time complexity increases to $O(n^7) \sim O(n^{12})$ depending on the complexity of β-sheet structures. However, they did not show how to incorporate residue-residue interaction preferences into the RCG-based methods. Furthermore, they posed the following question for proving NP-hardness of β-sheet prediction: "it remains to be seen whether such dependencies might be needed, for example, in calculating conformation counts for β-sheets."

In this paper, we propose a simple and flexible dynamic programming algorithm for prediction of antiparallel up-down β-sheets. This algorithm is based on RCG approach [5], where no experimental results on structure prediction were provided. It is noteworthy that our method explicitly takes pairwise interaction preferences into account and thus can be applied to real protein sequences. Hubbard [9] also used interstrand residue pairing preferences to predict β-strand contact maps, but did not show an original prediction algorithm specific for β-sheet prediction. Our prediction algorithm achieved good performance of overall per-residue accuracy $Q_3 \approx 80\%$ for nonhomologous protein sequences, where there are only two secondary structural states. Although types of β-sheet structures that can be handled by our method are restricted, the technique is extensible to more complex β-sheet structures including β-barrel. We also provide insight into an existing grammar-based method. Furthermore, we show that prediction of planar β-sheet structures is NP-hard. This result gives an answer to the question posed by Chiang et al. [5].

2 Methods

2.1 Ungapped Antiparallel β-Sheet

β-sheets are formed by pairwise interaction of several (consecutive) amino acids, called β-strands, in parallel and/or antiparallel way. Antiparallel β-structure is a fundamental topology of β-sheet, and many proteins include it in their domain. Although there are a large number of combinations of β-strands, it is known that the number of topologies of the class of antiparallel β-sheets is relatively few [4]. In this section, we are concerned with the simplest topology among them, called *up-down β-sheet,* where all strands have antiparallel topology via hydrogen bonding and they are connected by hairpin. In addition, suppose that every amino acid of β-strands is involved in hydrogen bonding, which we call *ungapped β-sheet.* This assumption enables us to design more efficient prediction algorithm in terms of computational complexity.

Let $a = a_1 a_2 \cdots a_n$ denote an amino acid sequence to be analyzed. We consider an ungapped up-down β-sheet that have N strands of the same length L where $N \leq \lfloor \frac{n}{L} \rfloor$. The reason why we can assume L is fixed is that we are concerned with only ungapped β-sheets. Because of this assumption, a β-sheet can be represented by an N-tuple of the start positions of β-strands $(p_1, p_2, \ldots, p_N)$ in the amino acid sequence a. Note that $p_i + L \leq p_{i+1}$ must be satisfied to prevent adjacent strands from overlapping each other. Let $s : (a_i, a_j) \to \mathbb{R}$ be a score (energy) function between two amino acid residues. Then, the ungapped up-down β-sheet prediction problem can be defined as follows:

Definition 1. (Ungapped up-down β-sheet prediction problem)
Input: An amino acid sequence $a = a_1 a_2 \cdots a_n$, the number of strands N, their common length L and a score function s.
Output: An ungapped up-down β-sheet $(p_1, p_2, \ldots, p_N)$ that minimizes $\sum_{i=1}^{N-1} \sum_{j=1}^{L} s(a_{p_i+j-1}, a_{p_{i+1}+L-j})$, subject to $p_i + L \leq p_{i+1}$ $(i = 1, 2, \ldots, N)$.

2.2 Dynamic Programming Algorithm

We provide a dynamic programming (DP) algorithm for predicting ungapped up-down β-sheets. In the experiments described later, we will predict β-sheet by changing the value of N, though N is fixed in the algorithm described below. Let $W(k, j)$ be the minimum free energy of up-down β-sheet for $a_1 \cdots a_j$, where j is the last position of the kth β-strand. $W(k, j)$ can be calculated by the following simple recursion formula:

$$W(k, j) = \min_i \{W(k-1, i) + S(i, j, L)\}$$

where $S(i, j, L) = \sum_{h=1}^{L} s(a_{i-L+h}, a_{j-h+1})$. The time complexity of the DP algorithm using this recursion is evaluated as $O(n^3)$. Obviously, the algorithm requires $O(n^2)$ space. Note that the optimal β-sheet itself can be constructed by a simple traceback procedure.

Although our DP algorithm can only handle ungapped up-down β-sheets, we can easily extend our method to predict more complicated structures, including consecutive parallel β-sheets, β-barrels as well as gapped structures.

In order to extend the algorithm for β-barrels, we compute the following:

$$W(k, j, i_0) = \min_i \{W(k-1, i, i_0) + S(i, j, L)\}$$

for each i_0 under the condition that

$$W(1, j, i_0) = \begin{cases} 0, & \text{if } j = i_0, \\ \infty, & \text{otherwise.} \end{cases}$$

Then, we compute the minimum of

$$W(N, j, i_0) + \sum_{h=1}^{L} s(a_{i_0-L+h}, a_{j-h+1}).$$

In this case, the time complexity increases from $O(n^3)$ to $O(n^4)$. More complicated β-sheet structures may be treated. However, the time complexity would increase as the complexity of β-sheet increases as suggested by the NP-hardness result in Section 5.

