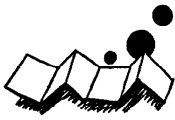


解説

3. ハードウェアアルゴリズムの設計法



3.4 セル構造オートマトンとハードウェアアルゴリズム†

丸 岡 章††

1. はしがき

セル構造オートマトンとは、同一の機能を持った有限オートマトンを規則的に配置し相互接続した系である。このモデルは、並列動作する一様構造系^{1)~7)}や力学系⁸⁾のひとつのモデルとして主に数理的な側面から広く研究されてきたが、最近では VLSI 技術の進歩に伴い、このモデルを VLSI で実現しようとする試みもなされている^{4), 9)}。プロセッサアレイ、セルアレイ等とも呼ばれる VLSI 向きのアルゴリズム—ハードウェアアルゴリズム—の計算のモデルは、基本的にはセル構造オートマトンそのものであり、したがってハードウェアアルゴリズムの良さの条件とされる (i) セルの機能が少数で単純、(ii) データや制御の流れが単純で規則的、(iii) 各々のセルが並列して意味のある計算をして時間を無駄に使わない、等の条件が¹⁰⁾、そのままセル構造オートマトンのアルゴリズムの特質であり、その良さの条件である。ハードウェアアルゴリズムの設計においては、同期のとり方、入出力機構、個々のセルの機能そのものの実現方法等、抽象化されたセル構造オートマトンの場合には必ずしも問題にする必要がなかった側面も考慮する必要がある。またハードウェアアルゴリズムの計算モデルとしては、隣接するセル間の局所的な信号の流れのみならず大域的な流れも許すとしたり、あるいはまた各セルの機能も一種類に限定しないでいくつかの機能を仮定する等、セル構造オートマトンと異なるものを想定する場合もある。これらの相違点はあるにしても、これまで良いハードウェアアルゴリズムとしてよく知られているいくつかのアルゴリズムが示すように、効率の良いハードウェアアルゴリズムを得るのに最も重要な点は、上

に述べた (i) と (ii) の制約のもとで、複数の種類のデータをセル間に巧みに流し、個々のセルにおける計算に費やされる時間とデータの移動に費やされる時間の両者を考慮して、いかに多数のセルを同時に効率良く使用するかということである¹⁰⁾。

ここでは、2. でセル構造オートマトンとは何かを述べた後、3. では、このモデルが単純であるにもかかわらずある意味で万能な普遍性のあるものであるということを、また 4. では、良いアルゴリズムでセル空間にデータが一樣に埋め込まれ、その結果並列性の高い処理が効率良く行われる様子を、それぞれ典型的ないくつかの問題を解くセル構造オートマトンを通してみていくことにする。文献は直接参照するものに限った。

ここで解説するものはセル構造オートマトンの一分野にすぎない。全般についての解説はたとえば文献 5)~7) 等を参照されたい。

2. セル構造オートマトン

1970 年 Conway により考案されたライフゲーム¹¹⁾ は、セル空間のひとつの例である。このゲームは、格子平面のますめに置かれた石の誕生や死滅に関する規則をあらかじめ与えておき、この規則を繰り返し適用して格子平面上に分布する石のパターンの変化を観察する一人遊びのゲームである。ますめには 1 個の石が置かれるか石が無いかのいずれかであるとし、ひとつのますめに (上下左右と対角線方向に) 隣接する 8 個のますめに現在置かれている石の個数により、次の時刻のそのますめの石の有無を次の規則で決める。すなわち、現在ますめに石がある場合は、隣接する石の個数が 2 個または 3 個の場合は次の時刻でもその石は存在し (生存)、隣接する石の個数が 0 か 1 か 4 個以上の場合は無くなる (死滅)。また現在ますめに石が無い場合は、隣接する石の個数がちょうど 3 個の場合次の時刻に新しく石が置かれる (誕生)。この規則を適用

† Cellular Automata and Hardware Algorithms (Department of Information Engineering, Tohoku University).

†† 東北大学工学部情報工学科

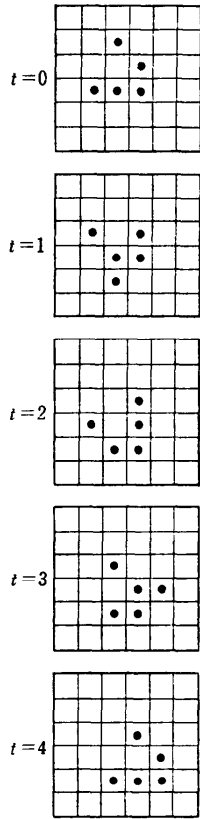


図-1 グライダ

する際に注意すべきことは、ある時刻の石の分布のパターンよりもまず次の時刻の石の有無をすべてのますめについて調べ、次の時刻に一斉にこの新しいパターンに移るということである。このようにして、石の分布の初期パターンを与えれば、引き続き表れるパターンが次々に定まることになる。ライフゲームのシミュレーションは方眼紙の上でもできるが、ディスプレイ装置に石の分布のパターンを次々に表示するプログラムを作ることにより、適当に与えられた初期パターンがあたかも生き物のように活発に活動した後、次第に勢いが衰えてついには小規模な安定パターン、またはある周期で繰り返すパターンとなることを観察することができる。図-1に、対角線方向右下に向い1/4ますめ/ステップの速度でシフトするグライダと呼ばれるパターンを示す。ライフゲームに関しては同好の士による研究も多いが、有限個の石からなる初期パターンで時間の経過とともに石の個数が際限なく増すものがあるかという Conway 自身による問題とか、ライ

