Vol.2009-BIO-17 No.12
2009/5/26

パイロシーケンシング法で決定されたDNA配列の読み取り誤差の訂正

並木洋平[†1]　秋山泰[†1]

DNA シーケンシング技術のひとつであるパイロシーケンシング法は，一度に大量の DNA 配列をシーケンシングできるという利点がある一方，長い DNA 配列のシーケンシングを進めていく過程で測定されるデータに特有の読み取り誤差が入る傾向がある．本研究では，パイロシーケンシング法で決定された DNA 配列データから読み取り誤差を取り除き，元の配列を復元するための手法を開発した．パイロシーケンシング法をシミュレーションするプログラムを繰り返し用いて実際のパイロシーケンシング法で得られた発光強度の測定データにより近いシミュレーション結果を出力する配列を類推し，元の配列を復元していく手法をとった．

Correcting read errors on DNA sequences determined by Pyrosequencing

Youhei Namiki[†1] and Yutaka Akiyama[†1]

Pyrosequencing, one of the DNA sequencing technologies, allows us to determine the order of nucleotides in a large amount of DNA at a time. However, this method has a tendency to contain some particular read errors in the result sequences when determining long DNA sequences. In this study, we developed a method correcting read errors on DNA sequences determined by Pyrosequencing. In our method, a simple pyrosequencing simulator is repeatedly used and a corrected sequence which gives a simulated pyrogram most similar to that of real experimental record is chosen.

†1 東京工業大学 大学院情報理工学研究科 計算工学専攻
Graduate School of Information Science and Engineering, Tokyo Institute of Technology

1. はじめに

生物の DNA の塩基配列を決定することをシーケンシングといい，DNA のシーケンシングを行うための装置をシーケンサーという．DNA シーケンシングの手法にはさまざまなものがあるが，本研究では DNA の相補鎖の合成を監視しながら塩基配列決定を行っていく手法のひとつである「パイロシーケンシング法」を取り上げる．

パイロシーケンシング法は 454 Life Sciences 社が開発したシーケンサー「GS20」などで使われている配列決定法で，増幅させた DNA 断片の各塩基を端から順に化学反応させていき，反応系が発する光の強さを観測することで DNA の配列決定を行う．一度に大量の DNA 断片をシーケンシングできるという利点がある一方で，発生させる反応の特性上，シーケンシングで得られた DNA 配列データに特有の読み取り誤差が入ることがある．この現象は長い DNA 断片のシーケンシング時に発生しやすく，正しい配列を得る上でしばしば問題となる．

本研究ではパイロシーケンシング法で決定された DNA 配列データから読み取り誤差を取り除き，元の配列を復元することを目的とする．元の DNA 配列の推測にはパイロシーケンシングシミュレータを用いる．これを利用してシミュレーションを繰り返しながら，DNA 配列に含まれる読み取り誤差を訂正していく手法を取る．

シーケンサーを動かして DNA 配列データを採取するには多額の費用がかかるため，シーケンシングで得られたデータの精度を計算機によって改善可能ならば，生物の DNA データを正確に収集する上で大きな助けになると考えられる．

2. パイロシーケンシング法

2.1 パイロシーケンシング法について

パイロシーケンシング法 (Pyrosequencing method) は DNA の配列決定の手法のひとつで，合成時解読 (sequencing-by-synthesis) という手法の一種に分類される (図 1)．1990 年代後半に Mostafa Ronagh らによって実用化された[1)2)]．

パイロシーケンシング法による DNA 配列決定の手順は以下の通りである．

(1) あらかじめ配列決定する一本鎖 DNA を増幅し，反応系に固定しておく．一本鎖 DNA を増幅しておくのは，シーケンシング時の合成反応で十分な発光量を得られるようにするためである．

(2) 反応系に A，T，G，C のいずれかの溶液を投入し，十分な時間をおく．投入した溶

Vol.2009-BIO-17 No.12
2009/5/26

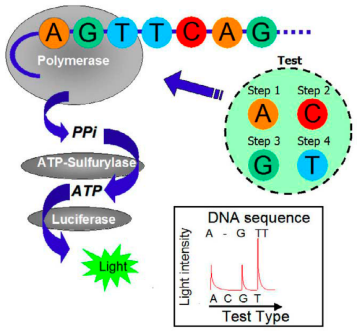

図 1 パイロシーケンシング法
Fig. 1 Pyrosequencing method.

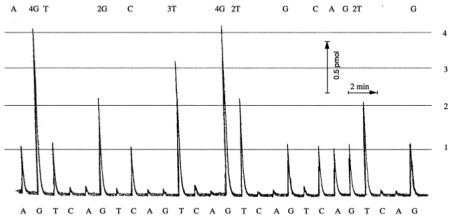

図 2 反応系が発する可視光の測定結果 (パイログラム)
Fig. 2 Pyrogram.

