数理解析研究所講究録
522巻 1984年 77-89



# $k \log n$ 決定性領域を必要とする問題

東海大学　岩田茂樹（Shigeki Iwata）
電気通信大学　笠井琢美（Takumi Kasai）

## 1. はじめに

問題の複雑さが tight な下限をもつようなものは，そう多く見つかっているわけではない。決定性多項式時間で解ける問題のうち，ある2人ゲームに関する先手必勝問題は $\Omega(n^k)$ 時間必要とすること[1]や，非決定性対数領域に属する問題のうち $k \log n$ 非決定性領域を必要とするような自然な問題が見つかっている[5]。これらはいずれも"石おきゲーム"[4]の概念を利用している。

本稿では finite transducer に関する問題を考え，その問題が $k \log n$ 決定性領域を必要とすること，すなわち，2個のテープ記号の決定性チューリングマシンで $k \log n$ 以下の領域では解けないことを示す。

Hong [2] は決定性領域で，$O(n)$ で解けるが，$o(n)$ では解けないような問題を構成した。本稿で考える問題（$FSTP_k$）

は，決定性領域で，任意の $\varepsilon>0$ について，$(k+\varepsilon)\log n$ で解けるが，$(k-4-\varepsilon)\log n$ では解けないことを示す。

## 2．準備

本稿で使用する計算モデルは決定性のチューリングマシンで，2方向読込専用入力テープと1本の記憶テープからなる。記憶テープのテープ記号は特にことわらない限り $\{0,1\}$ とする。計算のはじめは記憶テープのすべてのセルはすべて0とする。また，2個のテープヘッドは，決められた領域の外側に出ることはないと仮定する。本稿を通じて，$\Sigma=\{0,1\}$ とし，$|w|$ は列 $w$ の長さ，$\lceil a\rceil$ は $a$ より大きいか等しい最小の数を表わすものとし，すべての対数の底を2とする。

長さ $n$ の入力に対し，記憶テープの $S(n)$ セルしか使用しなければ，チューリングマシン $M$ は<u>$S(n)$ 領域で計算する</u>という。$L\subseteq\Sigma^*$ について，$L$ が $S(n)$ 領域で計算するチューリングマシンにより受理されるならば，$L$ は<u>$S(n)$ 領域で解ける</u>という。$DSPACE_2(S(n))$ を $S(n)$ 領域で解ける言語のクラスとする。

<u>定義</u>　$L\subseteq\Sigma^*$ を $\Sigma$ 上の<u>問題</u>という。問題 $L$ を受理するチューリングマシンの計算に要する領域が，長さ $n$ の無限に多くの入力について $S(n)$ 領域を越えるとき，$L$ は<u>$S(n)$ 領域を必要とする</u>という。すなわち，$\sup S'(n)/S(n)<1$ であるどの $S'(n)$ についても，$L\notin DSPACE_2(S'(n))$ のとき，$L$ は $S(n)$ 領域を必

要とする。

関数 $f:\Sigma^*\to\Sigma^*$ は，一方向書出し専用出力テープをもつチューリングマシンMが存在して，Mの入力 $w\in\Sigma^*$ に対し，記憶テープを $S(|w|)$ セルより多く使用することなしに，出力テープに $f(w)$ を出力して停止するとき，<u>$S(n)$ 領域計算可能であるという</u>。

<u>定義</u>　$S, Z$ を非負整数上の単調増加関数とする。$L_1, L_2\subseteq\Sigma^*$ について，$L_1$ は $L_2$ に <u>$(S,Z)$-reducible</u> であるとは，$S(n)$ 領域計算可能な関数 $f:\Sigma^*\to\Sigma^*$ が存在し，

