山口大学工学部電子工学科
卒業論文　1990年，3月

マルチプロセッサオートマトンに関する研究

山口大学工学部

電子工学科第五講座

角川裕次

ABSTRACT

Recently, a multiprocessor finite automaton was introduced, which consists of a number of finite automata called `processor'. The multiprocessor finite automaton may be considered as the simplest model of parallel computation.

In this paper, we introduce a multiprocessor pushdown automaton which consists of a pushdown automaton and several finite automata. We investigate some properties of these automata. It is shown that

(1) nondeterministic accepting type k-processor automata are equivalent to nondeterministic k-head automata,

(2) (k+1)-processor automata are more powerful than k-processor automata, etc.

Furthermore, closure properties of family of languages accepted by multiprocessor finite automata is examined.

目次

1. まえがき

最近マルチプロセッサ有限オートマトン（MPFA）が導入され（(2)）、いくつかの性質が明らかにされている。MPFAは最も簡単な並列計算のモデルと考えられる。MPFAは複数の有限オートマタより構成され、各有限オートマトンは各々並列に動作を行う。各々のオートマトンをプロセッサと呼ぶ。又、スイッチング関数と呼ばれるものによって、各々のプロセッサを動作させるか非動作にさせるかが決定される。本論文では一方向マルチプロセッサオートマトンの幾つかの性質について検討する。2. では、(2)で定義されたMPFAの受理条件を変えた「受理型MPFA」を導入する。そして、MPFAに加えて、プロセッサの一つがプッシュダウンオートマトンで他のプロセッサを有限オートマトンとしたマルチプロセッサプッシュダウンオートマトン（MPPDA）を導入する。これにともなって、異なった種類のオートマトンより構成されるマルチプロセッサオートマトン（MPA）の一般的な記法を定める。3. では、マルチプロセッサオートマトンとマルチヘッドオートマトンとの関係について調べる。そして、非決定性受理型MPFAと非決定性マルチヘッド有限オートマトンが、そして非決定性受理型MPPDAと非決定性マルチヘッドプッ

シュダウンオートマトンの受理能力が等価であることを示す。4. では、MPFAとMPPDAとの関係について調べ、これらの受理能力には真に差があることを示す。5. では停止型と受理型の関係について調べる。停止型k+1プロセッサMPFAは受理型kプロセッサMPFAを模倣可能なことを、そして受理型k+1プロセッサMPFAは停止型kプロセッサMPFAを模倣可能な事を示す。6. では、非決定性2プロセッサMPFA（MPPDA）では受理されるがいかなる決定性MPFA（決定性MPPDA）でも受理されない言語の存在を示す。これにより、決定性と非決定性とでは受理能力に真に差があることが示される。7. では、MPFA, MPPDAで受理される言語の階層性、即ち、プロセッサ数がkのときよりもk+1のときのほうが受理能力が強い事を示す。8. では、一方向と二方向との受理能力の違いを調べる。9. では、MPFAによって受理される言語の閉包性について調べる。

2. 準備

文献(2)にはマルチプロセッサ有限オートマトン（MPFA）の正式な定義が与えられている。後述のようにこのオートマトンは全てのプロセッサが非動作にな

ったとき入力を受理する。本稿ではこれに加えて受理条件を「全てのプロセッサが受理状態に入ったとき入力を受理する」と変更したオートマトンを考える。更にプロセッサのうち一つがプッシュダウンオートマトンで他が有限オートマトンであるようなマルチプロセッサプッシュダウンオートマトン（MPPDA）を導入する。以下に、これらのオートマトンの正式な定義を与える。

定義2．1　（停止型マルチプロセッサ有限オートマトン）

停止型1方向 k プロセッサ有限オートマトンは、

$$M=(Q, E, g, h, v_0)$$

と5つ組で表される。各成分は次の通りである。

Qは状態集合と呼ばれる有限集合で、各要素は状態である。Eは入力テープに書かれる記号の有限集合で、アルファベットと呼ばれる。但し、左端、右端記号である¢, $は含まれない。gは遷移関数と呼ばれ、$g: Q\times E\rightarrow Q\times\{0,1\}$なる写像をする。hはスイッチング関数と呼ばれ、$h:\{1,2,...,k\}\times Q^k\rightarrow\{0,1\}$なる写像をする。$v_0$は初期状態の組で、$v_0\in Q^k$である。

非決定性の場合、遷移関数gは$g: Q\times E\rightarrow 2^{Q\times\{0,1\}}$なる写像をする。他の成分については、決定性のときと同じである。

停止型MPFAは、プロセッサと呼ばれるいくつかの有限オートマトンとスイッチング関数よりなる。入力は左端記号¢と右端記号$とに囲まれた読み取り専用のテープに与えられる。動作の開始前には、各プロセッサは左端記号上にあり、各プロセッサの有限制御部には各々初期状態が設定される。そして各プロセッサが並列に動作を行っていくが、各々プロセッサを動作させるかどうかはスイッチング関数によって決められる。直感的には、スイッチング関数は全てのプロセッサの状態によって、次のステップでの各々のプロセッサの動作を行うか動作を行わないかを決める。あるプロセッサに対するスイッチング関数の値が1なら次の計算ステップでそのプロセッサは動作し、0ならば非動作となる。そして計算のある時点において全プロセッサが非動作になったとき、そのオートマトンは入力語を受理する。 □

