

## 平衡直線的プログラムに対するパターン照合アルゴリズム

平尾 昌啓<sup>†a)</sup> 篠原 歩<sup>††b)</sup> 竹田 正幸<sup>††c)</sup> 有川 節夫<sup>††d)</sup>

Pattern Matching Algorithm for Balanced Straight-Line Programs

Masahiro HIRAO<sup>†a)</sup>, Ayumi SHINOHARA<sup>††b)</sup>, Masayuki TAKEDA<sup>††c)</sup>,  
and Setsuo ARIKAWA<sup>††d)</sup>

あらまし テキストとパターンが直線的プログラムとその変種で記述された文字列の照合問題を扱う．直線的プログラムによって表される文字列の長さは，その記述長に対して指数的に長くなる可能性があるので，展開することなしに，いかに高速な照合を行うかが問題となる．直線的プログラムに対する既存の最も良い圧縮パターン照合アルゴリズムは  $O(n^2m^2)$  時間  $O(nm)$  領域で動作する．ここで， $n$  は圧縮テキストのサイズで， $m$  は圧縮パターンのサイズである．本論文では，平衡直線的プログラムという直線的プログラムの変種を提案し，それに対する高速な圧縮パターン照合アルゴリズムを示す．平衡直線的プログラムは直線的プログラムに比べ圧縮率では劣るが，直線的プログラムと同様に指数長の表現力をもつ．提案アルゴリズムは  $O(nm)$  時間  $O(nm)$  領域で動作する．

キーワード パターン照合，データ圧縮，圧縮パターン照合，直線的プログラム，平衡直線的プログラム

### 1. まえがき

ネットワーク技術の進展と情報の電子化によって，情報源符号化の技法がますます重要になっている．単一の情報源モデルに依存しないユニバーサルなデータ圧縮符号の研究は，実用的なデータ圧縮アルゴリズムとしてのみならず，モデル選択問題など様々な分野で応用され，情報理論の基本問題としての基礎研究から様々な応用研究に至るまで幅広く研究されている．

近年，Kieffer らによって，文法変換に基づいたデータ圧縮の手法が提案されている [2], [3], [5]．これは，原文のデータ文字列  $w$  に対し，まず  $w$  のみを生成する文脈自由文法  $G$  を構築し，その  $G$  を符号化することで  $w$  を記述しようとするものである (図 1)．

文字列  $w$  の中に繰り返し現れる部分文字列を抽出して非終端記号に置き換える操作を適用していくこと

によって， $w$  をよりコンパクトに表現する文法  $G$  を求めることが，効率の良い圧縮率を達成する鍵となる．彼らは，文法変換に基づいた損失のないユニバーサルなデータ圧縮アルゴリズムを示し，この圧縮アルゴリズムを実データに適用することによって Compress や Gzip をしのぐ圧縮率が達成できることを実証している．Nevill-Manning と Witten によって独立に提案された Sequitur [9] も本質的には同じアイデアに基づくものであり，データ圧縮の手法としてのみならず，1 次元データに内在する階層構造を自動的に抽出するためのツールとしても注目されている [10]．また，Larsson と Moffat による，頻度の高い記号対を新たな記号で置き換える操作を再帰的に繰り返す Re-pair [6] も同じく文法変換に基づいたデータ圧縮方式である．

一方，テキスト  $T$  中に現れるパターン  $P$  のすべての出現位置を求めるパターン照合は，文字列処理の最も基礎的な問題の一つである．このパターン照合問題

<sup>†</sup>九州大学大学院システム情報科学研究科，福岡市  
Department of Infomatics, Kyushu University, 6-10-1  
Hakozaki, Hakata-ku, Fukuoka-shi, 812-8580 Japan

<sup>††</sup>九州大学大学院システム情報科学研究院，福岡市  
Department of Infomatics, Kyushu University, 6-10-1  
Hakozaki, Hakata-ku, Fukuoka-shi, 812-8580 Japan

a) E-mail: hirao@i.kyushu-u.ac.jp

b) E-mail: ayumi@i.kyushu-u.ac.jp

c) E-mail: takeda@i.kyushu-u.ac.jp

d) E-mail: arikawa@i.kyushu-u.ac.jp

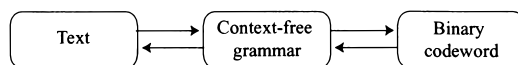


図 1 文法変換に基づくデータ圧縮  
Fig. 1 Data compression based on grammar transform.

において、特に最近注目を集めているのが、入力として圧縮された文字列が与えられたときのパターン照合問題（以降、圧縮パターン照合問題と呼ぶ）と呼ばれるものであり、主に二つの設定において研究がなされている（文献 [11] 参照）。

設定 1 (Compressed Pattern Matching)

入力としてパターン  $P$  とテキスト  $T$  の圧縮表現  $Compress(T)$  が与えられる。

設定 2 (Fully Compressed Pattern Matching)

入力としてパターン  $P$  の圧縮表現  $Compress(P)$  とテキスト  $T$  の圧縮表現  $Compress(T)$  が与えられる。

これらの設定のもとで、文法変換に基づいたデータ圧縮に対するパターン照合を考える際には、通常、文法上での照合問題を考えることが一般的である。そこで本論文の以降の議論では、入力として構成済みの文法  $G$  が与えられるものとする。つまり、 $G$  の表す文字列  $w$  を明示的に求めてからパターン照合するのではなく、 $G$  から直接的にパターンの出現位置を計算することを目的とする。そして、計算量の評価は文字列  $w$  の長さ  $N$  に対しては行わず、文法  $G$  のサイズ  $n$  に対して行うことにする。なお問題としては、文字列が与えられたとき、それを文法で表現してからパターン照合するといったものではない。

圧縮パターン照合に対する研究は、文字列照合の新たな研究課題としてだけでなく、既存の圧縮法を評価するための新たな指標を与えるものでもある。すなわち、今までの主な指標である圧縮率、圧縮時間、展開時間に加え、圧縮したままのパターン照合時間も重要な要素となる。例えば、Re-pair に対して規則の数を 256 に制限した Byte-Pair Encoding 法は、Lempel-Ziv 型の圧縮法に比べ圧縮率の点で劣るが、圧縮パターン照合に適した方式であるとして再評価されている [12], [13]。

本論文では、設定 2 のもとで、テキスト及びパターンがともに直線的プログラムとその変種で記述された文字列の圧縮パターン照合問題について考える。直線的プログラムは文字の連結を基本命令とした一種の文脈自由文法で、直線的プログラムのサイズ  $n$  に対して、極端な場合にはそれが表す文字列の長さ  $N$  は指数的に長くなり得る。したがって、展開してから文字列照合を行ったのでは最悪時には  $n$  に対する多項式時間で照合を行えない。この枠組みにおいて、宮崎らは直線的プログラムに対する高速な圧縮パターン照合アルゴ

