反復法に基づく並列処理方式の収束性†

横山 正明††

工業諸分野における設計計算が非常に大形化する傾向にあり，数 GFLOPS 級の能力をもつ超高性能計算機の出現が強く望まれている．この要望にこたえるためには演算ユニットを複数もつ並列処理計算機方式を採用する以外に解決策はなく，このような並列処理計算機に対して SIMD 形，MIMD 形のノイマン形計算機あるいはデータフロー計算機など数多くの並列処理計算機が提案されている．

設計計算の大半は有限要素法などによる偏微分方程式の数値計算であるが，このような計算を反復法に基づいて並列処理計算機で実行する際に，問題領域の要素分割により発生した節点群を複数ある演算ユニットへ割り当てる割当て方式ならびに割り当てられた節点間の計算順序付けの方式に工夫を施せば，計算効率の高い並列処理方式を確立することができる．それゆえ，本論文ではこの節点割当てと計算順序付けの方式に工夫を施した並列処理方式を取りあげ，その収束性などについて考察している．

1. 緒 言

近年，工業諸分野における設計計算が非常に大形化する傾向にあり，現在の高性能計算機のもつ性能をはるかに越えた数 GFLOPS 級の能力をもつ超高性能計算機の出現が強く望まれている．たとえば，NASA では風洞シミュレーション用にこのような超高性能計算機の設計提案を公募し，これに対して Burroughs 社とか CDC 社などが複数の演算ユニットをもつ超高性能の並列処理計算機を提案したときいている[1]．各社いずれも複数の演算ユニットをもつ計算機を提案したのは，単一の演算ユニットの計算機では演算速度の限界が見えてきており，演算速度を高めるためには複数の演算ユニットを結合し，それらを同時に動作させて，総合的に演算速度を高める方式を採用する以外に解決策が見出せないからであろう．

いずれにしても，このような超高性能計算機を必要とする分野の問題のほとんど大半は偏微分方程式の数値解を求める問題であり，通常，差分法，有限要素法，重みつき残差法あるいは境界要素法などにより問題の離散化を行い，得られた連立一次方程式を消去法あるいは反復法により解き，数値解を求める．

いまここで，複数の演算ユニット（以後 AU と略記する）をもつ並列処理計算機で，このような問題を反復法により数値計算する過程を考えると次のようになる．まず，問題領域の要素分割を行い，その結果発生した節点群につき一次方程式を作成し，作成された一次方程式群を連立して連立一次方程式を作成する．こののち，1行に1AU ずつ順次割り当てていき，この割当て操作を全行終わるまで繰り返す．続いて，SBOR 法の反復計算式の作成に必要な逆行列ならびに行列の積の計算を行い，反復計算式を作成する．そして，全 AU 同期をとりながら周知のごとき反復計算を所望の収束条件が満足されるまで繰り返す．

以上の計算過程にはかなりの計算量を必要とする逆行列ならびに行列の積の計算が含まれている．しかし，連立一次方程式の作成のとき，行および列を適切に入れ換えることによりこの計算を省略することが可能となる．この結果，計算効率の高い並列処理方式を確立することができると考えられる．現在までに反復法に基づく並列処理方式についてはいくつかの研究成果が報告されている[2)~4)]が，このような観点から検討した並列処理方式はまだ報告されていない．

連立一次方程式の行および列の入換えは AU への節点の割当てと計算の順序付けの入換えに相当する．それゆえ，本論文では AU への節点割当てと分担節点内の計算順序付けに工夫を施した，計算効率の高い，SBOR 法に基づいた並列処理方式を取りあげ，その収束性などについて考察している．

2. 反復法に基づく並列処理方式の収束性

最初に，AU の総数を N，問題領域の要素分割により発生した節点の総数を nN，各節点の物理量の個数を1と仮定する．

† Convergence of Parallel Processing by Iterative Method by MASAAKI YOKOYAMA (Dept. of Precision Machinery and Systems, Graduate School of Science and Engineering, Tokyo Institute of Technology).

