

TM-1116

並列推論マシンを用いた
タンパク質の配列解析手法

石川 幹人、星田 昌紀、鬼塚 健太郎
廣沢 誠、戸谷 智之、新田 克己

September, 1991

© 1991, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

並列シミュレーテッドアニーリングを用いたタンパク質の配列解析¹戸谷 智之, 星田 昌紀, 石川 幹人, 新田 克己²(財) 新世代コンピュータ技術開発機構³金久 實⁴京都大学 化学研究所⁵

(1) はじめに

多数のタンパク質のアミノ酸配列の類似性を解析する有効な手法であるマルチプルアライメントは組み合わせ最適化問題として定式化可能である[1]。また、シミュレーテッドアニーリング[2]は汎用の組み合わせ最適化問題の有効な解決手法のひとつである。シミュレーテッドアニーリングにおいて困難とされている事前の温度スケジューリングを並列化により不要としたものが、並列シミュレーテッドアニーリング[3]であり、我々はそれを用いてマルチプルアライメントの解決を試みた[4]。このシステムの特徴は、一度に比較的多数の配列をアライメントできるため、多数の配列にわたった類似性を考慮できる。また、同時に複数のアライメント例を生成できるので、異なるアライメントの可能性を検討することにも有用である。

上記のような特徴を持つマルチプルアライメントシステムを実現したので、以下にその詳細について述べる。

(2) シミュレーテッドアニーリング

シミュレーテッドアニーリングのアルゴリズムは、組み合わせ最適化問題でローカルミニマム(局所的にはコスト最小であるが、大局的にはそうでない点)に捕まらずに、グローバルミニマムを探索することを可能にするものである。元来、アニーリングとは、物理系の焼きなまし過程を意味する。つまり、ある物質を高温の状態から徐々に温度を下げることにより、非常に安定な物質が得られる過程を指している。シミュレーテッドアニーリングとは、この焼きなまし過程を模倣したアルゴリズムで、温度パラメータに依存して探索の範囲が決定される探索手法である。高温時には、その温度に依存して、コスト(評価値)の悪化する方向への探索をも許し、徐々に温度を下げていくにつれて探索範囲を絞り込む。つまりコストの良くなるような方向にしか探索しなくなるわけである。

以下に、具体的にアルゴリズムを説明する。初期解 X_0 から順に次の様に解系列を生成していき、徐々に最適解に近い解を得ていく。まず、ある解 X_n にランダムな微小変形を行うことで次の解の候補 Y_n を作る。最小化を目的とする評価関数を E とすると、評価値の変化は $\Delta E = E(Y_n) - E(X_n)$ となる。 $\Delta E \leq 0$ ならば、無条件に $X_{n+1} = Y_n$ とし、また、 $\Delta E > 0$ のような場合には、確率値として、 $P = \exp(-\frac{\Delta E}{T_n})$ を採用し、温度パラメータ T_n に依存させて、次の解を決定する。つまり、確率 P で、 $X_{n+1} = Y_n$ とし、確率 $(1-P)$ で $X_{n+1} = X_n$ とする。このオペレーションを多数回繰り返す。ここで温度パラメータ列 $\{T_n\}$ を適切に設定することにより、最適解を求めることができる。この温度パラメータ列 $\{T_n\}$ を温度スケジューリングと呼ぶ。

(3) シミュレーテッドアニーリングの並列化

限られた時間内に可能な限り最適解に近づくために、温度スケジューリングを最適化することは非常に難しい問題である。そこで、次に述べる並列化[3]を用いて、人手による温度スケジューリング設計を解消できる。ユーザは低温の状態を監視しておくだけで、手軽に良い解を得ることが可能である。各アニーリングプロセスごとに初期解を与え、それぞれのプロセスにおいては担当する温度パラメータで一定温度のアニーリングを行う。そして、温度を減少させることに対応させて、ある間隔置きに、隣接する温度を担当しているプロセス間で、解の交換を確率的に行う[図1]。

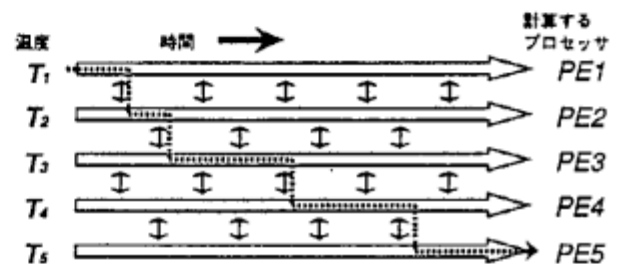


図1: 並列シミュレーテッドアニーリング

この場合、解交換を確率的に制御することにより、前もって温度スケジューリングを設計することを不要にし、適切な温度スケジューリングを経た解が最終的に低い温度プロセスに至ることが可能になる。温度パラメータ $T1$ において得られた解の評価値が $E1$ 、温度パラメータ $T2$ において得られた解の評価値が $E2$ の時 $\Delta E = E1 - E2$ 、 $\Delta T = T1 - T2$ とおく。この時、プロセス間の解の交換確率は、

$$p(T1, E1, T2, E2) = \begin{cases} 1 & \text{if } \Delta E \cdot \Delta T < 0 \\ \exp(-\frac{\Delta E \cdot \Delta T}{T1 \cdot T2}) & \text{otherwise} \end{cases}$$