フゲームは任意のチューリング機械をシミュレートできるかという問題等が肯定的に解かれている¹¹⁾。

ここでセル空間の形式的な定義を与えて少し概念を整理する。Zを整数の集合とし、整数の対の集合をZ²で表す。Qを有限集合とする。Z²の要素をセルといい、Qの要素を状態という、Z²からQへの写像を様相といい、その集合をCで表す。様相c∈CはZ²の各セルがどの状態をとるかを示すもので、セル(x, y)∈Z²の状態はc((x, y))である。したがって、様相はセル空間全体の状態とみなすこともできる。nを整数とする。様相の推移の仕方を、局所写像と呼ばれるQⁿからQへの写像fと、近傍形と呼ばれるZ²のn個の要素の組X=(z₁, ..., z_n)により定める。すなわち様相cは

$$c'(z) = f(c(z+z_1), \dots, c(z+z_n))$$

が任意のz∈Z²に対して成立するような様相c'へ推移する。ここで、z=(x, y), z_i=(x', y')∈Z²に対してz+z_i=(x+x', y+y')とする。この様相の推移の仕方をCからCへの写像とみなし、fとXから定まる並列写像Fとする。形式的には、セル空間はセル集合Z²、状態集合Q、近傍形X、及び局所写像fで規定されるもので、(Z², Q, X, f)と表される。なお、Qには静止状態と呼ばれる特定の状態qがあり、局所写像はf((q, ..., q))=qを満たすとすることが多い。静止状態以外の状態をとるセルの個数が有限であるような様相を、有限様相という。以上述べた2次元セル空間は容易に一般のd次元(d≥1)に拡張することもできる。

ライフゲームとセル空間の対応は明らかである。ライフゲームのますめ、石の有無、石の置き換えの規則が、それぞれセル空間のセル、状態、局所写像に対応する。Q={0, 1}とし、0は石が無い状態、1は石が有る状態を表すことにする。また、ライフゲームに対する近傍形をX=(z₁, ..., z₉)とおくと、{z₁, ..., z₉} = {(x, y)∈Z² | |x|≤1, |y|≤1}となるが、この近傍形は特に Moore 近傍形と呼ばれる。z₁=(0, 0)とおくと、ライフゲームの規則は、s=b₂+b₃+...+b₉とおくと

$$f((b_1, \dots, b_9)) = \begin{cases} 0, & s=0, 1 \text{ または } 4 \leq s \text{ のとき} \\ b_1, & s=2 \text{ のとき} \\ 1, & s=3 \text{ のとき} \end{cases}$$

と定義される局所写像として表される。ここに、b₁, ..., b₉∈{0, 1}である。

上に述べたセル空間をセル構造オートマトンということもあり、これらの用語は厳密に区別して用いられ

てはないように思われるが、ここではセル空間に少し修正を加えたり、あるいは適当な解釈を与えたりして入出力の機構を付加したものをセル構造オートマトンと呼ぶことにする。セル構造オートマトンの中には、セル空間のある有限の領域内のセルだけに注目し、領域の境界には時間に依存しない固定した信号を境界条件として与えるとするモデルもある。この境界条件を、注目する領域に接するセルの状態とみなすことにすれば、この種のセル構造オートマトンに対しても様相や並列写像等がセル空間の場合と同様に定義される。入出力機構の与え方にはいろいろのものがある。その中には、特定のセルにその局所写像を適当に変更して入出力機能を持たせ、入力(出力)をこのセルへの(からの)の信号の時系列とみなすものや、データを空間的に展開して入力や出力を適当な領域内の状態のパターンとみなすもの等がある。なお、本特集 3.3 でも解説される Kung らの行列の乗算を行うハードウェアアルゴリズムの入出力機構はこれらが組み合わされたものになっている。

$F^i(c) = F(F^{i-1}(c)), i \geq 2$, としよう。初期様相 c と局所写像が与えられたとき、様相の系列 $c, F(c), F^2(c), \dots$ の振舞がどうなるかという問題は、基本的に興味あるものである。ここで論じるセル構造オートマトンの設計の問題も、この系列の振舞に関する所望の条件が与えられたとき、それを実現するような局所写像や初期様相を求める問題とみることもできる。ところで、1次元セル空間をチューリング機械 M のテープに対応するとみなし、セルの状態として M のテープ記号 a のほかにテープ記号 a と M の状態 s の対 (a, s) も考え、後者は状態が s の M のヘッドが記号 a をみていることを表すものとすると、任意のチューリング機械に対してそれをシミュレートする近傍形が $(-1, 0, 1)$ の1次元セル構造オートマトンを構成できることは容易にわかる。したがって、この簡単な構造のセル空間が任意のアルゴリズムをシミュレートできることになり、その結果様相の系列に関する決定問題¹²⁾の多くは決定不能となる。たとえば、 F を適当にとると、与えられた c に対して、 $c, F(c), F^2(c), \dots$ がサイクルに入るか否か、すなわち $F^i(c) = F^j(c)$ となる $i, j (i \neq j)$ が存在するかを問う問題はそのようなもののひとつである。

最近 Wolfram¹³⁾ は、1次元セル空間の様相の時系列を計算機シミュレーションで求め、それを観察した結果、その時空間パターンの規則性等の様子によりセ

ル空間は4つのクラスに分類できることを指摘し、この分類と力学系との対応にも言及している。ただし、セルの個数が有限でしかも端の影響が現われないリング状の1次元セル空間のシミュレーションを行っており、また、初期様相 c としては、系列 $c(0), c(1), \dots$ がランダムとみなされるようなものを用いている。これらのクラスの明確な定義が与えられているわけではないが、あるセル空間に対してランダムな初期パターンをいろいろ変えてみると、その時空間パターンの細部は当然初期様相に応じて変るが、パターン全体の傾向は初期パターンによらずそのセル空間が属するクラスの特徴を保持するという点は興味深い。また、様相の時系列に関するものとしては、任意の有限初期様相が、いずれはある規則性を持った様相となるような1次元セル空間を構成した例¹⁴⁾もある。

