ゲノム解析に用いるDPマッチングの分割統治法による高速化

中川 匠[1) 吉原 郁夫[2) 山森 一人[3) 安永 守利[4)

A Divide-and-Conquer Method of Dynamic Programming for Genome Informatics

Takumi NAKAGAWA Ikuo YOSHIHARA Kunihito YAMAMORI
Moritoshi YASUNAGA

ABSTRACT

Alignment based on DP-matching is used to extract unknown feature pattern embedd ed in genome sequence. To extract feature pattern with base length n, complete set of similarity of c andidate pattern with base length $N = 2, 3, \cdots, n$ is required. When extending base length n, number of candidate pattern and execution time i ncrease exponential order $O(4^{n+1})$. This paper propose a new faster method of extracting feature pattern by reusing similartiy which is calculated past step. The method enable extracting feature pattern with base length $n = 10$ to speed up as much as 9 times than conventional method.

Key Words:
Genome Informatics, Dynamic Programming, Divide-and-Conquer , Alignment, Feature Extraction

1 はじめに

ヒトゲノム計画をはじめとするゲノム（genome）解析プロジェクトの進展により，さまざまな生物種のゲノム情報が大量に蓄積されている[1)]．現在，これら大量のゲノム情報からその働きや意味を明らかにすることが重要な課題になっている．

遺伝子とはタンパク質をコードしているDNA(Deoxyribo Nucleic Acid:デオキシリボ核酸)塩基配列の一部であり，実際に遺伝子が読み取られタンパク質が作られることを遺伝子の発現という．この過程において発現調節要素と呼ばれる部分塩基配列が重要な役割を果たしている事がわかっている．現在までにいくつかの発現調節要素が発見されているが，いまだに発見されていない発現調節要素もたくさん存在すると考えられている．これら未知の発現調節要素を発見することは遺伝子発現のメカニズムを解明するうえで重要な意味を持つ．そこでこれまで発現調節要素の候補となる特徴的な部分塩基配列 (特徴パターン) を自動的に抽出する研究を行ってきた[2)]．

特徴パターンの抽出にあたっては考慮すべき問題が2つある．1つは特徴パターンは常に同じ部分塩基配列ではなく通常は多少変化して存在するということである．もう1つは抽出しようとする特徴パターンの長さや塩基配列，出現位置が未知であると言う問題である．そこでアラインメントを用いた探索を総当たりに行う必要がある．

ゲノム解析の分野ではアラインメントを効率的に計算する方法として動的計画法 (Dynamic Programming:DP) によるDPマッチングが用いられる[3)]．しかし特徴パターン抽出では総当たりでアラインメントを求めなければならず，DPマッチングの計算量が増大し特徴パターン抽出に膨大な時間がかかる．そこで分割統治法の考えを用いてDPマッチングに要する計算量を削減する方法を提案する．それは抽出をおこなう候補パターンを短い部分配列に分割しDPマッチングを行う方法である．この手法を用いることで候補パターンの長さが短くなりDPマッチングの計算量が削減される．また一度計算したアラインメントスコアを保存し再利用することでDPマッチングの実行回数自

[1)]情報工学専攻学生
[2)]情報システム工学科教授
[3)]情報システム工学科助教授
[4)]筑波大学電子・情報工学系教授

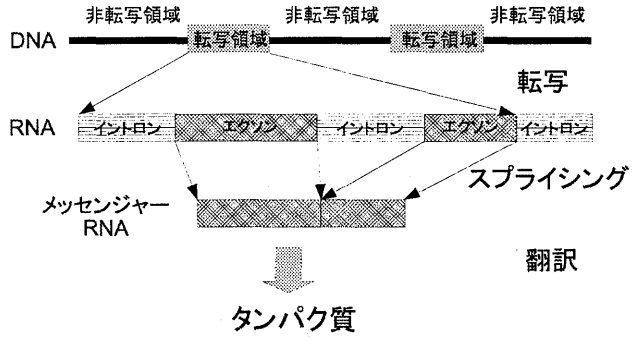

図. 1 遺伝子の発現

体を削減することができる.