リズムを提案した [8]。このアルゴリズムは  $O(n^2m^2)$  時間  $O(nm)$  領域で動作する。ここで、 $n$  は圧縮テキストのサイズで、 $m$  は圧縮パターンのサイズである。

これに対し我々は、直線的プログラムの変種であり、より高速な圧縮パターン照合が可能な平衡直線的プログラムを提案する。平衡直線的プログラムと直線的プログラムの違いは最後の代入文に接尾語の削除の操作を許している点と、それ以外の代入文において変数の連結は同じ長さの文字列を表す変数の対に限定されている点である。したがって、平衡直線的プログラムの評価木は根を除くすべての部分木が完全二分木をなす。文字列の長さが 2 のべき数のときは、この直線的プログラムによる表現は Kieffer らによってユニバーサルであることが示された二分アルゴリズム [3] で得られる文脈自由文法と等価である。平衡直線的プログラムは直線的プログラムに比べ圧縮率の点では劣る。しかしながら直線的プログラム同様に指数長  $O(2^n)$  の表現力をもっており、圧縮アルゴリズムが単純で高速である。本論文は、平衡直線的プログラムの定義と、それに対する  $O(nm)$  時間、 $O(nm)$  領域の圧縮パターン照合アルゴリズムを与える。

図 2 は喜田ら [4] が提案したコラージュシステム (collage system) とその部分クラス、そして平衡直線的プログラムとの関係を図示したものである。コラージュシステムは、様々な辞書式の圧縮方式を、圧縮文字列照合の観点から統一的にとらえる枠組みとして有用であることが示されている。正則コラージュシステム (regular collage systems) は連結操作のみからなるクラスで、直線的プログラムに対応している。また、単純コラージュシステム (simple collage system) は LZ78 系 (LZ78, LZW) の圧縮法を含む。なお、平衡直線的プログラムは最後の代入分に接尾語を削除する操作を含むので、正則コラージュシステムのクラスには含まれない。表 1 に本論文の結果も含めて関連する

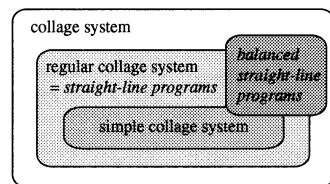


図 2 Collage system と平衡直線的プログラム  
Fig. 2 Collage system and balanced straight-line programs.

表 1 計算量の比較  
Table 1 Comparison of the complexity.

	Compressed Pattern Matching	Fully Compressed Pattern Matching
Collage Systems [4]	$O(n \text{ height}(T) + m^2)$ [4]	unknown
直線的プログラム	$O(n + m^2)$ [4]	$O(n^2 m^2)$ [8]
平衡直線のプログラム	$O(n + m^2)$ [4]	$O(nm)$

実行時間をまとめる .

2. 準備

アルファベット  $\Sigma$  上の直線的プログラム  $R$  とは、  
代入文の列

$$X_1 = \text{expr}_1; X_2 = \text{expr}_2; \dots; X_n = \text{expr}_n,$$

をいう . ここで ,  $X_i$  は変数で ,  $\text{expr}_i$  は終端記号  $a \in \Sigma$  若しくは変数  $X_\ell, X_r (\ell, r < i)$  の連結  $X_\ell \cdot X_r$  である . 直線のプログラム  $R$  の変数の個数  $n$  を  $\|R\|$  で表し ,  $R$  中の最後の変数  $X_n$  によって得られる文字列を  $R$  で表す . 特に紛らわしくない場合は , 変数  $X_i$  によって表される文字列も  $X_i$  と書く . 変数  $X$  によって表される文字列から , 長さ  $d$  の接尾語を削除した文字列を  $X^{[d]}$  と表す . 変数  $X$  の高さ  $\text{height}(X)$  を次のように定義する .

$$\text{height}(X) = \begin{cases} 1 & (X = a \in \Sigma \text{ の場合}) \\ 1 + \max(\text{height}(X_\ell), \text{height}(X_r)) & (X = X_\ell \cdot X_r \text{ の場合}) \end{cases}$$

文字列  $w$  の長さを  $|w|$  と表す . 文字列  $w$  に対して  $w[i : j] (1 \leq i \leq j \leq |w|)$  は  $w$  の  $i$  番目から  $j$  番目までの部分文字列である .  $w[i : i]$  を  $w[i]$  と略記する . また ,  $w[1 : j]$  は 1 を省略して  $w[: j]$  と書く . 同様に  $w[i : |w|]$  は  $w[i : ]$  と略記する .

文字列  $w$  の周期とは , すべての  $i \in \{1, \dots, |w| - p\}$  に対して ,  $w[i] = w[i + p]$  を満たす整数  $p (1 \leq p \leq |w|)$  である . 例えば , 文字列  $aabaaabaa$  は周期 4 , 7 , 8 , 9 をもち , 最小周期は 4 となる .

直線のプログラムで記述された文字列に対する圧縮パターン照合問題とは , パターン  $P$  とテキスト  $T$  が , 直線のプログラム  $\mathcal{P}$  と  $\mathcal{T}$  として与えられたときに , テキスト  $T$  に現れるパターン  $P$  の出現位置をすべて求める問題である . つまり , 次の集合を求める .

$$\text{Occ}(\mathcal{T}, \mathcal{P}) = \{i \mid T[i : i + |P| - 1] = P\}.$$

以降では ,  $X_i$  と  $Y_j$  をそれぞれ  $\mathcal{T}$  と  $\mathcal{P}$  の変数と

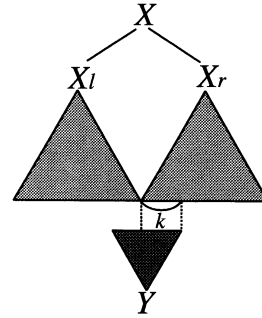


図 3  $\text{Occ}^\Delta(X, Y)$  は  $X_\ell$  と  $X_r$  をまたぐ  $Y$  の出現位置  $k$  の集合  
Fig. 3  $k \in \text{Occ}^\Delta(X, Y)$ , since  $Y$  covers the boundary between  $X_\ell$  and  $X_r$ .

し , テキストとパターンの直線のプログラムのサイズを  $\|\mathcal{T}\| = n, \|\mathcal{P}\| = m$  とする , また , 整数の集合  $U$  と整数  $k$  に対して ,  $U \oplus k = \{i + k \mid i \in U\}$  と  $U \ominus k = \{i - k \mid i \in U\}$  とする .