として、定義される。これにより、各プロセス上における Boltzmann 分布に従う平衡状態を崩さず解の交換を行うことが可能になり、十分に時間をかければ最適解を得られることが保証される。

(4) マルチプルアライメントへの適用

タンパク質のアミノ酸配列のマルチプルアライメントをシミュレーテッドアニーリング法で実現した。ここで、解の初期状態はアライメントしたい複数の配列であり、適当な数のギャップをあらかじめ配列の前後に付け加えたものである[図2]。

解に対する微小変形は次のように定義した。複数の配列のうちのある1本の配列に対して、ランダムにアミノ酸、

¹Protein Sequence Analysis by Parallel Simulated Annealing²Tomoyuki Toya, Masaki Hoshida, Masato Ishikawa, Kazumi Nitta³Institute for New Generation Computer Technology (ICOT)⁴Minoru Kanchisa⁵Kyoto University

ギャップをそれぞれ選択し、選択されたギャップを選択されたアミノ酸の反対側の隣に移動させる [図 3]。ただし、両サイドのギャップは配列中にギャップが入り過ぎることを考慮して、ギャップがいくつあっても一つと見なしで確率的に選択している。ランダムな選択については、合同法により生成される乱数列を使って選択を行っている。この微小変形操作により、ギャップが配列中に挿入された状態を生成し、残りの配列との比較により評価値を決定する。

```

-----GFILATPQTGRASKEVISHVIECLATIGKPTIKTDNGPGTTGKHFQDFCQKLIQI-----
-----YSEFTFATARTGKATKDVQLHQAQSFAYNGIPQKINTDAPATVSRISQIFLARV-----
-----GVNFATLITGKASTVYIQHCLAVSANGKPAIKTDNGPATTSQKFRQFCQNDVT-----
-----IVVTQGRVTSVAVQKRWATAIAYLGRPK-AIKTDNGSCFTSKSTREVLARNGIAH-----
-----SGAISATQKAKETSSEAISSILQIAHLEKPSYITDNGPATISQDFLWCTSLA-----
-----DTFSGAVSYSCCKKETSCTETISAVLQAIISLLEKPLKINTDNGPAFLSQEFQFCT-----
-----HSAKRGLTQTITIGLLRAIVELGRPKLITDQGANYSKTFVAFCCQFCQVSL5-----
(score = 877)

```

図 2: 初期解

```

-----GFILATPQTGRASKEVISHVIECLATIGKPTIKTDNGPGTTGKHFQDFCQKLIQI-----
-----YSEFTFATARTGKATKDVQLHQAQSFAYNGIPQKINTDAPATVSRISQIFLARV-----
-----GVNFATLITGKASTVYIQHCLAVSANGKPAIKTDNGPATTSQKFRQFCQNDVT-----
-----IVVTQGRVTSVAVQKRWATAIAYLGRPK-AIKTDNGSCFTSKSTREVLARNGIAH-----
-----SGAISATQKAKETSSEAISSILQIAHLEKPSYITDNGPATISQDFLWCTSLA-----
-----DTFSGAVSYSCCKKETSCTETISAVLQAIISLLEKPLKINTDNGPAFLSQEFQFCT-----
-----HSAKRGLTQTITIGLLRAIVELGRPKLITDQGANYSKTFVAFCCQFCQVSL5-----
(score = 886)

```

図 3: 初期解からの微小変形

評価値の計算においては、各カラムにおけるアミノ酸の全組み合わせについて Dayhoff マトリックス [5] の値を総和し、全カラムにわたって、それを合計したものを使用した。ギャップを含むベアのコストについては、長さ k のギャップに対して $a + bk$ のような一次式が与えられる。 a と b は可変であるが、上記のアミノ酸の Dayhoff 値との兼ね合いから、通常は、 $a = 4$, $b = 1$ としている。

以上のように定式化して、アライメントを行った。

(5) KL1 プログラムの試作

本システムは、並列推論マシン Multi-PSI 上に、並列論理型言語 KL1 で実装を行った。

本システムにおいて、プロセスとは指定温度において、一定回数の微小変形を行うものを意味する。プログラムでは、最初に各プロセスに初期解とそのプロセスが担当する温度を与え、そのプロセスを各プロセッサエレメントに割り当てる。各プロセスの内部では、各配列に一定回の微小変形を繰り返して行い、あらかじめ決めておいた回数を行うごとに、隣接温度プロセスと所保持している解を交換するということが相対的に解の間でそれぞれの解を所保持しているプロセスを交換することに等しい。後者の方が通信のオーバーヘッドが小さいので、解交換を行う場合は、この方式で実装を行った。

本システムでは、ある温度の解をリアルタイムに表示する機能を持っている。モニター温度は対話的に切り替えることが可能である。これはユーザからの入力に従って表示ルーチンへのストリームを切り替えることにより実現している。その結果、複数の低温部の解をモニターすることにより複数のアライメント可能性が分かったり、高温部をモニターすることによりアニーリングが効果的に行われているかどうか判断できるようになった。