3. von Neumann の自己増殖機械

von Neumann は生物が持つ自己増殖の機能をもつ、数学的に厳密に定義された人工の機械を構成することが可能かという問題を提起しこれを肯定的に解いた¹⁵⁾。

この問題は始め運動力学的モデルといわれる次のようなモデルに基づいて考えられた。このモデルは、特定の働きをする9種類の基本部品が組み合わされてできた“機械”が水の表面に浮遊していて、そのまわりでこれら9種類の基本部品がそれぞれ無限個ランダムな運動をしているとするものである。これらの基本部品は、論理和、論理積、否定の機能を持つゲートと遅延素子からなる計算のための部品と、これら計算用部品からの指令に基づいて種々の働きをする部品からなるが、後者は、他の部品の種類を識別する、他の部品を運搬する、結合している部品を離す、逆に離れているものを結合する、部品の集合体が剛体となるように継ぎ手としての働きをするといった機能を持つ。このモデルでは、機械は部品のリストとそれらの接続関係により記述できる。ここで、次のような、機械 M の記述 $\mathcal{D}(M)$ が書き込まれたテープ付きの万能工作機械 Mc を考える。 Mc は、まずテープ $\mathcal{D}(M)$ を複製し、その後 $\mathcal{D}(M)$ を解読して機械 M を自分の側に組立て、作られた機械 M とテープ $\mathcal{D}(M)$ を結合する。ここで、 Mc のテープを自分自身の記述 $\mathcal{D}(Mc)$ とすると、テープ $\mathcal{D}(Mc)$ 付き機械 Mc は自分自身と同じものを作り、自己増殖が行われたことになる(図-2 参照)。テープ $\mathcal{D}(M)$ や機械 Mc を組立てるための Mc の

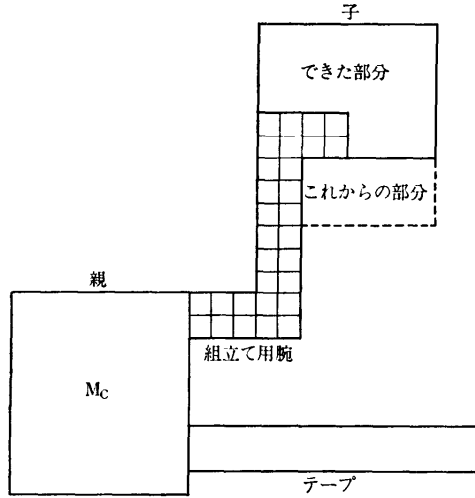


図-2 自己増殖中の機械

組立て用腕は、その先端に識別能力のある部品を付けているので、水面でランダムに動いている部品の中の必要な部品が接触するとその種類を識別して適当な場所に設置することができる。このように各基本部品は無限に供給可能であり、部品そのものを作り出す問題に煩わされる必要はない。

以上の設計方針に基づき自己増殖機械を構成するためには、各々の基本部品の機能を厳密に定義する必要がある。この煩雑でやっかいな問題を回避するため、運動力学的モデルの代りにセル空間が採用されるに至った。自己増殖の可能性を論理的側面から探るという初期の目的のためには、後者のより抽象化の進んだモデルの方が向いている。

von Neumann は自己増殖機械をセル空間に作りあげたが、図-2 に示すようにその設計の基本方針は上に述べたものと同じである。自己増殖機械を作るには、セル空間と、このセル空間に埋め込まれる自己増殖機械の2つを定めなければならない。このうち後者は、ある有限の領域内のセルの状態を指定することにより定められる。すなわち、自己増殖機械とは状態のパターンである。

ところで、上に述べた設計方針によると、自己増殖機械は、任意の機械 M に対してその記述 $\mathcal{D}(M)$ がテープ上に与えられたとき、それを解読して M を作る万能工作機械より得られる。そこで、その構成法について簡単に述べる。

用いるセル空間は、Neumann 近傍形 $((0, 0), (1, 0), (0, 1), (-1, 0), (0, -1))$ を持つ 29 状態のものであ

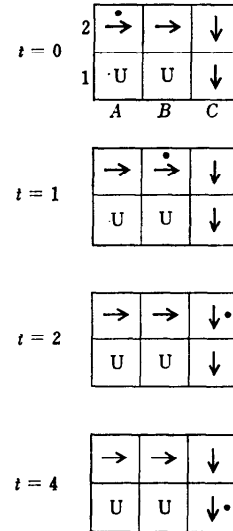


図-3 パルスを伝達する経路

る。局所写像は状態を表す記号にそれぞれ解釈を与えることにより指定する。29種類の状態は、Uで表される非活性状態、20種類の活性状態、及び8種類の過渡状態からなる。後に述べるように、この過渡状態は非活性状態が活性状態に推移する過程で現われる状態である。なお、逆に活性状態から非活性状態への推移は1ステップで行われ、途中にこの種の過渡状態は現われない。また20種類の活性状態は、9種類の静穏状態と11種類の興奮状態からなる。9種類の静穏状態は、 $\rightarrow, \uparrow, \leftarrow, \downarrow$ で表される4種類の一般伝達状態、 $\Rightarrow, \Uparrow, \Leftarrow, \Downarrow$ で表される4種類の特殊伝達状態、及び C_{00} で表される分岐状態からなる。また、11種類の興奮状態は、 \rightarrow, \Rightarrow 等のように“ \cdot ”の付いている8種類の興奮した伝達状態と、 C_{01}, C_{10}, C_{11} で表される3種類の興奮した分岐状態からなる。伝達状態と分岐状態は、それぞれ1単位時間、2単位時間の遅れで、1と0のパルス（それぞれ興奮と静穏の状態に対応する）を伝達する機能を持つことに注意すると、これらの状態の持つ意味は明らかであろう。