†† 東京工業大学総合理工学研究科精密機械システム専攻

各節点に関する一次方程式を適当に連立して

$$\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b} \tag{1}$$

なる連立一次方程式を作成する．$\boldsymbol{A}$ は nN 行 nN 列の係数行列，$\boldsymbol{x}$ は nN 行の解ベクトル，$\boldsymbol{b}$ は nN 行の定数項ベクトルである．こののち，1行に1AUずつ順次に割り当てていき，この割当て操作を全行終るまで繰り返す．各 AU には n 行割り当てられる．このようにすると，式（1）は**図1**のようにブロック分割されたものと同等になる．AU 総数は N であるから $\boldsymbol{A}_{ij}(i=1\sim n, j=1\sim n)$ は N 行 N 列の小行列である．

ここで，次式のような反復計算過程を考える：

$$\left.\begin{aligned}
&\tilde{\boldsymbol{x}}_1^{(\nu+1)}=\boldsymbol{A}_{11}^{-1}[\boldsymbol{b}_1-(\boldsymbol{A}_{12}\boldsymbol{x}_2^{(\nu)}+\boldsymbol{A}_{13}\boldsymbol{x}^{(\nu)}+\cdots\\
&\qquad+\boldsymbol{A}_{1n-1}\boldsymbol{x}_{n-1}^{(\nu)}+\boldsymbol{A}_{1n}\boldsymbol{x}_n^{(\nu)})]\\
&\boldsymbol{x}_1^{(\nu+1)}=\boldsymbol{x}_1^{(\nu)}+\omega(\tilde{\boldsymbol{x}}_1^{(\nu+1)}-\boldsymbol{x}_1^{(\nu)})\\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&\tilde{\boldsymbol{x}}_i^{(\nu+1)}=\boldsymbol{A}_{ii}^{-1}[\boldsymbol{b}_i-(\boldsymbol{A}_{i1}\boldsymbol{x}_1^{(\nu+1)}+\cdots\\
&\qquad+\boldsymbol{A}_{ii-1}\boldsymbol{x}_{i-1}^{(\nu+1)}+\boldsymbol{A}_{ii+1}\boldsymbol{x}_{i+1}^{(\nu)}+\cdots\\
&\qquad+\boldsymbol{A}_{in}\boldsymbol{x}_n^{(\nu)})]\\
&\boldsymbol{x}_i^{(\nu+1)}=\boldsymbol{x}_i^{(\nu)}+\omega(\tilde{\boldsymbol{x}}_i^{(\nu+1)}-\boldsymbol{x}_i^{(\nu)})\\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&\tilde{\boldsymbol{x}}_n^{(\nu+1)}=\boldsymbol{A}_{nn}^{-1}[\boldsymbol{b}_n-(\boldsymbol{A}_{n1}\boldsymbol{x}_1^{(\nu+1)}+\boldsymbol{A}_{n2}\boldsymbol{x}_2^{(\nu+1)}\\
&\qquad+\cdots+\boldsymbol{A}_{nn-1}\boldsymbol{x}_{n-1}^{(\nu+1)})]\\
&\boldsymbol{x}_n^{(\nu+1)}=\boldsymbol{x}_n^{(\nu)}+\omega(\tilde{\boldsymbol{x}}_n^{(\nu+1)}-\boldsymbol{x}_n^{(\nu)})
\end{aligned}\right\} \tag{2}$$

この反復過程は $\boldsymbol{A}_{ii}^{-1}(i=1\sim n)$，$\boldsymbol{A}_{ii}^{-1}\boldsymbol{A}_{ij}(i=1\sim n, j=1\sim n, i\neq j)$ などの計算が完了した段階で全 AU 同期をとって並列処理できる．もちろん，プログラムに工夫を施せば逆行列ならびに行列の積の計算も並列処理は可能である．式（2）は周知のように SBOR 法の反復計算式である．行列 $\boldsymbol{D}$ を**図2**のように定義するとき，$\boldsymbol{D}$ が正定値ならば SBOR 法が収束する必要十分条件は，$\boldsymbol{A}$が正定値行列で，$0<\omega<2$ であることである[5]．有限要素法から作成される係数行列 $\boldsymbol{A}$ は対称行列であり，したがってエルミート行列であり，かつ正定値である[6]．他方，行列 $\boldsymbol{A}$ が正定値エルミート行列であれば，行列 $\boldsymbol{A}$ の任意の分割は行列 $\boldsymbol{D}$ が正定値エルミート行列であることを保証している．ゆえに，有限要素法から作成される式（2）の SBOR 法に基づく並列処理方式は，全 AU 同期をとって反復過程が進行すれば，理論的には $0<\omega<2$ で必ず収束する．