(1) $w\in L_1 \Leftrightarrow f(w)\in L_2$,

(2) 任意の $w\in\Sigma^*$ について，$|f(w)|\leq Z(|w|)$

を満たすときをいう。

次の補助定理は，Seiferas [6] の証明と同じ手法により証明される。

<u>補助定理 2.1</u>　$L\subseteq\Sigma^*$ は，テープ記号 $\{0,1,\#\}$，# は記憶テープ上に高々 $k$ 個しかあらわれないような $S(n)$ 領域限定チューリングマシンにより受理されるとする。このとき $L$ はテープ記号 $\{0,1\}$ のチューリングマシンにより

$$S(n)+(2k+2)\lceil \log S(n)\rceil$$

領域で受理される。

次の補助定理は，Jones [3] と補助定理 2.1 よりえられる。

<u>補助定理 2.2</u>　$L_1, L_2 \subseteq \Sigma^*$ とする。$L_1$ が $L_2$ に $(S,Z)$-reducible で $L_2$ が $S_2(n)$ 領域で解けるならば，$L_1$ は

$$\hat{S}(n) + 16\lceil \log \hat{S}(n) \rceil$$

領域で解ける。ただし，$\hat{S}(n) = S_2(Z(n)) + 2\lceil \log Z(n) \rceil + S(n)$.

3. $k \log n$ 決定性領域を必要とする問題

finite state transducer の問題を考える。<u>finite state transducer</u> は5項組 $(K, \Gamma, \Delta, \delta, p_0)$ で $K, \Gamma, \Delta$ はそれぞれ，<u>状態</u>，<u>入力アルファベット</u>，<u>出力アルファベット</u>である。$\delta$ は $K \times \Gamma$ から $K \times \Delta$ への写像であり <u>next move function</u> という。$p_0$ は<u>初期状態</u>である。$\delta(q, a) = (p, b)$ は，S が状態 $q$ で入力記号 $a$ を読むと，状態 $p$ に行き出力記号 $b$ を出力することをあらわす。S は決定性であることに注意する。

$\delta$ を $K \times \Gamma^*$ からの関数になるように次のように拡張する。

(1) $\delta(q, \lambda) = (q, \lambda)$，ただし $\lambda$ は空系列をあらわす。

(2) ある $p'$ について $\delta(q, x) = (p', w)$, $\delta(p', a) = (p, b)$ ならば，$\delta(q, xa) = (p, wb)$。ただし，$x \in \Gamma^*$, $a \in \Gamma$, $w \in \Delta^*$, $b \in \Delta$.

$\Gamma = \Delta$ とする。$w \in \Gamma^*$, $q \in K$ とする。<u>j-fold $\delta$ 関数</u> $\delta^{(j)}$ は $K \times \Gamma^*$ から $K \times \Gamma^*$ への関数で，

$$\begin{cases} \delta^{(j)}(q, w) = \delta(\delta^{(j-1)}(q, w)) & j > 0 \\ \delta^{(0)}(q, w) = (q, w) \end{cases}$$

をみたす。

<u>長さ $k$ の finite state transducer の問題</u> ($FSTP_k$) とは，

入力： $S=(K,\Gamma,\Gamma,\delta,p_0)$ , $a,b\in\Gamma$

決定すること： ある整数 $j$ と $p\in K$ が存在して，

$$\delta^{(j)}(p_0,a^k)=(p,b^k)$$

となるかどうか。

<u>定理 3.1</u> $L\subseteq\Sigma^*$, $L\in DSPACE_2(k\log n)$ とする。このとき，定数 $c$ が存在して任意の $\varepsilon>0$ について，$L$ は $FSTP_{k+1}$ に $((1+\varepsilon)\log n,\ cn\log^2 n)$-reducible である。

<u>証明</u> $L$ を受理する $k\log n$ 領域限定チューリングマシンを $M$ とする。$M$ の状態集合を $Q$，$\Sigma$ を入力記号の集合とテープ記号の集合とする。$q_s$, $q_f$ をそれぞれ初期状態，ただ一つの受理状態とする。$M$ の next state function は $Q\times\Sigma\times\Sigma\times Q\times\{L,R,S\}\times(\Sigma\times\{L,R,S\})$ の要素であらわすことにし，$M$ の <u>instruction</u> ということにする。instruction $r=(q,\sigma,\eta,q',d,(\eta',d_w))$ は，$M$ が状態 $q$ で，入力記号 $\sigma$，テープ記号 $\eta$ を読むと，状態 $q'$ に移り，入力テープと記憶テープのヘッドをそれぞれ $d,d_w$ に移動することを意味する。