万能工作機械を組立てる際に最も基本になるのが、パルスを伝達する経路の形成である。図-3 に $A2$ から $C1$ へ至る一般伝達状態からなる経路をパルスが伝播する例があるが、特殊伝達状態も同様に経路を形成する。一般に、伝達状態にあるセル α からセル β への接続は、セル α の矢印の向きがセル β の方向で、かつセル β の矢印がセル α へは向いていないときに達成されるので、図-3 のセル $B2$ はセル $C2$ へ向けて

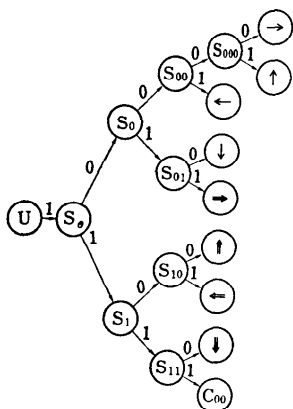


図-4 活性化の過程

接続される。分岐状態も経路を構成しうる（この場合分岐状態の前後は伝達状態でなければならず、接続の向きはこれらの伝達状態の向きで定まる）が、一般伝達状態と特殊伝達状態は隣接していても接続されない。また、伝達状態は経路の合流点になり得、分岐状態は合流点に分岐点にもなり得る。更に伝達状態と分岐状態は、パルスに対して合流点でそれぞれ論理和と論理積の機能をはたす。したがって、閉じた経路でフリップフロップの機能が実現できることに注意すると、有限オートマトンをこれまでに述べたセル空間に埋め込めることは容易に予想されるであろう。

有限オートマトンが構成されることがわかると、テープ内容の読み出しと、組立て用腕による機械の組立てが残された主な問題となるが、ここでは後者について簡単に述べる。非活性化状態にあるセル α の活性化は、 α に向う矢印の伝達状態にある、隣接するセルからパルス列を α に送り込むことにより図-4に示す過程を経て達成される。ここで、 S_9, S_0, \dots, S_{11} は過渡状態を表す。図に示すように、適当なパルス列により9種類の静穏な活性化状態のいずれをも得ることができる。また逆の非活性化は、一般伝達状態（特殊伝達状態）にあるセル α の状態を、 α に向う矢印の特殊伝達状態（一般伝達状態）にある隣接するセルからパルス1を α に送り込むことにより、Uにすることで達成される。これらの活性化と非活性化を用いて、図-5に示すようにD1, D2の状態をそれぞれ \uparrow, \Rightarrow に推移させることにより組立て用腕を伸ばすことができる。このように腕を伸ばしたり縮めたりしながら、テープに書き込まれた内容 $\mathcal{D}(M)$ に応じて、図-4に示す適当なパルス列を送ることにより静穏な状態からなる状態のパターン、すなわち、機械を作ることができ

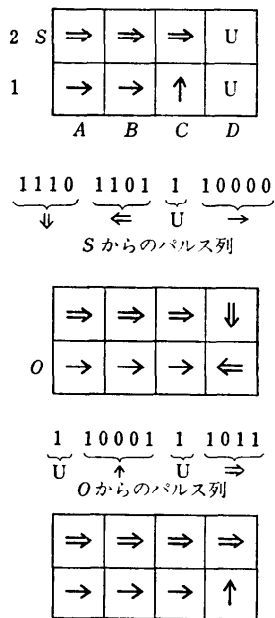


図-5 組立て用腕の伸び方

る。機械を完成させた後、万能工作機械はそれに始動信号を送り動かす。これまで述べた動作を行うよう局所写像を定めることは容易であろう。なお、セル空間は定義より並列動作を行うものであるが、こうして作られた自己増殖機械では実際に状態を変化させ活動するセルは興奮状態にある少数のセルの近傍に限られ、全体の動作は逐次的なものとなる。以上が von Neumann の自己増殖機械の概略である。

自己増殖機械の定義に関して次のような微妙な問題点がある。今仮に0, 1の状態からなるセル空間で、状態1をとるひとつのセルを機械とみなすものとする。局所写像として、Neumannの近傍形の少なくともひとつの近傍の状態が1のとき、値が1となるものを考えると、このセル空間で機械の複製が次々に作られる。しかし、これで自己増殖が行われたとは考えにくい（自己増殖とは考えにくい他の例として、初期様相として任意の状態パターン（有限様相）を与えるとそれを次々に複製するセル空間の報告もある¹⁶⁾）。von Neumannは、自己増殖がtrivialでないためには、機械自体が万能チューリング機械をシミュレートする能力を備えていなければならないと考え、この条件を満たすものを構成した。

このようにしていったん自己増殖機械が構成されることがわかると、状態数を減らすことが試みられたり¹⁷⁾、自己増殖がtrivialではないための条件をゆる

める検討がなされている¹⁸⁾。

4. 効率の良い並列アルゴリズムの設計

セル構造オートマトンで所望の処理を効率良く行わせるため、通常、時空間ダイアグラム上で信号をどのように伝播させ処理するかを考える。この場合、アルゴリズムの設計の立場からは、時間と空間というわれわれが利用できる資源の間の違いは、時間軸上では情報は時間が経過していく方向にしか伝播しないのに対して、空間軸上では任意の方向に伝播するという点だけである。したがって、時間軸上での観察の視点を時間の経過よりも速い速度で移動させれば、見かけ上時間軸上でも両方向に情報を伝播させることができ、両者の違いはなくなる。実際、このような考えに基づき、セル空間で、ある種のシミュレーションを行うのに時間軸と空間軸を入れ替えた方が都合がいい場合もある¹⁹⁾。