まず , 文献 [8] と本論文で述べる圧縮パターン照合アルゴリズムの概要を示す . はじめに , 集合  $\text{Occ}(X, Y)$  に対して簡潔な表現を考える .  $X = X_\ell \cdot X_r$  としたとき ,  $X_\ell$  と  $X_r$  をまたぐ  $Y$  の出現位置を  $\text{Occ}^\Delta(X, Y)$  とする . つまり ,

$$\text{Occ}^\Delta(X, Y) = \left\{ s \mid \begin{array}{l} 1 \leq s \leq |Y| + 1 \text{ かつ} \\ s + |X_\ell| - |Y| \in \text{Occ}(X, Y) \end{array} \right\}$$

と定義する ( 図 3 ) . 便宜上 ,  $X = a \in \Sigma$  に対して ,  $\text{Occ}^\Delta(X, Y) = \text{Occ}(X, Y)$  とする .

次の二つの補題は , 任意の直線のプログラムに対して成り立つ .

[ 補題 1 ] ( Miyazaki et al. [8] ) 直線のプログラム  $\mathcal{T}, \mathcal{P}$  中の任意の変数  $X, Y$  に対して , 集合  $\text{Occ}^\Delta(X, Y)$  は一つの等差数列で表される .

[ 補題 2 ] ( Miyazaki et al. [8] ) 直線のプログラム  $\mathcal{T}, \mathcal{P}$  中の変数  $X_i, Y$  に対して ,  $X_i = X_{\ell(i)} \cdot X_{r(i)}$  としたとき , 次式が成立する ( 図 4 参照 ) :

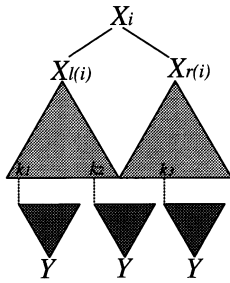


図 4  $k_1 \in Occ(X_{l(i)}, Y)$ ,  $k_2 \in Occ^\Delta(X_i, Y) \oplus |X_{l(i)}| \ominus |Y|$ ,  $k_3 \in Occ(X_{r(i)}, Y) \oplus |X_{l(i)}| \cdot k_1$ ,  $k_2, k_3$  はともに集合  $Occ(X_i, Y)$  の要素である  
 Fig.4  $k_1, k_2, k_3 \in Occ(X_i, Y)$ , while  $k_1 \in Occ(X_{l(i)}, Y)$ ,  $k_2 \in Occ^\Delta(X_i, Y) \oplus |X_{l(i)}| \ominus |Y|$  and  $k_3 \in Occ(X_{r(i)}, Y) \oplus |X_{l(i)}| \cdot$

$$Occ(X_i, Y) = Occ(X_{l(i)}, Y) \cup (Occ^\Delta(X_i, Y) \oplus |X_{l(i)}| \ominus |Y|) \cup (Occ(X_{r(i)}, Y) \oplus |X_{l(i)}|).$$

補題 2 により, 集合  $Occ(X_n, Y)$  は  $\{Occ^\Delta(X_i, Y)\}_{i=1}^n$  の組合せによって表すことができる. また, 補題 1 により,  $Occ^\Delta(X_i, Y)$  は等差数列をなすので初項, 公差, 項数の三つ組で表すことができる. これは, 任意の集合  $Occ^\Delta(X_i, Y)$  が定数領域で表せることを意味している. このようにして, 集合  $Occ(T, P) = Occ(X_n, Y_m)$  は  $O(n)$  領域の  $\{Occ^\Delta(X_i, Y_m)\}_{i=1}^n$  の組合せで表現できる. 以上により, 我々は集合  $Occ(T, P)$  を求める問題を,  $Occ^\Delta(X_i, Y_m)(i = 1, \dots, n)$  を求める問題に帰着する.

次の定理は直線的プログラムに対する圧縮パターン照合に関する最も良い結果である.

[定理 1] (Miyazaki et al. [8]) 直線的プログラム  $T$  と  $P$  が与えられると, テキスト  $T$  中のパターン  $P$  のすべての出現位置の集合  $Occ(T, P)$  は  $O(n)$  の領域で表現でき, この表現は  $O(nm)$  の領域を用いて  $O(n^2m^2)$  時間で計算できる.

### 3. 平衡直線のプログラム

平衡直線的プログラムは直線的プログラムの変種で, より高速な圧縮パターン照合を行うことができる.

[定義 1] 平衡直線のプログラム  $B$  とは, 代入文の列

$$X_1 = expr_1; X_2 = expr_2; \dots; X_n = expr_n,$$

をいう. ここで,  $X_i$  は変数で,  $expr_i$  は以下のいずれかである.

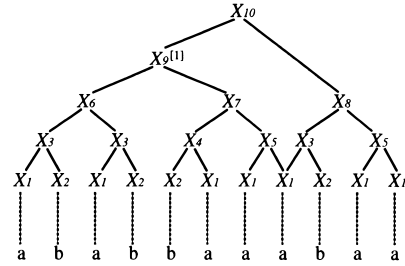


図 5 例 1 の  $R$  の評価木  
 Fig.5 Evaluation tree of  $R$ .

- $expr_i$  は終端記号  $a \in \Sigma$
- $expr_i = X_\ell \cdot X_r$ ,  
ただし  $|X_\ell| = |X_r|$  ( $\ell, r < i < n$ )
- $expr_n = X_\ell^{[d]} \cdot X_r$ ,  
ただし  $X_\ell[|X_\ell| - d + 1 :] = X_r[: d]$  かつ  $d \geq 0$ .

平衡直線のプログラムでは, 最後の代入文を除いて, 同じ長さの文字列を表す変数同士しか連結は許されない. よって, 文字列が与えられたとき, 最後の代入文における重なりが大きさ  $d$  が決められると, その文字列を表す平衡直線のプログラムは一意に決定される.

[例 1] 次のような平衡直線のプログラム  $B$  を考える:

$$\begin{aligned} X_1 &= a; X_2 = b; X_3 = X_1 \cdot X_2; X_4 = X_2 \cdot X_1; \\ X_5 &= X_1 \cdot X_1; X_6 = X_3 \cdot X_3; X_7 = X_4 \cdot X_5; \\ X_8 &= X_3 \cdot X_5; X_9 = X_6 \cdot X_7; X_{10} = X_9^{[1]} \cdot X_8. \end{aligned}$$

この例において,  $B = X_{10} = ababbbaaabaa$  となる. 図 5 に  $B$  の評価木を示す.  $X_9$  と  $X_8$  を根にもつ部分木は完全 2 分木をなし, これが高速な圧縮パターン照合を可能にする鍵となる.