このような式（2）の並列処理方式を採用する限りにおいては AU への節点の割当て方式も分担節点内

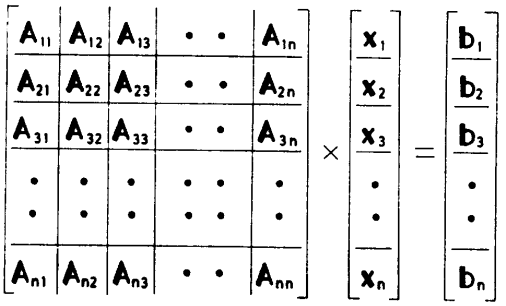

図 1 連立一次方程式のブロック分割
Fig. 1 Matrix equation partitioned into block submatrices.

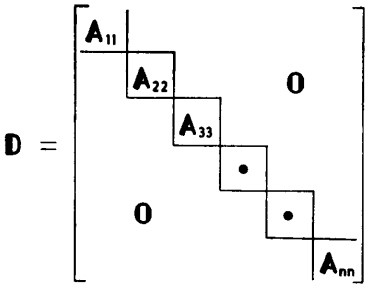

図 2 行列 $\boldsymbol{D}$ の定義
Fig. 2 Block diagonal matrix $\boldsymbol{D}$.

の計算順序付けの方式も問題にならない．しかし，前処理の段階で節点割当ておよび計算順序付けに工夫を施して，かなりの計算量を必要とする逆行列ならびに行列の積の計算を省略することが可能ならば，計算効率向上の観点からこの工夫を施すべきであると考える．戸川博士もその著書の中で指摘している[7]ように，SBOR 法の場合 $\boldsymbol{A}_{ii}^{-1}(i=1\sim n)$，$\boldsymbol{A}_{ii}^{-1}\boldsymbol{A}_{ij}(i=1\sim n, j=1\sim n, i\neq j)$ の計算量を考えずにその収束性だけを論ずるのは意味がない．

それではいかなる場合に逆行列ならびに行列の積の計算を省略することができるか．それは $\boldsymbol{A}_{ii}(i=1\sim n)$ が対角行列のときである．$\boldsymbol{A}_{ii}$ が対角行列のとき，$\boldsymbol{A}_{ii}^{-1}$ は $\boldsymbol{A}_{ii}$ の対角成分の逆数を対角成分とする対角行列となる．この結果，式（2）の反復計算式は次式のような形式になる．

$$\left.\begin{aligned}
&\tilde{x}_i^{(\nu+1)}=[b_i-(a_{i1}x_1^{(\nu+1)}+\cdots+a_{ii-1}x_{i-1}^{(\nu+1)}\\
&\qquad+a_{ii+1}x_{i+1}^{(\nu)}+\cdots+a_{in}x_n^{(\nu)})]/a_{ii}\\
&x_i^{(\nu+1)}=x_i^{(\nu)}+\omega(\tilde{x}_i^{(\nu+1)}-x_i^{(\nu)})
\end{aligned}\right\} \tag{3}$$

次に，$\boldsymbol{A}_{ii}$ が対角行列になるための条件について考える．$\boldsymbol{A}_{ii}$ に対応する $\boldsymbol{x}_i$ には N 要素あり，それら N 要素は同時点で同時に計算されるものである．同時に

計算される x_i に対応する A_{ii} が対角行列であるということは x_i の各要素に関する反復計算式の中に x_i に含まれるほかの要素は入っていないことを意味する．これを節点の観点からのべると

「全 AU 同時に計算する節点どうしは相互に近傍節点にならない」

ことと同等である．これを「近傍の非同時性」と仮称することにする．ここで，近傍とはある節点に注目するとき，その節点の物理量に直接影響を与える物理量を含む節点の集合を意味する．