一般に、時間と空間を効率良く使う並列性の高いアルゴリズムでは、対象とするデータを空間的に一様なものとしてセル空間に埋め込み、各々のセルは、一樣に分布するデータに局所的な処理を施し、中間的な出力や状況を作り出し、これを徐々に最終的なものに近づけていく。この場合、全体をうまく動作させるためには、中間結果の伝播の速度や各々のセルの計算のタイミングを時空間ダイアグラムの中で注意深く調整しなければならず、この種のアルゴリズムの正当性の証明は一般に複雑なものとなる。

4.1 一斉射撃の問題

一斉射撃の問題とは、一列に並んだ兵士の列の左端にいる隊長が撃てと号令をかけたとき、隣りのものしか届かない号令をたよりに、隊長および兵士全員がしばらくして一斉に射撃するようになるかという問題である。この問題は、 n 個のセルからなる1次元セル構造オートマトンを用いて正確に述べると次のようになる。様相（1次元セル空間の様相を n 個のセルに限定したもの）を、長さ n の状態の系列で表すとして、(i) 初期様相として $sg \in Q^n$ を与えると、ある時刻 t に $sf \in Q^n$ なる様相となり、しかも (ii) 時刻 $0, 1, \dots, t-1$ にはどのセルの状態も sf ではないとき、このセル構造オートマトンは時刻 t に一斉射撃するという。ここに、 sg, sf は特定の状態で、 q, sg, sf のどの2つも相異なるとする。一斉射撃の問題とは、近傍形を $(-1, 0, 1)$ としたとき、任意の n に対して長さ n の1次元セル構造オートマトンが一斉射撃するよう

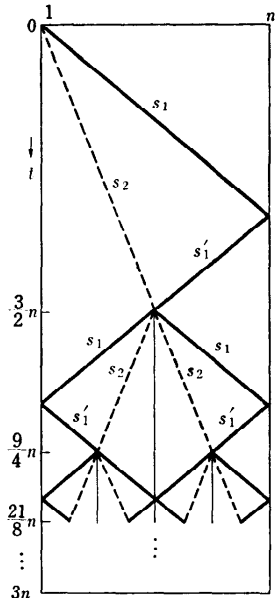


図-6 一斉射撃の $3n$ 時間の解

に、 Q を適当に定め局所写像 f を決定せよというものである。 $s_x \in Q$ を左右の境界に加えられる信号とすると、射撃が隊長からの号令が原因で引き起こされることを保障するためには、 $f((s_x, q, q)) = f((q, q, q)) = f((q, q, s_x)) = q$ という条件は満たされなければならない。

次に、一斉射撃のひとつの解を簡単に説明する。

図-6 に示すように、始め左端のセル（隊長）から右方向へ速度1の信号 s_1 と速度 $1/3$ の信号 s_2 を送る。ただし、速度の単位はセル/ステップとする。 s_1 は右端のセルに到達すると、 s_1' に変わり同じ速度で今度は左方向へ伝播する。この結果、 s_1' と s_2 が衝突する位置にあるセルは自分が中央にいることを知る。その後の信号の流れは、セル構造オートマトンがあたかも中央のセルで左半分と右半分に分割されたようなものとなり、右半分に対しては上に述べたものと同様の信号の流れが、また左半分に対しては左右を逆転した鏡像の関係にある信号の流れが生じる。その結果、それぞれの部分の中央のセルで再び分割される。この分割を各部分の長さが2となるまで繰り返す。各セルは各部分の長さが2であることを知ることができるので、その次の時刻に一斉に状態 sf に推移（射撃）すればよい。以上の動作をする、状態数が有限（文献20）には27状態の解がある）のセル構造オートマトンの局所写像を定めることができる。

上述のものは、射撃までに $3n$ 時間要する解であるが、後藤²³⁾ と Waksman²²⁾ は $2n-2$ 時間の解を与えている ($2n-2$ 時間より早くできないことも容易にわかる)。また、セルの結合関係を一般のグラフで表されるものに拡張した場合の一斉射撃の問題も論じられ興味ある結果が得られている^{23), 24)}。

4.2 2進数の乗算

Atrubin は、2進数で表した2つの整数 $a=(a_n \dots a_1 a_0)_2, b=(b_n \dots b_1 b_0)_2$ の積 $ab=(p_{2n+1} p_{2n} \dots p_1 p_0)_2$ を実時間で計算する $(-1, 0, 1)$ 近傍形の右半無限1次元セル構造オートマトンを構成した²⁵⁾。ここに、 $a_i, b_i, p_j \in \{0, 1\}$ である。このオートマトンの左端のセルに時系列 $(a_0, b_0), (a_1, b_1), \dots$ を入力すると、左端のセルから実時間で時系列 p_0, p_1, \dots が出力される。ここで、 p_0, p_1, \dots は