液の塩基が一本鎖 DNA の未合成の塩基のうち一番固定端側に近いものと相補的な関係である場合のみ対合して合成反応が起こり，相補鎖が伸長する．またこの合成反応の副産物であるピロリン酸が，反応系にある ATP スルフリラーゼやルシフェラーゼとの間で連鎖反応を起こしていくことにより，最終的に反応系で光が発生する．

(3) 反応系で発生した光をカメラで測定し，ソフトウェアで解析する．得られた発光強度データを**パイログラム**という (図 2)．発光強度は反応系で合成反応した塩基数にほぼ比例する．

(4) (2)〜(3) をひとつの**反応ステップ**とする．この反応ステップを各塩基の溶液について繰り返し行っていくことによって，一本鎖 DNA の全ての塩基を固定端側から順に合成させていき，パイログラムデータを得る．このパイログラムデータから一本鎖 DNA の塩基配列を決定する．

2.2 パイロシーケンシング法の問題点

パイロシーケンシング法では，以下の二点が原因でしばしば増幅された各 DNA 配列の合成の進み方に乱れが生じ，反応系で発する光の強さにノイズが入ることがある[3]．

(1) 不完全塩基対形成
(2) 余剰塩基除去ミス

これにより，本来合成するべき反応ステップで一本鎖 DNA と投入した溶液の塩基が対合せず合成反応が起こらなかったり，逆に合成しないはずの反応ステップで合成反応が起こる場合があり，反応系で増幅された個々の DNA 配列の合成反応の進み方に乱れが生じていく．合成反応の乱れは反応系で測定される光の強さに影響し，本来光が発生しないはずの反応ステップで光が発生して観測され，結果として誤った DNA 配列がシーケンシングされることがある．

これらのことが原因で，パイロシーケンシング法では比較的長い DNA 配列をシーケンシングすることが困難となっている．2005 年に発売された 454 Life Sciences 社のシーケンサーで配列決定できる DNA 配列は，改善が進んでいるが一本につき平均 100 塩基程度となっている (その後に発表された製品では，より長い配列が読めるとされているが，原理的な改善ではなく，複数回の測定での多数決に主に基づいた手法であるため，起きやすい誤りが多数決で残ってしまうおそれがある)．

2.2.1 不完全塩基対形成

不完全塩基対形成 (Incomplete-Hybridization) とは，反応系の一本鎖 DNA と本来反応するはずの塩基が対合せず合成反応が起こらないことである．溶液の塩基と一本鎖 DNA が接触しなければ合成反応は起こらないため，DNA の合成に遅延 (delay) が発生する．遅延した一本鎖 DNA は以後他の DNA と異なる反応の進み方をすることになり，本来反応すべきタイミングとは異なる時点で合成反応を起こし，光を発する．これによって発生した光はノイズとなって DNA 配列決定に影響を及ぼす．

2.2.2 余剰塩基除去ミス

余剰塩基除去ミス (Miss-Washing) とは，反応せずに残った余剰塩基をアピラーゼで除去するときに一部の塩基が分解に失敗し反応系に残留してしまうことである．反応系に残留した塩基は次の反応ステップの際に投入された塩基に紛れて一本鎖 DNA と対合し，他の増幅された一本鎖 DNA よりも合成反応が進んでしまう (gain) ことがある．

3. パイロシーケンシングシミュレータ

パイロシーケンシングシミュレータ (PyroSequencing Simulator) は，パイロシーケンシング法による DNA 配列決定のシミュレーションを行うソフトウェアで，当研究室で 2007 年に開発された[4]．パイロシーケンシングシミュレータは，シーケンシング対象となる DNA 配列と処理パラメータを入力として与えると，実際のパイロシーケンシングで得られると考えられるシーケンシング結果の DNA 配列と，合成反応時に得られるパイログラムがモンテカルロシミュレーションに基づき出力される．

処理パラメータには前述の**不完全塩基対形成の発生率**，**余剰塩基除去ミスの発生率**などの値を指定することができる．シミュレーションの際には，これらの値をシミュレーション対

Vol.2009-BIO-17 No.12
2009/5/26

象のシーケンサーに合わせて調整する必要がある．

4. 読み取り誤差訂正の手法

本研究では，パイロシーケンシング法で決定された DNA 配列データから読み取り誤差を取り除き，元の配列を復元するための手法を開発した．以下にその詳細を述べる．

4.1 誤差訂正に必要な情報

パイロシーケンシング法で決定された DNA 配列の読み取り誤差を訂正するためには以下の情報が必要である．

(1) 誤差訂正する DNA 配列 s_1

(2) シーケンサーから s_1 が得られたときのパイログラムデータ p_0

なお，パイログラムデータは反応系にある全ての増幅された一本鎖 DNA の 1 塩基が合成反応した場合に観測される発光強度を 1 とした場合の，各反応ステップでの発光強度を並べたベクトルデータとして与えられる．