定義2.2 （受理型マルチプロセッサ有限オートマトン）

受理型1方向 k プロセッサ有限オートマトンは、

$$M=(Q, E, g, h, v_0, F)$$

と6つ組で表される。Fは、$F\subseteq Q$で、受理状態集合である。他の成分は、停止型MPFAと同じである。非決定性の場合についても同様である。受理型MPF

Aは、受理条件を「計算のある時点において全プロセッサが受理状態に入っていたら入力語を受理する」、としたものである。 □

定義2. 3 (停止型マルチプロセッサプッシュダウンオートマトン)

停止型1方向kプロセッサプッシュダウンオートマトンは、1つのプッシュダウンオートマトンとk−1個の有限オートマトンよりなり、

$$M = (Q, E, \Gamma, g_1, g_2, h, v_0, Z_0)$$

と8つ組で表される。各成分は次の通りである。Qは状態集合と呼ばれる有限集合で、各要素は状態である。Eは入力テープに書かれる記号の有限集合で、アルファベットと呼ばれる。但し、左端、右端記号である¢, $は含まれない。Γはプッシュダウンストアのアルファベットである。g_1は有限オートマトンの遷移関数で、$g_1 : Q \times E \rightarrow Q \times \{0, 1\}$なる写像をする。$g_2$はプッシュダウンオートマトンの遷移関数で、$g_2 : Q \times E \rightarrow Q \times \{0, 1\} \times \Gamma^*$なる写像をする。hはスイッチング関数と呼ばれ、$h : \{1, 2, ..., k\} \times Q^k \rightarrow \{0, 1\}$なる写像をする。$v_0$は初期状態の組で、$v_0 \in Q^k$である。

非決定性の場合、遷移関数g_1, g_2はそれぞれ、$g_1 : Q \times E \times \Gamma \rightarrow 2^{Q \times \{0, 1\} \times \Gamma^*}$, $g_2 : Q \times E \rightarrow 2^{Q \times \{0, 1\}}$なる写像をする。他の成分については、決定性のとき

と同じである。受理条件は停止型MPFAと同じである。 □

定義2．4 （受理型マルチプロセッサプッシュダウンオートマトン）

受理型1方向kプロセッサプッシュダウンオートマトンは、1つのプッシュダウンオートマトンとk-1個の有限オートマトンよりなり、

$$M=(Q, \Sigma, \Gamma, g_1, g_2, h, v_0, Z_0, F)$$

と9つ組で表される。Fは、$F \subseteq Q$で、受理状態集合である。他の成分は、停止型MPPDAと同じである。また、受理条件は受理型MPFAのときと同じである。非決定性の場合についても同様である。 □

記法　本論文では次の記法を用いる。各$k \geqq 1$に対し、

FA　　　：有限オートマトン,

(k)HFA　：kヘッド有限オートマトン,

SP(k)HFA：シンプル kヘッド有限オートマトン,

（詳しくは(7)を参照されたい。）

PDA　　：プッシュダウンオートマトン,

(k)PDA　：kヘッドプッシュダウンオートマトン.

上の各種オートマトンを組み合わせたマルチプロセッサオートマトンを記述するのに次のような記法を用いる。

$H(A_1(k_1), A_2(k_2), \ldots, A_r(k_r))$

　：停止型マルチプロセッサオートマトン,

$A(A_1(k_1), A_2(k_2), \ldots, A_r(k_r))$

　：受理型マルチプロセッサオートマトン.

ここで、A_i $(1 \leqq i \leqq r)$はオートマトン、例えばFA, PDAであり、$A_i(k_i)$はオートマトンA_iをプロセッサとしてk_i個持つことを表す。但し、$k_i \geqq 1$とする。例えば、

H(FA(k))　：停止型マルチプロセッサ有限オートマトン,

A(PDA(1), FA(k−1))　：受理型マルチプロセッサプッシュダウンオートマトン.

である。更に決定性（非決定性）のものには頭に`D'（`N'）をつけ、1方向（2方向）のものには更にその前に`1'（`2'）を頭につける。例えば、

1DH(FA(k))　：1方向決定性H(FA(k)),

1NA(PDA(1), FA(k−1))：1方向非決定性A(PDA(1), FA(k−1)).

である。また、例えば1DH(FA(k))によって受理される言語族を$\mathcal{L}$[1DH(FA(k))]と表す。$\mathcal{L}$[1NA(PDA(1), FA(k−1))], $\mathcal{L}$[1NA(PDA(k))]などについても同様に定義される。

3.　マルチプロセッサオートマトンとマルチヘッドオ

ートマトンの関係

文献(2)ではマルチヘッド有限オートマトンはマルチプロセッサ有限オートマトンを模倣できる事が示されていた。本章ではこれより強い結果、非決定性受理型MPFAとマルチヘッド有限オートマトン及び非決定性受理型MPPDAと非決定性マルチヘッドプッシュダウンオートマトンとは受理能力が等価であることを示す。