2 遺伝子の発現と特徴パターン

2.1 ゲノムの構造と特徴

ゲノムとはある一つの生物が持っている完全な染色体のことで，染色体は一連の DNA 分子から構成されている[4]. DNA は 2 本のポリヌクレオチド鎖がらせん状に巻きあった構造である. ポリヌクレオチド鎖はヌクレオチド (nucleotide) と呼ばれる分子が鎖状につながってできている. ヌクレオチドは 1 個の糖，1 個のリン酸，1 個の塩基から成り立っており，塩基の部分だけが異なる 4 種類が存在する. 異なる塩基の部分はアデニン (adenine; A)，シトシン (cytosine; C)，グアニン (guanine; G)，チミン (thimine; T) の 4 種類があり，これら 4 つの塩基の配列によって DNA は表される.

DNA は多数の異なる遺伝情報を担う区画 (遺伝子) にわかれており，実際に遺伝子が読み取られタンパク質が作られることを遺伝子の発現という.

遺伝子発現は図 1 に示すメカニズムによって行われる[5]. まず DNA の塩基配列が RNA に移し変えられる. この過程は転写と呼ばれる. 次に，スプライシングという作業によってイントロンが取り除かれメッセンジャー RNA（mRNA）がつくられる。mRNA の塩基配列はアミノ酸をコードしており，そのアミノ酸が順に結合してタンパク質が生成される. この過程は塩基配列から質的にまったく異なるアミノ酸配列への変換であり，翻訳と呼ばれる.

2.2 発現調節要素と特徴パターン

DNA 塩基配列の中で転写される領域を転写領域といい，その始まりの位置を転写開始点という. 遺伝子の近傍にはいくつかの特徴的な短い配列が存在し，これらの適切な組合せがシグナルとなり転写開始点が決定されている[6]. これらは発現調節要素とよばれ，TATA ボックスや CCAAT ボックスなどが良く知られ

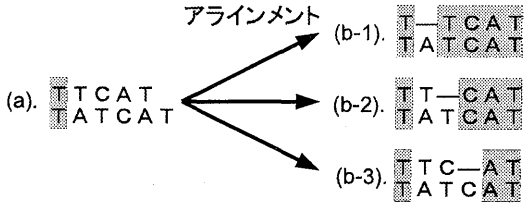

図. 2 アラインメント

ており，まだ発見されていないものもたくさん存在すると考えられる.

このような未知の発現調節要素の発見は遺伝子発現のメカニズムを解明する上で重要である. しかし大量に存在する DNA 塩基配列データの中から未知の発現調節要素を発見する事は非常に困難である.

これまでに見付かっている発現調節要素の特徴から，転写開始点を基準として一定の位置付近で頻出する塩基部分配列が発現調節要素の候補パターンと考えられる. そこで発現調節要素の候補となるパターンを計算機を用いて抽出し，それを元に生物学的な考察を加えて発現調節要素を発見する. この発現調節要素の候補となるパターンを特徴パターンと呼ぶ.

発現調節要素は常に同じ塩基配列で存在するとは限らず，通常は多少変化した塩基配列として存在することもある. そのため類似した塩基配列も特徴パターンとして抽出できるようなパターンマッチングを用いる必要がある.

また特徴パターンを抽出する上でその長さや出現位置に関する先験的知見は全くなく，塩基配列のどの位置にどんな特徴パターンが現れるかわからない. そのため総当たりのパターンマッチングを行わなければならない. 長さ N までの特徴パターンを抽出するには，長さ N までの全通りの塩基配列を候補パターンとして用意し DNA 塩基配列データの全ての位置でパターンマッチングする必要がある.

2.3 アラインメント

類似した塩基配列を特徴パターンとして抽出するため，アラインメントを用いたパターンマッチングを利用する. アラインメントとは 2 つの塩基配列に対して塩基の間にギャップと呼ばれるすき間を入れ，塩基配列の塩基の対応関係を求めたものである. [7].