平衡直線のプログラムに関する次の主定理を次章以降で証明する.

[定理 2] 平衡直線のプログラム  $T$  と  $P$  が与えられると, テキスト  $T$  中のパターン  $P$  のすべての出現位置の集合  $Occ(T, P)$  は  $O(n)$  の領域で表現でき, この表現は  $O(nm)$  の領域を用いて  $O(nm)$  時間で計算できる.

### 4. 完全 2 分木に対する計算

本章では,  $Occ^\Delta(X_i, Y_j)$  ( $1 \leq i < n, 1 \leq j < m$ ) の計算法について述べる. 変数  $X_i (= X_{l(i)} \cdot X_{r(i)})$  に対し, 高さ  $h$  ( $< height(X_i)$ ) の最右子孫  $rmd(X_i, h)$  と最左子孫  $lmd(X_i, h)$  を再帰的に定義する.

$$rmd(X_i, h)$$

$$= \begin{cases} rmd(X_{r(i)}, h) & \text{height}(X_i) > h+1 \text{ の場合,} \\ X_{r(i)} & \text{height}(X_i) = h+1 \text{ の場合,} \end{cases}$$

$$lmd(X_i, h)$$

$$= \begin{cases} lmd(X_{\ell(i)}, h) & \text{height}(X_i) > h+1 \text{ の場合,} \\ X_{\ell(i)} & \text{height}(X_i) = h+1 \text{ の場合.} \end{cases}$$

例えば,  $rmd(X_9, 2) = X_5$ ,  $rmd(X_8, 1) = X_1$ ,  $lmd(X_6, 2) = X_3$  となる (図 5 参照). 変数  $X_i$  ( $1 \leq i < n$ ) と高さ  $h$  ( $< \text{height}(Y)$ ) に対して,  $rmd(X_i, h)$  と  $lmd(X_i, h)$  のテーブルを前もって構築しておくことで, テーブル中の任意の  $rmd(X_i, h)$  と  $lmd(X_i, h)$  の値は  $O(1)$  で参照できる. このテーブルの構築には,  $O(nm)$  時間がかかる.

次の補題は  $Occ^\Delta(X_i, Y_j)$  に関する再帰的な関係式であり, 図 6 から導出できる.

[補題 3] 平衡直線的プログラム  $\mathcal{T}, \mathcal{P}$  中の変数  $X_i, Y_j$  ( $1 \leq i < n, 1 \leq j < m$ ) に対して,  $Y_j = Y_{\ell(j)} \cdot Y_{r(j)}$  とおくと, 集合  $Occ^\Delta(X_i, Y_j)$  に対して,

$$Occ^\Delta(X_i, Y_j) = Occ_\ell^\Delta(X_i, Y_j) \cup Occ_r^\Delta(X_i, Y_j)$$

が成立する. ここで,  $X_{\ell'(i)} = rmd(X_{\ell(i)}, \text{height}(Y_j))$ ,  $X_{r'(i)} = lmd(X_{r(i)}, \text{height}(Y_j))$  とおくと, 集合  $Occ_\ell^\Delta(X_i, Y_j)$  と  $Occ_r^\Delta(X_i, Y_j)$  は以下の式を満たす.

$$Occ_\ell^\Delta(X_i, Y_j) = Occ^\Delta(X_{\ell'(i)}, Y_{\ell(j)}) \cap Occ^\Delta(X_i, Y_{r(j)}),$$

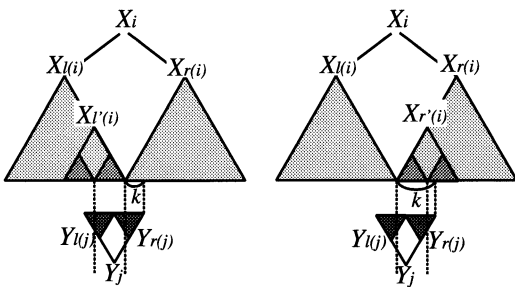


図 6 集合  $Occ^\Delta(X_i, Y_j)$  中の任意の要素  $k$  は,  $k \in Occ^\Delta(X_{\ell'(i)}, Y_{\ell(j)})$  かつ  $k \in Occ^\Delta(X_i, Y_{r(j)})$  (左図), 若しくは  $k - |Y_{r(j)}| \in Occ^\Delta(X_i, Y_{\ell(j)})$  かつ  $k - |Y_{r(j)}| \in Occ^\Delta(X_{r'(i)}, Y_{r(j)})$  (右図) を満たす

Fig. 6  $k \in Occ^\Delta(X_i, Y_j)$  if and only if either  $k \in Occ^\Delta(X_{\ell'(i)}, Y_{\ell(j)})$  and  $k \in Occ^\Delta(X_i, Y_{r(j)})$  (left case), or  $k - |Y_{r(j)}| \in Occ^\Delta(X_i, Y_{\ell(j)})$  and  $k - |Y_{r(j)}| \in Occ^\Delta(X_{r'(i)}, Y_{r(j)})$  (right case).

$$Occ_r^\Delta(X_i, Y_j) = Occ^\Delta(X_i, Y_{\ell(j)}) \oplus |Y_{r(j)}| \cap Occ^\Delta(X_{r'(i)}, Y_{r(j)}) \oplus |Y_{r(j)}|.$$

補題 1 より  $Occ^\Delta(X_i, Y_j)$  の要素は等差数列をなし, 和の演算は  $O(1)$  時間で計算できる. よって問題となるのは, 積の演算を効率良く行うことである. この問題を解く重要な鍵となる補題 5 の証明において, 次の周期性補題を利用する.

[補題 4] (周期性補題 (文献 [1], p.29 参照))  $p$  と  $q$  を文字列  $w$  の周期とすると,  $p+q-\text{gcd}(p, q) \leq |w|$  ならば, 最大公約数  $\text{gcd}(p, q)$  もまた  $w$  の周期である.

以降, 初項  $a$ , 公差  $d$ , 末項  $b$  の等差数列のなす集合を  $\langle a, d, b \rangle$  と表す.

[補題 5] 二つの等差数列の集合  $\langle a_1, d_1, b_1 \rangle = Occ^\Delta(X_{\ell'(i)}, Y_{\ell(j)})$  と  $\langle a_2, d_2, b_2 \rangle = Occ^\Delta(X_i, Y_{r(j)})$  の積集合  $\langle a, d, b \rangle = \langle a_1, d_1, b_1 \rangle \cap \langle a_2, d_2, b_2 \rangle$  は,  $O(1)$  時間で計算できる.