これらの情報から，読み取り誤差を取り除いた DNA 配列 s_0 の推定 $\hat{s}_0$ を求める．

$$s_0 \xrightarrow{\text{sequencing}} (s_1, p_0) \tag{1}$$

$$(s_1, p_0) \xrightarrow{\text{error correction}} \hat{s}_0 \tag{2}$$

4.2 誤差訂正の方針

誤差訂正のおおまかな方針を以下に示しておく (図 3)．

(1) 誤差訂正する DNA 配列 s_1 を編集して元の DNA 配列 $\hat{s}_0$ の候補を列挙する．

(2) 候補配列をパイロシーケンシングシミュレータでシミュレーションし，パイログラムを取得する．

(3) 元のパイログラムとシミュレーション結果のパイログラムを比較してスコアを算出し，スコアが良い候補配列をより元の配列に近いものとして採用する．

(4) (1)〜(3) を繰り返し，最終的にスコアが良い配列を元の配列 $\hat{s}_0$ の候補として出力する．

どの手法も以上の方針に基いて誤差訂正を行う．

4.3 提案する誤差訂正手法について

今回提案した誤差訂正の手法は以下の 3 つである．

(1) 全近傍探索法 (AS: All-neighbor Search method)

(2) 順次探索法 (SS: Sequenial Search method)

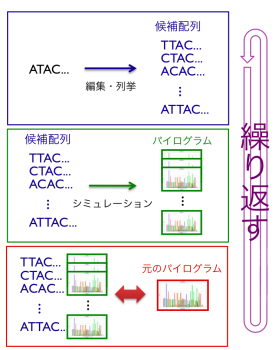

図 3 読み取り誤差訂正
Fig. 3 Error correction.

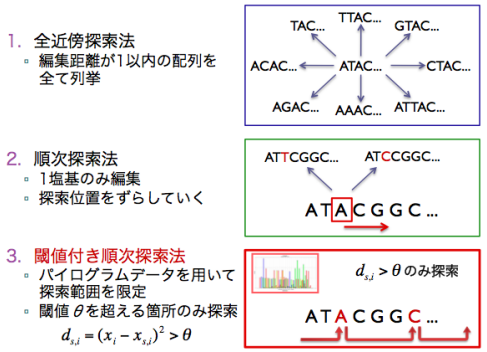

図 4 候補配列の列挙法
Fig. 4 Candidate sequences numeration method.

(3) 閾値付き順次探索法 (TSS: Threshold Sequential Search method)

各手法では元の配列 $\hat{s}_0$ の候補を列挙する方法がそれぞれ異なっている (図 4)．

4.4 手法 1：全近傍探索法

全近傍探索法 (AS: All-neighbor Search method) では次の手順で探索を行う．

(1) 前準備

(2) 候補配列の列挙

(3) 候補配列のシミュレーション

(4) シミュレーション結果のスコア計算

(5) 結果のスコア順ソート

(6) (2)〜(5) を繰り返す

4.4.1 前 準 備

S を DNA 配列の集合とする．入力として与えられた誤差訂正する DNA 配列 s_1 を S の要素に加える．

$$S = \{s_1\} \tag{3}$$

4.4.2 候補配列の列挙

C を候補配列の集合とし，$C = S$ とする．

各 $s(\in S)$ に以下の編集 (edit) を行うことで得られる DNA 配列全てを C に追加する．

(1) s に 1 塩基を挿入する (insertion)

(2) s 中の 1 塩基を別の塩基に置き換える (mutation)

(3)　s 中の 1 塩基を削除する (deletion)

$$C = S + \mathrm{edit}(S) \tag{4}$$

これにより，C には各 s から編集距離が 1 以内の DNA 配列が入った状態となる．s の長さを L とすると，各 s につき insertion で $4L$ 個，mutation で $3L$ 個，deletion で L 個，計 $8L$ 個の候補配列ができる．

4.4.3　候補配列のシミュレーション

各候補配列 $c(\in C)$ をパイロシーケンシングシミュレータの入力 DNA 配列としてシミュレーションを実行する．結果として得られるパイログラムデータ p_c を入力 DNA 配列と関連付けて保持しておく．