図 2 の (a) は 2 つの類似した塩基配列を上下に並べたものである. また同図の (b-1)，(b-2)，(b-3) はアラインメントの例を示しており，塩基配列中に挿入されている'–' はギャップを表す. このとき網掛けが施されている部分は，上下の塩基配列で塩基が一致している

表. 1 類似度計算テーブル

状態	類似度
一致	+1
不一致	-1
ギャップ	-2
連続ギャップ	-1

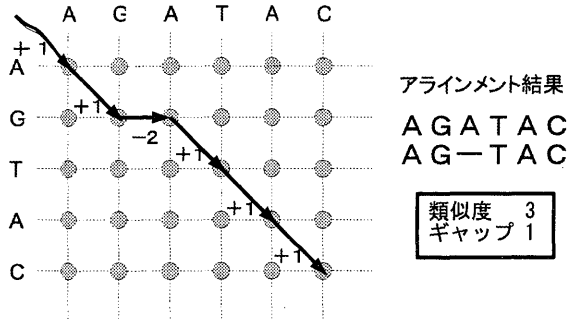

図. 3 DP マッチング

ことを表す．塩基が一致している部分が多いほどその2つの塩基配列はより似ている塩基配列であると考えられる．同図の (a) では塩基配列の上下で一致している塩基は1つに過ぎないが，ギャップを挿入する事で (b-1) では5つの塩基が縦に一致している．このようにギャップを挿入することで二つの塩基配列がどれだけ一致しているかをもとめることができる．

ここで対応づけられた塩基の一致・不一致を表1のように定義し，これをもとに類似度を求めることにする．

表1の Match・Mismatch はそれぞれ対応づけられた塩基が一致・不一致であるときの類似度である．また Open gap はギャップが挿入されたときの類似度であり，Extended gap はギャップが延長されたときの類似度である．これを用いると図2の (b-1) の2つの塩基配列で対応する各塩基の類似度の和は3である．この類似度の和をアラインメントスコアと呼ぶ．

図2の (b-1) 以外にも (b-2)(b-3) のように，アラインメントはギャップの位置を変えることでいくつも生成できる．このときもっともアラインメントスコアが高くなるアラインメントを最適アラインメントと呼ぶ．特徴パターンの抽出には最適アラインメントによるパターンマッチングを用いる．

3 特徴パターン抽出とその高速化

3.1 DP マッチングによる最適アラインメントの導出

最適アラインメントを効率的に計算する方法として DP マッチングが一般的に用いられる．

図3は二つの塩基配列 AGATAC と AGTAC の最適アラインメントを DP マッチングにより求める例である．ア

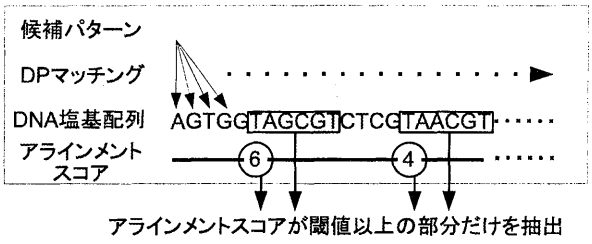

図. 4 アラインメントを用いた部分配列の抽出

ラインメントスコアの計算には表1を用いている．DP マッチングでは最適アラインメント問題を図3のような2次元の格子状にして表す．この図の横方向は1つ目の塩基配列に対応し，縦方向は2つ目の塩基配列に対応する．また右下方向の矢印は対応する塩基の類似度が対応づけられていて，横方向と縦方向の矢印にはギャップの類似度が対応づけられている．そして，左上の点から右下の点までの類似度の和が最大となる経路を求めれば，最適アラインメントが得られる．この例では最適アラインメントとして AGATAC と AG-TAC が得られ，そのときのアラインメントスコアは3となる．