(証明) 集合  $\langle a_1, d_1, b_1 \rangle, \langle a_2, d_2, b_2 \rangle$  の要素数がいずれも 2 以下の場合と,  $d_1 = d_2$  の場合は明らかである.  $d_1 \neq d_2$  の場合, 一般性を失うことなく  $d_1 < d_2$  と仮定できるが, このとき次の二つの命題が成り立つことを証明する.

[命題 1]  $\langle a, d, b \rangle$  の要素数はたかだか 1 である.

[命題 2]  $\langle a, d, b \rangle \subseteq \{a_1, a_2, b_1, b_2\}$ .

以下の命題の証明において,  $X_{\ell'(i)} = X_L \cdot X_R$  とおき,  $T_1$  と  $T_2$  を次のように定義する.

$$T_1 = X_{\ell'(i)}[a_1 + 1 : b_1 + |X_L|],$$

$$T_2 = X_R[a_2 + 1 : b_2] \cdot X_{r(i)}[: b_2].$$

また,  $T_1$  と  $T_2$  の共通部分文字列を  $T'$  とすると,  $T' = X_R[a_2 + 1 : b_1]$  となる (図 7).

(命題 1 の証明)  $d_1$  と  $d_2$  をそれぞれ,  $T_1$  と  $T_2$  の最小の周期とする.  $\langle a, d, b \rangle$  が二つの異なる要素  $t_1$  と  $t_2$  ( $t_1 < t_2$ ) を含むと仮定し, 矛盾を導く. まず,  $t_1$  と  $t_2$  はいずれも, 集合  $\langle a_1, d_1, b_1 \rangle$  と  $\langle a_2, d_2, b_2 \rangle$  の要素なので,  $t_1 \geq a_2, t_2 \leq b_1$  となり,  $|T'| = b_1 - a_2 \geq t_2 - t_1$  が成り立つ.

次に,  $t_2 - t_1 \geq d_1 + d_2 - \text{gcd}(d_1, d_2)$  となることを示す.  $t_1, t_2 \in \langle a_1, d_1, b_1 \rangle \cap \langle a_2, d_2, b_2 \rangle$  なので, ある  $c_1, c_2 (\geq 1)$  に対して  $t_2 - t_1 = d_1 \cdot c_1 = d_2 \cdot c_2$  が成り立つ. ここで,  $c_1$  が  $c_2$  で割り切れるならば  $\text{gcd}(d_1, d_2) = d_1$  となり, 割り切れない場合は  $t_2 - t_1 = d_2 \cdot c_2 \geq 2d_2 > d_1 + d_2$  となる. いずれの場合においても  $t_2 - t_1 \geq d_1 + d_2 - \text{gcd}(d_1, d_2)$  とな

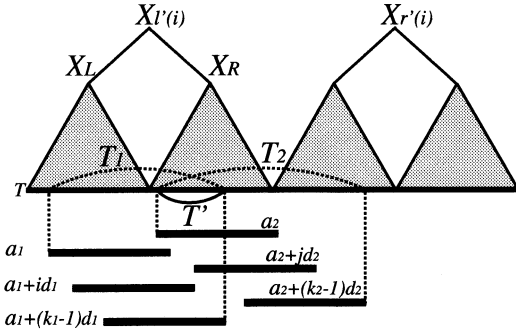


図 7 文字列  $T_1, T_2$  と  $T'$  の関係.  $d_1$  は  $T_1$  の周期で  $d_2$  は  $T_2$  の周期である  
 Fig. 7 Strings  $T_1, T_2$  and  $T'$ . Note that  $d_1$  is a period of  $T_1$  and  $d_2$  is a period of  $T_2$ .

り,  $|T'| \geq d_1 + d_2 - \gcd(d_1, d_2)$  が成り立つ.  
 $d_1$  と  $d_2$  は  $T'$  の周期なので, 周期性補題より,  $\gcd(d_1, d_2)$  も  $T'$  の周期となる. ところが  $\gcd(d_1, d_2) \leq d_1 < d_2$  なので,  $d_1$  と  $d_2$  が  $T'$  の最小の周期であるという事実に矛盾する. よって  $\langle a, d, b \rangle$  の要素数はただか一つである.  $\square$   
 (命題 2 の証明) 集合  $\langle a, d, b \rangle - \{a_1, a_2, b_1, b_2\}$  の要素  $t$  を仮定する. 集合  $\langle a_1, d_1, b_1 \rangle$  と  $\langle a_2, d_2, b_2 \rangle$  の要素数をそれぞれ  $k_1, k_2$  とする.  $\langle a, d, b \rangle = \langle a_1, d_1, b_1 \rangle \cap \langle a_2, d_2, b_2 \rangle$  なので, ある  $i, j$  ( $1 \leq i \leq k_1 - 2, 1 \leq j \leq k_2 - 2$ ) に対して,  $t = a_1 + i \cdot d_1 = a_2 + j \cdot d_2$  となる. よって

$$\begin{aligned} |T'| &= b_1 - a_2 \\ &= a_1 + (k_1 - 1) \cdot d_1 - a_2 \\ &= (t - a_2) + (k_1 - 1) \cdot d_1 - (t - a_1) \\ &= j \cdot d_2 + (k_1 - 1 - i) \cdot d_1 \\ &\geq d_1 + d_2. \end{aligned}$$

$d_1$  と  $d_2$  は  $T'$  の周期なので, 周期性補題より  $\gcd(d_1, d_2)$  もまた  $T'$  の周期である.  $\gcd(d_1, d_2) \leq d_1 < d_2$  なので  $\gcd(d_1, d_2)$  は  $T_2$  の最小の周期となり矛盾. よって  $\langle a, d, b \rangle \subseteq \{a_1, a_2, b_1, b_2\}$  が成り立つ.  $\square$

これらの命題により  $a_1, b_1 \in \langle a_2, d_2, b_2 \rangle$  と  $a_2, b_2 \in \langle a_1, d_1, b_1 \rangle$  を検証するだけで  $\langle a, d, b \rangle$  は求まるので  $\langle a, d, b \rangle$  は定数時間で計算でき, 補題 5 は成立する.  $\square$

補題 5 により,  $Occ_{\ell}^{\Delta}(X_i, Y_j)$  と  $Occ_{r}^{\Delta}(X_i, Y_j)$  の

積演算は  $O(1)$  時間で行える.  $Occ^{\Delta}(X_i, Y_j)$  を再帰的に求める際, 同じ集合  $Occ^{\Delta}(X'_i, Y'_j)$  ( $i' < i, j' < j$ ) が複数回参照されるので, 動的計画法を用いる. つまり, 集合  $Occ^{\Delta}(X_i, Y_j)$  の要素を表す三つ組を格納する  $n$  行  $m$  列の表  $App[i, j]$  をボトムアップに構築する. この表  $App$  は  $O(nm)$  領域を占める.  $1 \leq i < n$  かつ  $1 \leq j < m$  なる  $i$  と  $j$  の領域に対しては, この章の議論によって  $O(nm)$  で構築できることが示された. 次の章で,  $i = n$  または  $j = m$  の領域に対して,  $O(nm)$  時間で表の要素を計算できることを示す.