In order to extend the algorithm for gapped antiparallel β-sheets, it is enough to modify the definition of $S(i, j, L)$ so that it denotes the score of an optimal *alignment* between $a_{i-L+1} \cdots a_i$ and $a_j \cdots a_{j-L+1}$. In this case, the total time complexity increases to $O(n^4)$. Of course, we can extend it for prediction of gapped β-barrels. In that case, the time complexity remains $O(n^4)$.

3 Experimental Results

3.1 Data

In our experiments on up-down β-sheet prediction, we used real protein sequences with known structure available in PDB_SELECT (2007) [7] as the test sets (see Table 1). The criteria for selecting test data are as follows: (1) The test sequences are contained in the 25% threshold list of PDB_SELECT, where no two proteins have more than 25% sequence identity; (2) They have no α-helix; (3) They have at least four β-strands specified in DSSP [10]. Note that we do not count a residue involved in an isolated β-bridge as one strand; (4) All but at most one pair of adjacent β-strands in the primary sequence are involved in hydrogen bonding. This constraint results from lack of a perfect set of up-down β-sheets in the list.

3.2 Tests

Since the sequences selected above actually have different strand lengths, we set the strand length constant L by rounding the mean of their actual lengths. We used a contact potential table derived from 785 proteins described in [6] as the score function s. Implementation of the prediction algorithm was carried out in Java (version 1.6.0_03) on a machine with Intel Core2 CPU 6700 2.66GHz, 1.57GHz and 2.99GB RAM. To evaluate prediction accuracy of our algorithm, we measured per-residue accuracy Q_3, Q_E and Q_E^{pred}. Q_3 is the ratio of correctly predicted residues in overall secondary structural elements. Note that there are only two secondary structural states in this case (i.e., strand and other), and observed structures that we referred to are specified in DSSP. Q_E is defined as the ratio of the number of correctly predicted residues of the β-strands to the total number of residues of the strands in the observed structure, which corresponds to sensitivity. Q_E^{pred}, corresponding to specificity, is the ratio of the number of correctly predicted residues of the β-strands to the total number of predicted residues of the strands. Prediction results are shown in Table 1. Computation time is fairly short (0.56 seconds on average).

Table 1: Accuracy of up-down β-sheet prediction.

PDBID	N	n	L	Q_3 [%]	Q_E [%]	Q_E^{pred} [%]
2B9K	4	47	7	72.34	77.78	75.00
1AUU	4	55	4	83.64	70.59	75.00
1NY4	4	82	6	84.15	72.00	75.00
1TPN	5	50	4	68.00	61.11	55.00
2E6Z	5	59	4	74.58	61.90	65.00
2DIG	5	68	5	82.35	74.07	80.00
2JN4	6	66	5	87.88	89.29	83.33
2BT9	8	90	8	80.00	88.33	82.81
1G90	8	176	11	82.39	81.32	84.09
Average				79.48	75.15	75.03

4 Remarks on Grammatical Modeling

4.1 Definitions

Range concatenation grammar [3] is defined as a deductive system on sequences. A (positive) range concatenation grammar (RCG) is a 5-tuple $G = (N, T, V, P, S)$, where N, T, V and P are finite sets of predicate names, terminals, variables, rules, respectively, and $S \in N$ is the start predicate. For each predicate name $A \in N$, a nonnegative integer $\dim(A)$ is specified. Each rule in P has the shape $\psi_0 \to \psi_1 \cdots \psi_k$. This rule means that ψ_0 holds when all of $\psi_1, \ldots, \psi_k$ hold. Each ψ_i $(0 \leq i \leq k)$ in the rule is a predicate of the shape $A_i(\alpha_{i1}, \ldots, \alpha_{i\,\dim(A_i)})$, where $A_i \in N$ and each α_{ij} $(1 \leq j \leq \dim(A_i))$ is just a variable in V if $1 \leq i \leq k$. If every variable occurs at most once in the left-hand side (rsp. right-hand side) of a rule, the rule is called *left linear* (rsp. *right linear*). Let $\Rightarrow$ denote the one-step derivation relation and let $\overset{+}{\Rightarrow}$ denote the transitive closure of $\Rightarrow$. The language generated by an RCG G is $\{w \mid S(w) \overset{+}{\Rightarrow} \varepsilon\}$.

