

セル構造オートマトンによる並列処理

京大理学部 西尾英之助

II. 序論

セル構造オートマトンは 1) 同一の素子が 2) 多数結合されて
できる系である。Von Neumann はこの様な系の 逐時的動作 に
よってオートマトンやキューリング機械の埋込みの問題を扱い
、自己増殖の論理モデルを作った。¹⁾ 逐時動作というのは、
ある時点で状態変化をする素子が 1 個であり、かつその素子
が空間的に隣合った素子に順に移ってゆく動作形式のことだ
である。残りの素子は状態変化をしない。このような静状
態の素子は、しかるべき時期が来るまで待っていると考えら
ば、やはり処理目的に貢献していることになる。

Von Neumann の仕事の後から始まった Moore, Myhill など
による“エレンの圈問題”は“斉射撃問題”(同期問題)²⁾を生んだ
。これは同時に全素子が同じ状態になることを要求するの
だから少なくとも最後の瞬間には全素子の 並列動作が必要だ
である。さらに問題は 最短時間解を求めることになったから
、並列動作の必要性が高まった。

他方一般に情報処理系については原理的に処理目的の達成
されるか否かの問題と同時に所要時間の問題がある。計算

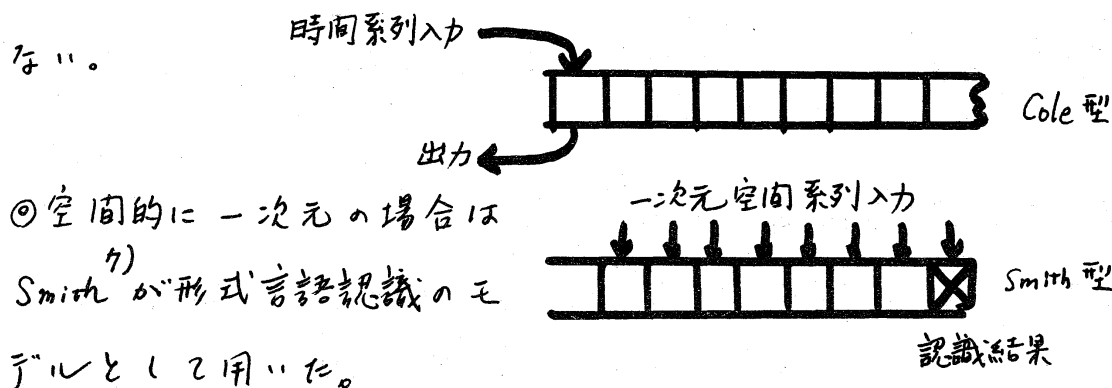
機システムが大型化して多数の装置が組込まれるにつれて、それらを並列動作させるモニターが必要になった。こうしてメモリと処理時間を短かくするために並列処理が必要になったように見えるが、本来処理されるべき情報の空間的振かりを持っていきから並列に処理される場合もある。視覚情報の処理はさうで、人工の装置で逐時処理をするのは、経済性と考え易さのためである。

2. 処理目的とセル構造のモデル

系への入力の子之才と結果の取出し方の違いにより、解析方法や論じよう命題、所要時間の長短など、大いに違ってくる。従来扱われてきたモデルに言及しよからこの臭を整理してやる。

1) 入力情報の一次元の場合:

①時間的に一次元(空間的に0次元)で方向性を持った、形式言語のような情報を空間的に一次元(あるいは n 次元)のセル構造で処理するモデル——これは言語認識モデルとして Code³⁾ が用いたのと、Code型と云ふ。Fischer⁴⁾ が実時間い“素数タイミング”を発生する系をつくつたし、Byxutab⁵⁾ や有沢⁶⁾ が実時間で計算できる関数を論じた。これら一次元のセル構造の並列処理能力を利用していきこといふやうなとき



入力系列は系の初期状態として与えられる、系は自律動作をする。結果はある特定の素子に出る。後述べる“並列・実時間ソート”のモデル⁸⁾では結果は全セルの最終状態として与えられる。一斉射撃や“French Flag問題”⁹⁾もこのような入出力の与え方の例である。まとめて、一次元空間パターンの(実時間)変換の問題と云えよう。Cole型で時間系列の実時間変換を扱った論文は手にない。

2) 入力情報の二次元の場合:

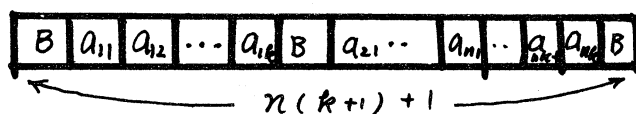
⑨ 時間+空間で二次元の情報はまたセル構造で扱われている。ない。Hennie¹⁰⁾の iterative array も主な結果は時間的に振る舞いのない場合についてである。関¹¹⁾がこのような情報の認識問題を考えているがまた明確な結果が出てない。

