統計数理(2008)
第 56 巻 第 2 号 169–184

特集「データマイニングと統計数理」
[原著論文]

遺伝子発現データからの接尾辞木に基づく疑似バイクラスタ抽出

難波 徹郎[†]・原口 誠[†]・大久保 好章[†]

(受付 2008 年 1 月 4 日；改訂 2008 年 4 月 15 日)

要　　旨

本研究では，遺伝子発現データをはじめとする，時系列データを対象としたバイクラスタリングについて考察する．時系列性を考慮したバイクラスタリングでは，通常，データ行列の行と列を同時にクラスタリングすることで，ある連続した時間区間において同様の変動を示す個体群を極大バイクラスタとして抽出する．特に，接尾辞木を利用することで，これらはデータ行列サイズの線形オーダで抽出可能なことが知られている．本研究ではこの枠組を拡張し，生物学的により興味あるバイクラスタの抽出を目指す．具体的には，疑似バイクラスタの概念を導入し，ある時間区間まで同様な発現変動を示す遺伝子群が，その後枝分かれをして異なる変動を示す様子を捕まえることを試み，こうした疑似バイクラスタを接尾辞木を用いて抽出する多項式時間アルゴリズムを提案する．ホヤの遺伝子発現データを用いた計算機実験により，期待した様子が観察可能な疑似バイクラスタが得られることを確認する．

キーワード：バイクラスタリング，疑似バイクラスタ，接尾辞木，遺伝子発現データ，時系列データ．

1. はじめに

近年，特定の DNA 配列を検出するプローブをスライドガラスなどの基板上に配置し，細胞中で発現している遺伝子を検出する DNA マイクロアレイ技術により，数千から数万個の遺伝子を一度に測定することが可能となった．これにより，大量の発現データを短期間で解析し結果を理解することが重要な課題となっている．特に，生物の各発生段階での変動を測定した**遺伝子発現データ**の解析においては，『同様の発現変動を示す遺伝子群の抽出』が重要なタスクとなる(例えば Masuda, 2005)．そのために，データマイニング分野では，類似した遺伝子群の抽出を目的とした**クラスタリング**(Gan et al., 2007; Jain et al., 1999)が行なわれて来たが，遺伝子発現時系列データでは，特定の遺伝子が特定の時間ステージにおいて発現することが多く，従来のクラスタリング手法を用いた遺伝子クラスタリングでは有用な結果を得にくい問題があった．この問題を解決するために，データ行列の行と列を同時にクラスタリングする，すなわち，遺伝子群と，ある時間区間を同時に切り出す**バイクラスタリング**(Madeira and Oliveira, 2004)の適用が試みられている(Cheng and Church, 2000; Madeira and Oliveira, 2005)．一般に，バイクラスタリングとは，データ行列の行と列の任意の一部を同時に抽出する処理を指すが，特に遺伝子発現データのバイクラスタリングにおいては，『生物学的な発現プロセスは連続した

[†] 北海道大学大学院 情報科学研究科：〒060–0814 北海道札幌市北区北 14 条西 9 丁目

時間の発現値の増減で示される』との仮定から，連続した列を持つバイクラスタに限定した解析が行なわれる．

この様に限定された問題に対して，Madeira and Oliveira (2005) では，**接尾辞木** (Weiner, 1973; Gusfield, 1997) に基づく線形時間バイクラスタリングアルゴリズムが提案されている．そこでは，各遺伝子の発現データを符号化してひとつの文字列に変換し，その**一般化接尾辞木**をもとに，それぞれに共通な部分文字列を探索することで極大なバイクラスタを効率良く抽出する．しかし，抽出される極大バイクラスタは，行あるいは (連続した) 列において，多くの重複が観測される．遺伝子発現データを考えた場合，こうしたクラスタ間の列の重複は，ある一定の時間区間においては類似した発現変動を示すが，それ以外では異なる変動を示す遺伝子群の存在を意味する．例えばこのことは，ある遺伝子群が，同様の発現変動をした後，枝別れをして，異なる部位の形成に関与した様子を示唆しているかもしれない．この様な観察は，生物学的に非常に興味深い貴重な知見を与える可能性があり，こうした変動パターンを積極的に自動抽出することは極めて意義深い．

そこで本研究では，生物の遺伝子発現時系列データの解析に際し，重複が観測される極大バイクラスタをひとつにまとめ上げた**疑似バイクラスタ**の概念を導入し，その抽出を試みる．疑似バイクラスタは接尾辞木のリンク構造を利用することで容易に抽出可能であることを示し，その多項式時間アルゴリズムを与える．また，計算機実験により，疑似バイクラスタの抽出が，上述した遺伝子群の興味深い挙動の解析に有用となり得ることを確かめる．

データ行列における行と列のクラスタを同時に抽出する枠組は**共クラスタリング**とも呼ばれ，文献 Dhillon et al. (2003) 等で議論されている．そこでも，列方向のクラスタに対して連続性を要請することで，ある時間区間の発現変動のみに注目した遺伝子クラスタを抽出することは可能であるが，クラスタリングの善し悪しは，クラスタリング後の 2 次元の分割が持つ相互情報量，もしくは，クラスタリング前後のクロス表間の Kullback-Leibler 情報量によって評価される．これは，ひとつのクラスタリングにおける複数のバイクラスタを平均的に評価することを意味している．一方，本研究で抽出対象とする疑似バイクラスタは，複数のバイクラスタの重複を重視したものであり，目標とする疑似バイクラスタを構成する要素バイクラスタは，それ以外の部分の評価とあわせた平均評価においては最適とは限らない．すなわち，共クラスタリングの意味における最適なバイクラスタリングと，本研究で目標とするバイクラスタとは異なる．同様のことは，説明変数と被説明変数間の相互情報量劣化と説明変数の情報圧縮のバランス問題の解決を意図する情報ボトルネック (Tishby et al., 1999) についても言える．

接尾辞木は，高速な文字列検索や部分文字列の列挙等を可能とするデータ構造であり，自然言語処理はもちろんのこと，近年はバイオインフォマティクス等の分野でも利用されている (例えば，Kurtz et al., 2001)．そこでは，パターンの出現回数を数える，あるいは，繰り返し出現する (頻出する) パターンを抽出するといった用途に用いられることが多く，主に**頻度**に注目したパターン抽出が行なわれる．これに対し，Madeira and Oliveira (2005) では，接尾辞木がこうした頻度に基づくパターン抽出のみならず，バイクラスタとしての**極大性を考慮したパターン**の効率的な抽出においても有効に利用出来ることが示された．これを基礎として，本研究では，**極大パターン間の重複**に注目した新たなパターン抽出問題を提示し，その接尾辞木に基づく計算アルゴリズムを与える．抽出対象が単なる頻出パターンではなく，重複を重視した新たなパターンであること，および，接尾辞木がそうした新たなパターン抽出においても有用なデータ構造であることを示した点において，従来研究との明白な差異を認めることが出来る．