この証明を通して $m=\lceil\log n\rceil$ とする。記憶テープの領域 $k\log n$ を長さ $\log n$ の $k$ 個のブロックに分割する。

Mの<u>様態</u>を $C=(q, h, z_0 z_1 \cdots z_{k-1})$ であらわす。ただし，$q$ はMの状態，$h$ は入力テープのヘッドの位置，$0 \leq h < n$，$z_i$ は記憶テープの $i$ 番目のブロックの内容とヘッドの位置をあらわす。すなわち，記憶テープのヘッドが $u$ 番目のブロックの $v$ 番目のセル上にあるならば，$i \in \{0, 1, \cdots, k-1\} - \{u\}$ については $z_i \in \Sigma^m$，また $z_u \in \Sigma^v \uparrow \Sigma^{m-v}$ である。

Mの2本のテープのヘッドの位置と記憶テープの内容を長さ $k+1$ の記号列 $\alpha = [z_0][z_1] \cdots [z_{k-1}]\langle h \rangle$ で表わす。$[z_0][z_1] \cdots [z_{k-1}]$ は記憶テープの内容とヘッドの位置を表わす。$[z_i]$ を1つの記号として扱う。$z_i \in \Sigma^m \cup (\bigcup_{i=0}^{m-1} \Sigma^i \uparrow \Sigma^{m-i})$ である。$\langle h \rangle$ は入力テープのヘッドの位置を表わす。$\langle h \rangle$ も1つの記号として扱う。

$S_{M,x}$ の入出力アルファベット $\Gamma$ を次のようにする：

$$\Gamma = \{a, b\} \cup \{\langle h \rangle \mid 0 \leq h < n\} \cup \{[z] \mid z \in \Sigma^m \cup (\bigcup_{i=0}^{m-1} \Sigma^i \uparrow \Sigma^{m-i})\}$$

$S_{M,x}$ の next move function $\delta$ は次の3つの部分，(1) 初期化，(2) シミュレーション，(3) 後処理よりなる。$\delta$ の図示のため，$\delta(q, a) = (p, b)$ を図1に示す。

<u>(1) 初期化</u>　$a^{k+1}$ を $\alpha_0 = [\uparrow 0^m][0^m] \cdots [0^m]\langle 0 \rangle$ に変換し，状態を $q_{k+1}$ にするためのもので，$\delta$ は図2に示す move を含むものとする。

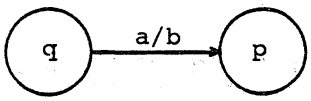


図1 An illustration of $\delta(q,a) = (p,b)$

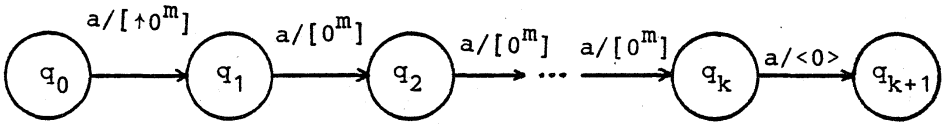


図2 Initialization part of the next move function of $S_{M,x}$

<u>(2) シミュレーション</u> ここでは，Mの動きをシミュレートする。$S_{M,x}$ の状態で，$\langle q, v_1, v_2 \rangle$ の形のものを使用する。ただし，$q$はMの状態，$v_1 \in \{p_1, p_2, r_1, \cdots, r_k\}$，$v_2 \in \{0,1\} \cup \{L, R, S\}$ とする。$v_1$ は補助的に使用する。$q_{k+1} = \langle q_s, p_1, \sigma_0 \rangle$ とする。ただし，$\sigma_0$ はMの入力テープの左端の記号（ビット）とする。