### 5. 最後の代入文に対する計算

この章では最後の変数を含めた  $Occ(X_n, Y_m)$  の計算手法について述べる. 以下の議論では  $Y_{\ell(j)} \geq Y_{r(j)}$  と仮定するが, それ以外の場合も同様に証明される.

平衡直線的プログラム  $\mathcal{T}, \mathcal{P}$  中の変数  $X_i, Y_j$  ( $1 \leq i \leq n, 1 \leq j < m$ ) と整数  $k$  ( $1 \leq k \leq |X_n|$ ) に対し,

$$S(X, Y, k) = \left\{ s \mid \begin{array}{l} k - |Y| + 1 \leq s \leq k + 1 \text{ かつ} \\ X[s : s + |Y| - 1] = Y \end{array} \right\}$$

とする.  $S(X, Y, k)$  は, テキスト  $X$  のある位置  $k$  をまたいで現れるパターン  $Y$  の出現位置を表す集合で, 補題 1 と同じ議論により, その要素は等差数列をなす. なお, 定義より  $Occ^{\Delta}(X, Y) = S(X, Y, |X_{\ell}|) \oplus (|Y| - |X_{\ell}|)$  が成り立つ.

[補題 6] 平衡直線的プログラム  $\mathcal{T}, \mathcal{P}$  中の変数  $X_i, Y_j$  ( $1 \leq i < n, 1 \leq j < m$ ) と整数  $k$  が与えられたとき, 集合  $S(X_i, Y_j, k)$  は表  $App[i, j]$  ( $1 \leq i < n, 1 \leq j < m$ ) を参照しながら  $O(\text{height}(X_i))$  時間で計算できる.

(証明)  $X_{\ell(j)} > X_{r(j)}$  の場合について示す. それ以外の場合も同様に証明できる.

$$\begin{aligned} X_{\ell'(i)} &= rmd(X_{\ell(i)}, \text{height}(Y_{\ell(j)})), \\ X_{r'(i)} &= lmd(X_{r(i)}, \text{height}(Y_{\ell(j)})) \end{aligned}$$

とし,  $k$  の値による場合分けを行う.

場合 (1)  $1 \leq k < |X_{\ell(i)}| - |Y_j|$  の場合,

$$S(X_i, Y_j, k) = S(X_{\ell(i)}, Y_j, k).$$

場合 (2)  $|X_{\ell(i)}| - |Y_j| \leq k < |X_{\ell(i)}|$  の場合,

$$S(X_i, Y_j, k)$$

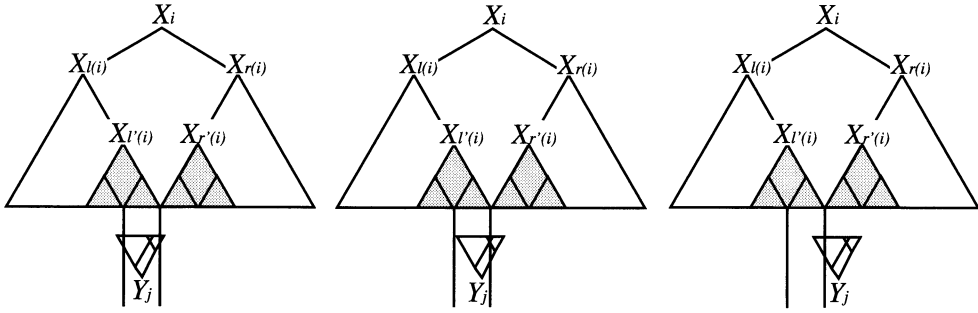


図 8  $Occ_1^\Delta(X_i, Y_j)$  (左),  $Occ_2^\Delta(X_i, Y_j)$  (中央),  $Occ_3^\Delta(X_i, Y_j)$  (右)  
 Fig. 8  $Occ_1^\Delta(X_i, Y_j)$  (left),  $Occ_2^\Delta(X_i, Y_j)$  (center) and  $Occ_3^\Delta(X_i, Y_j)$  (right).

$$= \left\{ s \left| \begin{array}{l} k - |Y_j| + 1 \leq s \leq k + 1 \text{ かつ} \\ s \in (Occ^\Delta(X_i, Y_j) \oplus |X_{\ell(i)}| \oplus |X_{\ell'(i)}| \\ \cup Occ^\Delta(X_{\ell'(i)}, Y_j) \oplus |X_{\ell(\ell'(i))}| \oplus |Y_j|) \end{array} \right. \right\}.$$

場合 (3)  $|X_{\ell(i)}| \leq k \leq |X_{\ell(i)}| + |Y_j|$  の場合,

$$S(X_i, Y_j, k) = \left\{ s \left| \begin{array}{l} k - |Y_j| + 1 \leq s \leq k + 1 \text{ かつ} \\ s \in (Occ^\Delta(X_i, Y_j) \oplus |X_{\ell(i)}| \oplus |Y_j| \\ \cup Occ^\Delta(X_{r'(i)}, Y_j)) \oplus |X_{\ell(i)}| \\ \oplus |X_{\ell(r'(i))}| \oplus |Y_j| \end{array} \right. \right\}.$$

場合 (4)  $|X_{\ell(i)}| + |Y_j| < k \leq |X_i|$  の場合,

$$S(X_i, Y_j, k) = S(X_{r(i)}, Y_j, k - |X_{\ell(i)}|) \oplus |X_{\ell(i)}|.$$

(1) と (4) については  $O(\text{height}(X_i))$  時間で、(2) と (3) については定数時間でそれぞれ  $S(X_i, Y_j, k)$  が計算できることがわかる。よって補題は成立する。□