(6) 結果

約 60 個のアミノ酸から構成される蛋白質の前後にあらかじめギャップを 10 個ずつ付け加えたもの、7本を 1 時間アニーリングした結果、以下の良好な解 [図 4] を得た。こうした実験を行った結果、このシステムにおいては、全ての配列を対等に扱っているため、全配列に対して類似性が高いカラムが非常に効率良く揃うことが分かった。

```

-----GFILATPQTGRASKEVISHVIECLATIGKPTIKTDNGPGTTGKHFQDFCQK-LQI-----
-----YSEFTFATARTGKATKDVQLHQAQSFAYNGIPQKINTDAPATVSRISQIFLARV-----
-----GVNFATLITGKASTVYIQHCLAVSANGKPAIKTDNGPATTSQKFRQFCQNDVT-----
-----IVVTQGRVTSVAVQKRWATAIAYLGRPK-AIKTDNGSCFTSKSTREVLAR-NGIAH-E-----
-----SGAISATQKAKETSSEAISSILQIAHLEKPSYITDNGPATISQDFLWCTSLA-----
-----DTFSGAVSYSCCKKETSCTETISAVLQAIISLLEKPLKINTDNGPAFLSQEFQFCT-----
-----HSAKRGLTQTITIGLLRAIVELGRPKLITDQGANYSKTFVAFCCQFCQVSL5-----
(score = -1206)

```

図 4: マルチプルアライメント結果

また、低温部の他の温度プロセスには、異なった温度スケジュールを経た解が生成され、複数のアライメント結果を得ることが可能であることが分かった。アライメントが多様な評価基準を持つという観点から見ると、複数のアライメント結果を獲得できることは、このシステムが生物学者の研究を支援するツールとして有用であることを示している。

(7) おわりに

類似性の低い配列のマルチプルアライメントには、数時間から数日かかることもしばしばで、さらなる高速化、効率化の必要性がある。そのため、高温における解評価を大雑把にやって評価時間をかせぐ方法、低温における微小変形を小さくして無駄な変形を少なくする方法、ギャップの集約はまとめて動かす微小変形を導入する方法 [6] を検討している。また、温度パラメータの範囲や間隔、解交換を行う頻度などの一層のチューニングを行うという課題も残されている。

参考文献

- [1] 金久: シミュレーテッドアニーリングを用いたマルチプルアライメント法, 分子生物学会年会, 1989.
- [2] S.Kirkpatrick, C.D.Gelatt and M.P.Vecchi, "Optimization by Simulated Annealing," Science, vol.220, no.4598, pp.671-681, 1983.
- [3] 木村、瀧: 時間的一様な並列アニーリングアルゴリズム, 電子情報通信学会 NC90-1, 1990.
- [4] 石川、戸谷、星田、新田、金久: 並列シミュレーテッドアニーリングを用いたマルチプルアライメント, '知識情報処理技術とヒトゲノム計画' 講演要旨集, 1990, A-4.
- [5] Dayhoff, Hunt and Hurst-Calderone "Composition of Protein" in Atlas of Protein Sequence and Structure 5:3, Nat. Biomed. Res. Found., Washington, D.C., 1978, pp.363-373.
- [6] 萩原、金久: パターン認識を取り入れたマルチプルアライメント法, 日本生物物理学会年会, 1990.

3次元ダイナミックプログラミングを用いたタンパク質の配列解析¹石川 幹人, 星田 昌紀, 広沢 誠, 戸谷 智之, 新田 克己²(財) 新世代コンピュータ技術開発機構³

(1) はじめに

似たような性質や構造をもつタンパク質同士は、互いに似たアミノ酸配列をもつことから、タンパク質のアミノ酸配列の類似性解析がよく行われる。なかでも、多数のアミノ酸配列の類似性を解析するマルチプルアライメントは、構造や機能において重要な配列部分の抽出や、進化系統の分析に用いられる代表的な手法である。

本方式は、ある評価基準が与えられたとき、アミノ酸配列3本の最適なアライメントを実用的時間内で得ることができるものである。従来、2次元ダイナミックプログラミング法を用いたアミノ酸2本のアライメント[1]は、生物学者の間でよく用いられていた。しかし、それを組み合わせてマルチプルアライメントを求める[2]には、精度が十分でなく、難しいところは熟練した生物学者の勘に頼っていたのである。本方式の特徴は、従来のものを3次元(配列3本)に拡張し、計算量の増大の克服を図るために、処理の並列化を行ったことである。

これまでにも、ダイナミックプログラミングの多次元化を試みた例が存在しないわけではないが[3,4]、いずれも部分的に計算を間引いた近似計算であり、最適なアライメントを得るものではなかった。本方式を用いると、アライメントされた3本の組を複数組合せて、配列4本以上のマルチプルアライメントを比較的精度良く行うことができる。また、この組み合わせ法には独自のアルゴリズムを開発して使用している。

(2) ダイナミックプログラミング

ダイナミックプログラミング(DP)は段階的に決定を行う特徴を持つ最適化問題を解くためのアルゴリズムのひとつである。いくつもの段階で決定を行う必要があり、その各段階における決定が直前の段階の決定にのみ依存するという形式に最適化問題を定式化できるとき、DPを用いると非常に効率良く最適解を求めることができる。

アミノ酸配列のアライメントにDPを用いるときの概念的説明を、図1を用いて行う。たとえば、ADHE, AHIEという2つの配列をアライメントする場合、この2つの配列を図のような2次元のネットワークの辺に対応させる。斜め方向のアーキ(矢)は、そのアーキの位置に対応する2つのアミノ酸の類似度がコストとして割り振られる。また縦および横方向のアーキはギャップに対応し、ギャップを挿入するときのコストが割り振られる。