この近傍の非同時性を実現するために，AU への節点割当てと分担節点内の計算順序付けの方式に工夫を施す必要があるので，次にこの問題を考える．まず，問題領域の要素分割であるが，議論を簡単にするために，二次元問題では正方形あるいはそれに同方向の対角線を引いた三角形要素，三次元問題では立方体あるいはそれを分割した四面体要素を用いることにする．このような要素分割方式によって発生した節点群に対して前述の近傍の非同時性を実現する最も簡単な方法は次のように行うことである．すなわち，

$$p \geqq p_0 \text{ かつ } q \geqq q_0 \text{ かつ } r \geqq r_0 \qquad (4)$$

を満足するように問題領域を x 方向に p 節点，y 方向に q 節点，z 方向に r 節点を含む直方体の網目状に分割し，この直方体（以後この直方体あるいは二次元問題のときの長方形をセルと仮称する）を座標値の大きさにしたがって順次 AU に割り当てていく．こののち，全 AU は同じ回に割り当てられたセル内の同じ位置にある節点に関して同時に計算するようにする．これにより近傍の非同時性を実現することができる．$p=3$, $q=2$, $r=1$, $N=4$, $n=18$ の場合の節点割当てと計算順序付けの例を**図3**に示す．

近傍の非同時性が実現されていないときに，式(3)の形式の反復計算式を採用しても $0<\omega<2$ で必ずしも収束するとは限らない．これは非同時性が実現されていない節点については式（3）中の $x^{(s+1)}$ を $x^{(s)}$ にしなければならず，厳密な SBOR 法にならないからである．ちなみに，次章に挙げる問題（A）では $p=q=1$ のとき近傍の非同時性を実現しないとき $\omega>1.3$ 程度で発散してしまう．また，Odd-even SOR 法および Typewriter ordering 法がよく使用されていると言われている[8]が，近傍の非同時性が実現されておらず，かつ式（3）の形式の反復計算式を用いるならばこの両法とも SBOR 法にはならず，収束速さがおそいばかりか，$0<\omega<2$ での収束性は必ずしも保証されない．Odd-even SOR 法は節点配列を NODE (I, J) とすると，$I+J=$odd の節点すべてについて同時に計算し，次に $I+J=$even の節点すべてについて同様に計算する方式であるが，図3のような三角形要素分割のときは同時に計算している節点どうしが相互に近傍になっており，その結果 SBOR 法にならない．また，Typewriter ordering 法も同様な理由から SBOR 法にはならない．

以上説明したように，近傍の非同時性が実現され，式（3）の形式の反復計算式を採用した並列処理方式は $0<\omega<2$ で必ず収束する．この近傍の非同時性を実現するためには，前述のような均等な要素分割方式については式（4）を満足するセル方式を採用し，セル内の同じ位置にある節点に関して同時に計算すればよい．しかし，実際問題に関してはこのような節点割当ておよび計算順序付けは不可能であり，より拡張した方式でなければならないが，これについては次章において説明する．

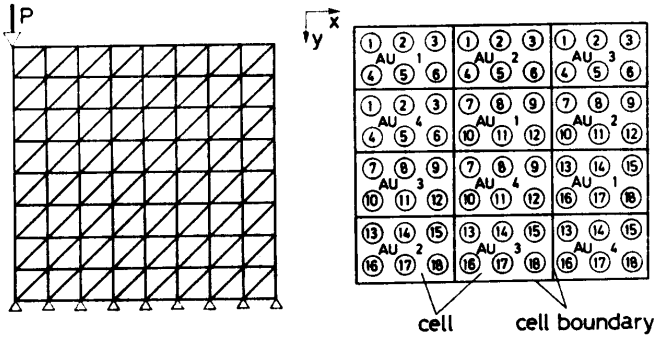

図 3 問題領域の要素分割，AU へのセルの割当ておよび計算順序（円内の数字）例

Fig. 3 Division of two-dimensional body into triangular elements, allocation of nodal points to AUs and the sequence of computations (numbers in circles).