次の補題は、補題 2 の一般化である。

[補題 7] 平衡直線のプログラム  $\mathcal{T}, \mathcal{P}$  中の変数  $X_i (1 \leq i \leq n), Y_m$  に対して、 $X_i = X_{\ell(i)}^{[d_x]} \cdot X_{r(i)}$  としたとき、次式が成立する：

$$Occ(X_i, Y) = Occ(X_{\ell(i)}, Y_m) \cup (Occ^\Delta(X_i, Y_m) \oplus |X_{\ell(i)}| \oplus |Y_m|) \cup (Occ(X_{r(i)}, Y_m) \oplus |X_{\ell(i)}| \oplus d_x).$$

表  $App$  の要素  $App[i, m] (1 \leq i < n)$  は次の補題により求まる。

[補題 8]  $Occ^\Delta(X_i, Y_m) (1 \leq i < n)$  の各要素は  $O(m)$  時間で計算できる。

(証明) 平衡直線のプログラム  $\mathcal{T}, \mathcal{P}$  中の変数  $X_i = X_{\ell(i)} \cdot X_{r(i)}$  と  $Y_j$  について、 $X_{\ell'(i)} = rmd(X_{\ell(i)}, \text{height}(Y_j))$ ,  $X_{r'(i)} = lmd(X_{r(i)}, \text{height}(Y_j))$  とおく。このとき、 $Y_j = Y_{\ell(j)}^{[d_y]} \cdot Y_{r(j)}$  に対して、

$$Occ^\Delta(X_i, Y_j) = Occ_1^\Delta(X_i, Y_j) \cup Occ_2^\Delta(X_i, Y_j) \cup Occ_3^\Delta(X_i, Y_j),$$

が成り立つ(図 8 参照)。ここに

$$\begin{aligned} Occ_1^\Delta(X_i, Y_j) &= Occ^\Delta(X_{\ell'(i)}, Y_{\ell(j)}) \oplus d_y \oplus |Y_{r(j)}| \\ &\quad \oplus |Y_{\ell(j)}| \cap Occ^\Delta(X_i, Y_{r(j)}), \\ Occ_2^\Delta(X_i, Y_j) &= Occ^\Delta(X_i, Y_{\ell(j)}) \oplus d_y \oplus |Y_{r(j)}| \\ &\quad \cap Occ^\Delta(X_i, Y_{r(j)}), \\ Occ_3^\Delta(X_i, Y_j) &= Occ^\Delta(X_i, Y_{\ell(j)}) \oplus d_y \oplus |Y_{r(j)}| \\ &\quad \cap Occ(X_{r'(i)}, Y_{r(j)}) \oplus |Y_{r(j)}|. \end{aligned}$$

集合  $Occ_1^\Delta(X_i, Y_j), Occ_2^\Delta(X_i, Y_j)$  は補題 5 により定数時間で計算できる。よって、 $Occ_3^\Delta(X_i, Y_j)$  が  $O(m)$  時間で計算できることを示せば十分である。 $Occ^\Delta(X_i, Y_{\ell(j)}) \oplus d_y \oplus |Y_{r(j)}|$  を  $\langle a, d, b \rangle$  とおき、 $Occ(X_{r'(i)}, Y_{r(j)}) \oplus |Y_{r(j)}|$  を  $C$  とおく。補題 7 より、集合  $C$  中の最小の要素  $c$  は、 $App[i, j]$  から  $O(\text{height}(X_{r'(i)}))$  時間で求まる。集合  $\langle a, d, b \rangle$  の分割  $A, B$  を次のように定義する。

$$\begin{aligned} A &= \{x \mid x \in \langle a, d, b \rangle \text{ かつ } x \leq b - |Y_{r(j)}| + d_y\}, \\ B &= \{x \mid x \in \langle a, d, b \rangle \text{ かつ } x > b - |Y_{r(j)}| + d_y\}. \end{aligned}$$

$d$  は  $X_i [|X_{\ell(i)}| - |Y_{\ell(j)}| + 1 : |X_{\ell(i)}| + |Y_{\ell(j)}|]$  の周期なので、集合  $A \cap C$  は  $c \in A$  の場合、

$$A \cap C = \left\{ x \left| \begin{array}{l} x \in \langle c, d, b \rangle \text{ かつ} \\ x \leq b - |Y_{r(j)}| + d_y \end{array} \right. \right\}$$

となり、それ以外の場合は空集合になる。よって集合  $A \cap C$  は  $O(1)$  時間で求められる。一方、

$$B \cap C = B \cap S(X_{r'(i)}, Y_{r(j)}, b-1) \oplus |Y_j|$$

が成り立つが、補題 6 より集合  $S(X_{r'(i)}, Y_{r(j)}, b)$  は  $O(\text{height}(X_{r'(i)})) = O(\text{height}(Y_j)) = O(m)$  時間で求まり、その要素は等差数列をなすので、補題 5 から  $B \cap C$  は  $O(m)$  時間で計算できる。よって集合  $Occ_3^\Delta(X_i, Y_j)$  は  $O(m)$  時間で求まる。以上より、補題は成り立つ。□

表  $App$  の要素  $App[n, j] (1 \leq j \leq m)$  は次の補題により求まる。

[補題 9]  $Occ^\Delta(X_n, Y_j) (1 \leq j \leq m)$  の各要素は  $O(n)$  時間で計算できる。

(証明)  $X_{\ell'(i)} = \text{rmd}(X_{\ell(i)}, \text{height}(Y_j))$  とおくと、

$$\begin{aligned} Occ^\Delta(X_n, Y_j) &= Occ_\ell^\Delta(X_n, Y_j) \cup Occ_r^\Delta(X_n, Y_j), \end{aligned}$$

ここに、

$$\begin{aligned} Occ_\ell^\Delta(X_n, Y_j) &= Occ^\Delta(X_{\ell'(n)}, Y_{\ell(j)}) \cap Occ^\Delta(X_n, Y_{r(j)}), \\ Occ_r^\Delta(X_n, Y_j) &= Occ^\Delta(X_n, Y_{\ell(j)}) \oplus |Y_{r(j)}| \\ &\quad \cap Occ(X_n, Y_{r(j)}) \oplus |Y_{r(j)}| \ominus |X_{\ell(n)}|. \end{aligned}$$

集合  $Occ^\Delta(X_n, Y_j)$  は補題 1 より等差数列をなし、 $Occ_\ell^\Delta(X_n, Y_j)$  と  $Occ_r^\Delta(X_n, Y_j)$  の和演算は定数時間で計算できる。また、 $Occ_\ell^\Delta(X_n, Y_j)$  は補題 5 より定数時間で、 $Occ_r^\Delta(X_n, Y_j)$  は補題 8 中の  $Occ_3^\Delta(X_i, Y_j)$  の計算法と同様に  $O(\text{height}(X_n)) = O(n)$  時間で計算できる。よって補題は成り立つ。□