4.4.4　シミュレーション結果のスコア計算

x_i を入力として与えられたパイログラムデータ p_0 の i ステップ目の値，$x_{c,i}$ をシミュレーション結果のパイログラムデータ p_c の i ステップ目の値とする．また p_0 と p_c の全ステップ数をそれぞれ M_0，M_c とし，$M = \max\{M_0, M_c\}$ とする．

$$p_0 = (x_1, x_2, ..., x_{M_0}) \tag{5}$$

$$p_c = (x_{c,1}, x_{c,2}, ..., x_{c,M_c}) \tag{6}$$

これらを用いて，以下の式でシミュレーション結果と**観測された発光強度の一致性のスコア** d_c を計算する．

$$d_c = \sum_{i=1}^{M} (x_i - x_{c,i})^2 \tag{7}$$

ただし，$i > M_0$ のとき $x_i = 0$ とし，$i > M_c$ のとき $x_{c,i} = 0$ とする．

この d_c をシミュレーション結果の評価とし，値が小さいほど元のパイログラム p_0 に近いものとする．すなわち，このスコアの値が小さければ小さいほど当該シミュレーションの入力の DNA 配列 c は本来パイロシーケンシング法で配列決定した元の DNA 配列に類似している (読み取り誤差が含まれていない) ということになる．

4.4.5　結果のスコア順ソート

4.4.4 で計算したスコアを元にシミュレーション結果をソートし，スコアがよい (d_c の値が小さい) c のうち上位 N 個のみを次のステップにおける新たな S の要素とする．その後再び 4.4.2 に戻り，新たな S から候補配列を列挙してシミュレーションを行う，という手続きを繰り返す．

この 4.4.2 から 4.4.5 までの過程を一定回数繰り返し，最終的に残った S を元の DNA 配列の候補として出力し，読み取り誤差訂正を終了する．一度のループで DNA 配列の編集距離が 1 だけ変わるため，s_0，s_1 間の編集距離の回数だけ 4.4.2 〜 4.4.5 を実行しなければならない．

4.5　手法 2：順次探索法

順次探索法 (SS: Sequential Search method) では次の手順で探索を行う．

(1)　前準備
(2)　候補配列の列挙
(3)　候補配列のシミュレーション
(4)　シミュレーション結果のスコア計算
(5)　結果のスコア順ソート
(6)　(2)〜(5) を繰り返す

4.5.1　前　準　備

S を DNA 配列の集合とする．入力として与えられた誤差訂正する DNA 配列 s_1 を S の要素に加える．

$$S = \{s_1\} \tag{8}$$

また，$i = 1$ とする．

4.5.2　候補配列の列挙

C を候補配列の集合とし，$C = S$ とする．

各 $s(\in S)$ について以下の編集 (edit) を行うことで得られる DNA 配列全てを C に追加する．

(1)　s の i 塩基目の前に 1 塩基を挿入する (insertion)
(2)　s の i 塩基目を別の塩基に置き換える (mutation)
(3)　s の i 塩基目を削除する (deletion)

$$C = S + \mathrm{edit}(S, i) \tag{9}$$

各 s につき insertion で 4 個，mutation で 3 個，deletion で 1 個，計 8 個の候補配列ができる．

4.5.3　候補配列のシミュレーション

全近傍探索法と同様である．

4.5.4　シミュレーション結果のスコア計算

全近傍探索法と同様である．

4.5.5　結果のスコア順ソート

4.5.4 で計算したスコアを元にシミュレーション結果をソートし，スコアがよい (d_c の値が

小さい) c のうち上位 N 個のみを S の要素とする．ここで，新たな S の要素がシミュレーション前の S と同じ場合，つまり i 塩基目について編集を行ってもスコアが改善されない場合，i の値を 1 増やして次の塩基の編集を行うようにする．

この 4.5.2 から 4.5.5 までの過程を i の全ての $s(\in S)$ の配列長よりも大きな値になるまで繰り返す．そして，最終的に残った S を元の DNA 配列の候補として出力し，読み取り誤差訂正を終了する．

4.6 手法 3：閾値付き順次探索法

閾値付き順次探索法 (TSS: Threshold Sequential Search method) は，パイログラムデータから読み取り誤差が含まれる部分を推測し，その部分を中心に配列の編集・シミュレーションを行う手法である．前述の手法と比較して精度を落とさず高速に誤差訂正をすることができる．

閾値付き順次探索法では以下の手順で探索を行う．

(1) 前準備
(2) パイログラムの差分の計算
(3) i ステップ目にシーケンシングされた部位の導出
(4) 候補配列の列挙
(5) 候補配列のシミュレーション
(6) シミュレーション結果のスコア計算
(7) 結果のスコア順ソート
(8) パイログラムの差分が閾値以上のステップがなくなるまで (2)〜(7) を繰り返す