補題3．1　各$k \geq 2$，$X \in \{D, N\}$，$Y \in \{H, A\}$に対し、

(1) $\mathcal{L}$[1XY(FA(k))] $\subseteq$ $\mathcal{L}$[1X(k)HFA]，

(2) $\mathcal{L}$[1XY(PDA(1)，FA(k-1))] $\subseteq$ $\mathcal{L}$[1X(k)PDA]．

（証明）　(1) Y=H（停止型）の場合は文献(2)の定理2．1で示されている。Y=A（受理型）の場合でも、文献(2)の定理2．1の証明と同様にして示される。

(2) この場合も、文献(2)の定理2．1の証明と同様にして示される。 □

定理3．1　各$k \geq 2$に対し、

(1) $\mathcal{L}$[1NA(FA(k))] = $\mathcal{L}$[1N(k)HFA]，

(2) $\mathcal{L}$[1NA(PDA(1)，FA(k-1))] = $\mathcal{L}$[1N(k)HFA]．

（証明）　(1) 補題3．1 (1)より $\mathcal{L}$[1NA(FA(k))] $\subseteq$ $\mathcal{L}$[1N(k)HFA]であるので、$\mathcal{L}$[1N(k)HFA] $\subseteq$ $\mathcal{L}$[1NA(FA(k))] を示す。1N(k)HFA Mのk本のヘッドを$h_1, h_2, \ldots,$ h_kとし、1NA(FA(k)) Nのk個のプロセッサを$p_1, p_2, \ldots,$

p_kとする。プロセッサp_1でMの有限制御部の状態及びh_1の動作を模倣する。p_1は$h_1, h_2, \ldots, h_k$によって読まれる記号を非決定的に推測する。これらの記号とp_1が読んでいる記号及びp_1の有限制御部の状態によって動作していく。このとき$h_1, h_2, \ldots, h_k$の各ヘッドが右に動くか停止するかをも有限制御部の中に貯えておく。p_1は推測したk-1個の記号が正しいか否かを$p_2, p_3, \ldots, p_k$で読んだ記号に依存した各プロセッサ内の状態及びスイッチング関数によってチェックする。推測が誤っていた場合は全てのプロセッサを非動作にする。プロセッサ$p_2, p_3, \ldots, p_k$はヘッド$h_2, h_3, \ldots, h_k$が右に1コマ動くか停止するかを非決定性的に推測をし、どちらであるかを有限制御部に貯える。この推測が正しいか否かはこれらのプロセッサの状態と先にp_1中に貯えておいた$h_2, h_3, \ldots, h_k$のヘッド動きとスイッチング関数によって比べることで行う。推測が誤っていた場合には、全てのプロセッサを非動作にする。p_1の状態がMの受理状態になればNは入力を受理する。こうしてNはMの動作を模倣できる。

(2) 補題3. 1 (2)より、$\mathcal{L}$[1NA(PDA(1), FA(k-1))]$\subseteq$ $\mathcal{L}$[1N(k)PDA]である。又、$\mathcal{L}$[1N(k)PDA]$\subseteq\mathcal{L}$[1NA(PDA(1), FA(k-1))]であることは(1)での証明を拡張することで示せる。 □

4. マルチプロセッサ有限オートマトンとマルチプロセッサプッシュダウンオートマトンの関係

本章ではMPPDAはMPFAより受理能力が真に強いことを示す。即ち、プッシュダウンストアの有無による受理能力の差について調べる。

定理4.1 各 $k \geq 2$, $X \in \{D, N\}$, $Y \in \{H, A\}$ に対し、

$$\mathcal{L}[1XY(FA(k))] \subsetneq \mathcal{L}[1XY(PDA(1), FA(k-1))].$$

(証明) 各 $k \geq 2$, $X \in \{D, N\}$, $Y \in \{H, A\}$ に対し、

$\mathcal{L}[1XY(FA(k))] \subseteq \mathcal{L}[1XY(PDA(1), FA(k-1))]$ は明らか。

$L = \{ w_1 c \cdots c w_m 2 w_m^R c \cdots c w_1^R \mid m \geq 1, w_i \in \{0,1\}^* (1 \leq i \leq m)\}$ とする。文献(8)の定理6.2において $L \notin \bigcup_{1 \leq k < \infty} \mathcal{L}[1N(k)HFA]$ が示されている。この事実と補題3.1より、 $L \notin \bigcup_{1 \leq k < \infty} \mathcal{L}[1NH(FA(k))]$, $L \notin \bigcup_{1 \leq k < \infty} \mathcal{L}[1NA(FA(k))]$ が成り立つ。ところが、明らかに $L \in DCFL$(決定性分脈自由言語族)なので、各 $k \geq 2$ に対し、 $L \in \mathcal{L}[1DH(PDA(1), FA(k-1))]$, $L \in \mathcal{L}[1DA(PDA(1), FA(k-1))]$ である。よって本定理が成立する。 □

5. 停止型と受理型の関係

停止型と受理型の受理能力にはどのような関係があるかは興味ある点である。本章では、MPFA, MP

PDAそれぞれについて停止型と受理型の関係を調べる。

定理5.1　各$k \geq 2$に対し、

(1) $\mathcal{L}[1NH(FA(k))] \subseteq \mathcal{L}[1NA(FA(k))]$,

(2) $\mathcal{L}[1DH(FA(k))] \subseteq \mathcal{L}[1DA(FA(k+1))]$,

(3) $\mathcal{L}[1NH(PDA(1), FA(k-1))]$

$\subseteq \mathcal{L}[1NA(PDA(1), FA(k-1))]$,

(4) $\mathcal{L}[1DH(PDA(1), FA(k-1))] \subseteq \mathcal{L}[1NA(PDA(1), FA(k))]$.