以上により、定理 2 の証明が得られる。

(定理 2 の証明) 平衡直線的プログラム  $\mathcal{T}$ 、 $\mathcal{P}$  が与えられたとき、出現位置を表す集合  $Occ(\mathcal{T}, \mathcal{P}) = Occ(X_n, Y_m)$  は補題 7 より、 $O(n)$  領域の  $\{Occ^\Delta(X_i, Y_m)\}_{i=1}^n$  の組合せで表現できる。また、各  $Occ^\Delta(X_i, Y_j) (1 \leq i \leq n, 1 \leq j \leq m)$  は補題 8, 9 より計算でき、全体の計算時間は  $O(nm)$  時間領域となる。□

## 6. む す び

本論文で導入した平衡直線のプログラムは、本質的には Kieffer らの MPM (Multilevel Pattern Matching) 符号 [3] と同一であるが、もとの文字列の長さが 2 のべき数 (以降、べき文字列と呼ぶ) でないときの対処法に違いがある。MPM では、任意の文字列をべき文字列の単純な連結として表しているが、この場合、文字列の長さ  $n$  に対して連結の回数は  $O(\log n)$  となり、我々のアルゴリズムを直接適用することは困難である。平衡直線のプログラムにおいては、最後の代入文に切り取り命令を導入することによって、たかだか二つのべき文字の和として任意の文字列を表している。

本論文では、文法に基づく圧縮法という観点においては「中間表現」である文法 (プログラム) を入力とする照合アルゴリズムを考察してきた。一方、Miyazaki ら [7] は、ハフマン符号による符号語上を直接まとめ読みしながらパターン照合機械の状態遷移を行うことによって高速な圧縮文字列照合を行っている。文法に基づく圧縮法の枠組みにおいても、符号語上の操作を含めたより深い洞察を行うことが今後の課題である。

謝辞 文献 [2], [3], [5] の紹介をはじめとして非常に有益なコメントを頂いた査読者の方々に深謝致します。

## 文 献

- [1] M. Crochemore and W. Rytter, Text Algorithms, Oxford University Press, New York, 1994.
- [2] E.-H. Yang and J.C. Kieffer, "Efficient universal lossless data compression algorithms based on a greedy sequential grammar transform," IEEE Trans. Inf. Theory, vol.46, no.3, pp.755-777, 2000.
- [3] E.-H. Yang, G.J.N. John, C. Kieffer, and P. Cosman, "Universal lossless compression via multilevel pattern matching," IEEE Trans. Inf. Theory, vol.46, no.4, pp.1227-1245, 2000.
- [4] T. Kida, Y. Shibata, M. Takeda, A. Shinohara, and S. Arikawa, "A unifying framework for compressed pattern matching," Proc. 6th International Symposium on String Processing and Information Retrieval, pp.89-96, 1999.
- [5] J.C. Kieffer and E.-H. Yang, "Grammar-based codes: A new class of universal lossless source codes," IEEE Trans. Inf. Theory, vol.46, no.3, pp.737-754, 2000.
- [6] N.J. Larsson and A. Moffat, "Offline dictionary-based compression," Proc. Data Compression Conference '99, pp.296-305, IEEE Computer Society, 1999.
- [7] M. Miyazaki, S. Fukamachi, M. Takeda, and T. Shinohara, "Speeding up the pattern matching machine for compressed texts," Trans. Information Processing Society of Japan, vol.39, no.9, pp.2638-2648,

1998.

- [8] M. Miyazaki, A. Shinohara, and M. Takeda, "An improved pattern matching algorithm for strings in terms of straight-line programs," Proc. 8th Annual Symposium on Combinatorial Pattern Matching, vol.1264, Lecture Notes in Computer Science, pp.1-11, Springer-Verlag, 1997.
- [9] C. Nevill-Manning and I. Witten, "Compression and explanation using hierarchical grammars," Comput. J., vol.40, no.2/3, pp.103-116, 1997.
- [10] C. Nevill-Manning and I. Witten, "Identifying hierarchical structure in sequences: A linear-time algorithm," J. Artificial Intelligence Research, vol.7, pp.67-82, 1997.
- [11] W. Rytter, "Algorithms on compressed strings and arrays," Proc. 26th Annual Conference on Current Trends in Theory and Practice of Informatics, vol.1725 of Lecture Notes in Computer Science, pp.48-65, Springer-Verlag, 1999.
- [12] Y. Shibata, T. Kida, S. Fukamachi, M. Takeda, A. Shinohara, T. Shinohara, and S. Arikawa, "Speeding up pattern matching by text compression," Proc. 4th Italian Conference on Algorithms and Complexity, vol.1767 of Lecture Notes in Computer Science, pp.306-315, Springer-Verlag, 2000.
- [13] Y. Shibata, T. Matsumoto, M. Takeda, A. Shinohara, and S. Arikawa, "A Boyer-Moore type algorithm for compressed pattern matching," Proc. 11th Annual Symposium on Combinatorial Pattern Matching, vol.1848 of Lecture Notes in Computer Science, pp.181-194, 2000.

(平成12年9月25日受付, 13年1月9日再受付)



竹田 正幸

1987 九大・理・数学卒. 1989 同大大学院総合理工学研究科情報システム学専攻修士課程了, 同年より同大工学部電気工学科助手. 1996 より同大大学院システム情報科学研究科情報理学部門助教授, 現在に至る. 博士(工学). パターン照合アルゴリズム, テキスト圧縮, 発見科学, 情報検索等の研究に従事.



有川 節夫

1964 九大・理・数学卒. 1966 同大大学院理学研究科数学専攻修士課程了. 同大理学部助手, 京都大学数理解析研究所助手等を経て, 現在, 九州大学大学院システム情報科学研究科情報理学部門教授. この間, 大型計算機センター長, 評議員, 附属図書館長等を歴任. 理博. 現在, 機械学習と発見科学, 情報検索システム等の研究に従事.



平尾 昌啓 (学生員)

1999 九大・理・物理卒. 2001 同大大学院システム情報科学研究科情報理学専攻修士課程了. 同年同大学院システム情報科学府情報理学専攻博士後期課程進学, 現在に至る. パターン照合アルゴリズムや情報検索を中心に研究する.



篠原 歩

1988 九大・理・数学卒. 1990 同大大学院総合理工学研究科情報システム学専攻修士課程了. 同年より同大理学部附属基礎情報科学研究施設助手. 1994 より同施設助教授. 1996 より同大大学院システム情報科学研究科情報理学部門助教授, 現在に至る. 博士(理学). 発見科学, ゲノム情報処理, 文字列照合等に興味をもつ.