3.2 従来法による特徴パターンの抽出

従来の研究で用いられてきた特徴パターンの抽出法について説明する．

DNA 塩基配列中に含まれている候補パターンを抽出するために，候補パターンと DNA 塩基配列の DP マッチングを行う．図4に示すように，DP マッチングは DNA 塩基配列の先頭から1塩基ずつずらしながら，DNA 塩基配列の終端に達するまで局所的に行う．その中からアラインメントスコアが一定の閾値 *Threshold* を越えた部分の位置とスコアを保存する．このようにして DNA 塩基配列から候補パターンと同じか非常に似ている部分配列が含まれている位置とそのアラインメントスコアを抽出する．長さ N の特徴パターンを抽出するときは，長さ N のありうる全ての塩基配列を候補パターンとして以上の操作を行う．

Threshold は候補パターンを特徴パターンとして抽出する最低のアラインメントスコアであり，以後は特徴抽出閾値と呼ぶ．いま候補パターンの長さを l，アラインメントに含むことが許されるギャップの数を G とする．またアラインメントにおける塩基の一致による類似度を *Similarity(match)*，ギャップ挿入による類似度を *Similarity(opengap)* とする．このとき *Threshold* は式 (1) のように表される．

$$Threshold = Similarity(match) \times (l - G) - Similarity(opengap) \times G \quad (1)$$

DNA塩基配列データは1つではなく大量に存在し，特徴パターンを抽出するには他のDNA塩基配列データにおいても頻出する部分配列を見つけ出さねばならない．そこで全てのDNA塩基配列データに対し候補パターンかそれに類似した部分配列が含まれている位置とそのアラインメントスコアを抽出する．そして抽出された情報をもとに特定の位置に頻出するパターンを見つけ出し，特徴パターンとする．

しかし先に述べたように生物進化の過程で塩基の挿入・欠損・変異などが起こるため，なんらかの意味があるパターンが存在したとしてもそれが全く同じ位置には存在せず，少しずれた位置に存在することもある．そこで一定の範囲内に類似度を相対的に表す2種類のスコア，$ScoreM$ と $ScoreT$ が提案されている[2]．M個のDNA塩基配列データが存在する時，塩基配列の位置 i におけるそれぞれのスコアを求める式を以下に示す．

$$Score_M(i) = \max_{i-M \le j \le i+M} (Similarity(j))$$

$$Score_T(i) = \sum_{j=i-M}^{i+M} (Similarity(j))$$

これらのスコアをもとに塩基配列の特徴を抽出する．

3.3 特徴パターン抽出に必要な計算量

塩基長 L のDNA塩基配列データ M 個から長さ l の特徴パターン抽出を行うことを考える．比較する2本の文字列の長さが I,J のときDPマッチングの実行に必要な計算量 c_1 は式(2)のようになる[8]．

$$c_1 = \frac{I}{2}(J+1)(J+2) \quad (2)$$

ここでアラインメントに含まれるギャップの数を g とした時，$0 \le g \le G$ を特徴パターンとして抜き出す条件とする．候補パターンの長さが l の時，高々 $l+G$ のDNA塩基配列と比較すれば良い．そこで式(2)の $I = l+G$，$J = l$ として，長さ l の候補パターンのDPマッチングに必要な計算量 c_2 を式(3)に表す．

$$c_2 = \frac{l+G}{2}(l+1)(l+2) \quad (3)$$

候補パターンの長さが l の時，考えられる候補パターンの数は 4^l である．長さ L の塩基配列データの全ての位置で全ての候補パターンとDPマッチングを行うため，DPマッチングの実行回数は $4^l L$ となる．また M 個のDNA塩基配列データについてこれを行うので，DPマッチングを実行する回数は $4^l LM$ となる．これらのことから長さ l の特徴パターンを抜き出すために必要な計算量 c_3 は式(4)のようになる．