3. 数値実験とその結果ならびに一般的指針

3.1 数値実験とその結果

前章において述べたようにセル方式は実際問題には適用できない．しかし，実用的な節点割当て方式について多くのことを示唆すると考えられるので，まずセル方式を取りあげる．セルの大きさは種々考えられるが，計算効率の観点から考えて反復過程を最もはやく収束させるものを選択することが望ましいので，ここで収束速さとセルの大小の関係について調べてみた

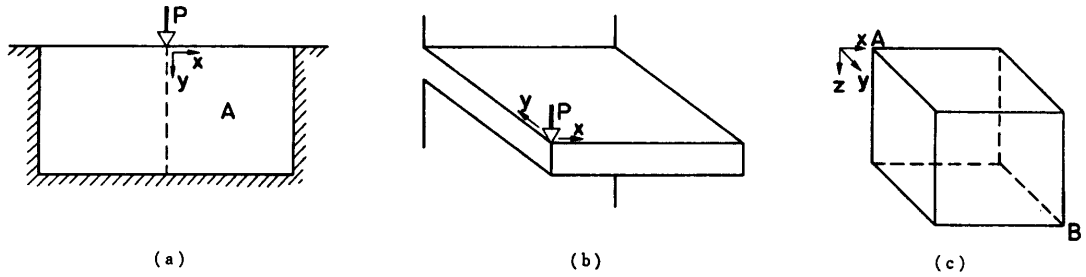

図 4 数値実験に用いた問題

Fig. 4 Problems subjected to numerical experiments.

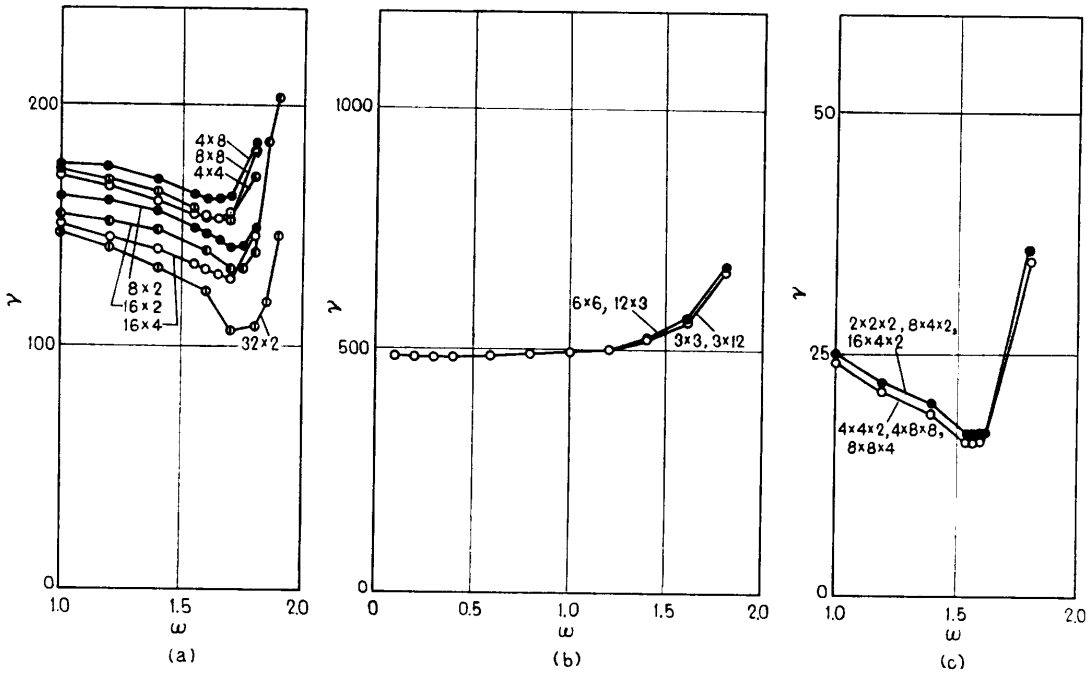

図 5 収束までの計算次数 ν と加速係数 ω の関係 ($\varepsilon<10^{-3}$)

Fig. 5 Relation between the number of iterations ν and relaxation factor ω ($\varepsilon<10^{-3}$).