(証明)　(1)　定理3.1と文献(2)の定理2.1より明らか。

(2)　いかなる1DH(FA(k)) Mに対しても、Mを模倣する1DA(FA(k+1)) Nを構成できることを示せばよい。

Mのプロセッサ$p_1, p_2, \ldots, p_k$の動作をNのプロセッサ$p_1', p_2', \ldots, p_k'$で模倣する。Nの$k+1$番目のプロセッサp_{k+1}'は左端記号`¢'上で初期状態のままスイッチング関数で非動作にしておく。Mが与えられた入力を受理したとする。このとき、$p_1, p_2, \ldots, p_k$は全て停止しており、各プロセッサの状態を$q_1, q_2, \ldots, q_k$とする。次の集合Hを定義する。

$H = \{q_1, q_2, \ldots, q_k \mid \forall i(1 \leq i \leq k)$ [p_iは非動作で、かつその状態はq_iである] $\}$.

Nの受理状態集合を $H \cup \{q_a\}$ $(q_a \notin H)$ とする。Nのある時点で $\forall i(1 \leq i \leq k)$ [(p_iの状態) $\in H$] とな

ったとする。Nはスイッチング関数を使って、p_iの状態の組合せがMが入力を受理するときの組合せとなっているかをチェックしそうであればp_{k+1}'を動作させその状態をq_aに遷移させる。そうでない場合にはp_{k+1}'は非動作のままにして、NはMの模倣を続けていく。こうしてNはMの動作を模倣できる。

(3),(4)　(1),(2)の証明と同様にして示せる。 □

定理5. 2　各$k \geq 2$, $X \in \{D, N\}$に対し、

(1)$\mathcal{L}[1XA(FA(k))] \subseteq \mathcal{L}[1XH(FA(k+1))]$,

(2)$\mathcal{L}[1XA(PDA(1), FA(k-1))] \subseteq \mathcal{L}[1XH(PDA(1), FA(k))]$.

(証明)(1)　いかなる1XA(FA(k)) Mに対しても、Mを模倣する1XH(FA(k+1)) Nを構成できることを示せばよい。Mのプロセッサ$p_1, p_2, \ldots, p_k$の動作をNのプロセッサ$p_1', p_2', \ldots, p_k'$で模倣する。Nのk+1番目のプロセッサp_{k+1}'は左端記号`¢'上で初期状態のままスイッチング関数で非動作にしておく。Mが与えられた入力を受理したとする。このとき$p_1, p_2, \ldots, p_k$は全て予め定められた受理状態に入っている。この場合Nはスイッチング関数によって$p_1', p_2', \ldots, p_k'$およびp_{k+1}'を非動作にして入力を受理する。計算のある時点でMのプロセッサが全て非動作になったとする。この場合Nはスイッチング関数によってp_{k+1}'を動作させて、無限ループに入る。(停止型MPFAは、全てのプロセッサが

非動作になったときに入力を受理することに注意。）
こうしてNはMの動作の模倣ができる。

(2)　(1)と同様にして示される。 □

以上より次の系を得る。

系5.1　各 $k \geq 2$, $X \in \{D, N\}$ に対し、

(1) $\bigcup_{1 \leq k < \infty} \mathcal{L}[1XH(FA(k))] = \bigcup_{1 \leq k < \infty} \mathcal{L}[1XA(FA(k))]$,

(2) $\bigcup_{1 \leq k < \infty} \mathcal{L}[1XH(PDA(1), FA(k-1))]$

$= \bigcup_{1 \leq k < \infty} \mathcal{L}[1XA(PDA(1), FA(k-1))]$.

6.　決定性と非決定性の差異

本章では、MPFA, MPPDA共に停止型及び受理型に関わらず、非決定性は決定性より強力であることを示す。

定理6.1　各 $k \geq 2$, $Y \in \{H, A\}$ に対し、

(1) $\mathcal{L}[1DY(FA(k))] \subsetneq \mathcal{L}[1NY(FA(k))]$,

(2) $\mathcal{L}[1DY(PDA(1), FA(k-1))\}$

$\subsetneq \mathcal{L}[1NY(PDA(1), FA(k-1))]$.

（証明）

$L = \{ x_1 ¢ y_1 * \cdots * x_n ¢ y_n \# x_n' ¢ y_n' * \cdots * x_1' ¢ y_1' \mid$

$[$ある $i, j (1 \leq i, j \leq n)$ に対し $x_i = x_j'$ かつ $y_i \neq y_j']$

$\& [x_i, x_i', y_i, y_i' \in \{0, 1\}^*] \& [n \geq 1] \}$

とする。 $L \notin \bigcup_{1 \leq k < \infty} \mathcal{L}[1D(k)PDA]$が文献(3)の定理

2'で示されているので、Lはいかなる1DH(FA(k)),1DA(FA(k))によっても受理されない。よって、本定理を示すにはLを受理する1NH(FA(2)),1NA(FA(2))の存在を示せば良い。Lを受理する1NH(FA(2)) Mの動作を以下に示す。なお、同様な考え方でLを受理する1NA(FA(2))を構成できるので、この場合の証明は省略する。Mの動作は大別して4つの動作よりなる：(1)移動,(2)待ち,(3)比較,(4)無限ループ. なお、Mの各プロセッサはそれぞれどの動作を行なっているかを有限制御部に記憶する。Mの2つのプロセッサをそれぞれp_1,p_2とする。まず、p_2は中心の記号`#'に移動する。この間$p_1$はスイッチング関数で非動作になっている。次に$p_1$,$p_2$は並列に右へ「移動」をしていく。そして$p_1$,$p_2$は、`*'を読むとその右にある部分語の「比較」を行なうか、更に「移動」を続けるかを非決定的に選ぶ。「比較」すると決めたら、他方のプロセッサが比較の準備ができるまで（比較する部分語を決定するまで）「待ち」動作にはいる。両方のプロセッサが「待ち」動作にはいるまで、早く「待ち」動作にはいった方がスイッチング関数で非動作となる。そして、両プロセッサが共に「待ち」動作にはいったらスイッチング関数は同時にp_1,p_2を動作させ、「比較」動作を開始させる。あとはp_1,p_2を用いて2つの部分語$x_i¢y_i$,$x_j'¢y_j'$に対して