遺伝子発現データを対象としたクラスタリング (Kerr et al., 2008; Jiang et al., 2004; Gan et al., 2007) では，多変量データ解析手法やグラフスペクトル法の適用等も含め，実に様々なアプローチが取られるが，基本的には全空間を分割する意味でのクラスタリングである．一方，

本研究では，数あるクラスタの中から重要と考えるクラスタのみをピンポイントに抽出する点で，従来の意味でのクラスタリングを行なっていることとは明らかに異なる．なお，こうしたクラスタのピンポイント抽出の枠組は，文書データや医療データに対しても試みられている（Haraguchi and Okubo, 2006, 2007; Sato et al., 2007）．

本論文の構成は以下の通りである．次章では，後の議論で必要となる基本語彙の定義や表記を与える．3 章ではバイクラスタリングについて述べ，4 章で接尾辞木に基づくバイクラスタ抽出手法の概略を紹介する．続く 5 章で，本研究で提案する接尾辞木に基づく疑似バイクラスタ抽出手法の詳細について議論し，抽出アルゴリズムを与える．6 章では実験結果を示し，それをもとに考察を行ない，最後の 7 章で稿をまとめる．

2. 準備

ここでは，本論文で用いる用語の定義を与える．

M 行 N 列の行列 $D=(a_{ij})$ において，a_{ij} を，個体 x_i に関する属性 A_j の値（属性値）と解釈し，D をデータ行列と呼ぶ．任意の a_{ij} が離散値である時，特に離散データ行列と言う．

文字（letter）あるいは 記号（symbol）の（空でない）有限集合をアルファベットと呼び，これを Σ で表わす．文字 $\mathtt{s}_i \in \Sigma$ から構成される有限の列 $S=\mathtt{s}_1\cdots\mathtt{s}_n$ を Σ 上の文字列と言い，n をその長さと定める．特に，S の i 番目の文字 $\mathtt{s}_i$ は $S[i]$ で参照される．以下では，Σ 上の文字列のことを単に文字列と呼ぶ．後に議論するが，本研究で扱う遺伝子発現データにおいては，各時刻における発現量をある記号に符号化することで，各遺伝子の発現変動をひとつの文字列として表現する．

長さ n の文字列 $S=\mathtt{s}_1\cdots\mathtt{s}_n$ において，$S[i]$ で始まり $S[j]$ で終る S の部分文字列を $S[i;j]$ と表記する．特に，$S[i;n]$ を S の接尾辞（Suffix）と呼ぶ．例えば，文字列 $S=\mathtt{banana}$ の接尾辞は，$S[1;6]=\mathtt{banana}$，$S[2;6]=\mathtt{anana}$，$S[3;6]=\mathtt{nana}$，$S[4;6]=\mathtt{ana}$，$S[5;6]=\mathtt{na}$，$S[6;6]=\mathtt{a}$ の 6 つが存在する．

3. バイクラスタリング

3.1 バイクラスタ

所与のデータ行列と類似性尺度のもと，従来のクラスタリングでは，類似した個体群，あるいは，類似した属性（値）群のいずれかの抽出を試みる．つまり，データ行列における行のクラスタリング，あるいは，列のクラスタリングを単独に行なう．

これに対し，個体と属性のクラスタリングを同時に行なう，すなわち，行と列の両方向のクラスタリングを同時に行なう枠組が近年活発に研究されており（例えば，Cheng and Church, 2000; Madeira and Oliveira, 2005），こうした枠組はバイクラスタリングと呼ばれる．そこでは，ある特定の属性群に関して類似した個体群を考えていることから，まとまりとしての意味がより明確なクラスタ抽出が可能となる．

文献 Madeira and Oliveira (2004) に従うと，バイクラスタは次の 4 つのタイプに分類される．

(1) 行および列に関してすべて同一数値であるバイクラスタ
(2) 行あるいは列に関して同一数値であるバイクラスタ
(3) 行・列に関して一貫した関係のある数値から成るバイクラスタ
(4) 行・列に関して一貫した変動を示す記号から成るバイクラスタ

(1) はすべての要素が同一値である最も素朴なバイクラスタであり，(2) は行方向，あるいは，列方向に限って同一値をとるものである．(3) は異なる行や列との間に，一定の差や一定の比が認

	A	B	C	D	E
1	a1	b1	c1	d2	e2
2	a3	b3	c1	d1	e2
3	a2	b2	c2	d1	e1
4	a2	b3	c2	d1	e2

図 1. バイクラスタ.

められるものであり，これを離散値に拡張したものが(4)であると考えられる．本研究では，各遺伝子の発現値の変動を文字列として捉えることから，(4)のタイプのバイクラスタを扱うものとする．

図 1 に，個体集合 {1,2,3,4}，属性集合 {A,B,C,D,E} に関する 4×5 の離散データ行列におけるバイクラスタの一例を示す．

個体 2 と 4 は，属性 B，D および E において，全く同じ値 b3，d1，e2 を取る．この時，個体群 {2,4} および属性値群 {b3,d1,e2} の組を**バイクラスタ**と呼び，$\langle\{2,4\},\{b3,d1,e2\}\rangle$ と表記する．あるバイクラスタ C を完全に包含するバイクラスタが存在しない場合，C は**極大**であると言われる．例えば，いまの例において，バイクラスタ $\langle\{2,4\},\{d1,e2\}\rangle$ は $\langle\{2,4\},\{b3,d1,e2\}\rangle$ に完全に含まれるので極大ではない．一方，$\langle\{2,4\},\{b3,d1,e2\}\rangle$ を含むバイクラスタは存在しないので，これは極大バイクラスタである．

3.2 時系列データにおけるバイクラスタ

先に述べた通り，本研究では，時系列性を有する遺伝子発現データからのクラスタ抽出を試みる．すなわち，データ行列の属性群は，左から右へ時間的に連続した流れを持つものであると仮定する．つまり，$M \times N$ のデータ行列 $D=(a_{ij})$ における列 $(a_{i1},a_{i2},\ldots,a_{iN})$ は，時刻の経過 $t_1,t_2,\ldots,t_N$ に伴う i 番目の個体(遺伝子)の発現値変動を表わすものと考える．こうした時系列データにおけるクラスタを考える上で，ここでは次の生物学的な視点を考慮したバイクラスタを考える:

生物学的な発現プロセスは，連続した時間区間における発現値の変動により示される．

つまり，ここでは，連続した時間の流れに伴い遺伝子の発現値がどの様に変動したかを重視し，時間的に不連続な発現変動には注目しない．上述した通り，データ行列の属性群(列)は，時間的に連続した流れを持つと仮定していることから，本稿では，列のクラスタが連続して隣り合った属性値から成るバイクラスタのみを考える．以下の議論におけるバイクラスタは，特に断りが無い限りこれらを指すものとする．