Mの各 instruction $r = (q, \sigma, \eta, q', d, (\eta', d_w))$ について，$S_{M,x}$ の next move function はすべての $w \in \Sigma^*$，$z \in \bigcup_{i=0}^{m-1} \Sigma^i \uparrow \Sigma^{m-i}$，$h$ $(0 \le h < n)$ について図3に示すものを含む。ただし，$h+d$ は $r$ の実行後の入力テープのヘッド

の位置を表わし，$\sigma_h$ は入力テープの $h$ 番目の記号とする。図3を参照せよ。もし $r$ の実行後記憶テープのヘッドの位

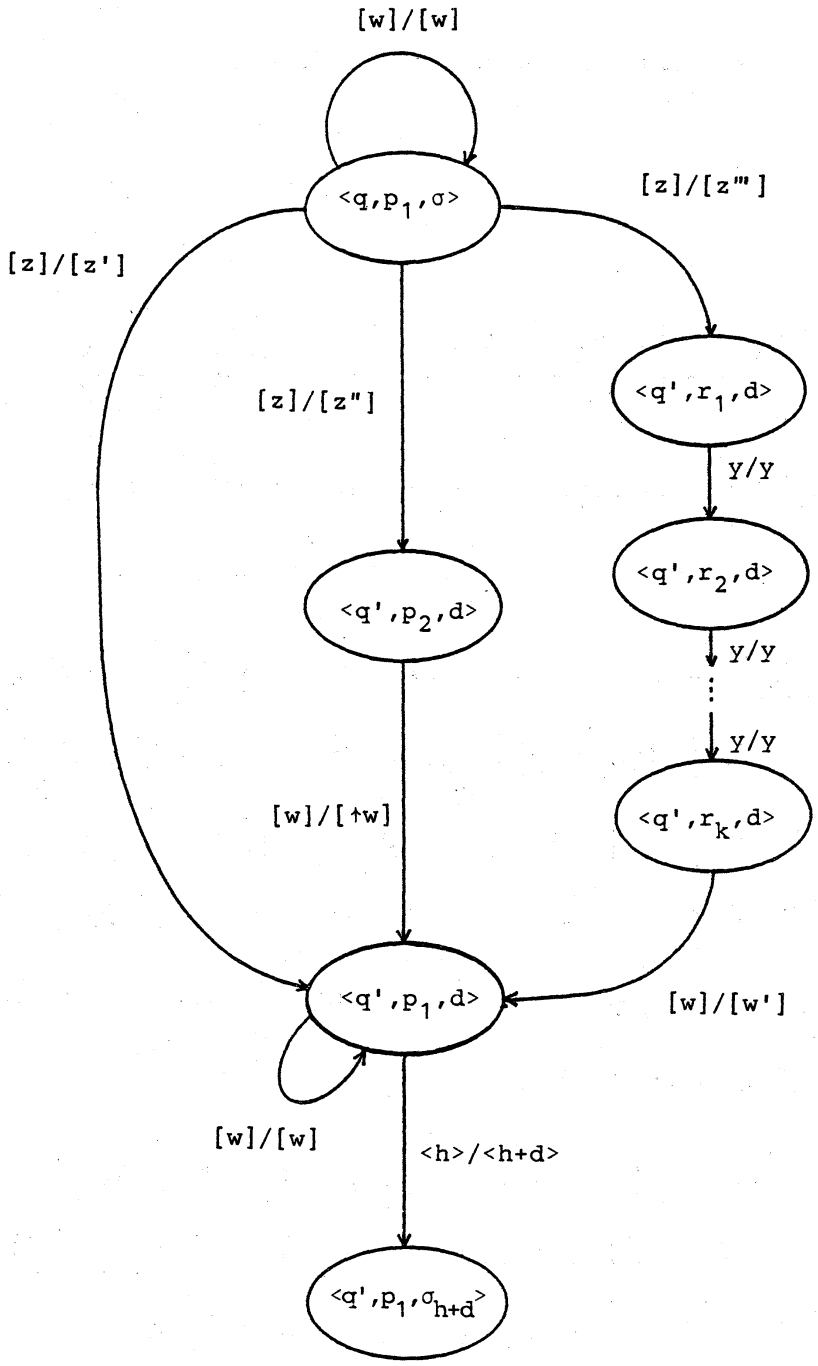


図 3 Simulation part of the next move function of $S_{M,x}$

置が実行前と同じブロックであれば，$\delta$は次を含む。

$$\delta(\langle q, p_1, \sigma\rangle, [z]) = (\langle q', p_1, d\rangle, [z'])$$

