# 遺伝的アルゴリズムによる塩基配列アラインメントの検討

河本　敬子† 水野　陽介‡ 一野　天利† 谷澤　一雄† 堀部　和雄†
近畿大学　生物理工学部　知能システム工学科† 学生‡

## 1．はじめに

配列アラインメントは、異なる遺伝子やタンパク質をデータベースから類似の配列を検索するために用いられ、進化類縁関係や機能を推測するときの基礎となる。塩基配列またはアミノ酸がアラインできた場合、両者は類似した遺伝子やタンパク質であると一般にいえ、既知の構造あるいは機能をもつという推測が成り立つ。この情報は病気に対抗する新薬の設計に繋がる。

本研究では、マルチプルアラインメントに対する遺伝的アルゴリズムを用いた解法を検討することを目的としている。

## 2．マルチプルアラインメント

マルチプルアラインメントでは、3本もしくはそれ以上の文字列を各文字ごとに照合し、一致数が最大となるように整列させる。

図1は3本の配列時のマッチングとギャップの挿入例を示す。一致数をより大きくするために文字列間にギャップ（“-” ハイフンで表す）を挿入させる。文字列が一致したときやギャップを挿入したときに与える点数の合計をアラインメントスコアと呼び、最大となるアラインメントスコアが導かれるような最適なアラインメントを求める[2]。また、本研究では文字間に挿入するギャップはランダムに入れる。今回は以下のような前提と定義を用いた。

- 対象は塩基配列とする。
- スコア設定は以下のようにする。
  マッチ時　　スコアに +2点
  ミスマッチ　スコアに −2点
  ギャップ　　スコアに −1点

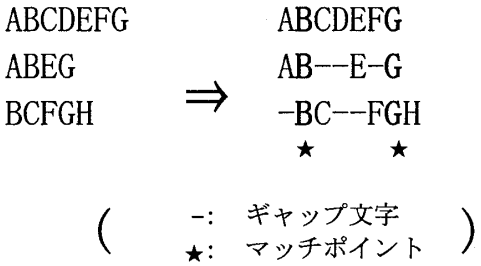


図1．アラインメントの例

## 3．遺伝的アルゴリズム

遺伝的アルゴリズム（Genetic Algorithm：GA）とは生物進化（選択淘汰・突然変異）の原理に着想を得たアルゴリズムであり、最適化の一手法である。歴史的にみるとGAはHollandのAdaptation in Natural and Artificial Systems（1975）において導入された手法である[1]。一般的なGAの処理手順は以下のようになる。このような一連の処理の繰り返しによって、個体集団は全体として適応度の高い個体の集団へと収束する。

Step 1. 個体の集団である個体集団を初期化する。
Step 2. 個体集団の各個体を目的関数に従って評価し、適応度を求める。
Step 3. 各個体の適応度の低い個体を個体集団から取り除き、逆に適応度の高い個体をその高さに応じて増やす。これを淘汰と呼ぶ。但し、個体集団のサイズを変えない。
Step 4. 個体集団の各個体をランダムに2つずつペアにし、このペアに突然変異、交叉を施して新しい個体を作る。
Step 5. Step 2〜4が繰り返しの単位であり、世代と呼ぶ。現在の世代の処理が終わると次の世代の処理が終わると次の世代の処理に移るため、Step 2へ戻る。

本研究での交叉方法は、2点交叉を用いた。図2は、本研究で使用した交叉方法を示す。本研究では、配列4つを1つの個体として表す。2個体（両親）に対して2箇所の交叉点をランダムに

Examination of Multiple Alignment Problem by Genetic Algorithm
Keiko Kohmoto†, Yosuke Mizuno‡, Tadatoshi Ichino†, Kazuo Tanizawa†, Kazuo Horibe†
† Department of Intelligent Systems, School of Biology-Oriented Science and Technology, Kinki University
‡ Undergraduate Student, Kinki University

| | 親1 | | | 子1 |
|---|---|---|---|---|
| 配列1 | TA-\|ACA\|G-T | | 配列1 | TA-\|A-CA\|G-T |
| 配列2 | -AT\|CT-A\|GC | ⇒ | 配列2 | -AT\|CT-A\|GC |
| 配列3 | AT-\|GT-A\|G-A | | 配列3 | AT-\|G-TA\|G-A |
| 配列4 | -GC\|AAA-\|G-C | | 配列4 | -GC\|A-AA\|G-C |
| | 親2 | | | 子2 |
| 配列1 | TA\|A-CA\|-GT | | 配列1 | TA\|ACA\|-GT |
| 配列2 | AT\|CT-A\|GCG | ⇒ | 配列2 | AT\|CT-A\|GCG |
| 配列3 | AT-\|G-TA\|--G | | 配列3 | AT-\|GT-A\|--G |
| 配列4 | GC-\|A-AA\|-GC | | 配列4 | GC-\|AAA-\|-GC |