$$c_3 = 4^l LM\frac{l+G}{2}(l+1)(l+2) \quad (4)$$

長さ N までの特徴パターンを抽出するためには，長さ2から N までの特徴パターン抽出を行わなければならない．よって長さ L のDNA塩基配列データ M 個から長さ N の未知の特徴パターンを抽出するのに必要な計算量 c_4 は式(5)で表される．

$$c_4 = \sum_{l=2}^{N} 4^l LM\frac{l+G}{2}(l+1)(l+2) \quad (5)$$

3.4 分割統治法を用いた高速化

従来法では抽出したい特徴パターンの長さが延びるに従いDPマッチングにかかる計算量が指数関数的に増大する．そのため長い特徴パターンの抽出には膨大な時間が必要となる．そこで分割統治法の考えを用い計算量を削減する方法を提案する．

3.4.1 候補パターンの分割

提案手法は候補パターンを分割し，分割された部分配列についてDPマッチングを行う方法である．長さ l の候補パターンを途中で分割し，$l = a+b$ となる長さ a と b の部分配列にする．分割してできた部分配列についてそれぞれDPマッチングを行い，それぞれのアラインメントスコアを計算する．

候補パターンを分割してDPマッチングを行う時の計算量を求める．長さ l の候補パターンのDPマッチングに必要な計算量は式(3)で表される．そこで候補パターンを途中で2分割し，$l = a+b$ となる長さ a と b の部分配列 A,B にする．提案手法は分割されてできた部分配列についてDPマッチングを行い，その結果をもとにアラインメントスコアを計算する方法である．この時の計算量 c_5 は式(6)で表される．

$$c_5 = \frac{a+G}{2}(a+1)(a+2) + \frac{b+G}{2}(b+1)(b+2) \quad (6)$$

候補パターンの数は全部で 4^l 個存在し，長さ L のDNA塩基配列データの全ての位置でDPマッチングを行う必要がある．よって長さ l の特徴パターン抽出全体で必要となる計算量 c_6 は式(7)で表される．

$$c_6 = 4^l LM\left\{\frac{a+G}{2}(a+1)(a+2)+\frac{b+G}{2}(b+1)(b+2)\right\} \tag{7}$$

3.4.2 候補パターン全体のDPマッチング

部分配列AとBのアラインメントスコアをそれぞれ$score(A)$と$score(B)$とする．

特徴パターン抽出の条件としてG個のギャップ挿入が許されているとしたとき，分割した候補パターンに許されるギャップの数もGとなる．そこでギャップ挿入によるペナルティを$P(G)$，部分配列A,Bがとりうる最大のアラインメントスコアを$MAXscore(A)$,$MAXscore(B)$とすると，式(8)の条件が満たされたとき候補パターン全体のDPマッチングを実行する．

$$\begin{cases} score(A) \geqq MAXscore(B)-P(G) \\ score(B) \geqq MAXscore(B)-P(G) \end{cases} \tag{8}$$

3.4.3 アラインメントの保存と再利用

候補パターンを部分配列に分割しアラインメントスコアを求めた後，次の候補パターンについても同じ様に分割してアラインメントを求める．

このとき過去に計算したアラインメントを再利用できる場合がある．いま長さ6の特徴パターン抽出を行うことを考える．このとき候補パターンを**AAAAAA**, **AAAAAG**, **AAAAAC**, ⋯, **TTTTTC**, **TTTTTT**と変化させ，前後3塩基の部分配列に分割してDPマッチングを行うとする．このとき前半の部分配列について何度も同じアラインメントを計算することになる．そこで一度計算した部分配列のアラインメントを保存し再利用することにする．

アラインメントの保存と再利用による計算量の変化を求める．長さlの候補パターンを$l=a+b$となる長さaとbの部分配列に分割してDPマッチングを行うとする．このときの計算量c_7は式(9)のように求められる．

$$c_7 = 4^a LM\frac{a+G}{2}(a+1)(a+2)+4^b LM\frac{b+G}{2}(b+1)(b+2) \tag{9}$$

3.5 並列処理による高速化

第3.4節では候補パターン長が延びるに従い指数関数的に増加するDPマッチング呼び出し回数をアルゴリズム的に減少させ，高速化する方法を提案した．本研究ではこれに加え処理の並列化を行うことでさらなる高速化を図る．