ギャップコストは経験上、ギャップの長さ k に対して、 $a + b \cdot k$ のような一次式を与えるのが適当であり、DPにも通常そうしたギャップコストが実装される。ギャップコストの効率の良い実装法や、それに伴う計算量の分析も考察されている[5]。

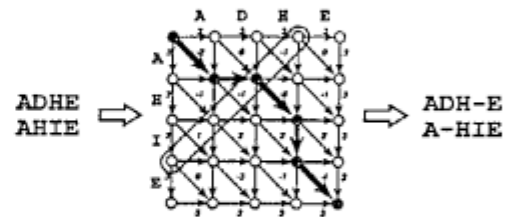


図1: 2次元 DP によるアライメント

このように問題を定式化すると、最適なアライメントを求めることは、このネットワーク上のコスト最小の経路を求めることに対応する。図の例では太いアーキで表された経路がコスト最小となる。この太いアーキで表された経路を順に見ていくと、図の右側のアライメントが得られる。

コスト最小の経路は左上の端から右下の端に向って、各ノードに至る最短経路を段階的に決定していくことにより求めることができる。各段階は図1の右上がり斜め線上に存在するノード群に対応する。段階的に各ノードへ至る最短経路を求めていくと、いったん求めた部分的最短経路は、もはや変更されることがない。それゆえこの部分的最短経路を用いて次の段階の計算を行うことができる。この性質があるために、全体の計算量が段階数の指数オーダーになることを回避できるわけである。

(3) 並列3次元 DP の実装

2次元 DP によるアライメントを3次元に拡張すると、こんどは図2のような3次元のネットワーク内の最短経路問題として表現できる。3次元 DP によるアライメントでは各段階が図の点線で囲まれた面の領域に存在するノード群に対応する。また2次元 DP によるアライメントでは各ノードは直前の3つのノードと連結されていたが、3次元 DP では7つのノードと連結されている。この計算量の多い3次元 DP を、並列推論マシン上で並列に動作させるプログラムを開発することにより、実行時間を削減させることができた。詳細は[6]を参照されたい。

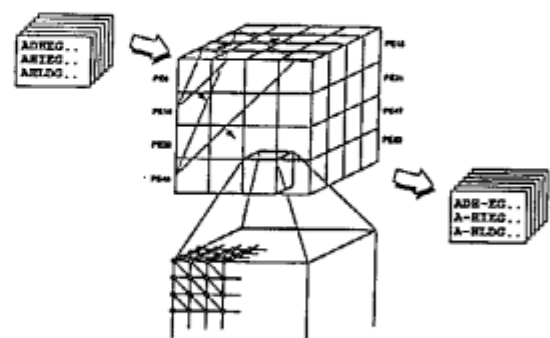


図2: 3次元 DP によるアライメント

図3に示すようなネットワークを並列言語 KL1 で構築し、マルチ PSI の要素プロセッサ (PE) 64 台に、均等に

¹ Protein Sequence Analysis by 3-dimensional DP-matching² Masato Ishikawa, Masaki Hoshida, Makoto Hiroseawa, Tomoyuki Toya, Katsumi Nitta³ Institute for New Generation Computer Technology (ICOT)

蛋白質配列間距離解析を用いた蛋白質の相同性解析システム

広沢 誠、星田 昌紀、石川 幹人

(財) 新世代コンピュータ技術開発機構 (ICOT)

1 はじめに

蛋白質の相同性解析の技術であるマルチプル・アライメントは、蛋白質の機能・構造予測、生物種の進化系統樹の作成の際に欠かせない技術である。従来は、複数本の蛋白質配列のマルチプル・アライメントは、人手により行われていたが、決定された蛋白質配列が増大するにつれて計算機によるマルチプル・アライメントが導入されつつある。

現在のマルチプル・アライメントシステムは2次元のDP [1] (DPによる2本の蛋白質配列のアライメント) を基本とするものである [2]。2次元DPに基づくマルチプル・アライメントは、計算量が少ないという点で優れているが、類似性の低い蛋白質間のアライメントの品質は良くないという欠点がある。

我々が蛋白質の相同性解析システムとして開発した統合アライメントシステム [3] は、3次元DPを基本とするものである。具体的には、3次元DPの結果を組み合わせてマルチプル・アライメントを作り、この結果をシミュレテッド・アニーリングにより洗練化する。この時、3次元DPの結果を組み合わせる順序は重要であり、これを決定する際に蛋白質配列間距離解析を行っている。今回は配列間距離解析 (“組み合わせ順序決定”) を中心に統合アライメントシステムについて発表を行う。

2 統合アライメントシステム

我々が開発した統合アライメントシステムは、(1) “初期アライメントの作成”、(2) “アライメントの洗練化”、(3) “アライメント評価”の3モジュールにより構成される。