「比較」を行なう。もし$x_i = x_j'$かつ$y_i \neq y_j'$であったら両プロセッサを非動作にして、入力を受理する。そうでなければ、このオートマトンは「無限ループ」に入る。 □

7. プロセッサ数に基づく階層性

文献(2)では、$\mathcal{L}[1DH(FA(1))] \subsetneq \mathcal{L}[1DH(FA(2))]$が示されている。本章では各$k \geq 1$に対し、停止型、受理型及び決定性、非決定性の場合でも$k+1$プロセッサはkプロセッサより受理能力が強い事を示す。

定理7.1 各$k \geq 1, X \in \{D, N\}, Y \in \{H, A\}$に対し、

(1) $\mathcal{L}[1XY(FA(k))] \subsetneq \mathcal{L}[1XY(FA(k+1))]$,

(2) $\mathcal{L}[1XY(PDA(1), FA(k-1))] \subsetneq \mathcal{L}[1XY(PDA(1), FA(k))]$.

(証明) $L(b) = \{ w_1 c \cdots c w_b 2 w_b c \cdots c w_1 \mid w_i \in \{0, 1\}^* \}$とする。$L(k(k-1)/2) \notin \mathcal{L}[1N(k-1)PDA]$が文献(3)に示されている。よって本定理が成立するには$L(k(k-1)/2) \in \mathcal{L}[1DH(FA(k))]$, $L(k(k-1)/2) \in \mathcal{L}[1DA(FA(k))]$を示せば良い。帰納法を用いてこのことを示す。まずは停止型の場合を示す。

・基底段階 $k=2$のとき、$L(1) = \{ w_1 * w_2 \mid w_1, w_2 \in \{0, 1\}^*, w_1 = w_2 \}$は、次のように動作する$1DH(FA(2))$ Mによって受理される。Mの2つのプロセッサをそれ

ぞれp_1, p_2とする。まず、p_2を部分語w_2の左の`*'上に移動する。そしてp_1とp_2は同時に右に移動する。このとき、各プロセッサは読んだ記号、および各セグメント内でのプロセッサの位置が左端から数えて偶数か奇数かを状態として記憶する。スイッチング関数はp_1, p_2の状態を比べることで各プロセッサが読んだ記号の比較を行う。もし読んだ記号が異なっていたら、p_1を非動作にし、同じであったら更に比較を続ける。読んだ記号が異なっている場合には、次の計算のステップではp_1のw_1の左端からの位置と、p_2のw_2の左端からの位置は異なっている。つまり、一方のプロセッサは偶数番目のコマに、他方のプロセッサは奇数番目のコマ上にあることをそれぞれ記憶している。そして、スイッチング関数は各プロセッサの読んだ記号には関係なく常にプロセッサを動作させて各プロセッサを右に動かす。Mが入力語を受理するのは、p_1とp_2がそれぞれ`*'と`¢'を同時に読み、かつp_1, p_2が共に奇数番目のコマにいるか、偶数番目のコマにあるときのみとする。それ以外はこれらのプロセッサは無限の動作を続ける。

・帰納段階　$k=t$のとき、$L(t(t-1)/2)$を受理する1DH(FA(2)) Mが存在すると仮定する。$k=t+1$のとき。Mのプロセッサを$p_1, p_2, \ldots, p_{t+1}$とする。まず$p_{t+1}$を部分語$w_{(t+1)t/2-t+1}$の左の`*'に移動させる。そして$p_{t+1}$

が、その右にあるt個の部分語

$w_{(t+1)t/2-t+1}, w_{(t+1)t/2-t+2}, \ldots, w_{(t+1)t/2}$

を読み、$p_1, p_2, \ldots, p_t$がそれぞれ部分語$w_1, w_2, \ldots, w_t$を読んで対応する部分語の比較を行う。もし対応する部分語が異なっていたら、無限の動作にはいる。この比較は、$k=2$のときの動作と同じ考え方で行う。比較がうまくいけば更に（再帰的に）比較を行う動作にはいる。即ち$p_1, p_2, \ldots, p_t$を用いて部分語$w_{t+1}, w_{t+2}, \ldots, w_{t(t-1)/2}$の比較を行う。帰納法の仮定より、この比較を行うことができる。

受理型の場合も停止型の場合と同様にして示される。但し、対応する部分語を比較しているときに、部分語が異なっていると分かったときには、その時点でスイッチング関数で全プロセッサを非動作にする。対応する部分語が全て同じであればそれぞれのプロセッサは受理状態に遷移して入力語を受理する。□