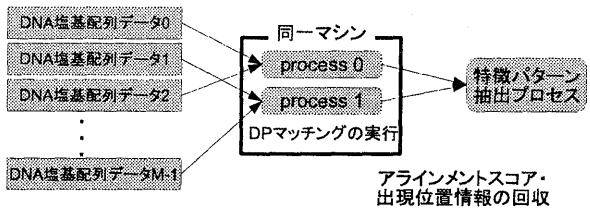

図. 5 DPマッチングの並列処理

表. 2 実験に使用した計算機の仕様

ハードウエア

CPU	Intel Xeon 3.2GHz DualCPU
2次キャッシュ	2MB
メインメモリ	4GB

ソフトウエア

OS	Redhat Enterprise Edition Linux WS 3.0
コンパイラ	Intel C++ Complier Linux Version 9.0

処理は図5のように行われる．

まずDPマッチング専用のプロセスであるprocess[0]とprocess[1]を生成する．これらのプロセスに均等にDNA塩基配列データを与え，候補パターンとのDPマッチングを行う。

特徴パターン抽出プロセスは全てのDPマッチング専用プロセスからDPマッチングの結果を受け取り，特徴パターン抽出処理を行う．

4 提案手法の性能評価

4.1 実験データと実験条件

特徴パターン抽出実験に用いるDNA塩基配列データには細胞性粘菌の非転写領域を用いる．細胞性粘菌は生活様式が特徴的な生物で，培養が容易であることなどから細胞分化・形態形成のメカニズム，細胞集合，細胞間通信などの生理・生化学，遺伝学，微細構造学の有力な研究材料として用いられている[9]．なお使用する細胞性粘菌のデータは筑波大学の漆原研究室によって独自に作成されたものである．今回は細胞性粘菌の塩基配列から転写開始点の上流2000塩基を切り出したデータ332個を用意した．特徴パターンに含まれるギャップの数Gは2とし，特徴抽出閾値$Threshold$は式(1)より求める．また類似度の計算には表1を用いる．実験に用いた計算機の詳細は表2のとおりである．

4.2 実験

細胞性粘菌の塩基配列データから長さ10の特徴パターン抽出実験を行う．特徴パターン抽出は従来手法を用いたプログラムと提案手法を用いたプログラムの両方で行い，その実行時間とDPマッチング処理の呼び出し回数を比較する．なお提案手法を用いたプログラムでは候補パターンの分割数を2に設定し，部分配

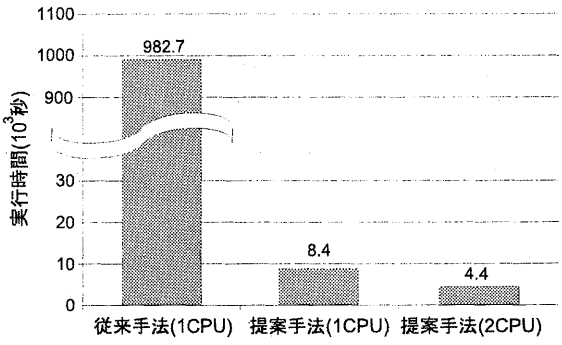

図. 6 手法による実行時間の変化

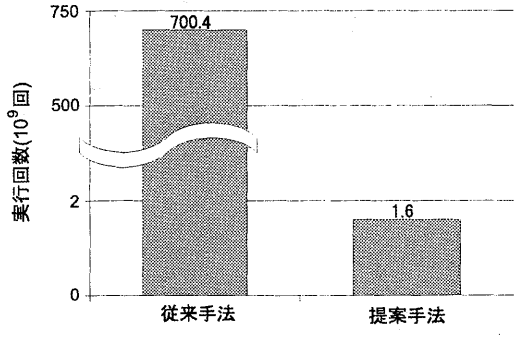

図. 7 提案手法による DP マッチング実行回数の変化

列の長さは 5 とする.