4. 接尾辞木に基づくバイクラスタリング

一般に，バイクラスタの抽出は 高コストの計算を要するが，上述した時系列性(属性の連続性)を考慮したバイクラスタは，接尾辞木を利用して効率良く抽出可能であることが知られている(Madeira and Oliveira, 2005)．ここでは，接尾辞木に基づくバイクラスタリング手法の概略について述べる．

4.1 接尾辞木

接尾辞木(Suffix Tree)は，所与の文字列の接尾辞を索引単位とするデータ構造であり，これにより，様々な文字列処理を効率良く行なうことが出来る(Weiner, 1973; Gusfield, 1997)．

S を長さ n の文字列とする．S の接尾辞木とは，以下の条件を満たす木である：

- 各内部ノードは少なくとも 2 つの子ノードを持つ．
- 各枝は，(非空の)文字列でラベル付けされている．特に，あるノードとその子を結ぶ枝のラベルの先頭文字は，それぞれ必ず異なる．
- 根ノードから葉ノードへの各パスは，S の接尾辞と 1 対 1 に対応し，特に，パス上の各枝のラベルを連結した文字列がその接尾辞を表わす．

ここで，接尾辞木中，ノード v がノード w の子孫である時，そのことを $v \prec w$ と表記する．

通常は，S に出現しない文字(例えば `$`)を**終端記号**として S の末尾に連結したものを文字列と考えて接尾辞木を構築する．接尾辞木中のノード数は，その構造から $O(n)$ であることが分かる．

接尾辞木は，Ukkonen により提案されたアルゴリズムに従うと，文字列の長さ n の線形時間オーダで構築可能であることが知られている(Ukkonen, 1995)．そこでは特に，線形時間構築を実現するために接尾辞リンクが導入されている．

長さ n の文字列 S について，Ukkonen のアルゴリズムでは，S の先頭から文字をひとつずつ読み込みながら，木構造を徐々に拡張する処理を繰り返して S の接尾辞木を得る．いま，$S[1;\ell]$ に関して木構造が構築されているとする．特に，この木構造には，根からのパスが $S[1;\ell]$ の接尾辞に相当するノードがすべて含まれると仮定する．さらに，任意の $S[i;\ell]$ と $S[i+1;\ell]$ に対応するノード間に，前者から後者へ向かうリンクが張られ，これを**接尾辞リンク** (Suffix Link)と呼ぶ．ここで，次の文字 $\alpha = S[\ell+1]$ の追加に伴い，必要に応じて適切な葉ノードを作成しながら木構造を拡張することを考える．こうした葉ノードを作成する場所は，言うまでもなく $S[1;\ell]$ の各接尾辞に対応するノードの下であるが，それらは，最も長い接尾辞，すなわち，$S[1;\ell]$ に対応するノードから，接尾辞リンクを順次辿って行くことで，n の線形時間ですべて訪れることが出来る．追加された葉ノードに関して，新たに適切な接尾辞リンクを作成し，α に関する拡張処理を終える．同様の拡張処理を繰り返すと，最終的に S の接尾辞木が構築される．図 2 に，文字列 `cacao` (`cacao$`)に関する接尾辞木の構築過程を示す．図中の一点鎖線が接尾辞リンクを表わし，例えば，$S[1;2] = \mathtt{ca}$ に対応するノードから，$S[2;2] = \mathtt{a}$ に対応するノードへリンクが張られる．なお，Ukkonen のアルゴリズムの詳細については，文献 Ukkonen (1995) を参照されたい．

複数の文字列 $S_1, \ldots, S_k$ に関する接尾辞木は，**一般化接尾辞木**(Generalized Suffix Tree)と呼ばれ，その構築は，単一文字列のそれと同様である．具体的には，どの S_i にも出現しない k

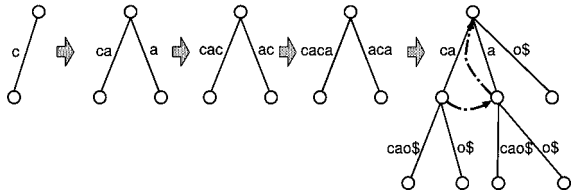

図 2. 接尾辞木.

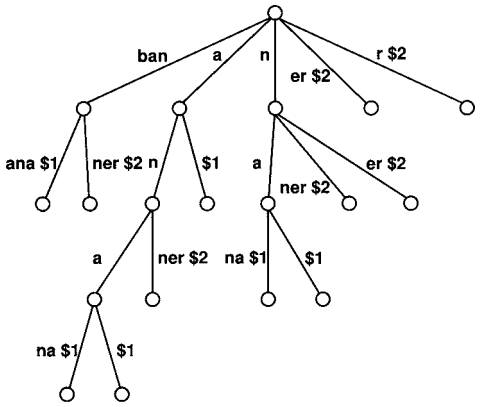

図 3. 一般化接尾辞木.

個の文字$\$_1,\ldots,\$_k$ を用意して各文字列の最後に終端記号として加え，それら文字列 $S_i\$_i$ を順次入力として，ひとつの接尾辞木を構築すればよい．図 3 に，文字列 banana (banana$\$_1$) および banner (banner$\$_2$) に関する一般化接尾辞木を示す(繁雑さを避けるため接尾辞リンクは省略する)．

4.2 接尾辞木に基づく極大バイクラスタ抽出

M 行 N 列の離散データ行列 $D=(a_{ij})$ を考える．いま，各属性 A_j $(j\in\{1,\ldots,N\})$ が取り得る値の集合(すなわち A_j のドメイン)を $dom(A_j)$ とすると，任意の $i,j\in\{1,\ldots,N\}$ $(i\neq j)$ について，$dom(A_i)\bigcap dom(A_j)=\phi$ であると仮定する．つまり，データ行列の各行に同一の属性値が複数回出現することはない．また，異なる行に出現する同一の属性値については，それらが出現する列(属性)は一致する．

D の各 i 行を $\mathtt{a}_{i1}\ \mathtt{a}_{i2}\cdots\ \mathtt{a}_{iN}$ なるひとつの文字列と見做し，D を文字列の集合と考える．この文字列集合 D に関する一般化接尾辞木を T とする．

ここで，T 中の各葉ノードは D のある行の接尾辞に対応することに注意しよう．T 中の内部ノード v は，根ノードから v に至るパス上の文字列が，v 以下の葉ノードに対応する接尾辞すべてにおいて共通して現れることを示している．すなわち，各内部ノード v は，D のあるバイクラスタを定めていることが分かる．より正確には，v 以下の葉ノードの数を $L(v)$，根ノードから v へ至るパス上の文字列の長さを $P(v)$ で参照すると，v は $L(v)$ の行と，連続した $P(v)$ の列から成るバイクラスタに対応する(図 4 参照)．