$$s_i = \sum_{j=0}^i a_{i-j} b_j \quad (1)$$

とおくと

$$p_i = (s_i + c_i) \bmod 2 \quad (2)$$

$$c_{i+1} = ((s_i + c_i) - (s_i + c_i) \bmod 2) / 2$$

と表される ($i \geq 0$)。ただし、 $c_0 = 0$ とし、 $x \bmod 2$ は x を2で割ったときの余りを表す。Atrubin のセル構造オートマトンは、(2)の p_i を、(1)のコンボリューションを中ほどの項から両端に向けて、ひと桁下の計算の途中結果からの桁上げを加えながら計算するが、そのために各セルは T, R_0, R_1, P, C のレジスタを持っている。 (a_i, b_i) を g_i で表すことにする。図-7に示すように、左端のセル1に入力系列 g_0, g_1, \dots が入ると、レジスタ T を通して速度1で右へシフトされ、 g_i がセル $\lfloor i/2 \rfloor + 1$ に到達したら (このことはそのセルのレジスタ R_0 または R_1 の内容が*であることよりわかる。ここに、*はデータが入る前にレジスタに蓄えられている記号である)、そのセルのレジスタ $R_{i \bmod 2}$ に永久に蓄えられる。ただし、 $\lfloor x \rfloor$ は x を越えない最大の整数を表す。また、各レジスタの値が図-8に示すようなものとするとき、時刻 $t+1$ におけるセル k のレジスタ P と C の値はそれぞれ

$$e = a_{2k-1} b_{i-2k+1} + a_{2k} b_{i-2k} + a_{i-2k} b_{2k} + a_{i-2k+1} b_{2k-1} + p + c$$

とおくと

$$p' = e \bmod 2, \quad c' = (e - e \bmod 2) / 2$$

と計算される。ただし、セル1や、入力データが蓄えられているセルのうちで最も右のセルに対しては、この部分の計算は多少変更される。時刻 $t+1$ にセル k

	セル1	2	3	4	5	6	7		
$t=12$		g_{12}	g_{11}	g_{10}	g_9	g_8	*	*	T
		g_1	g_3	g_5	g_7	*	*	*	R_1
		g_0	g_2	g_4	g_6	*	*	*	R_0
$t=13$		g_{13}	g_{12}	g_{11}	g_{10}	g_9	*	*	T
		g_1	g_3	g_5	g_7	*	*	*	R_1
		g_0	g_2	g_4	g_6	g_8	*	*	R_0
$t=14$		g_{14}	g_{13}	g_{12}	g_{11}	g_{10}	*	*	T
		g_1	g_3	g_5	g_7	g_9	*	*	R_1
		g_0	g_2	g_4	g_6	g_8	*	*	R_0
$t=15$		g_{15}	g_{14}	g_{13}	g_{12}	g_{11}	g_{10}	*	T
		g_1	g_3	g_5	g_7	g_9	*	*	R_1
		g_0	g_2	g_4	g_6	g_8	*	*	R_0

図-7 入力データの流れ

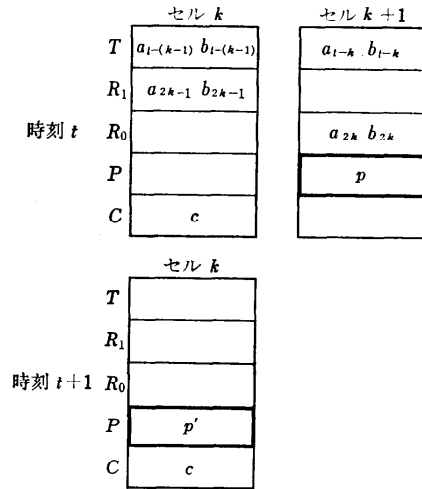


図-8 コンボリューションの計算

のレジスタ P に蓄えられるこの値 p' は、最終的には p_{i+k} になる値であるが、上の計算が繰り返して適用されながら速度1で左方向へレジスタ P を通して伝播していき、時刻 $t+k$ に p_{i+k} となり、セル1のレジスタ P より出力される。上の計算の1ステップで、レジスタ T の内容自身が右へ速度1で移動しているから T から2個の入力データ ($g_{i-(k-1)}$ と g_{i-k}) が、またレジスタ R_0, R_1 には1セル当り2個の入力データが蓄えられているから R_0, R_1 からは2個の入力データ (g_{2k-1} と g_{2k}) が、またレジスタ C から下の桁を一部分計算した結果でくる桁上げ (c) が、それぞれレジスタ P に取り込まれることに注意されたい。結局 p_i の計算は、'おおよそ' 時刻 $i-(i/4)$ にセル $i/4$ のレジスタ P で始まり、1ステップに4個の入力データをレジスタ

T, R_0, R_1 から取り込みながら左方向に速度1で進み、 $i/4$ 時間後の時刻 i にセル1で終了することになる。

4.3 文脈自由言語の認識

Kosaraju は、セル空間に Cocke, Kasami, Younger のアルゴリズム^{26), 27)} を埋め込むことにより、文脈自由言語の認識問題を線形時間で解く Moore 近傍形の2次元セル構造オートマトンを構成した²⁸⁾。

$G=(V_N, V_T, P, S)$ を文脈自由文法とする。ここに、 V_N, V_T を、それぞれ非終端記号の集合、終端記号の集合とし、 $S \in V_N$ を初期記号とする。 P は、 $A \rightarrow a$ か $A \rightarrow BC$ のタイプの書き換え規則の有限集合とする。ここに、 $a \in V_T, A, B, C \in V_N$ である。 P が $A \rightarrow \gamma$ を含むとき、任意の $\alpha, \beta \in (V_N \cup V_T)^*$ に対して $\alpha A \beta \Rightarrow \alpha \gamma \beta$ とし、関係 \Rightarrow の反射的、推移的閉包を $\stackrel{*}{\Rightarrow}$ と表す。ここに、一般に T^* は集合 T の要素の有限の長さの系列の集合である。 $a_1 a_2 \dots a_n \in V^*, n \geq 1$, が与えられているとし

$$D_i^j = \{A | A \in V_N, A \stackrel{*}{\Rightarrow} a_i \dots a_j\}, 1 \leq i \leq j \leq n$$