4.6.1 前 準 備

S を DNA 配列の集合とする．入力として与えられた誤差訂正する DNA 配列 s_1 を S の要素に加える．

$$S = \{s_1\} \tag{10}$$

また，許容するパイログラムの差分の閾値を θ とする．さらに $i = 1$ とする．

4.6.2 パイログラムの差分の計算

i ステップ目にシーケンシングされた領域に読み取り誤差が含まれている可能性のある配列の集合を C' とする．

各 $s(\in S)$ をシミュレータの入力としてシミュレーションを実行し，結果得られたパイログラムを $p_s = (x_{s,1}, x_{s,2}, ..., x_{s,M_s})$ とする．また，$M = \max\{M_0, M_s\}$ とする．

次に，シーケンサーから s が得られたときのパイログラム p_0 と s のパイログラム p_s の i

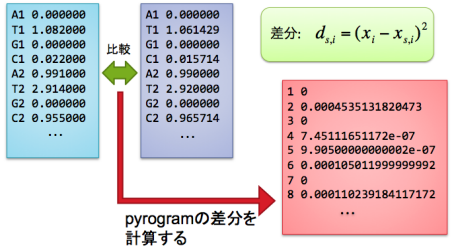

図 5 パイログラムの差分の計算
Fig. 5 Calculation of the deference between two pyrograms.

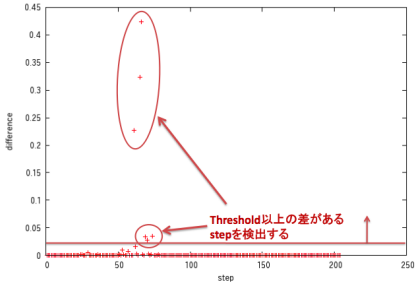

図 6 二つのパイログラム間の差分
Fig. 6 The deference between two pyrograms.

ステップ目の差分 $d_{s,i}$ を計算する．

$$d_{s,i} = (x_i - x_{s,i})^2 \tag{11}$$

ただし，$i > M_0$ のとき $x_i = 0$ とし，$i > M_s$ のとき $x_{s,i} = 0$ とする．

そして，$d_{s,i} \geq \theta$ となる s を C' に加える．$d_{s,i} \geq \theta$ となる s がひとつもない (つまり $C' = \emptyset$) 場合は i の値を 1 増やし，4.6.2 の最初に戻って処理を繰り返す．

4.6.3 i ステップ目にシーケンシングされた部位の導出

各 $c'(\in C')$ について，i ステップ目にシーケンシングされた部位を導出する．

4.6.4 候補配列の列挙

C を候補配列の集合とし，$C = S$ とする．

$c'(\in C')$ の i ステップ目にシーケンシングされた領域について，i ステップ目にシーケンシングされる塩基 b_i の insertion, deletion を行ったものを C に追加する．i ステップ目には b_i についてのシーケンシングしか行われないため，候補配列を列挙するときは b_i に関してだけ編集を行えば十分である．

$$C = S + \mathrm{edit}(C', i, b_i) \tag{12}$$

各 c' につき insertion で 1 個，deletion で 1 個，計 2 個の候補配列ができる．

4.6.5 候補配列のシミュレーション

全近傍探索法と同様である．

4.6.6 シミュレーション結果のスコア計算

全近傍探索法と同様である．

4.6.7 結果のスコア順ソート

4.6.6 で計算したスコアを元にシミュレーション結果をソートし，スコアがよい (d_c の値が小さい) c のうち上位 N 個のみを S の要素とする．ここで，新たな S の要素がシミュレーション前の S と同じ場合，つまり i ステップ目にシーケンシングされた塩基について編集を行ってもスコアが改善されない場合は i の値を 1 増加させる．

そして 4.6.2 に戻り，同様の探索を繰り返す．

5. マウスのゲノムを用いた誤差訂正実験

前章で提案した 3 つの手法 (AS, SS, TSS) を Perl スクリプトで実装し，マウスの染色体の DNA 配列データを用いて読み取り誤差訂正の性能を測る実験を行った．本章ではこの実験の手順と結果について述べる．

5.1 計算機環境

本実験では東京工業大学学術国際情報センター (GSIC) のスーパーコンピュータ TSUBAME を利用した．

5.2 利用したデータ

UCSC Genome Bioinformatics グループの Web サイトから取得したマウスの 1 番染色体の DNA 配列データのうち，長さ 50 塩基 (50b) と 75 塩基 (75b) の部分列をランダムに 500 本選び，実験用データとして用意した．なお，DNA 配列データに N(A, T, G, C のいずれか判明していない塩基) が含まれる領域は実験用データに採用しないようにした．

5.3 実験の前準備

まず，各実験用 DNA 配列データ s_i をパイロシーケンシングシミュレータの入力としてシミュレーションを行い，読み取り結果 DNA 配列データ s'_i とパイログラムデータ p_i を準備した．ここで不完全塩基対形成発生率と余剰塩基除去ミス発生率のパラメータはシミュレータのデフォルトの値を用いている．これはパイロシーケンシングシミュレータが 2007 年に開発された時に過去の文献を基に決められたものであり，本研究でもこの値を利用して実験を行った．