4.2 Modeling by RCG

Chiang et al. [5] presented the following RCG to generate linearly ordered β-sheets:

$$\begin{array}{ll}
\mathit{Beta}(xy) \to B(x, y), & B(xyz, y') \\
 & \quad \to B(x, y)\mathit{Adj}(y, y'), \\
B(yz, y') \to \mathit{Adj}(y, y'), & \\
\mathit{Adj}(x, y) \to \mathit{Anti}(x, y), & \mathit{Adj}(x, y) \to \mathit{Par}(x, y), \\
\mathit{Anti}(ax, y\bar{a}) \to \mathit{Anti}(x, y), & \mathit{Anti}(\varepsilon, \varepsilon) \to \varepsilon, \\
\mathit{Par}(ax, \bar{a}y) \to \mathit{Par}(x, y), & \mathit{Par}(\varepsilon, \varepsilon) \to \varepsilon,
\end{array}$$

where $a, \bar{a} \in T$ stand for amino acid residues that are connected with each other by hydrogen bond. (We extend the notion $\bar{u}$ for a sequence u.) Par and Anti generate parallel and antiparallel strands, respectively. $B(u, v)$ means that uv is a β-sheet where the second argument v is the "last" strand. Thus, the second rule says that if xy is a β-sheet (with y the last strand) and (y, y') constitutes a pair of adjacent strands, then $xyzy'$ is also a β-sheet (with y' the last strand) for an unpaired subsequence z. In this rule, the right nonlinearity plays a crucial role that expresses the constraints that the last strand y should be one component y of pair strands (y, y'). The time complexity of the structure prediction based on parsing of RCG is easily derived by counting the independent positions that appear in the arguments of the left-hand side for each rule, and taking the maximum of them. For example, the independent positions are marked by $*_i$ $(1 \leq i \leq 5)$ for the second rule as $B({}_{*_1}x_{*_2}y_{*_3}z_{*_4}, y'_{*_5})$. This is the maximum among all the above rules, thus the complexity is $O(n^5)$ where n is the length of an input sequence.

Returning to the problem of this paper, we assume that the length of each strand is L. This means that $|y| = |y'| = L$ in the second rule, implying that the position $*_3$ and $*_5$ is determined by $*_2$ and $*_4$, respectively. Thus, the time complexity becomes $O(n^3)$, which is the same order as our algorithm in Section 2. Note that the formalism in [5] does not incorporate residue-residue interaction preferences. Implementation or experimental results on β-sheet prediction based on RCG has not been reported as far as the authors know. On the other hand, we have performed experiments with real protein sequences. Although our algorithm currently considers only antiparallel β-sheets, it is not difficult to extend our proposed algorithm so that parallel structures and other complicated structures can be treated, as described in Section 2.2.

5 Hardness Result

Although we have presented an $O(n^3)$ time dynamic programming algorithm in Section 2, it remains a question whether *generalized* ungapped β-sheets can be predicted in polynomial time or not. To discuss the complexity of such a prediction problem, we define the corresponding decision problem as follows:

Definition 2. (Ungapped β-sheet prediction problem, UGBETA)
Input: An amino acid sequence, a topology diagram and a real number e.
Output: "Yes" if and only if there exists an ungapped β-sheet with some free energy e or less.

We can show that UGBETA is NP-complete by reducing the longest common subsequence problem that is known to be NP-complete [13], but omit a detailed proof as space is limited.

Definition 3. (Longest common subsequence problem, LCS)
Input: m sequences over an alphabet and a positive integer k.
Output: "Yes" if and only if there exists a common subsequence of length k or more, which is not necessarily consecutive.

Theorem 1. UGBETA is NP-complete even if the topology diagram is planar. □

References

[1] Abe, N. and Mamitsuka, H., *Machine Learning*, 29, 275–301 (1997).

[2] Asai, K., Hayamizu, S. and Handa, K., *Bioinformatics*, 9, 141–146 (1993).

[3] Boullier, P., *Proc. Sixth Intl. Workshop on Parsing Technologies (IWPT2000)*, 53–64 (2000).

[4] Branden, C. and Tooze, J., *Introduction to Protein Structure, Second Edition*, Garland Publishing (1999).

[5] Chiang, D., Joshi, A.K. and Searls, D.B., *J. Comp. Biol.*, 13, 1077–1100 (2006).

[6] Dosztányi, Z., Csizmók, V., Tompa, P. and Simon, I., *J. Mol. Biol.*, 347, 827–839 (2005).

[7] Hobohm, U., Scharf, M., Schneider, R. and Sander, C., *Protein Sci.*, 1, 409–417 (1992).

[8] Hua, S. and Sun, Z., *J. Mol. Biol.*, 308, 397–407 (2001).

[9] Hubbard, T.J.P., *Proc. the Twenty-Seventh Annual Hawaii Intl. Conf. System Sciences*, 336–344 (1994).

[10] Kabsch, W. and Sander, C., *Biopolymers*, 22, 2577–2637 (1983).

[11] Krogh, A., Brown, M., Mian, I.S., Sjölander, K. and Haussler, D., *J. Mol. Biol.*, 235, 1501–1531 (1994).

[12] Lin, K., Simossis, V.A., Taylor, W.R. and Heringa, J., *Bioinformatics*, 21, 152–159 (2005).

[13] Maier, R., *J. ACM*, 25, 322–336 (1978).

[14] Muggleton, S., King, R. and Sternberg, M., *Protein Eng.*, 5, 647–657 (1992).

[15] Rost, B. and Sander, C., *J. Mol. Biol.*, 232, 584–599 (1993).