とする。また、 $X, Y \subseteq V_N$ に対して $X * Y = \{A | \exists B, C, B \in X, C \in Y, A \rightarrow BC \in P\}$ とする。Cocke, Kasami, Younger のアルゴリズムは

$$D_i^j = \{A | A \in V_N, A \rightarrow a_i \in P\}, 1 \leq i \leq n$$

$$D_i^j = \bigcup_{i \leq k \leq j} D_i^k * D_{k+1}^j, 1 \leq i \leq j \leq n \quad (3)$$

の計算を行い、 D_i^j が S を含むか否かにより $a_1 \dots a_n$ を G が生成するか否か、すなわち $a_1 \dots a_n \in L(G) = \{x | x \in V^*, S \stackrel{*}{\Rightarrow} x\}$ か否かを決定するものである。

Kosaraju のセル構造オートマトンは、入出力のための入力テープと入力ヘッド、及びセル空間を動くセルヘッドを持つ。詳しい説明は省略するが、始めに入力 a_1, \dots, a_n を読み込み、図-9 の $\delta=0$ の対角線上の

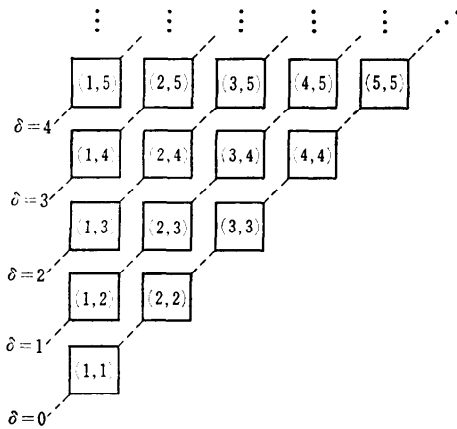


図-9 文脈自由言語を認識するセル構造オートマトン

セル $(1, 1), (2, 2), \dots$ にそれぞれ D_{11}, D_{22}, \dots を蓄える。次いで、これらのセル $(1, 1), \dots, (n, n)$ を1次元セル構造オートマトンとみなし、一斉射撃を利用して一斉に発火させる。この発光を合図に各セルは一斉にこのアルゴリズムの主たる計算過程である(3)のコンボリューションを求める過程に入る。この計算過程では、各セル (i, j) は最終的には D_i^j を計算する ($1 \leq i \leq j \leq n$). $j=i+\delta$ とおくとし、この計算を、 $\delta=1$ のセル、 $\delta=2$ のセル、 \dots と順次(3)に基づきそのまま計算するものとする、セル $(1, n)$ が D_1^n を計算し終るまで、 c を適当な定数として cn^2 時間を要する。そこで、計算時間を線形(すなわち、適当な定数 c に対して cn 時間)にするため、4.2のアルゴリズムのように、各セル (i, j) が D_i^j を完全に計算し終るのを待たず、(3)の右辺の一部の項からなる答を得たら計算を先に進めるという方針をとる。すなわち、各セル (i, j) は、 $D_i^l * D_{l+1}^j$ のうち l が i と j の中ほどのある区間内に入るものの和集合を、時間とともにこの区間幅を増やしながらかし計算し(1ステップに区間幅を2だけ増やす)、最終的に全区間 $[i, j-1]$ にわたるようにする。結局、セル (i, j) は時刻 $2(j-i)-1$ に D_i^j の計算を完了する。ただし、セル $(1, 1), \dots, (n, n)$ が一斉に(3)のコンボリューションを計算し始めた時刻を0とする。この部分の計算の概略は次のようになる。

$j-i$ が偶数か奇数かでいくぶん状況が異なるので、ここでは $j-i$ を奇数とし、 $j-i=\delta=2d+1$ とおく。セル (i, j) は(3)のコンボリューションの実質的な計算を時刻 $3d+2$ から $4d+1$ にわたって行うが、時刻 $t=3d+k+1, 1 \leq k < d$, にセル (i, j) のひとつのレジスタは(3)の計算の途中結果である

$$\bigcup_{-k \leq m \leq k} D_i^{i+d-m} * D_{i+d-m+1}^j \quad (4)$$

を蓄えている。一方、時刻 $2(d-(k+1))-1, 2(d+k+1)-1$ にセル $(i, i+d-(k+1)), (i, i+d+k+1)$ でそれぞれ計算された $D_i^{i+d-(k+1)}, D_{i+d+k+1}^j$ を、巧妙に速度を調整しながら垂直方向(上)に伝播させ、時刻 t に同時にセル $(i, j-1)$ に到達するようにする。同様に、 $D_{i+d-(k+1)+1}^j, D_{i+d+k+2}^j$ をそれぞれ水平方向(左)に伝播させ、時刻 t に同時にセル $(i+1, j)$ に到達するようにする。セル (i, j) は、これらの信号を元に

$$D_i^{i+d-(k+1)} * D_{i+d-(k+1)+1}^j \cup D_{i+d+k+1}^j * D_{i+d+k+2}^j$$

を計算し、これと(4)の和集合をとり

$$\bigcup_{-(k+1) \leq m \leq k+1} D_i^{i+d-m} * D_{i+d-m+1}^j$$

を時刻 $t+1=3d+k+2$ に得る。時刻 $4d+1=2(2d+1)-1$ には上式の $i+d-m$ の値の範囲は $[i, j-1]$ にわたり、上式の値は d_i^j となる。このようにして、セル $(1, n)$ が d_i^j を計算した時点で、 $S \in d_i^j$ か否かにより $a_1 \cdots a_n$ を受理するか否かの出力を出せばよい。

チューリング機械に対する線形加速定理¹²⁾は明らかにセル空間に対しても成立するので、上に述べたことより、任意の $\varepsilon > 0$ 、任意の文脈自由言語 $L(G)$ に対して、 $L(G)$ を $(1+\varepsilon)$ 時間で認識する2次元セル構造オートマトンが構成されることになる。