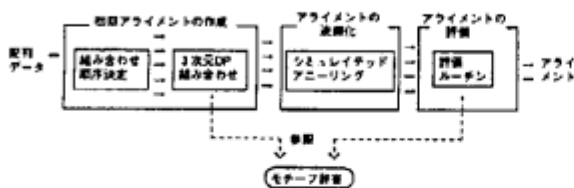


図1: 統合アライメントシステム

“初期アライメントの作成”のモジュールは、“組み合わせ順序決定”と“3次元DP組み合わせ”という2つのサブモジュールにより構成される。このモジュールは、明確なモチーフ（後述）を捕らえているアライメントを作成することを目的とするものである。ここでは、3次元DPの結果を組み合わせてマルチプル・アライメントを作成してゆく（組み合わせ法については [4] を参照のこと）。しかしながら、アライメントの結果は、3次元DPの結果の組み合わせ順序により異なってくるので、配列間距離解析により適切な組み合わせ順序を決定する (“組み合わせ順序決定”) については、後に詳細に説明す

る。“3次元DP組み合わせ”は、各組み合わせ順序に対して1つの初期アライメントの作成を行う。

“アライメントの洗練化”は、“初期アライメントの作成”で求めた各初期アライメントをシミュレテッド・アニーリング (以下SAと呼ぶ) を用いて洗練化する (各初期アライメントは、SAによるマルチプル・アライメント [5] の初期状態に対応している)。“アライメントの洗練化”は、さらに詳細なモチーフを捕らえているアライメントを作成することを目的としている。

“アライメント評価”は、“アライメントの洗練化”が出力した複数のマルチプル・アライメントの中から優れているアライメントを数個選択する。そして、選択されたアライメントを序列化し、結果をユーザに示すものである。このモジュールは、モチーフ辞書等を参照する。優れているアライメントを選択する具体的な基準としては、保存されている残基が有るか、それがモチーフであるか、アライメントの Dayhoff Score が良いかなどを用いる。

3 “初期アライメントの作成”

“初期アライメントの作成”では、“組み合わせ順序決定”において、蛋白質配列間距離解析を用いて3次元DPの結果の組み合わせ順序を決定し、この順序に従い“3次元DP組み合わせ”によりアライメントを作成する。以下、“3次元DP組み合わせ”の概略を述べてから“組み合わせ順序決定”を説明する。

3.1 “3次元DP組み合わせ”

“3次元DP組み合わせ”は“組み合わせ法” [4] により、DPにより求めた3次元アライメントを組み合わせてマルチプル・アライメントを作っていく。“組み合わせ法”自体は、2つのアライメント (例えば、 n 次元と m 次元のアライメント) より本数の多い高いアライメント ($n+m-1$ 次元) を作る手法である。“3次元DP組み合わせ”では、 m の値として3、1の値として2を採用している。つまり、 n 次元と3次元のアライメントより $n+1$ 次元のアライメントを作る。

例を用いて説明をする。配列0 から配列5までの6本の配列があるとすると、 “組み合わせ順序決定” において選択された組み合わせ順序として $\{\{4,5,3\}, \{4,5,2\}, \{4,5,1\}, \{4,1,0\}\}$ が選択されたとする。この場合は、まず、必要なアライメントとして4本の3DP ($\{4,5,3\}, \{4,5,2\}, \{4,5,1\}, \{4,1,0\}$) を行う。ここで、 $\{4,5,3\}$ は、配列4、配列5、配列3のアライメントを意味する。

この後、“組み合わせ法”により、4つの3次元アライメントより6次元アライメントを作っていく。まず、 $\{4,5,3\}$ と $\{4,5,2\}$ から共通配列である配列4と配列5を利用して $\{4,5,3,2\}$ をつくる。そして、 $\{4,5,3,2\}$ と $\{4,5,1\}$ より $\{4,5,3,2,1\}$ を作る。最後に、 $\{4,5,3,2,1\}$ と $\{4,1,0\}$ より $\{4,5,3,2,1,0\}$ を作る。

3.2 “組み合わせ順序決定”

配列が n 本ある場合に組み合わせ順序は $C_{n,3}(\prod_{i=1}^{n-1} C_{i,2})(n-3)!$ 通りある。例えば、6本の場合には28800通りの膨大な組み合わ

¹The Protein Similarity Analysis System using Distance Analysis

²Makoto Hirose, Masaki Hoshida, Masato Ishikawa

³Institute for New Generation Computer Technology (ICOT)

Seq0(copia17.6): ILDFHEKLLHPGIQKTKLFGETYYFPNSQLLIQNIINECSICNLAKTEHANTDMPTKTT
 Seq1(M-HULV): LLDLEQLTSLFSKMKALLERSHSPYYMLNRDRTLKNITETCKACAQVNAKSAVKQGTTR
 Seq2(ETLV): LTDALLITPVLQLSPAELHSFTHCGQTALTLQGATTTEASNLRSCHACRGNGPQHMPRGHI
 Seq3(RSV): VADSQATFQAYPLREKDLHTALHIGPRALSACNISMQQAREVVQTCPCNSAPALEAGVN
 Seq4(MMTV): ISDPIHEATQANTLHNLNANTLRLLYKITREQARDIVKACKQCVVATPVPHLGVN
 Seq5(SMRV): ILTALESAQESHALHQAALRFQFHITREQAIREIVKLCPCPDWGSAPQLGVN