接尾辞木の定義から，これらは特に右方向に極大なバイクラスタであることに注意する．すなわち，このバイクラスタを完全に含んだまま，さらに右へ列(属性値)を追加することは出来ない．以下では，ノード v が定めるバイクラスタを C_v で参照する．

言うまでもなく，極大となるバイクラスタはこれらの一部であるが，T の接尾辞リンクを利用することで，極大バイクラスタは線形時間で抽出可能なことが示されている(Madeira and Oliveira, 2005)．

いま，T の内部ノード v_1 から v_2 への接尾辞リンクが存在するとする．この時，接尾辞木および接尾辞リンクの定義から，C_{v_2} の連続した列は C_{v_1} の最左列を取り除いたものと一致する．また，v_1 以下の葉ノードに対応する接尾辞を有する行は，必ず v_2 以下の葉ノードに対応する接尾辞を有することから，$L(v_1)\leq L(v_2)$ が成り立つ．これらのことから，

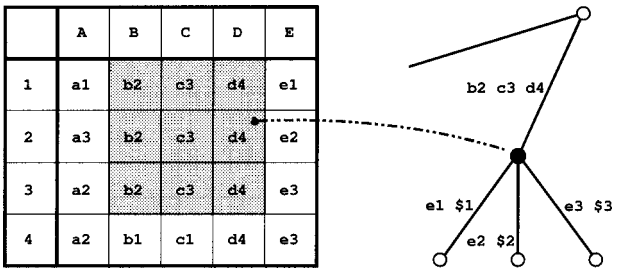

図 4. 接尾辞木の内部ノードと対応するバイクラスタ.

内部ノード v_1 から v_2 への接尾辞リンクが存在して $L(v_1) < L(v_2)$ が成り立つ時，かつ，その時に限り，C_{v_2} は極大バイクラスタである

ことが分かる．接尾辞木中の各内部ノードについて，この条件が成り立つか否かを調べることで，すべての極大バイクラスタを容易に見つけることが出来る．D 中の各文字列の長さ総和を n とすると，先に述べた通り(一般化)接尾辞木のノード総数は，n の線形オーダであることから，極大バイクラスタの探索は n の線形オーダで行なうことが出来る．また，接尾辞木の構築も Ukkonen のアルゴリズムに従い n の線形オーダで構築可能である．よって，接尾辞木に基づく極大バイクラスタ抽出処理は n の線形時間で行なえる．なお，詳細については文献 Madeira and Oliveira (2005) を参照されたい.

5. 接尾辞木に基づく疑似バイクラスタの抽出

本章では，本研究での抽出対象となる疑似バイクラスタの概念を導入し，接尾辞木のリンク構造を利用したそれらの抽出手法について議論する.

5.1 疑似バイクラスタ

一般に，所与の離散データ行列から多くの極大バイクラスタが抽出されるが，それらは行あるいは(連続した)列において多くの重複が観測される．遺伝子発現データを考えた場合，異なるクラスタ間の列の重複は，ある時間区間においてはそれらに属する遺伝子が同様の発現変動を示すことを意味する．よって，これらクラスタをひとつにまとめることで，ある時点までは同様の変動を示すが，その後異なる変動を示すクラスタの抽出が可能となる．ここでは，重複のある極大バイクラスタをまとめたものを，疑似バイクラスタと呼び，以下の通り定義する.

定義：疑似バイクラスタ．極大バイクラスタ $C_1 = \langle X_1, Y_1 \rangle, \ldots, C_k = \langle X_k, Y_k \rangle$ を考える．この時，$\bigcap_{i=1}^{k} X_i \neq \phi$ かつ $\bigcap_{i=1}^{k} Y_i \neq \phi$ ならば，

$$PC = \left\langle \bigcup_{i=1}^{k} X_i, \bigcup_{i=1}^{k} Y_i \right\rangle$$

を**疑似バイクラスタ**(Pseudo-bicluster)と呼び，すべての C_i の共通(重複)部分，すなわち，

$$\left\langle \bigcap_{i=1}^{k} X_i, \bigcap_{i=1}^{k} Y_i \right\rangle$$

を，PC の**核**(Core)と呼ぶ.

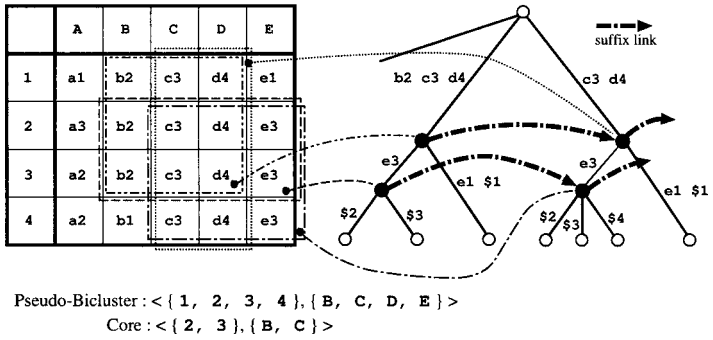

図 5. 疑似バイクラスタ.

疑似バイクラスタは，文献 Haraguchi and Okubo (2006) における疑似クリークの概念を拡張したものと捉えられる.

図 5 に，疑似バイクラスタの一例を示す．4 つの極大バイクラスタ $\langle\{1,2,3\},\{\mathtt{b2},\mathtt{c3},\mathtt{d4}\}\rangle$, $\langle\{1,2,3,4\},\{\mathtt{c3},\mathtt{d4}\}\rangle$, $\langle\{2,3\},\{\mathtt{b2},\mathtt{c3},\mathtt{d4},\mathtt{e3}\}\rangle$ および $\langle\{2,3,4\},\{\mathtt{c3},\mathtt{d4},\mathtt{e3}\}\rangle$ が，疑似バイクラスタ $\langle\{1,2,3,4\},\{\mathtt{b2},\mathtt{c3},\mathtt{d4},\mathtt{e3}\}\rangle$ を構成し，これらすべての重複部分 $\langle\{2,3\},\{\mathtt{b2},\mathtt{c3}\}\rangle$ がその核となる.

定義より, 極大バイクラスタ数を N とすると, 可能な疑似バイクラスタの総数は高々 2^N-N-1 であるが，実際の抽出対象はこれらの極一部となる．後にも述べるが，本研究では共通した発現変動の後に変動パターンが枝分かれする様子を捉えることを目指すため，列数が小さ過ぎる疑似バイクラスタは興味深いとは考え難い．よって，単に上記定義を満たす疑似バイクラスタを列挙するのではなく，何らかの制約を満たすものを選択的に抽出するアプローチが現実的であろう.

5.2 疑似バイクラスタの抽出

接尾辞木 T における極大バイクラスタノード v および v' を考え，それぞれに対応する極大バイクラスタを $C_v=\langle X,Y\rangle$ および $C_{v'}=\langle X',Y'\rangle$ とする．接尾辞木および接尾辞リンクの定義から次の性質が成り立つ.