ただし，$z'$は$r$の実行後のそのブロックの内容とヘッドの位置をあらわす。すなわち，$z = w_1 \uparrow \eta w_2$, $w_1 w_2 \in \Sigma^{m-1}$, $\eta \in \Sigma$ ならば，$z'$は$z$の$\eta$を$\eta'$におきかえ，更に$\uparrow$の位置を$d_w \in \{L, R, S\}$により左，右に移動したり，そのままにしてえられる記号列である。

記憶テープのヘッドの位置が$r$の実行後右のブロックに移るならば，$\delta$は次を含む。

$$\delta(\langle q, p_1, \sigma\rangle, [z]) = (\langle q', p_2, d\rangle, [z''])$$

すべての$w \in \Sigma^m$について

$$\delta(\langle q', p_2, d\rangle, [w]) = (\langle q', p_1, d\rangle, [\uparrow w])$$

ただし，$z''$は$z$の右端の2記号$\uparrow\eta$を$\eta'$におきかえてえられる記号列である。

記憶テープのヘッドの位置が$r$の実行後左のブロックに移るならば，$\delta$は次を含む。

$$\delta(\langle q, p_1, \sigma\rangle, [z]) = (\langle q', r_1, d\rangle, [z'''])$$,

すべての$i$ $(1 \leq i < k)$,

$y \in \{[z] \mid z \in \Sigma^m \cup (\bigcup_{i=0}^{m-1} \Sigma^i \uparrow \Sigma^{m-i})\} \cup \{\langle h\rangle \mid 0 \leq h < n\}$について

$$\delta(\langle q', r_i, d\rangle, y) = (\langle q', r_{i+1}, d\rangle, y),$$

すべての$w \in \Sigma^m$について，

$$\delta(\langle q', r_k, d\rangle, [w]) = (\langle q', p_1, d\rangle, [w'])$$

ただし，$z'''$ は $z$ の左端の2記号 $\uparrow\eta$ を $\eta'$ におきかえてえられる記号列で，また，$w = w_1\xi$，$w_1 \in \Sigma^{n-1}$，$\xi \in \Sigma$ ならば $w' = w_1 \uparrow \xi$ である。

記憶テープのヘッドの位置は $r$ の実行によって，同じブロックにあるか，左かまたは右のブロックに移るかのいずれかである。よって，以上で定めた $\delta$ は決定性であることに注意する。

<u>(3) 後処理</u>　$S_{M,x}$ の状態が，ある $\sigma \in \Sigma$ について $\langle q_f, p_1, \sigma\rangle$ になると，$b^{k+1}$ をつくり出すための next move function である。$\delta$ はすべての $u \in \Gamma$ について図4に示すものを含む。

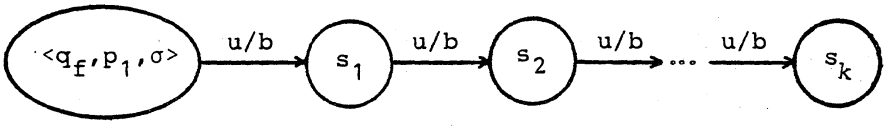


図4　Final treatment part of the next move function of $S_{M,x}$

$S_{M,x}$ の状態集合 $K$ は，図2，図3，図4にあらわれるすべての状態よりなる。

以上のように定めた $S_{M,x}$ については，

$M$ は $x$ を受理する $\Leftrightarrow$ ある $j$ が存在して $\delta^{(j)}(q_0, a^{k+1}) = (s_k, b^{k+1})$