次に各 s_i と s'_i を比較し，読み取り誤差が発生したもの，すなわち $s_i \neq s'_i$ を満たすものを列挙した．$s_i \neq s'_i$ となるものは，長さ 50b の場合 500 本中 352 本，長さ 75b の場合 500 本中 490 本であった．この中から長さ 50b および 75b ごとにそれぞれランダムに 200 本選び，s'_i および p_i を誤差訂正するための入力データセット D とした．

$$D = \left\{(s'_i, p_i) | s_i \neq s'_i\right\} \tag{13}$$

表 1 手法 1(AS) の実行結果
Table 1 The result of method1(AS).

配列長	正解	準正解
50b	169/200 (84.5%)	200/200 (100%)
75b	161/200 (80.5%)	199/200 (99.5%)

表 2 手法 2(SS) の実行結果
Table 2 The result of method2(SS).

配列長	正解	準正解
50b	168/200 (84%)	196/200 (98%)
75b	161/200 (81.5%)	185/200 (92.5%)

表 3 手法 3(TSS) の実行結果
Table 3 The result of method3(TSS).

配列長	閾値 θ	正解	準正解
50b	0.2	183/200 (91.5%)	183/200 (91.5%)
50b	0.1	186/200 (93%)	186/200 (93%)
50b	0.05	186/200 (93%)	186/200 (93%)
50b	0.01	183/200 (91.5%)	188/200 (94%)
50b	0.005	179/200 (89.5%)	192/200 (96%)
75b	0.2	141/200 (70.5%)	141/200 (70.5%)
75b	0.1	176/200 (88%)	176/200 (88%)
75b	0.05	179/200 (89.5%)	180/200 (90%)
75p	0.01	170/200 (85%)	180/200 (90%)
75p	0.005	171/200 (85.5%)	182/200 (91%)

5.4 実　　験

D の各 s'_i および p_i を誤差訂正スクリプトに入力として与え，読み取り誤差訂正実験を行った．一組の s'_i および p_i ごとに誤差訂正スクリプトのバッチジョブをひとつ立ち上げ，TSUBAME のベストエフォートキューに投入し，200 本の DNA 配列の誤差訂正を分散して行った．各ジョブには 1CPU を割り当てた．

5.5 結　　果

各手法で誤差訂正実験を行った結果を表 1〜3 に示す．ここで,「正解」は誤差訂正スクリプトが提示した誤差訂正結果の配列の候補のうち 1 位のものが元の配列と一致した実験結果の数を,「準正解」は 1 位から 5 位までのうちいずれかが元の配列と一致した実験結果の数を表す．また，全入力数 (200) に対する正解数の割合と準正解数の割合を百分率で示している．さらに，各手法の正解数と準正解数の棒グラフを図 7 と図 8 に示す．ここで，棒グラフの下にあるラベルは各手法とその閾値を表しており，`M{` 手法の番号 `}_T{` 手法 3 の閾値 θ の値 `}` の形で表記している (例：`M1` = 手法 1，`M3_T0.01` = 手法 3, 閾値 $\theta = 0.01$)．

各実験結果を検証する．正解数 (match) と準正解数 (weak match) の項目があるため，それぞれ順に確認していく．

まず正解数 (誤差訂正スクリプトが提示した誤差訂正結果の配列の候補のうち 1 位のもの

Vol.2009-BIO-17 No.12
2009/5/26

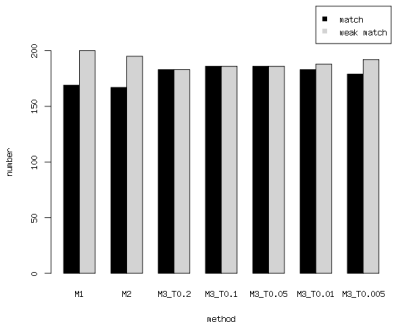

図 7 各手法の正解数 (match) と準正解数 (weak match) 配列長 50b
Fig. 7 The numbers of matches and weak matches(50b).

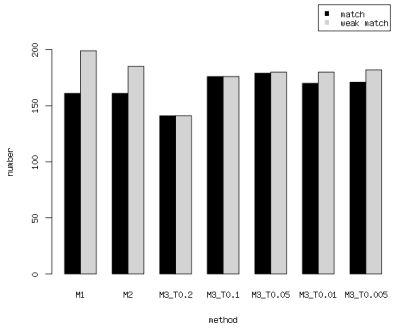

図 8 各手法の正解数 (match) と準正解数 (weak match) 配列長 75b
Fig. 8 The numbers of matches and weak matches(75b).