性質 1： $v\prec v'$ の時，C_v と $C_{v'}$ は疑似バイクラスタ $\langle X',Y\rangle$ を構成し，その核は，$\langle X,Y'\rangle$ となる (図 6 (a)).

性質 2： v から w への接尾辞リンクのパスが存在して $w\prec v'$ の時，C_v と $C_{v'}$ は疑似バイクラスタ $\langle X',Y\rangle$ を構成し，その核は，$\langle X,Y'\rangle$ となる (図 6 (b)).

性質 3： v から w への接尾辞リンクのパスが存在して $v'\prec w$ の時，$X\bigcap X'\neq\phi$ であるならば，C_v と $C_{v'}$ は疑似バイクラスタ $\langle X\cup X',Y\cup Y'\rangle$ を構成し，その核は，$\langle X\bigcap X',Y\bigcap Y'\rangle$ となる (図 6 (c)).

図 6 に示す極大バイクラスタ間の関係より，上記性質が成り立つことは容易に確認出来る.

接尾辞木の構造に基づくこれら性質を利用して，次の基本戦略に従って疑似バイクラスタを抽出することを提案する.

ある極大バイクラスタノード v を基準とし，そこから接尾辞リンクを辿って到達可能なノードを u とする．$u\prec w$ あるいは $w\prec u$ なるノード w が定める極大バイクラスタ C_w と C_v と

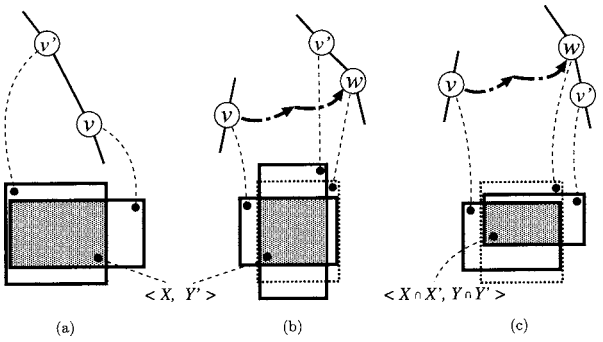

図 6. 極大疑似バイクラスタ間の関係.

の重複の様子を上記性質をもとに記録する．各極大バイクラスタノードを基準として，こうした処理を繰り返す．その後，各極大バイクラスタについて，それと重複のある極大バイクラスタを集め，共通部分(すなわち核)が空でなければ，それらを疑似バイクラスタとしてまとめた上で出力する(図 5 も参照)．

先に述べた通り，本研究では共通した発現変動が後に枝分かれする様子を捉えることを目指しているため，疑似バイクラスタがある程度の長さの列を有することを期待したい．よって，ここでは，閾値 δ を与え，δ 以上の列から成る極大バイクラスタから構成される疑似バイクラスタを抽出対象とする．

上記の手順に従う疑似バイクラスタの抽出アルゴリズムを図 7 にまとめる．図中，$suffixlink$ は，任意のノードに対して，その接尾辞リンクの先のノードを返す関数であり，そのノードが接尾辞リンクを持たない場合は，NULL が返されるものとする．また，任意のバイクラスタ $C = \langle X, Y \rangle$ において，X および Y をそれぞれ $raw(C)$ および $col(C)$ で参照している．

各極大バイクラスタを基準として，その接尾辞リンクおよび親子関係を辿って到達可能なそれぞれの極大バイクラスタについて，重複度合をチェックする．よって，データ行列のサイズを n，極大バイクラスタ数を m とすると，本アルゴリズムの最悪時間計算量は，$O(mn^2)$ となる．

本アルゴリズムは，疑似バイクラスタを構成する極大バイクラスタに δ に関する制約を課していることから，すべての疑似バイクラスタを漏れなく抽出するものではなく，その意味で完全ではない．しかし，こうした制約により，列数の少ない不要と思われる多くの疑似バイクラスタの抽出を回避することが出来る．なお，本アルゴリズムが疑似バイクラスタを抽出する健全な手続きであることは明らかである．

6. 実験

本章では，実データを対象に疑似バイクラスタの抽出実験を行なった結果を示す．

実験には，ホヤの遺伝子発現時系列データを用いた．これは 2340×14 の実数値のテーブルで与えられており，2340 の遺伝子それぞれについて，14 の時間ステージにおける発現値が記録されたものである．なお，この元となるデータの詳細については文献 Azumi et al. (2007) を参照されたい．

図 8 は，これを，横軸に時間ステージ，縦軸に発現値をとってグラフ表示したものである．

Input: a generalized suffix tree T for a data matrix D, a threshold δ
Output: a set of pseudo-biclusters each of which consists of maximal biclusters in D meeting the requirement w.r.t. δ

Procedure: $FindPseudoBiclusters(T, \delta)$
$V_{max} = \{v \in T \mid C_v$ is a maximal bicluster $\wedge\ col(\mid C_v \mid) \geq \delta\}T$;
for each $v \in V_{max}$ **do**
$w \leftarrow v$;
while $w \neq$ NULL **do**
if $raw(C_v) \bigcap raw(C_w) \neq \phi$ **then**
if $\frac{|col(C_v) \bigcap col(C_w)|}{|col(C_v)|} \geq 0.5$
then Define $overlap(C_v, C_w)$ as 1; **else** Define $overlap(C_v, C_w)$ as 0;
endif
if $\frac{|col(C_v) \bigcap col(C_w)|}{|col(C_w)|} \geq 0.5$
then Define $overlap(C_w, C_v)$ as 1; **else** Define $overlap(C_w, C_v)$ as 0;
endif
endif
for each w' such that $w' \prec w$ or $w' \prec w$ **do**
if $raw(C_w) \bigcap raw(C_{w'}) \neq \phi$ **then**
if $\frac{|col(C_w) \bigcap col(C_{w'})|}{|col(C_w)|} \geq 0.5$
then Define $overlap(C_w, C_{w'})$ as 1; **else** Define $overlap(C_w, C_{w'})$ as 0;
endif
if $\frac{|col(C_w) \bigcap col(C_{w'})|}{|col(C_{w'})|} \geq 0.5$
then Define $overlap(C_{w'}, C_w)$ as 1; **else** Define $overlap(C_{w'}, C_w)$ as 0;
endif
endif
endfor
$w \leftarrow suffixlink(w)$;
endwhile
endfor
for each $v \in V_{max}$ **do**
$Cand = \{C_{v'} \mid v' \in V_{max} \wedge overlap(C_v, C_{v'}) = 1\}$
if $\bigcap_{C \in Cand} raw(C) \neq \phi \wedge \bigcap_{C \in Cand} col(C) \neq \phi$ **then**
Output $\langle \bigcup_{C \in Cand} raw(C), \bigcup_{C \in Cand} col(C) \rangle$;
endif
endfor

図 7. 接尾辞木に基づく疑似バイクラスタ抽出アルゴリズム.