5. む す び

セル構造オートマトンに関する研究のうち、設計に関するものをいくつか取りあげ、解説した。セル構造オートマトンは、始め von Neumann により自己増殖機械のモデルとして用いられたが、Yamada, Amoroso の文献 29) 等を契機に計算のモデルとして広く研究されるようになった。ここでは触れなかったが、この分野には、局所写像が時間と共に変化してもよい（空間的には一様）とするテセレーションオートマトンに関するもの、セル空間の情報伝達能力に関するもの、局所写像と並列写像の関連に関するもの、セルの間の結合の仕方を一般的にした一斉射撃に関するもの、セル空間と力学系の関連に関するもの等、日本の研究者により独自に提起された問題や深く掘り下げられたテーマが多い。詳しくは、文献 5), 7) 等を参照されたい。

参 考 文 献

- 1) Hennie, F.C.: Iterative Arrays of Logical Circuits, MIT Press (1961).
- 2) Aladyev, V.Z.: Mathematical Theory of Homogeneous Structures and Their Applications, Tallinn Valgus (1980).
- 3) *Proceedings of the International Symposium on Uniformly Structured Automata and Logic*, Tokyo (1975).
- 4) Farmer, D., Toffoli, T. and Wolfram, S. (eds.): Cellular Automata, North-Holland (1983).
- 5) Nishio, H.: A Classified Bibliography on Cellular Automata Theory—with Focus on Recent Japanese References—, *Proceedings of International Symposium on Uniformly Structured Automata and Logic*, Tokyo, pp. 206-214 (1975).
- 6) Smith III, A.R.: Introduction to and Survey of Polyautomata Theory, in Automata, Languages, Development (Lindenmayer, A. and Rozenberg, G. (eds.)), pp. 405-422, North-Holland (1976).
- 7) 丸岡 章: セル構造オートマトン, 信学誌 Vol. 61, No. 10, pp. 1073-1083 (1978).
- 8) Hedlund, G.A.: Endmorphisms and Automorphisms of the Shift Dynamical System, *Math. Syst. Theory*, Vol. 3, No. 4, pp. 320-375 (1969).
- 9) Vollmar, R. and Legendi, T. (eds.): Cell-processors and Cellgorithms, Braunschweig (1981).
- 10) Foster, M.J. and Kung, H.T.: The Design of Special-purpose VLSI chips, *Computer*, Vol. 13, No. 1, pp. 26-40 (1980), 邦訳: 特殊用途向け LSI チップの設計アルゴリズム, 日経エレクトロニクス, Vol. 11, No. 10, pp. 150-169 (1980).
- 11) Gardner, M.: Wheels, Life, and Other Mathematical Amusement, Freeman (1983).
- 12) Hopcroft, J.E. and Ullman, J.D.: Formal Languages and Their Relation to Automata, Addison-Wesley (1969).
- 13) Wolfram, S.: Universality and Complexity in Cellular Automata, in Cellular Automata ((4)), pp. 1-35 (1983).
- 14) Maruoka, A., Kimura, M. and Shoji, N.: Pattern Decomposition for Tessellation Automata, *Theor. Comput. Sci.*, Vol. 14, pp. 211-226 (1981).
- 15) Burks, A.W. (ed.): Essay on Cellular Automata, Univ. of Illinois Press (1970).
- 16) Ostrand, T.J.: Pattern Reproduction in Tessellation Automata of Arbitrary Dimension, *J. Comput. Syst. Sci.*, Vol. 5, pp. 623-628 (1971).
- 17) Codd, E.F.: Cellular Automata, Academic Press (1968).
- 18) Langton, C.G.: Self-reproduction in Cellular Automata, in Cellular Automata ((4)), pp. 135-144 (1983).
- 19) Takahashi, H.: Undecidable Questions About the Maximal Invariant set, *Int. Control*, Vol. 33, No. 1, pp. 1-34 (1977).
- 20) 小林孝次郎: オートマトン理論とパズル・一斉射撃の問題, 別冊・数理科学「パズル」I, pp. 106-110 (1976).
- 21) 後藤英一: 情報科学への道, 共立出版 (1966).
- 22) Waksman, A.: An Optimal Solution to Firing Synchronization Problem, *Inf. Control*, Vol. 9, pp. 66-78 (1966).
- 23) Kobayashi, K.: On the Minimal Firing Time of the Firing Squad Synchronization Problem for Polyautomata Networks, *Theor. Comput.*

- Sci.*, Vol. 7, pp. 149-167 (1978).
- 24) Nishitani, Y. and Honda, N.: The Firing Squad Synchronization Problems for Graphs, *Theor. Comput. Sci.*, Vol. 14, pp. 39-61 (1981).
- 25) Atrubin, A. J.: A One-dimensional Real-time Iterative Multiplier, *IEEE Trans. Comput.*, Vol. EC-14, pp. 394-399 (1965).
- 26) Kasami, T.: An efficient Recognition and Syntax Algorithm for Context-free Languages, *Scientific Report AFCRL-65-758*, Air Force Cambridge Research Laboratory (1965).
- 27) Younger, D. H.: Recognition of Context-free Languages in Time n^3 , *Inf. Control*, Vol. 10, No. 2, pp. 189-208 (1967).
- 28) Kosaraju, S. R.: Speed of Recognition of Context-free Languages by Array Automata, *SIAM J. Comput.*, Vol. 4, No. 3, pp. 331-340 (1975).
- 29) Yamada, H. and Amoroso, S.: Tessellation Automata, *Inf. Control*, Vol. 14, pp. 299-317 (1969).

(昭和 60 年 1 月 9 日受付)