い．しかし，その関係を理論的に調べることはむずかしい．なぜなら，収束速さは最適加速係数において比較すべきであるが，この最適加速係数を理論的に見出す方法を現在は見出せない[9]からである．それゆえ，収束速さとセルの大小の関係は数値実験の結果から推測せざるを得ない．このために**図4**に示すような問題に関して，有限要素法のみならず差分法による数値実験をも行った．(A)は剛体に埋め込まれた弾性体の三角形要素の有限要素法による変形解析，(B)は重調和方程式の 13 点差分による片持平板のたわみ解析，(C)は調和方程式の 7 点差分による固体内の温度解析である．境界条件は（B）では固定部にてたわみおよびこう配を0，(C)では温度をA点で1，B点で0，そのほかの面で断熱としている．要素分割により発生した節点数は（A）ではA領域で 32×32，（B）では 24×24，(C)では 16×16×16 である．反復計算の収束判定には次式で定義される相対誤差を用い，$\varepsilon<10^{-3}$ および $\varepsilon<10^{-4}$ としている．

$$\varepsilon=\max|x^{(\nu+1)}-x^{(\nu)}|/\max|x^{(\nu+1)}| \qquad (5)$$

使用計算機は FACOM M 160 で，計算は単精度で実行した．なお，本実験では $N=16$ としている．

数値実験結果を**図5**と**表1**に示す．図5はセルの大きさを変化させたときの収束までの計算次数 ν と加速係数 ω の関係を示す．表1はセルの大きさを変化させたときの，最適加速係数 ω_{opt} における収束までの計算次数 ν をまとめて示す．なお，ω_{opt} は $\varepsilon<10^{-3}$ の実験よりもとめたもので，$\nu(10^{-4})$ はこの ω_{opt} における値である．

表 1 数値実験結果

Table 1 Results of numerical experiments.

番号	問題	セルの大きさ	$\nu(10^{-3})$	$\nu(10^{-4})$	ω_{opt}
1	A	2×2	152	460	1.625
2		4×2	148	420	1.650
3		8×2	141	401	1.700
4		16×2	130	352	1.700
5		32×2	106	286	1.700
6		4×4	153	423	1.675
7		8×4	153	415	1.700
8		16×4	128	368	1.700
9		4×8	161	443	1.650
10		8×8	153	455	1.650
11		4×16	160	488	1.650
12		2×32	146	403	1.650
13	B	3×3	481	4,987	0.250
14		4×3	481	4,986	0.250
15		6×3	481	5,013	0.275
16		12×3	482	5,013	0.275
17		3×4	481	4,987	0.250
18		3×6	481	5,014	0.275
19		6×6	481	5,013	0.275
20		3×12	481	5,014	0.275
21	C	2×2×2	17	125	1.575
22		4×2×2	17	125	1.575
23		4×4×2	16	125	1.575
24		8×2×2	17	125	1.575
25		8×4×2	17	124	1.575
26		8×8×2	17	125	1.575
27		16×2×2	17	125	1.575
28		16×4×2	17	125	1.575
29		16×8×2	16	125	1.575
30		8×8×4	16	124	1.575
31		4×8×8	16	124	1.575

以上の数値実験から推測するに，セルの大小は反復過程の収束性の良否に決定的な差異は与えないように思われる．

3.2 一般的指針

前節の数値実験ではセルの形は長方形あるいは直方体であり，セル内の節点の位置と個数は全セルについて同じであった．しかし，実験問題については要素の大きさならびに節点のちらばりは一般に不規則であり，したがって問題領域をこのようなセル群で埋めつくすことは不可能である．それでは，どのような節点割当ての方式を採用すればよいのか．前節の数値実験の結果から，セルの大小は反復過程の収束速さにはあまり差異は与えないことが推測できるから，基本方針として，「全 AU につき分担節点数がほぼ均等になるように，形状は適当とし節点をグループ分けし，各 AU に割り当てる方式」を採用すればよい．次に，各 AU の分担節点間の計算順序付けであるが，全 AU 同時に計算している節点どうしが相互に近傍にならないように，AU 1から始めて順次 AUN までその分担節点に関して計算順序付けする．これにより近傍の非同時性は実現され，式（3）の形式の反復計算式を採用した並列処理過程は SBOR 法になり，ある程度の収束速さと $0<\omega<2$ における収束性は保証される．**図 6** に問題（A）について，式（4）に違反する 32×1 形セルの場合に，この指針にしたがって計算順序付けを行った一例を示す．なお，要素分割は図 3 のように分割している．この例では $\omega_{opt}=1.75$，$\nu(10^{-3})=111$，$\nu(10^{-4})=254$ であり，表 1 の結果に比較して勝るとも劣るものではなく，セル形状の似ている 32×2 形セルの結果にほぼ一致している．