⑩ 空間的に二次元の場合は Smith型で図形認識が扱われている。Unger¹²⁾, Papert, Minsky¹³⁾, Beyer¹⁴⁾, 稲垣¹⁵⁾など。しかし、処理時間や並列処理の点で系統立つ議論は少ない。Smith¹⁶⁾。いずれも認識問題で、変換問題としては、一斉

射撃の二次元への(本質的にはよい)拡張があるからである。¹⁷⁾

3. 2進数の実時間ソート⁸⁾

n 個の k -bit 2進数の列 $\{a_1, a_2, \dots, a_n\}$ が与えられたとき、図のように Smith 型の系を用いて、大至小の順に並べ換える問題。ただし数 a_i は



$a_{i1} a_{i2} \dots a_{ik}$ と表現

され、数の区切り記号 B を用いる。系の素子は n, k に依存しない有限オートマトンで、ソート終了時点で系は自動的に停止せねばならない。

入力情報の量は $n(k+1)+1$ だから、 n, k の order の時間でソートの完了する解を求める。

◎ 解の概要:

a) local three-sort T : まず図 5) に数の 3 組の列の変換 T を考える。

$$X = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \gamma_0 \end{pmatrix} \begin{pmatrix} d_1 \\ \beta_1 \\ \gamma_1 \end{pmatrix} \begin{pmatrix} d_2 \\ \beta_2 \\ \gamma_2 \end{pmatrix} \dots \begin{pmatrix} d_i \\ \beta_i \\ \gamma_i \end{pmatrix} \dots \begin{pmatrix} d_n \\ \beta_n \\ \gamma_n \end{pmatrix} \begin{pmatrix} \infty \\ \infty \\ \infty \end{pmatrix} \begin{pmatrix} \infty \\ \infty \\ \infty \end{pmatrix}$$

$$= x_{-1} x_0 x_1 x_2 \dots x_i \dots x_n x_{n+1} x_{n+2}$$

$$T(X) = x_{-1}' x_0' x_1' \dots x_i' \dots x_n' x_{n+1}' x_{n+2}'$$

ただし

$$x_i' = \begin{pmatrix} d_i' \\ \beta_i' \\ \gamma_i' \end{pmatrix} = \begin{pmatrix} \min(\gamma_{i-1}, \beta_i, d_{i+1}) \\ \min(\gamma_{i-1}, \beta_i, d_{i+1}) \\ \max(\gamma_{i-1}, \beta_i, d_{i+1}) \end{pmatrix} \quad i = 0, 1, \dots, n+1$$

$$x_{-1}' = x_{-1}, \quad x_{n+2}' = x_{n+2}$$

$0, \infty$ はそれぞれ数より小、大の数を示す記号。

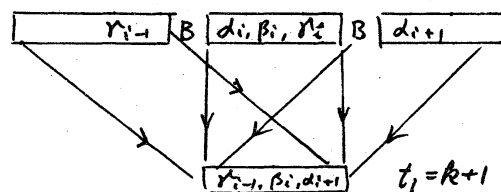
「ソート可能な数」の列 $\{a_1, \dots, a_n\}$ に対し、 $x_i = \begin{pmatrix} a_i \\ a_i \\ a_i \end{pmatrix}$ なる 3 組に対応させ、その列 $x = x_1, x_2, \dots, x_n$ となる。 x に local three sort T を施すと、やはり 1 同様の 3 組の列が得られる。この T は 3 の性質を持つ。 [命題] T を高々 $1.5n$ 回施せば、 $\beta_1'' \beta_2'' \dots \beta_n''$ は $a_1, a_2, \dots, a_n$ のソート結果になっている。(証明未定)

b) セル構造オートマトンによる T の実現:

初期状態: 各素子は、 a_{ij} ($0 \leq i \leq 1$) の 3 組 $\begin{pmatrix} a_{ij} \\ a_{ij} \\ a_{ij} \end{pmatrix}$, あるいは印記号 B の状態とする。数 a_i に対応する素子の部分を i -block と呼ぶ。

phase 1: $d_{ij}, \beta_{ij}, \gamma_{ij}$ は単位時間の遅れで、天々左、中、右へ走る。(k+1) 時間後には、 i -block 上で $\gamma_{i-1,j}, \beta_{ij}, d_{i,j}$ ($j=1, \dots, n$) が衝突する。

phase 2: bit ごとく大小比較をし、同一ブロックの全素子にその結果を配る。可なり。



$\gamma_{i-1,1}, \dots, \gamma_{i-1,n} = x_1, x_2, \dots, x_n$, $\beta_{i,1}, \dots, \beta_{i,n} = y_1, y_2, \dots, y_n$, $d_{i+1,1}, \dots, d_{i+1,n} = z_1, z_2, \dots, z_n$ とすれば、例として、 $\xi = \xi_1, \dots, \xi_n$ $\xi_i = \phi$ ($x_i = y_i$), $\xi_i = 0$ ($x_i < y_i$), $\xi_i = 1$ ($x_i > y_i$) の 5 行ベクトルを作れる。これは 1 時間遅れで出る。もし $x > y$ なら、 ξ_i の最高位 0 はじめに ϕ がない桁が 1 となる。これを i -block