図2: アライメントをするべき配列

Seq0: -----IL-DF----HEKLLHPGIQKTKL-F--GET--YY-FPNSQLLIQNIINECSICNL-AKT-EHR--W-TDMPTKTT
 Seq1: -----LL-DFL---EQ-LTHLSFSKM-KALLERSHSPYYMLNRDRTL-KNITETCKACAQ-VNA-SKS--A-VKQGTTR--
 Seq2: LTDALLITPVLQLSPA-ELHS-FTECGQTAL-T-LQ-----GATTTEA--SNILRSCHACRG-NGPQHMPRGHI-----
 Seq3: VADSQATFQAYPLREKDLHT-ALHIGPRAL-S-KA-----CNISMQQA--REVVQTCPCN-W-SAPA--LEAG-VN-----
 Seq4: -----ISDPIH-EAT-QANT-LHNLNANTL-R-LL-----YKITREQA--RDIVKACKQCVV-ATPVPHL--G-VN-----
 Seq5: -----ILTALE-SAQ-ESHA-LHQAAL-R-FQ-----FHITREQA--REIVKLCPCPDWGSAPQLGVN-----

図3: 統合アライメントシステムによる結果

せがある。この組み合わせから最終的に品質の良いアライメントを作ると予想される少数の組み合わせを選び出す必要がある。このために配列間距離解析を用いる(図4を参照)。

Step1 "3次元DP組み合わせ"で基本となる3次元DPによるアライメントの品質を配列間距離解析により評価する。配列*i*、配列*j*、配列*k*の品質 $R(i,j,k)$ を以下のように定義する。ここで、 $Cost(i,j)$ は配列*i*、配列*j*をDPによるアラインした時のコストである。 $a(i,j)$ は配列間距離の重みづけであるが、ここでは1に設定する。

$$R(i,j,k) = a(i,j)R(i,j) + a(j,k)R(j,k) + a(k,i)R(k,i) \\ [R(i,j) = -(Cost(i,k) + Cost(k,j)) / 2]$$

Step2 $n-2$ 階層 (n は配列の本数) の探索木を作成し、各枝にコストを割り振る。

まず、第一階層の葉として、三組 $\{i,j,k\}$ を $R(i,j,k)$ が大きい順に並べる。探索木の根と第一階層の葉を結ぶ枝にコストを割り当てる。現在の表現法では、 $l^*(n-1-m)$ を割り当てる。ここで l は葉の順番 m は階層の番号 (ここでは1) を割り当てる。

次に、各第一階層の葉の下に第二階層の葉として並べられる三組 $\{i,j,k\}$ (第一階層の三組との間に2つの配列が共通である必要がある) を $R(i,j,k)$ が大きい順に並べる。そして第一階層の葉との間の枝に上と同様にコストを割り当てる。以下、第 $n-2$ 階層まで同様の処理を行う。

Step3 探索木の根から葉にいたる各パスはおのの1つの組み合わせ方に対応する。各組み合わせ方のコストはパスの上の枝のコストを合計したものである。このコストを利用して初期アライメントを求める。

4. 結果

図2に示した配列を統合システムを用いアライメントを行った。構築した探索木を図4に示す。コストが13以下の組み合わせ方として7つのものを選択した。この6つに対して初期アライメントを求め、さらに"アライメントの洗練化"を行い、最終的に6つのアライメントを求めた最後に"アライメント評価"により図3に示したアライメントを最適なものとして選択した。このアライメントは、HH.....CC という Zinc Finger モチーフを捕らえている。Zinc Finger は蛋白質がDNAやRNAに結合する機能を持つ。このような部位をモチーフと呼んでいる。

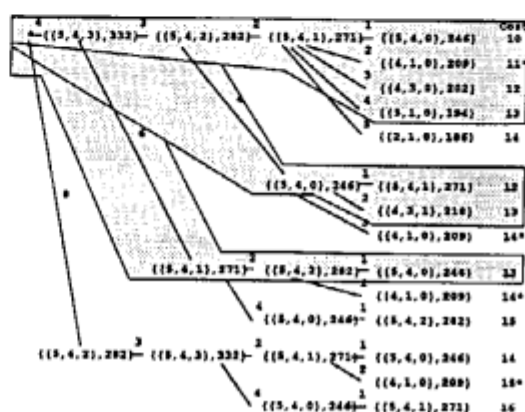


図4 例題の場合の探索木: 第2引数はアライメントの品質 $R(i,j,k)$ を示している。Zinc Finger を捕らえた組み合わせ順序を*で示している。

5 おわりに

3次元DPを基本とした蛋白質の相同性解析システムとこのシステムで重要な役割を果たしている配列間距離解析を紹介した。今後はさらに相同性の低い蛋白質配列の相同性解析にも利用できるようにシステムを改良していく予定である。

参考文献

- [1] Needleman and Wunsch "A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins", in J. Molecular Biology 48,1970, pp443-453
- [2] Jotun Hein "United Approach to Alignment and Phylogenies" in Methods in Enzymology Volume 183 Academic Press, 1990, pp.626-645
- [3] 石川、星田、広沢、戸谷、鬼塚、新田、金久: "並列推論マシンを用いたタンパク質の配列解析", 情報処理学会 情報処理基礎研究会報告23-2, 1991
- [4] 石川、星田、広沢、戸谷、新田: "3次元ダイナミックプログラミングを用いたタンパク質の配列解析" 情報処理学会 第43回全国大会論文集
- [5] 石川、戸谷、星田、新田、金久: "並列シミュレートッドアニーリングを用いたマルチプルアライメント", 情報処理技術とヒトゲノム計画 講演要旨集, 1990, A-4