が元の配列と一致した実験結果の数) については，配列長が 50b と 75b の両方の場合とも手法 3 で閾値 θ が 0.05 付近の場合が一番多く，正解数の割合は約 90% となっている．つまり，パイログラムの差分の二乗が 0.05 を超えたステップのみ探索を行う場合が一番正解数が多くなる．手法 1，2 および手法 3 の閾値が小さい場合と比べて探索が粗いにもかかわらず正解数が多くなる理由は以下のように考えられる．

手法 1，2 および手法 3 の閾値が小さい場合は，読み取り誤差が含まれておらずパイログラムの差分が十分小さい部位についても候補配列の列挙とシミュレーションを行う．シミュレータはモンテカルロ法に基づきパイロシーケンシングのシミュレーションを行うため，シミュレータが出力するパイログラムの値は実際のパイロシーケンシング法でもそうであるように確率的に若干変動するようになっている．そのため，まれに元の配列と類似しているが異なる配列のシミュレーションを行ったときに，その配列のパイログラムのスコアが元の配列のパイログラムのスコアよりもよくなる場合がある．

探索をより細かくして候補配列の探索数を増加させると探索の途中でそのような事象が発生する確率が上がるのに対して，手法 3 の閾値が大きい場合はパイログラムの差分が十分に大きい部分のみ候補配列の列挙とシミュレーションを行うため，元の配列と類似しているが違う配列をシミュレーションする頻度が少なくなり，準正解になる可能性は減少する．実際，読み取り誤差が含まれているステップのパイログラムの差分は比較的大きな値となるため (DNA 配列や乱数により変動するため，どの程度の値なのかを一概に決めることはできない)，手法 3 の閾値が大きい場合の方が読み取り誤差の入っている領域のみをピンポイントで探索することができ，余分な候補配列の探索が少なくなる．このことから，元の配列とは違うにもかかわらずパイログラムのスコアがよい配列を見つける確率が減り，正解数が多くなるものと考えられる．このような性質は，逆に正しくシーケンシングされ誤差が含まれていない配列を誤って編集し，新たな誤差を含めてしまうといったことを防ぐことができるという点でも重要である．ただし，閾値の値が 0.1 を超え大きくなりすぎると，本来誤差訂正すべき箇所も探索をスキップしてしまう場合が増えるため，正解数は減少する．

次に準正解数 (1 位から 5 位までのうちいずれかが元の配列と一致した実験結果の数) については，正解数の場合と逆で手法 1，手法 2，および手法 3 の閾値が小さい場合の方が多くなっている．これは先程の正解数が多い場合と同様に考えれば納得のいく結果である．

各手法の性能の評価指標として正解数と準正解数があるが，実際に誤差訂正した時に 2 位以下の結果を元の配列の候補として採用することは実用上あまり考えられない．このため正解数が多い方が誤差訂正の精度が高いものとして考えると，配列長が 50b と 75b の両方の場合で**手法 3 の閾値 $\theta = 0.05$** の場合が一番正解数が多く，性能がよいと判断できる．

5.6 実行時間

各手法で誤差訂正実験を行った時の実行時間を表 4〜6 に示す．それぞれ各手法・配列長での実行時間 (秒) の分布の最小値，第一四分位数，中央値，平均，第三四分位数，最大値を示している．また，実行時間の分布を図 9 と図 10 に箱ひげ図で示す．

実行時間について検証する．表 4〜6 と図 9〜10 とから，入力配列により実行時間にばらつきがあるものの，手法 1 よりも手法 2，手法 2 よりも手法 3 の方が実行時間が圧倒的に短く，より高速に読み取り誤差訂正ができることが分かる．また，手法 3 の各閾値については，若干の誤差はあるものの閾値が大きい方が実行時間が短くなると考えられる．ただし，閾値が十分に大きくなると実行時間の差があまり見られなくなる．

以上より，閾値によるが**手法 3** を用いた場合が一番高速に読み取り誤差訂正できると判断できる．

6. 結　　論

本研究ではパイロシーケンシング法で得られた配列の読み取り誤差を訂正する手法を提案・実装し，パイロシーケンシングシミュレータで用意した DNA 配列データの誤差訂正実験をすることで性能の評価を行った．その結果，手法 3：閾値付き順次探索法 (Threshold

Vol.2009-BIO-17 No.12
2009/5/26

表 4 手法 1(AS) の実行時間 (秒)
Table 4 The runtime(sec) of method1(AS).

配列長	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
50b	309	1,852	2,832	3,523	4,411	12,830
75b	2,584	8,226	11,970	12,220	15,580	26,800

表 5 手法 2(SS) の実行時間 (秒)
Table 5 The runtime(sec) of method2(SS).