8. 一方向と二方向の差異

本章では二方向は一方向よりも真に受理能力が強い事を示す。

定理8.1　各$k \geq 2$, $X \in \{D, N\}$, $Y \in \{H, A\}$に対し、

(1) $\mathcal{L}[1XY(FA(k))] \subsetneq \mathcal{L}[2XY(FA(k))]$,

(2) $\mathcal{L}[1XY(PDA(1), FA(k-1))]$

$$\subsetneq \mathcal{L}[2XY(PDA(1), FA(k-1))].$$

(証明)　(1)　$\mathcal{L}[1XY(FA(k))] \subseteq \mathcal{L}[2XY(FA(k))]$は明白。言語Lを定理4.1の証明で定めたものとすると、定理4.1の証明より各$X \in \{D, N\}$, $Y \in \{H, A\}$に対し、$L \notin \bigcup_{1 \le k < \infty} \mathcal{L}[1XY(FA(k))]$が云える。ところが、$L \in \mathcal{L}[2DH(FA(2)]$, $L \in \mathcal{L}[2DA(FA(2)]$は定理7.1の証明での技法を用いる事により示す事ができる。以上の事実より$\mathcal{L}[1XY(FA(k))] \subsetneq \mathcal{L}[2XY(FA(k))]$である。

(2)　各$k \ge 2$, $X \in \{D, N\}$, $Y \in \{H, A\}$に対し、$\mathcal{L}[1XY(PDA(1), FA(k-1))] \subseteq \mathcal{L}[2XY(PDA(1), FA(k-1))]$は明らか。$L = \{w_1\# \cdots \#w_b\$w_b\# \cdots \#w_1 \mid b \ge 1, w_i \in \{0, 1\}^*\}$とするとき、$L \notin \bigcup_{1 \le k < \infty} \mathcal{L}[1N(k)PDA]$は文献(3)の結果より導ける。ところが、Lを受理する2DH(PDA(1), (FA(1)), 2DA(PDA(1), FA(1))が存在する事は定理7.1の証明での技法を拡張する事で示せる。これらの事実と補題3.1(2)より、本定理が成立する。□

9.　閉包性

本章では、MPFAによって受理される言語族の閉包性について調べる。非決定性受理型MPFAの場合は、非決定性マルチヘッドオートマトンと受理能力が

等価であるを既に示したので、マルチヘッドオートマトンでの結果がそのまま成立する。本章ではその他の場合について議論する。

定理9.1 各$k \geq 2$に対して、$\mathcal{L}[1DH(FA(k))]$, $\mathcal{L}[1DA(FA(k))]$は、次の演算に関して閉じていない。

(1) 交差, (2) 和, (3) 連接, (4) クリーネ閉包, (5) ε無し準同型写像, (6) 代入, (7) 反転.

(証明) $L_0, L_1, L_2, G(f)$を、次のように定める。

$$L_0(b, r) = \{ w_1 * \cdots * w_b c w_{b+1} * \cdots * w_{2b} \mid [w_r = w_{2b-r+1}] \ \& \ [w_i \in \{0, 1\}^*] \ \& \ [1 \leq i \leq 2b] \},$$

$$L_1 = \{a, b\}^* c \cup \{\varepsilon\},$$

$$L_2 = \{ udud \mid u \in \{a, b, c\}^* \} \cup \{\varepsilon\},$$

$$G(f) = \{ w_1 c \cdots c w_f c w_f c \cdots c w_1 \mid w_i \in \{a, b\}^*, 1 \leq i \leq f \} \cup \{\varepsilon\}.$$

(1) 各$k \geq 2, 1 \leq i \leq k(k-1)/2+1$に対し、$L_0(k(k-1)/2+1, i) \in \mathcal{L}[1DH(FA(k))]$, $L_0(k(k-1)/2+1, i) \in \mathcal{L}[1DA(FA(k))]$は明らか。$L_3$を7.で定義した言語 Lの中心の区切り記号を`*'から`c'に変えた言語とすると、

$$L_0(k(k-1)/2+1, 1) \cap L_0(k(k-1)/2+1, 2) \cap \cdots \cap L_0(k(k-1)/2+1, k(k-1)/2+1) = L_3(k(k-1)/2+1)$$

となる。定理7.1の証明より、$L_3(k(k-1)/2+1)$は$\mathcal{L}[1NH(FA(k))]$, $\mathcal{L}[1NA(FA(k))]$に含まれないことが云

えるので、交差に関して閉じていない。

(2) L_2および$\{a,b\}^*cG(k(k-1)/2)$が$\mathcal{L}[1DH(FA(k))]$, $\mathcal{L}[1DA(FA(k))]$に含まれることは容易にわかる。ところで、$L_2\cup\{a,b\}^*cG(k(k-1)/2)$が$\mathcal{L}[1D(k)FA]$に含まれていないことは文献(5)の定理5に示されている。したがって、この言語は$\mathcal{L}[1DH(FA(k))]$にも$\mathcal{L}[1DA(FA(k))]$にも含まれない。よって、和に関して閉じていない。

(3) $L_1, L_2, G(k(k-1)/2)$は全て$\mathcal{L}[1DH(FA(k))]$, $\mathcal{L}[1DA(FA(k))]$に含まれることは明らか。一方、これらの連接$L_1L_2G(k(k-1)/2)$は$\mathcal{L}[1D(k)HFA]$に含まれていないことが文献(5)の定理2に示されている。よって、$\mathcal{L}[1DH(FA(k))]$, $\mathcal{L}[1DA(FA(k))]$は連接に関して閉じていない。

(4) 各$k\geq 2$に対し$L_4=\{e\}L_2\cup\{a,b\}^*cG(k(k-1)/2)\cup\{e\}$と定める。明らかに$L_4$は$\mathcal{L}[1DH(FA(k))]$, $\mathcal{L}[1DA(FA(k))]$に含まれるが、$\{L_4\}^*$は$\mathcal{L}[1D(k)HFA]$には含まれないことが文献(5)の定理4で示されている。この事実より$\{L_4\}^*$は、$\mathcal{L}[1DH(FA(k))]$, $\mathcal{L}[1DA(FA(k))]$に含まれない。よって、クリーネ閉包に関して閉じていない。