の全素子に配る。同様に、他の2組($x-z$, $y-z$)の比較結果も並行して全素子に配る。これを $(\tilde{x}, \tilde{y}, \tilde{z})$ とすれば、下記の判定表により、 x, y, z の大小順が求められる。

min	mid	max	
x	y	z	$\tilde{x}, \tilde{y}, \tilde{z}$ が1を含まないとき。
y	x	z	$\tilde{x}=1, \tilde{y} \neq 1, \tilde{z} \neq 1$
x	z	y	$(\tilde{x}, \tilde{y}, \tilde{z}) = (\emptyset \emptyset 1), (\emptyset 1 \emptyset), (001), (0 \emptyset 1)$
z	x	y	$\tilde{x} \neq 1, \tilde{y} = \tilde{z} = 1$
y	z	x	$\tilde{x} = \tilde{y} = 1, \tilde{z} \neq 1$
z	y	x	$\tilde{x} = \tilde{y} = \tilde{z} = 1$

従って各素子(bit)ごとには新しく $\alpha_{ij}, \beta_{ij}, \gamma_{ij}$ が求められる。bitごとの大小比較と数の大小比較は、全素子に配り、上記表をいく回操作し、合わせて k 時間と済む。この時間は phase 1 と合わせて、 $k+1+k=2k+2$ であり、local three sort の一回分の終了した時刻である。

同期の問題: 各素子は phase 1, 2 の終了時刻ごとの処理に移るわけであるから、その時刻を一意に知り必要がある。 k は任意に与え、素子内には時計を持たないとは出来ない。この一意解算の手法を利用する。已に報告中のセ〜の両端にあるから、長さ k の各プロセス内での k 時間と同期をとれる。これは上述の phase 1, 2 から $k+1$ 時間と済むのと矛盾し

ない。最後に、[命題]により、 $1.5n$ 回の T の適用でポートが終了するから、この時刻、 $1.5n \cdot 2(k+1) = 3n(k+1)$ を全素子を使って同期問題の解として求めることができる。

結局、求める素子は、本稿の処理 (phase 1, 2), と 2 種類の同期部分の直積の形で構成されることになる。

② この問題の最短時間解は $2n$ 以上だから、 n に近づく方法以外にはより良い設計があるかも知れない。local three out の拡張 (m-out) もこの一つの可能性である。Cole 型の実時間ポートはまた知られていない。

4. おわりに.

いくつかの例から明らかなように、セル構造ポートでは、必要な情報をしかるべき位置へ伝達し処理するが、伝達に要する時間・総時間の限界を定める。これに対して処理装置が1個で補助記憶を持つ系では、必要な情報を100%の時間でしかかる。今後セル構造の並列処理に適った問題を発見して、解くことにしよう。この系の特徴のよりよく理解しよう。しかし、並列処理を単に、処理速度を高めるという側面のみではなく、逐時処理の原理で可能な状態遷移をよせよう、果たすことも調査を行わねばならない。

[文献]

- 1) von Neumann: A.W. Burles. ed. Self-reproducing automata (1966)
- 2) 例212 Waksman, A: Information and Control (1966)
- 3) Cole, S.N: IEEE C-18 (1969)
- 4) Fischer, P.C: JACM 12 (1965)
- 5) Быхватов, Ю.А.: Проблемы Кибернетики, (1967)
- 6) 有次: 通信学会誌 54-C (1971)
- 7) Smith II, A.R: JCSS 6 (1972)
- 8) 西尾: 通信学会全国大会 S8-8 (1973)
- 9) Wolpert, L: in Towards a Theoretical Biology I. (1968)
- 10) Hennie, III, F.C: Iterative arrays of logical circuit, MIT Press (1964)
- 11) 豊: 私信
- 12) Unger, S.H: Proc. IEEE 47 (1959)
- 13) Minsky, M & Papert, S.: Perceptions, MIT Press (1969)
- 14) Boyer, T: PhD dissertation, MIT (1970)
- 15) 箱田: 九州大学シンポジウム (1972)
- 16) Smith II, R.A: IEEE Switching and automata Theory (1971)