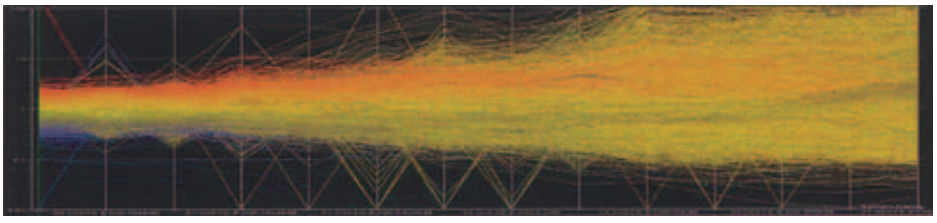

図 8. ホヤの遺伝子発現データ.

時間ステージが進むにつれて，発現の変動幅が大きくなっていく様子が見て取れる.

なお，実験環境は CPU:Pentium4 2.8 GHz，Memory:2 GB，OS:WindowsXP Professional であり，システムは Java で実装した.

6.1 一般化接尾辞木の構築

疑似バイクラスタの抽出にあたり一般化接尾辞木を構築する必要がある．そのためには上述した 2340×14 の実数値データ行列を，符号化(文字列化)しなければならない．ここでは，各ステージ i における発現値を，含まれる遺伝子数が均等になる様 10 の区間に離散化し，各区間にそれぞれ $\mathtt{A}_i$ から $\mathtt{J}_i$ の記号を割り当てるものとする．すなわち，各遺伝子 g は

$$g = s_1 s_2 \cdots s_{13} s_{14}$$

なる文字列で表わされ，s_i は $\mathtt{A}_i, \mathtt{B}_i, \ldots, \mathtt{J}_i$ のいずれかである．

こうして文字列化された各遺伝子 g_i に対して，終端記号$\mathtt{\$}_i$ を用意し，文字列集合 $\{g_1\mathtt{\$}_1, g_2\mathtt{\$}_2, \ldots, g_{2339}\mathtt{\$}_{2339}, g_{2340}\mathtt{\$}_{2340}\}$ に関する一般化接尾辞木を構築した．なお，得られた一般化接尾辞木のノード総数は 48077，接尾辞リンク総数は 12836 であった．

6.2 実験結果

抽出された疑似バイクラスタ

ここでは，$\delta = 3$，すなわち，少なくとも 3 つの時間ステージを有する極大バイクラスタのみから構成される疑似バイクラスタの抽出を試みた．

構築した一般化接尾辞木から，$\delta = 3$ のもとで 750 の極大バイクラスタが抽出され，それらが 98 の疑似バイクラスタを形成した．ここで，ひとつの疑似バイクラスタを構成する極大バイクラスタ数の平均は 3.3，核のステージ数(列数)は平均 2.3 であった．

抽出された疑似バイクラスタの一例を示す．8 つの極大バイクラスタ

$\langle(12), \{\mathtt{B}_8, \mathtt{A}_9, \mathtt{A}_{10}, \mathtt{A}_{11}, \mathtt{A}_{12}, \mathtt{A}_{13}, \mathtt{A}_{14}\}\rangle$　　$\langle(149), \{\mathtt{A}_{10}, \mathtt{A}_{11}, \mathtt{A}_{12}\}\rangle$
$\langle(97), \{\mathtt{A}_{10}, \mathtt{A}_{11}\mathtt{A}_{12}, \mathtt{A}_{13}\}\rangle$　　$\langle(143), \{\mathtt{A}_9, \mathtt{A}_{10}, \mathtt{A}_{11}\}\rangle$
$\langle(101), \{\mathtt{A}_9, \mathtt{A}_{10}, \mathtt{A}_{11}, \mathtt{A}_{12}\}\rangle$　　$\langle(64), \{\mathtt{A}_9, \mathtt{A}_{10}, \mathtt{A}_{11}, \mathtt{A}_{12}, \mathtt{A}_{13}\}\rangle$
$\langle(12), \{\mathtt{C}_7, \mathtt{B}_8, \mathtt{A}_9, \mathtt{A}_{10}, \mathtt{A}_{11}\}\rangle$　　$\langle(99), \{\mathtt{A}_{11}, \mathtt{A}_{12}, \mathtt{A}_{13}\}\rangle$

が疑似バイクラスタ

$$\langle(193), \{\mathtt{C}_7, \mathtt{B}_8, \mathtt{A}_9, \mathtt{A}_{10}, \mathtt{A}_{11}, \mathtt{A}_{12}, \mathtt{A}_{13}, \mathtt{A}_{14}\}\rangle$$

を形成し，その核は

$$\langle(5), \{\mathtt{A}_{11}\}\rangle$$

である．ここで，$\langle(N), \{\cdots\}\rangle$ なる表記は略記であり，そのバイクラスタや核が N の遺伝子から構成されることを意味する．

図 9 は，この(符号レベルの)疑似クラスタを，もとの数値データとしてグラフ化したものである．ステージ番号 10 から 11 にかけてそれぞれ共通した発現変動を示した後，変動パターンが枝分かれする様子が明確に観察出来るであろう．疑似バイクラスタの核には，ステージ番号 10 において各遺伝子が同様の発現を示すことが現れていないが，符号化後の各遺伝子のステージ番号 10 における値(記号)を調べたところ，大部分の遺伝子が $\mathtt{A}_{10}$ で，それ以外の極一部の遺伝子が $\mathtt{B}_{10}$ であった．$\mathtt{A}_{10}$ と $\mathtt{B}_{10}$ は，数値レベルでは隣同士の区間であるため，符号化の仕方によっては，これらはすべて同じ記号と成り得たものであり，そうした場合は，図 9 に観測されるステージ番号 10 から 11 にかけての共通発現変動を，疑似バイクラスタの核として符号レベルでも捉えることが可能である．以上より，本手法により概ね期待した通りのクラスタが抽出出来たと考える．

データサイズと計算時間の関係

データサイズ(文字列長)と計算時間の関係を観察するために，クラスタ抽出対象とする遺伝子数を変化させた場合の疑似バイクラスタ抽出に掛かる計算時間を計測した．結果を図 10 (左)