がなりたつ。

次に $S_{M,x}$ の表現の長さと，Mと$x$から $S_{M,x}$ を構成するのに要する領域量を計算する。Mの状態数は $n$ に無関係な定数個であるので，$S_{M,x}$ の状態数も定数個である。$\{[z] \mid z \in \bigcup_{i=0}^{m-1} \Sigma^i \uparrow \Sigma^{m-i}\}$ の要素の数は $n \cdot m$ 個である。よって $\Gamma$ 中の記号の数は，$c_1 n \log n$ 個以下である。ただし，$c_1$ は定数。$\Gamma$ の記号は任意の $\varepsilon > 0$ について長さ $(1+\varepsilon)\log n$ で表わすことができる。よって $S_{M,x}$ の長さは，定数 $c$ が存在して，$cn\log^2 n$ 以下で表わすことができる。

次に $S_{M,x}$ の構成に要する領域量を計算する。(1) 初期化，(3) 後処理の部分の $\delta$ を構成するには $\log n + \log\log n$ 領域でできる。(2) シミュレーションの部分の構成をするには，$z$ をつくり出すためのカウンタに $\log n + \log\log n$ 領域，更にそのカウンタの長さをコピーするのに $\log(\log n + \log\log n)$ 領域が必要である。よって $S_{M,x}$ の構成には，任意の $\varepsilon > 0$ について，$(1+\varepsilon)\log n$ 領域でできる。 □

<u>系</u>　$FSTP_k$ は $(k-4)\log n$ 領域を必要とする。

<u>証明</u>　ある $\varepsilon_1 > 0$ について，$FSTP_{k+1}$ が $(k-3-\varepsilon_1)\log n$ 領域で解けると仮定する。前定理より，任意の $k \log n$ 領域で解ける問題Lに対して，ある定数 $c$ が存在して，任意の $\varepsilon > 0$ について，Lは $FSTP_{k+1}$ に $((1+\varepsilon)\log n, cn\log^2 n)$-

reducible である。補助定理 2.2 より L は $\hat{S}(n)+16\lceil \log \hat{S}(n)\rceil$ 領域で解ける。ただし,

$$\hat{S}(n) = \lceil (k-3-\varepsilon_1)\log(cn\log^2 n)\rceil + 2\lceil \log(cn\log^2 n)\rceil + \lceil (1+\varepsilon)\log n\rceil .$$

すなわち, L は, 任意の $\varepsilon_2>0$ について, $\lceil (k+\varepsilon_2-\varepsilon_1)\log n\rceil$ で解ける。$\varepsilon_2$ は任意だから, L は $k\lceil \log n\rceil$ より小さい領域で解けることになる。よって $k\log n$ 領域以下で解けるすべての問題が, $k\lceil \log n\rceil$ より小さい領域で解けることになり, 矛盾する [6].

ゆえに, $FSTP_k$ は $(k-4)\log n$ 領域を必要とする。□

注 $FSTP_k$ は, 任意の $\varepsilon>0$ について, $(k+\varepsilon)\log n$ 領域で解けるので, $FSTP_k$ は

$$DSPACE_2((k+\varepsilon)\log n) - DSPACE_2((k-4-\varepsilon)\log n)$$

に属する。

<参考文献>

[1] A. Adachi, S. Iwata, and T. Kasai, Some combinatorial game problems require $\Omega(n^k)$ time, to appear in Journal of the ACM, 31 (1984)

[2] J. W. Hong, On some deterministic space complexity problems, SIAM J. on Comput., 11, 591-601 (1982)

[3] N. Jones, Space-bounded reducibility among

combinatorial problems, Journal of Comput. and Syst. Sci. 11, 68-85 (1975)

[4] T. Kasai, A. Adachi, and S. Iwata, Classes of pebble games and complete problems, SIAM J. on Comput., 8, 574-586 (1979)

[5] T. Kasai, and S. Iwata, Gradually intractable problems and nontrivial polynomial time lower bounds, Research Report, Dept. of Computer Science, University of Electro-Communications, 1983

[6] J. I. Seiferas, Relating refined space complexity classes, Journal of Comput. and Syst. Sci., 14, 100-129 (1977)