図 6 は特徴パターン抽出に要した時間を示すグラフである. なお提案手法による実行時間は並列化を行わずに 1 CPU で実行した場合と, プロセスを 2 つに分け 2CPU を用いた場合の 2 つを測定している.

図 7 は従来手法と提案手法のプログラム実行時に呼び出された DP マッチングの回数を示す.

4.3 考察

従来手法を用いたプログラムの実行時間は 982.7×10^3 秒であるが, 1CPU を用いた提案手法では 8.4×10^3 秒で実行を終了した. このことから分割統治法を用いたアルゴリズムにより 117 倍の高速化を達成したと考えられる. また並列化処理により 2CPU を用いた提案手法による実行時間は 4.4×10^3 秒でありおよそ 223 倍の高速化を達成した. これらのことから提案手法の有効性が確認された. またプログラム実行時に呼び出される DP マッチングの回数は従来手法では 7000 億回以上であったが, 提案手法ではおよそ 16 億回に削減されている. DP マッチング呼び出し回数の削減率はおよそ $\frac{1}{430}$ である. DP マッチングの呼び出し回数が $\frac{1}{430}$ となったにもかかわらず実行時間がおよそ 117 倍にしかならなかった. これはは, 提案手法においてアラインメントの保存のための処理が追加されたためだと考えられる.

なお並列化により実行時間は約 $\frac{1}{2}$ に短縮されており, 利用 CPU の増加がそのまま実行時間の短縮つながっていることがわかる. これは提案手法で用いた並列化がプロセス複製によるものであり, 一般的な並列計算で起こる通信によるオーバーヘッドが発生しないことが理由と考えられる.

5 おわりに

本研究は特徴パターン抽出に必要な処理の主要部分である DP マッチングの計算量を削減するため, 候補パターンを分割した DP マッチングとアラインメントスコアの再利用による手法を提案した. またアルゴリズムによる高速化に加えて, 並列処理による高速化手法を提案した.

これにより長さ 10 の特徴パターン抽出実験では, 従来手法に対して 117 倍の高速化を達成した. さらに処理の並列化を用いることで, 従来手法に対しておよそ 223 倍の高速化を達成した. このことから本手法を用いることで特徴パターン抽出が高速に行われることを示した.

今後の課題として高速化により可能となった長い特徴パターンの抽出を行い, なんらかの特徴パターンを抽出することが挙げられる.

参考文献

[1] 林田秀宜: "DNA データベース", 情報処理学会誌, Vol. 31, No. 7, pp. 875–877 (1990).

[2] T. ONITANI, I. YOSHIHARA, K. YAMAMORI and M. YASUNAGA: "Extraction of feature patterns embedded in non-transcribed region of dictyostelium discoideum", The 5th International Conference on Simulated Evolution and Learning(SEAL'04), Vol. 123, (2004).

[3] D. W.Mount, 岡崎康司, 坊農秀雅: バイオインフォマティクスゲノム配列から機能解析へ, メディカル・サイエンス・インターナショナル 2002.

[4] 菊池韶彦, 榊佳之, 水野猛, 伊庭英夫（編）: 遺伝子 第 7 版, 東京化学同人 2002.

[5] 美宅成樹, 金久實: ヒトゲノム計画と知識情報処理, 培風館 1995.

[6] 村松正實, 木南凌（編）: ヒトの分子遺伝学 第 3 版, メディカル・サイエンス・インターナショナル 2005.

[7] 岡崎康司, 坊農秀雅: ゲノム情報はこう活かせ!, 羊土社 2005.

[8] 上坂吉則, 尾関和彦: パターン認識と学習のアルゴリズム, 文一総合出版 1990.

[9] G. Glöckner, L. Eichinger, K. Szafranski, et al.: "Sequence and analysis of chromosome 2 of dictyostelium discoideum", Nature, Vol. 418, (2002).