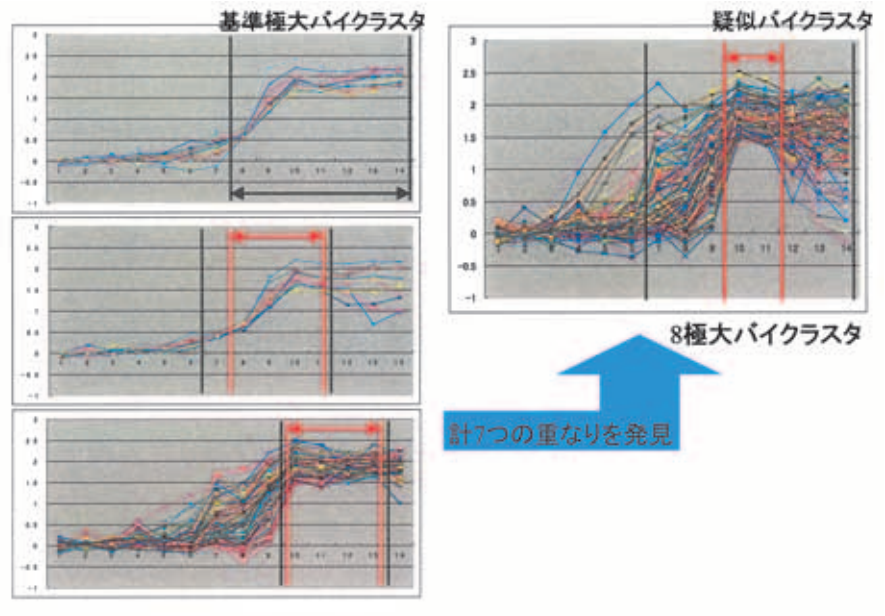

図 9. ホヤの遺伝子発現データにおける疑似バイクラスタ.

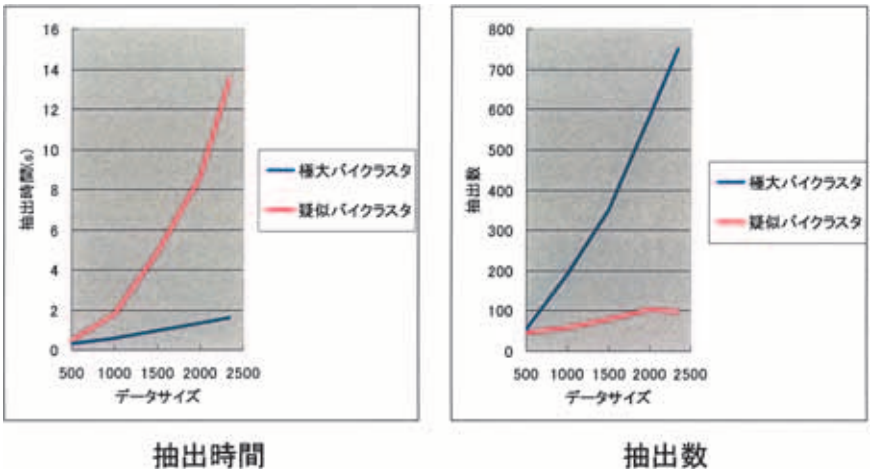

図 10. データサイズと計算時間・バイクラスタ総数.

に示す.

結果から，データ数 n の増加に伴い，計算時間の増加は n^3 程度に比例していることが分かる．前章で議論した通り，文字列長を n，極大バイクラスタ数を m とすると，疑似バイクラスタ抽出の最悪時間計算量は $O(mn^2)$ であり，概ねそれを支持する結果が観測されたと言えよう.

データサイズとバイクラスタ総数の関係

データサイズと，そこから抽出されるバイクラスタ総数の関係を観察するために，クラスタ抽出対象とする遺伝子数を変化させた場合の(疑似)バイクラスタ総数を計測した．結果を図 10 (右)に示す．

データサイズ n の増加に伴い，極大バイクラスタ総数は n^2 に近い増加を示している．一方，疑似バイクラスタ総数は，ほぼ n に比例する増加に留まっている．これは，極大バイクラスタ数が増加しても，疑似バイクラスタの核となる共通変動パターンは，それほど増加しないことを意味している．このことは，細胞の成長過程における各遺伝子の変動パターンは決して出鱈目なものではなく，いくつかの意味のある変動パターンが存在していることを示唆すると考えられ，直観にも合致する観測と言えるだろう．

6.3 考察

本実験に用いたホヤの遺伝子発現データの出典は文献 Azumi et al. (2007)であるが，そこでも k-means 法による遺伝子群のクラスタリングが行なわれている．まずは，文献 Azumi et al. (2007)において抽出されたクラスタと，先に示した疑似バイクラスタとを比較し，本手法の有効性を述べる．

先の疑似バイクラスタは，ステージ番号 11–13 (胚発生中−後期)に発現の亢進が顕著な遺伝子から成り，そこには，アクチンやミオシン等の筋肉関連遺伝子が多く含まれていること，および，コラーゲン遺伝子がある程度まとまって含まれていることが専門家により確認された．一方，文献 Azumi et al. (2007)においても，同様に筋肉関連遺伝子を多く含むクラスタが抽出されたが，コラーゲン遺伝子については，複数のクラスタに分散し，ひとつのクラスタにはまとまって現れていない．胚発生中−後期に，筋肉遺伝子やコラーゲン遺伝子の発現が亢進していることは，生物学的に合理性のある現象であり，それらをひとつのクラスタとして抽出出来たことは，本手法の大きな特徴であると言える．抽出した疑似バイクラスタは，筋肉遺伝子とコラーゲン遺伝子がある時間区間においてほぼ同様な発現変動を示した後，変動が枝分かれする様子を捉えており，枝分かれ後の発現変動をさらに詳細に分析し，それらを核として含む疑似バイクラスタとの関係を調べることで，発現変動のカスケード構造の把握などが可能となる．これにより，伝統的な k-means 法等では捉えることが出来ない遺伝子間の新たな相関を見出せる可能性がある．本手法が遺伝子データ解析における有用な基盤技術となり得ることを期待している．

750 の極大バイクラスタが存在する中，抽出された疑似バイクラスタ数は 98 であった．こうした出力数の削減も，疑似バイクラスタを抽出対象とする大きなメリットであると考えている．詳細(厳密)な多数の出力結果は，データ全体を概観する際にはむしろ障害となる場合も少なくない．実際にデータを解析する際には，比較的数の少ない疑似バイクラスタにより，全体の大まかな様子を捕え，必要に応じて興味ある部分のみを詳細化・厳密化すると言った戦略が現実的であろう．

データの符号化において，ここではデータ値を 10 区間に離散化した場合の結果を示したが，より粗い 5 区間への離散化，および，より細かい 20 区間への離散化においても，これとほぼ同様の結果が得られた．ただし，図 8 に示した通り，データの特性上，ステージが進むにつれて発現値の変動が激しくなることから，時間の経過に伴い離散化のレベルを詳細化するといった工夫の余地はある．

離散化の方法はこうした単純なものだけではない．例えば，ある幅の窓を考え，窓内で観測される部分系列のクラスタリングを行なうことで，連続する時間ステージ間の関係を考慮した離散化が可能となる．その際，単純なユークリッド距離だけでなく，密度の概念を用いてクラ

スタリング(Ester et al., 1996)すること等も大変興味深い．こうした離散化の詳細については，生物学的な視点を取り込むことも含め，さらに深く考察を進める必要がある．