フラクタル解析を応用したタンパク質の立体構造解析¹鬼塚 健太郎, 石川 幹人²(財) 新世代コンピュータ技術開発機構³

1 はじめに

タンパク質の立体構造の決定、予測は大変難しい問題である。ここでは、その予測技術の前段階として重要な立体構造解析について、今回開発した多変量解析を利用してタンパク質の局所構造を認識する方法、および、この方法にフラクタル解析を応用して、大域的構造を解析する方法の2つについて述べる。

2 タンパク質の構造解析予測技術の現状

2.1 タンパク質のアミノ酸残基配列と立体構造

タンパク質は20種類のアミノ酸が多数(およそ50から1000程度)鎖状に結合した高分子物質である。ただ単に、鎖状であるだけでなく、それが折れ畳まって、複雑な立体構造を形成する。

タンパク質のアミノ酸残基配列(互いに結合してタンパク質を形成しているアミノ酸を「アミノ酸残基」と呼ぶ)は自動的に配列決定を行う技術が開発されているため、既に数万種に及ぶタンパク質のアミノ酸残基配列が決定されているが、立体構造の決定は、現在、主にX線解析や核磁気共鳴法で行われており、どちらも、実験、データ解析双方に多くの困難を伴うため、未だに、数百種類のタンパク質についてしか満足な立体構造データが得られていない。

したがって、タンパク質のアミノ酸残基配列と立体構造の間の関係が分かれば、配列から立体構造を直接予測することが出来るようになり、分子生物学を応用した薬品の設計などの非常に広い応用分野が期待出来る。

2.2 理論的な構造予測法

比較的小さな分子の立体構造決定は、分子動力学理論や分子軌道理論などにより、分子の化学式から立体構造形成をシミュレーションによって求めることも可能であるが、タンパク質の折り畳みシミュレーションは、分子量が非常に大きいことや、折り畳みが水溶液中で行われるため、周囲の水分子の影響も考慮する必要があり、計算量が膨大なものとなる。そのため、現在のスーパーコンピュータによる数値計算技術をもってしても数千時間以上必要である。また、実際のタンパク質の折り畳みにはリボゾーム(タンパク質を合成する巨大なRNA分子とタンパク質の複合体)などが複雑に関与しているため、このようなシミュレーションによって得られた立体構造が正確である保証もない。現在、折り畳みシミュレーションについては計算量削減のために様々な努力が成されているが、どれも満足な結果を得るには至っていない。

2.3 発見的、帰納的構造予測法

既に立体構造が既知であるもののアミノ酸残基配列と立体構造との関係を発見的、帰納的に求め、構造を予測する研究も盛んに行われている。

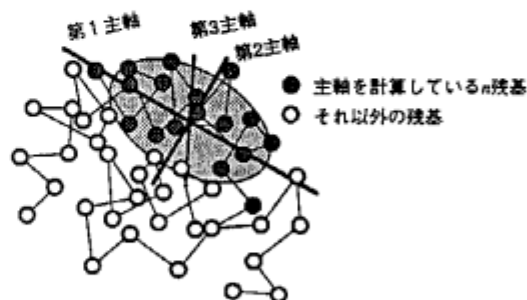
タンパク質の立体構造の中で、その主鎖(全てのアミノ酸残基でほぼ共通の化学構造を持つ部分で、鎖状のタンパク質の骨格となる部分)の比較的規則性をもった局所的構造として、コイル状の α -helixや、長く伸びた β -strand、それが複数並行に結合した平面状の β -sheet、主鎖が局所的に折れ曲がっているturnなどがある。立体構造中のこれらの局所的構造をタンパク質の2次構造(配列を1次構造、立体構造を3次構造としたとき、その間の構造として認識されるからである)と呼ぶ。

タンパク質内のアミノ酸残基は側鎖の種類によって、取りやすい2次構造が異なる。これを2次構造形成能と言う。立体構造既知のタンパク質から、それぞれのアミノ酸残基の2次構造形成能を統計的に求めて、その結果から2次構造を予測することがChouとFasman(1974)によって考えられた。しかし、この方法は様々な工夫(例えばFasman(1989)に詳しい)が試されたにも関わらず、予測精度が70%を超えることが無い。このことから、そもそも、立体構造を抽象化して表現する際に、2次構造をその単位とすること自体に問題があるのではないかと考えられる。

3 多変量解析およびフラクタル解析による構造の認識

これまで成されて来た構造予測の研究から、タンパク質の立体構造や局所的な構造をどのように抽象表現し、分類すべきか、という立体構造解析方法の開発が、構造予測方法の開発に先だって重要であると考えられる。

そこで、局所構造をより統計的、あるいは大域的に認識するために、アミノ酸残基の空間分布を多変量解析する方法を開発した。さらに、局所的な構造と大域的構造との関連を求めるための方法としてフラクタル解析を応用した方法を開発した。以下にその詳細を述べる。

図1: 配列上連続する n 残基の空間分布の主軸¹Protein Structure Analysis by Applied Fractal Analysis²Kentaro Onizuka, Masato Ishikawa³Institute for New Generation Computer Technology (ICOT)