(5), (6) 各$k\geq 2$に対し、$L_5=\{e\}L_2\cup\{g\}\{a, b\}^*\{c\}G(k(k-1)/2)$とする。明らかに$L_5$は$\mathcal{L}[1DH(FA(k))]$,

$\mathcal{L}$[1DA(FA(k))]に含まれている。 ε無し準同型写像hを、$h(a)=a, h(b)=b, h(c)=c, h(d)=d, h(e)=h(g)=e$ とする。hによるL_5の写像は$\mathcal{L}$[1D(k)HFA]には含まれないことが文献(5)の定理6で示されているので、 $\mathcal{L}$[1DH(FA(k))], $\mathcal{L}$[1DA(FA(k))]には含まれない。 よって、 $\mathcal{L}$[1DH(FA(k))], $\mathcal{L}$[1DA(FA(k))]はε無し準同型写像に関して閉じていない。

(6) 言語$L_2 \cup \{a, b\}^* c G(k(k-1)/2)$は$\mathcal{L}$[1D(k)HFA]に属していないことが文献(5)の定理3に示されている。よってこの言語は$\mathcal{L}$[1DH(FA(k))], $\mathcal{L}$[1DA(FA(k))]のどちらにも属していない。 ところがこの言語の反転$L_2^R \cup G(k(k-1)/2) c \{a, b\}^*$は1DH(FA(k)), 1DA(FA(k))のどちらによっても受理される。 したがって$\mathcal{L}$[1DH(FA(k))], $\mathcal{L}$[1DA(FA(k))]はともに反転に関して閉じていない。

□

定理9. 2 各$k \geq 2$に対し、 $\mathcal{L}$[1NH(FA(k))], $\mathcal{L}$[1NA(FA(k))]は次の演算に関して閉じていない。

(1) 交差, (2) 代入, (2) 補集合.

(証明) (1) 定理9. 1の証明と全く同じ。

(2) 各$k \geq 2$に対し、 $L_3(k(k-1)/2)$の記号'c'に言語$L_0(1)$を代入すると、 $L(k(k-1)/2+1)$となる。 よって定理7. 1の証明よりこの言語は$\bigcup_{1 \leq k < \infty} \mathcal{L}$[1NH(FA(k))], $\bigcup_{1 \leq k < \infty} \mathcal{L}$[1NA(FA(k))]に属さないことがいえ

る。したがって代入に関して閉じていない。

(3) $\mathcal{L}$[1NH(FA(k))], $\mathcal{L}$[1NA(FA(k))] 共に交差に関して閉じていないが、和に関しては閉じている。よって、ドモルガンの法則より補集合に関して閉じていないことが導かれる。 □

定理9. 3 各$k \geqq 2$に対し、$\mathcal{L}$[1NH(FA(k))], $\mathcal{L}$[1NA(FA(k))]は次の演算に関して閉じている。

(1) 和, (2) 反転 (但し、$\mathcal{L}$[1NA(FA(k))]についてのみ).

(証明) (1) 容易に証明できる。

(2) $\mathcal{L}$[1N(k)FA]は反転に関して閉じていることが文献(1)で示されている。この事実と定理3. 1(1)より$\mathcal{L}$[1NA(FA(k))]は反転に関して閉じていることが導かれる。 □

最後にMPFAで受理される言語全体についての閉包性について調べるが、その前に次の定義が必要である。

定義 決定性MPFA Mに対して、$Q^k \times (\Sigma \cup \{¢, \$\})^k$の部分集合S(M)を次のように定義する。ここで、QはMの状態集合、ΣをMのアルファベットとする。

$(q_1, \cdots, q_k, e_1, \cdots, e_k) \in S(M)$

⇔ (1) 各i ($1 \leqq i \leqq k$)に対し、ある$1 \leqq t_{i0} \leqq |Q|^k$が存在して、全てのt ($0 \leqq t < t_{i0}$)に対して、

$\delta(q_i(t), e_i) = (q_i(t+1), 0)$ & $q_i = q_i(t_{i0})$

ただし、各i $(1 \leqq i \leqq k)$に対し、$q_i(0) = q_i$.

(2) ある$t_0 \leqq |Q|^k$が存在して、

$(q_1, \ldots, q_k) = (q_1(t_0), \ldots, q_k(t_0))$

となり、全ての$0 \leqq t \leqq t_0$に対してあるiが存在して$h(i, q_1(t), \ldots, q_k(t)) \neq 0$となる。

なお、$q_i(t)$は時刻tにおけるプロセッサiの状態とする。

このS(M)は、Mが無限ループとなるときの各プロセッサの状態と、各々のプロセッサが読んでいる記号の対の集合を意味している。この集合S(M)の計算は容易に確かめられるように、有限回のステップ($O(|Q|^{2k} \cdot |\Sigma|^k)$)で行える。 □

定理9.4 各X∈{D,N}, Y∈{H,A}に対し、$\bigcup_{1 \leq k < \infty} \mathcal{L}[1XY(FA(k))]$は次の演算に関して閉じている。

(1) 交差, (2) 和, (3) 補集合.