7. おわりに

本研究では，遺伝子発現データの解析にあたり，重複のある極大バイクラスタをひとつにまとめた疑似バイクラスタを抽出対象とする枠組を提案し，接尾辞木の構造を利用したそれらの抽出アルゴリズムを与えた．疑似バイクラスタを抽出することで，ある時間区間においては同様の発現変動を示す遺伝子群が，その後枝別れをして別の挙動を示すといった，生物学的にも興味深いと思われる様子が明確に観測出来る．

今後の課題として，まずは計算量の軽減が挙げられる．現在の疑似バイクラスタの定義は非常に一般的なものであり，本来欲しかった疑似バイクラスタ以外のものも多く含まれる．クラスタの構造にさらに制約をかけて抽出対象を絞り込むことで，アルゴリズムをより効率化したい．

疑似バイクラスタの定義に従うと，ひとつの疑似バイクラスタが異なる核を有する場合がある．核は，その疑似バイクラスタをひとつの意味のあるまとまりと見做す根拠を与えるものであることから，複数の核の存在は，クラスタの多様な解釈に密接に関係する極めて興味深い示唆を与えるものとなろう．今後はこうした視点からの考察も進めたい．

本研究では，遺伝子発現データを対象に疑似バイクラスタ抽出を試みたが，手法の適用範囲をそれに限定するものではない．本手法は一般の離散的な時系列データに対して適用可能であり，他のドメインにおける本手法の有効性を考察することも大変興味深い．

謝　辞

本研究においては，北海道大学創生科学共同研究機構・安住薫助教よりホヤの遺伝子発現データを御提供頂き，生物学的な視点から，実験結果に関する大変貴重な御助言を頂きました．北海道大学大学院情報科学研究科・喜田拓也准教授には，接尾辞木構築アルゴリズムに関する様々な御助言を頂きました．また，査読者の方々からも，論文を改善するにあたり大変有益な御指摘や御助言を数多く頂きました．ここに深く感謝し御礼申し上げます．

参 考 文 献

Azumi, K., Sabau, S. V., Fujie, M., Usami, T., Koyanagi, R., Kawashima, T., Fujiwara, S., Ogasawara, M., Satake, M., Nonaka, M., Wang, H., Satou, Y. and Satoh, N. (2007). Gene expression profile during the life cycle of the urochordate *Ciona intestinalis*, *Developmental Biology*, **308**, 572–582.

Cheng, Y. and Church, G. M. (2000). Biclustering of expression data, *Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology*, 93–103.

Dhillon, I. S., Mallela, S. and Modha, D. S. (2003). Information-theoretic co-clustering, *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining — KDD'03*, 89–98.

Ester, M., Kriegel, H., Sander, J. and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial database with noise, *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining — KDD'96*, 226–231.

Gan, G., Ma, C. and Wu, J. (2007). *Data Clustering — Theory, Algorithms, and Applications*, SIAM,

Philadelphia.

Gusfield, D. (1997). *Algorithms on Strings, Trees and Sequences*: *Computer Science and Computational Biology*, Cambridge University Press, New York.

Haraguchi, M. and Okubo, Y. (2006). A method for pinpoint clustering of Web pages with pseudo-clique search, *Federation over the Web, International Workshop, Dagstuhl Castle, Germany, May 1–6, 2005, Revised Selected Papers*, Springer-LNAI 3847, 59–78.

Haraguchi, M. and Okubo, Y. (2007). An extended branch-and-bound search algorithm for finding top-N formal concepts of documents, *New Frontiers in Artificial Intelligence, JSAI 2006 Conference and Workshops, Tokyo, Japan, June 5-9, 2006, Revised Selected Papers*, Springer-LNCS 4384, 276–288.

Jain, A. K. and Murty, M. N. and Flynn, P. J. (1999). Data clustering: A review, *ACM Communication Surveys*, **31**(3), 264–323.

Jiang, D., Tang, C. and Zhang, A. (2004). Cluster analysis for gene expression data: A survey, *IEEE Transactions on Knowledge and Data Engineering*, **16**(11), 1370–1386.

Kerr, G., Ruskin, H. J., Crane, M. and Doolan, P. (2008). Techniques for clustering gene expression data, *Computers in Biology and Medicine*, **38**(3), 283–293.

Kurtz, S., Choudhuri, J. V., Ohlebusch, E., Schleiermacher, C., Stoye, J. and Giegerich, R. (2001). REPuter: The manifold applications of repeat analysis on a genomic scale, *Nucleic Acids Research*, **29**(22), 4633–4642.

Madeira, S. C. and Oliveira, A. L. (2004). Biclustering algorithms for biological data analysis: A survey, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **1**(1), 24–45.

Madeira, S. C. and Oliveira, A. L. (2005). A linear time biclustering algorithm for time series gene expression data, *Proceedings of the 5th International Workshop on Algorithms in Bioinformatics — WABI'05*, 39–52.

Masuda, S. (2005). Analysis of ascidian gene expression data by clique Search, Master's Thesis, Graduate School of Engineering, Hokkaido University, March (in Japanese).

Sato, K., Okubo, Y., Haraguchi, M. and Kunifuji, S. (2007). Data mining of time-series medical data by formal concept analysis, *Proceedings of The 11th International Conference on Knowledge-based Intelligent Information and Engineering Systems — KES'07*, Springer-LNCS 4693, 1214–1221.

Tishby, N., Pereira, F. C. and Bialek, W. (1999). The information bottleneck method, *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, 368–377.

Ukkonen, E. (1995). On line construction of suffix trees, *Algorithmica*, **14**(3), 249–260.

Weiner, P. (1973). Linear pattern matching algorithm, *Proceedings of the 14th Annual IEEE Symposium on Switching and Automata Theory*, 1–11.

184 Proceedings of the Institute of Statistical Mathematics Vol. 56, No. 2, 169–184 (2008)

Extracting Pseudo-biclusters from Gene Expression Data Based on Suffix Tree

Tetsuro Namba, Makoto Haraguchi and Yoshiaki Okubo

Division of Computer Science, IST, Hokkaido University

This paper describes a method for finding *Pseudo-Biclusters* of gene expression data. For time series data, a linear time algorithm with the help of a *suffix tree* has been proposed. Although this algorithm can efficiently enumerate all maximal biclusters, we often observe many overlapping clusters. By combining such clusters, we can interestingly observe that all genes in the combined cluster behave quite similarly within a common time span, but differently after that. This observation is expected to provide valuable suggestions to experts. Thus, we introduce a notion of *pseudo-biclusters*. A pseudo-bicluster consists of several maximal biclusters with some overlap. We design a polynomial time algorithm for finding them with a suffix tree. Some experimental results for gene expression data of ascidian (Hoya) are also presented, showing an interesting actually-extracted cluster.

Key words: Biclustering, pseudo-bicluster, suffix tree, gene expression data, time series data.