3.1 多変量解析によるタンパク質の立体構造解析法について

配列上で連続する n 個のアミノ酸残基の主鎖の中の α 炭素原子 (アミノ酸の種類を決定する側鎖が結合している主鎖の炭素原子) の空間分布の 3 つの主軸 (互いに直交する 3 本の主軸) を主成分分析法で求め [図 1]、各主軸へ α 炭素原子の座標位置を投影したときの軸上での分布の標準偏差 σ を求めると、 n 残基が空間的にどのような形で分布しているかが分かる。標準偏差 σ は、主軸を求める際の行列の固有値の平方根と等しいので、比較的少ない計算量で求めることが出来る。

タンパク質の末端から連続する n 残基をずらしながらとり、それらの各主軸上での標準偏差を計算して配列に沿ってグラフにすることによって、そのタンパク質の構造の概略を表現することが出来る。

3.2 タンパク質の立体構造のフラクタル解析

配列上の色々な位置から n 個の残基を取りだし、それらの残基の空間分布を上述の方法で主軸上での標準偏差 σ として求め、それらの平均値 $\langle \sigma(n) \rangle$ を求めて、残基数 n とで両対数グラフ上にプロットしてグラフの傾きを求めることによりフラクタル次元が計算出来る。

4 解析システムの試作

本システムは ICOT で開発された並列論理型言語 KL1 で記述した。プログラム上は特に並列化を意識して作成した訳ではないが、KL1 処理系では並列処理が可能な部分は自動的に並列あるいは並行に実行されるので、今後の拡張性や並列化を考慮すると有利である。

5 結果

計算結果の例を [図 2] と [図 3] に示す。

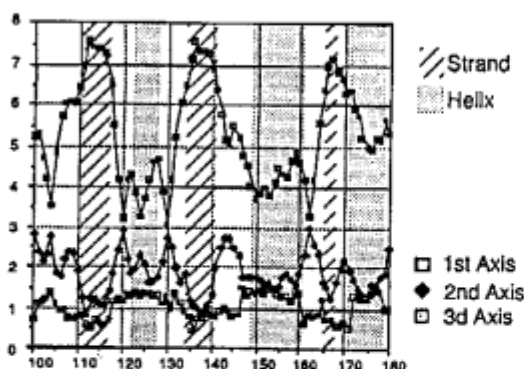


図 2: 8 残基の空間分布の主軸上での標準偏差

[図 2] は、3.1 で述べた方法 (ここでは $n = 8$) を用いて得られた空間分布の 3 つの主軸上で標準偏差のそれぞれの値を、配列に沿ってグラフ化したものである。 α -helix と β -strand の位置では、主軸上での標準偏差 σ の値は、それぞれある範囲内の値を取っていることが分かる。このことから 3 次元構造を立体表示して人手に頼って認識

していた 2 次構造が [図 2] のグラフから簡単に読みとれることが分かる。

また、この解析方法によって得られるデータは、異なるタンパク質の立体構造の類似性解析を行う際にも有益である。これまでは、3 次元座標 (全残基数を N としたときにパラメータ数 $3N$) を用いる場合のアフィン変換の任意性を避けるために各残基間の距離を計算した距離マップ (パラメータ数 $(N^2 + N)/2$) を使っていたので、大きな計算量を必要としたが、今回開発した解析法によるデータ (パラメータ数 $3N$ 以下) を利用すれば、少ない計算量で類似度を計算することが出来る。

[図 3] は 3.2 で述べた方法で行ったフラクタル解析の結果得られたグラフである。どのタンパク質においても、 $n = 20$ あたりを境にしてフラクタル次元が異なることが分かった。これは、統計的には、20 残基程度からなる局所的構造が大局的立体構造の構成要素となっていることを表している。また、sheet を多く含むタンパク質にくらべ、helix を多く含むタンパク質の方が、フラクタル次元が大きいかも分かった。このことから、この解析法によって、タンパク質を分類することが可能となる。

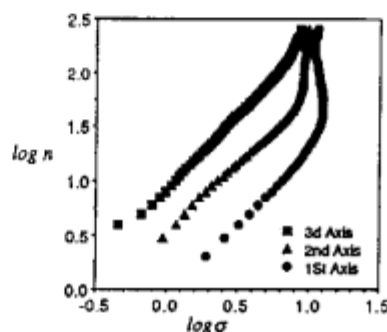


図 3: 残基数と空間分布の主軸上での標準偏差の平均

6 おわりに

本システムによって得られた構造データは数値データである。その点が、これまでの 2 次構造分類とは違う点である。したがって、構造予測の際も配列を 2 次構造分類上のクラスにあてはめる予測をするのではなく、配列から数値を予測することになる。その場合には、重回帰分析や数値的な学習手法を用いることが出来るため、これまでの予測方法に比べ、柔軟な予測が可能となる。

今後の課題としては、重回帰分析によってアミノ酸残基に適切な数値を割り振り、その数値から空間分布の主軸上での標準偏差を予測する方法を開発することが挙げられる。

参考文献

- Chou, P.Y. and Fasman, G.D., 1974, 'Prediction of protein conformation', *Biochemistry* 13, 222-244.
- Fasman, G.D.(editor), 1989, *Prediction of Protein Structure and the Principles of Protein Conformation*, New York: Plenum Publishing Corporation.
- Von Heijne, G., 1987, *Sequence Analysis in Molecular Biology*, San Diego: Academic Press, Inc.