図2. 交叉方法の例

指定し、それらの箇所で両親の遺伝子を交叉させ、子を生成する。

交叉によって生成された子はエリート選択によって、適応度の高い個体を次世代に残し適応度の低い個体を淘汰させる。

## 4. 数値実験

本実験では、配列アラインメントに対する遺伝的アルゴリズムを用いた解法について検討するために、文字数の違いによるスコアの増加傾向を調べた。試行回数は 100、個体数は 10、世代数は 1000 とした。突然変異は行わず、選択方法はエリート選択とした。また任意の配列を 4 つ用意し、文字数は 1 つの配列に対して 40、60、80、100、200、400、800 個とした。スコアは、配列 1 と 2、その後 1 と 3、1 と 4 を比較し、それぞれの点数を合計したものを示している。

表 1 は文字数 n の違いによる実験結果を示す。best は 100 回の試行回数で得られた最良解、average は解の平均値、best_gene_average は最良解が得られたときの平均世代を示す。表1から、n が小さければすぐに収束し、大きいほど多くの世代数を必要とすることが分かった。

図 3 は文字数 800 でのスコア分布を示す。スコアは 100 世代までに急激に高くなり、その後は緩やかに少しずつ増加することが分かる。早い段階で収束してしまうのは、文字数が少ないからだと考えられる。

## 5. 終わりに

本研究では、配列アラインメントに対する遺伝的アルゴリズムを用いた解法について検討した。今回の実験方法では、早い世代でスコアが収束し、その後はほとんど収束しないことが分かった。今後の課題としては、突然変異、交叉方法・選択方法の検討、様々な塩基配列やタンパク質のアラインメントを行う予定である。

表1. 文字数別での実験結果

| n | best | average | best_gene_average |
|---|---|---|---|
| 40 | 8 | -8.77 | 35.30 |
| 60 | 4 | -13.41 | 40.25 |
| 80 | 3 | -12.98 | 46.21 |
| 100 | 2 | -14.68 | 69.10 |
| 200 | 0 | -15.50 | 95.68 |
| 400 | 0 | -18.45 | 195.78 |
| 800 | 0 | -13.88 | 411.78 |

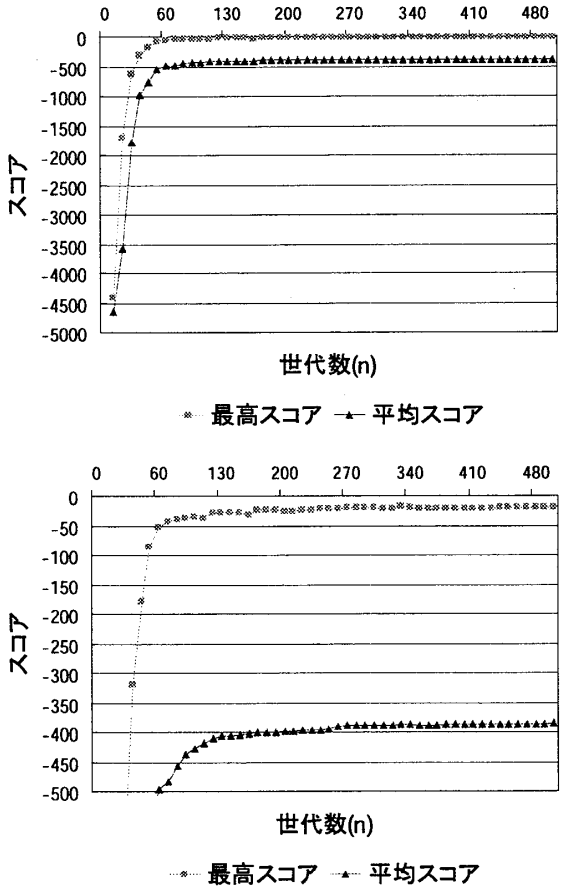


図3 文字数800でのスコア分布


参考文献

[1] 北野宏明, “遺伝的アルゴリズム”, 産業図書, 1995.

[2] 吉田孝廣, 牧之内顕文, “類似配列検索のための配列アラインメントアルゴリズムの高速化”, 電子情報通信学会, vol.104, no.176, pp.55-59, 2004.