(証明) 各X∈{D,N}に対し、$\bigcup_{1 \leq k < \infty} \mathcal{L}[1XH(FA(k))]$が交差, 和に関して閉じている事が文献(2)の定理2.3で示されている。この事と系5.1より、本定理を示すには$\bigcup_{1 \leq k < \infty} \mathcal{L}[1XH(FA(k))]$が補集合を取る演算に関して閉じている事を示せば良い。これを以下に示す。任意の1DH(FA(k)) Mに対して、Mの受理する言語の補集合を受理する1DH(FA(k+1)) Nが構成可能な事を

示せば良い。Mのプロセッサを$p_1, \ldots, p_k$、Nのプロセッサを$p_1', \ldots, p_k', p_{k+1}'$とする。QはMの状態集合、ΣをM，Nの入力アルファベットとする。Nは次のようにして構成される。Nの各プロセッサp_i'の状態qi'は次の形をしている。

$q_i' = (q_i, e, s)$, $q_i \in Q \cup \{q_{k+1,0}, q_{k+1,f}\}$

但し $q_{k+1,0}, q_{k+1,f} \notin Q$, $e \in \Sigma \cup \{¢, \$\}$, $s \in \{1, 2\}$

Nの各プロセッサは、状態の第一成分で対応するMのプロセッサの状態を貯え、第二成分ではプロセッサの位置の入力記号を貯える。そして第三成分でNの動作のフェーズを貯える。なお、Nの各プロセッサp_i'の初期状態は、

$q_{i,0}' = (q_{i,0}, ¢, 1)$, $1 \leqq i \leqq k$: p_iの初期状態

$q_{k+1,0}' = (q_{k+1,0}, ¢, 1)$: p_{k+1}'の初期状態

とする。Nの一動作は、2つのフェーズよりなる。フェーズ1では、状態の第一成分に従って、各p_i'はp_iを模倣して第一成分とヘッドの位置を変える。なおスイッチング関数はMのときと同じ振舞いをする。フェーズ2では、各p_i'はヘッドの下の入力記号を読み、状態の第二成分にその記号を貯える。なお、p_{k+1}'はスイッチング関数で非動作となったままである。こうして、$p_1', \ldots, p_k'$ は、それぞれMのプロセッサの動作を模倣

する。この時点tでのNの計算状況※C(t, x)を(入力をxとすると)、C(t, x)=((q_1, e_1, 1), … ,(q_k, e_k, 1), (q_{k+1}, e_{k+1}, 1), 1_1, … , 1_k, 1_{k+1})とする。そして、(q_1, … , q_k, e_1, … , e_k)∈S(M)であれば、スイッチング関数は全プロセッサを非動作にして入力を受理する。(この時Mは無限ループに入っている。)また状態の組合せ(q_1, … , q_k)が、Mにおいて全プロセッサが非動作となるような状態の組合せ(入力を受理するとき)であれば、p_1', … , p_k'を非動作、p_{k+1}'を動作させて無限ループ入り入力を非受理にする。これら2つのうちいずれにも当てはまらない場合は更に計算を続ける。

□

※時刻tにおける入力x上でのMの計算状況C(t, x)とは、Mの各プロセッサの状態と各プロセッサの入力テープ上でのヘッドの位置との対である。

10. むすび

未解決の問題をいくつか与えて本稿を終える。

各 $k \geqq 1$ に対し、

(1) $\mathcal{L}[1DH(FA(k))]$ と $\mathcal{L}[1DA(FA(k))]$ の関係はどうなのか?

(2) $\mathcal{L}[1NH(FA(k))] \subsetneq \mathcal{L}[1NA(FA(k))]$ であるか?

(3) $\mathcal{L}[1DH(FA(k))] \subsetneq \mathcal{L}[1D(k)HFA]$ であるか?

謝辞

本研究を進めるにあたって、御指導いただきました井上克司教授、大島商船高等専門学校の松野浩嗣先生に深く感謝いたします。また有益な御助言を賜りました高浪五男教授、伊東暁先生に感謝いたします。

参考文献

(1) Allen, E.:"Solution for Problems P90 and P91", EATCS, 243－244, June (1985)

(2) Buda., A.C:"Multiprocessor automata, Inform. Process. Lett. 25 257－261 (1987)

(3) Chorobak, M., Li, M. : "K+1 Heads Are Better than k for PDAs", J. Comput. System Sci. 37, 144－155 (1988)

(4) Hopcroft, J.E., Ullman, J.D. : "Introduction to Automata Theory, Languages, and Computaion", Addison-Wesley, Reading, Mass (1979)（邦訳 野崎, 高橋, 町田, 山崎 訳: オートマトン 言語理論 計算論 I, II, サイエンス社, 1984年）

(5) Hromkovič, J., :"One-way multihead deterministic finite automata", Acta Informatica, 19, 377－384 (1983)

(6) Ibarra, O.H., Kim, C.E. : "On 3-head versus 2-head finite automata", Acta Informatica 4, 173－200 (1975)

(7) Inoue, K., Takanami, I., Nakamura, I., Ae, T. : "One-way simple multihead finite automata", Theoret. Comput. Sci. 9 311－328 (1979)

(8) 井上・中村・高浪 :"シンプルマルチヘッドオート

マタ並びにマルチヘッドオートマタに関する2，3の性質",信学会オートマトンと言語研資 AL77-9 (1977)

(9) Rosenberg,A.L. : " On Multihead Finite Automata",IBM J.R and D. 10, 388－394 (1966)

(10) 谷口・井上・高浪 :"シンプルマルチヘッド有限オートマタのあるクラスについて",信学技報 AL78－64 (1978)

(11) Yao,A.C., Rivest,R.L.:"K+1 Heads Are Better than k", J.ACM 25, 337－340 (1978)