配列長	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
50b	176	309	439	502	626	1,469
75b	439	980	1,329	1,618	1,950	9,067

表 6 手法 3(TSS) の実行時間 (秒)
Table 6 The runtime(sec) of method3(TSS).

配列長	閾値 θ	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
50b	0.2	0.4	1.5	3.2	5.9	8.8	31
50b	0.1	0.4	1.5	3.1	5.5	7.2	25
50b	0.05	0.6	1.8	4.4	10	11	92
50b	0.01	0.6	3.6	8.7	19	22	209
50b	0.005	0.5	4.4	8.7	13	18	116
75b	0.2	1.9	11	21	24	36	64
75b	0.1	1.8	13	22	27	39	113
75b	0.05	1.7	16	31	43	48	253
75b	0.01	1.8	29	58	75	94	570
75b	0.005	4.0	30	51	61	84	233

Sequential Search method) を閾値 $\theta = \mathbf{0.05}$ で用いた場合が性能がよく，配列長 50b の DNA 配列の約 93%を，75b の DNA 配列の約 90%を誤差訂正することができた．454 Life Sciences 社などのシーケンサーから実際に得られた DNA 配列データについての誤差訂正実験は実施できなかったが，シーケンサーに対応するシミュレータを用意することができれば今回提案した手法で DNA 配列の読み取り誤差訂正を行うことができるのではないかと考えられる．

また，本実験では誤差を含む配列の生成と誤差訂正時の両方で同じ処理パラメータを用いたが，実環境においては，訂正時に使う処理パラメータは，シーケンサーの動作状況をモニタしながら推定値として与えるしかなく，実際のシーケンサーの出力とシミュレーション結果との間に誤差を生じる可能性がある．実際のシーケンサーからどのようにして定期的に処理パラメータの推定値を得るか，また処理パラメータ推定値に系統誤差が入った時にどのようなふるまいをするのか等については，本研究の今後の課題である．

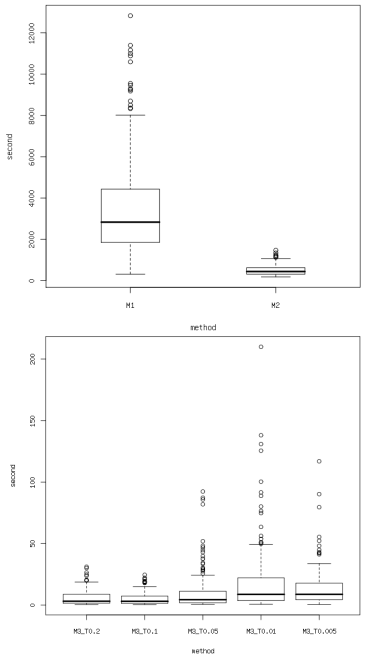

図 9 各手法の 200 個の入力に対する実行時間の分布 (配列長 50b)
Fig. 9 The deviation of the runtime(50b).

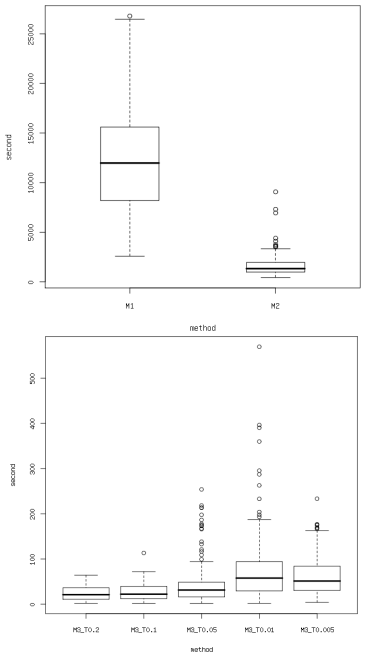

図 10 各手法の 200 個の入力に対する実行時間の分布 (配列長 75b)
Fig. 10 The deviation of the runtime(75b).

参 考 文 献

1) M. Ronaghi, M. Uhlen, P. Nyren: “A Sequencing Method Based on Real-Time Pyrophosphate”, *Science* 281:363-365 (1998).
2) Mostafa Ronaghi: “Pyrosequencing Sheds Light on DNA Sequencing”, *Genome Research*, 11:3-11 (2001).
3) Helmy Eltoukhy, Abbas El Gamal: “Modeling and Base-Calling for DNA Sequencing-By-Synthesis”, *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, 2:II-II (2006).
4) Yutaka Akiyama: “SimPyro: Pyrosequencing simulation software for analyzing random process with millions of reactions. Poster presentation”, *The 2nd International Workshop on Approaches to Single-Cell Analysis*, Tokyo (2007).