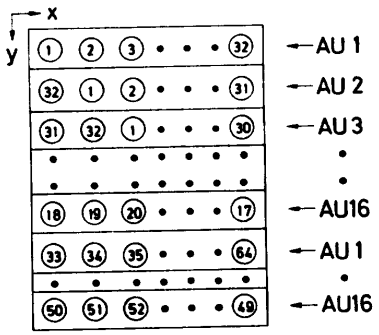

図 6 32×1 形セルの場合の計算順序例（問題(A)）

Fig. 6 An example of the sequence of computations in the case of 32×1 type cell (problem (A)).

以上において，一般的な指針を述べたが，実際問題でこれが本当に可能であるのかどうかに関しては，著者らの計画している並列処理計算機による大形有限要素解析問題の反復法に基づく並列処理方式に関連させて次の機会に報告したい．

4. 結 言

AU を複数有する並列処理計算機で，反復法に基づいて有限要素法あるいは差分法などによる数値解析を並列処理する際には問題領域の要素分割により発生した節点群の AU への割当て方式ならびに割り当てられた節点間の計算順序付けの方式をいかにすべきであるかが反復計算過程の収束速さのみならず前処理過程における逆行列ならびに行列の積の計算の必要性とも関連して大きな問題であるので，本論文ではこの問題を取りあげて，検討ならびに考察を加えた．

その結果，反復計算過程の収束性を保証し，かつかなりの計算量を必要とする逆行列ならびに行列の積の計算を省略し，計算の高効率化をはかるためには，近

傍の非同時性を実現することが必要であることを明らかにし，そのための節点割当ての方式と計算順序付けの方式について一般的指針を示した．

終りに，有益な助言をいただいた査読者に厚く謝意を表す．また，日頃有益な助言とご協力をいただいている本学大学院総合理工学研究科精密機械システム専攻林国一教授に厚く謝意を表す．

参考文献

1) 村岡洋一，坂間保雄：並列処理技術(技術展望)，電子通信学会誌，Vol. 63, No. 10, pp. 1064-1071 (1980)．
2) 金田悠紀夫：並列処理システムによる連立一次方程式と楕円形偏微分方程式の数値計算法，情報処理，Vol. 16, No. 2, pp. 122-129 (1975)，および，環状結合型超多重プロセッサシステムによる大次元連立一次方程式の並列計算，情報処理学会論文誌，Vol. 21, No. 5, pp. 402-406 (1980)．
3) Conrad, V. and Wallach, Y.: Iterative Solution of Linear Equations on A Parallel Processor System, IEEE Computer, Vol. C-26, No. 9, pp. 838-847 (1977)，および，On Block-Parallel Methods for Solving Linear Equations, IEEE Computer, Vol. C-29, No. 5, pp. 354-359 (1980)．
4) Anita, K.J. and Schwarz, P.: Experience Using Multiprocessor Systems-A Status Report, ACM Computing Surveys, Vol. 12, No. 2, pp. 121-165 (1980)．
5) Varga, R.S.: Matrix Iterative Analysis, p. 80, Prentice Hall, New Jersey (1962)．
6) 平野菅保，戸川隼人，藤井 宏，三好哲彦：計算技術および数値計算法，p. 180，培風館，東京 (1971)．
7) 戸川隼人：マトリックスの数値計算，オーム社，p. 86，東京 (1971)．
8) 星野 力：並列計算機のもたらす技術計算へのインパクト(解説)，日本原子力学会誌，Vol. 22, No. 4, pp. 204-212 (1980)．
9) 文献 7)，p. 71.

(昭和56年2月13日受付)
(昭和56年5月20日採録)