数理解析研究所講究録
第548巻 1985年 147-168

移流拡散方程式に対するベクトル計算機向きPCG法

日立ソフト　後 保範（Yasunori Ushiro）

1. はじめに

移流項を含まない拡散方程式を離散化すると対称疎行列が得られる。移流項を含む拡散方程式を離散化すると非対称疎行列が得られる。対称疎行列を係数とする連立一次方程式の反復解法としてICCG法[1]（Incomplete Cholesky Conjugate Gradient Method）が注目されてきている。スカラ用ICCG法と収束が同一なベクトル計算機向きICCG法[2]は規則的及び不規則的な疎行列のいずれもリストベクトルを使用して作成できることが分っている。非対称疎行列を係数とする連立一次方程式の反復解法としてPCG法（Preconditioned Conjugate Gradient Method）の一種であるILUCR法[4]（Incomplete LU decomposition Conjugate Residual Method）及びILUBCG法[3]（Incomplete LU decomposition BiConjugate Gradient Method）が注目されて来ている。ここではスカラ用ILUCR法及びILUBCG法と収束が同一なベクトル計算機

向き計算方法について報告する。さらにILUCR法及びILUBCG法を改良したMILUCR法(Modified[5] ILUCR法)及びMILUBCG法(Modified[5] ILUBCG法)についても報告する。

数値実験は三次元直方体領域における移流拡散方程式を7点中央差分公式及び風上差分公式で離散化した行列を対象に行った。ベクトル計算機としてHITAC S-810を使用した。

今回提案したベクトル計算機向きILUCR法及びILUBCG法は有限要素法による離散化などで多く発生する不規則的疎行列にも適用できる。ここではそのうちのILUCR法について報告する。

2. 規則的疎行列における計算方法

2.1 ILUCR法とILUBCG法の計算方法

ILUCR法(*Incomplete LU decomposition Conjugate Residual Method*)の計算手順を図2.1に示す。CR法(*Conjugate Residual Method*)は方向ベクトルP_iが元の行列Aに対して，k個前までのP_{i-k}とA^TA直交させる方法がありCR(k)と呼ばれる。ここで示したILUCR法はkを1としたCR(1)に基づいて作成したものである。図2.1に示したILUCR法は$\widetilde{A}=[LU]^{-1}A$と置くと下記のような基本性質を持つ。

(a) $(\widetilde{A}P_i, \widetilde{A}P_{i-1})=0$

(b) $(r_i, \tilde{A}p_{i-1}) = 0$

(c) $(r_i, \hat{A}p_i) = (r_i, \hat{A}r_i)$

(d) $(r_i, \hat{A}r_i) = 0$

ILUBCG法（Incomplete LU decomposition BiConjugate Gradient Method）の計算手順を図2.2に示す。ILUBCG法は下記のような基本的性質を持つ。

(a) $(r_i, r_j^*) = 0 \quad \text{for } i \neq j$

(b) $(Ap_i, p_j^*) = 0 \quad \text{for } i \neq j$

行列Aを不完全LDU分解する

$r = [LDU]^{-1}(b - Ax)$

$p = r, \quad q = [LDU]^{-1}Ar$

収束まで以下を反復する。

$\mu = (q, q)$

$\alpha = (r, q)/\mu$

$x = x + \alpha p, \quad r = r - \alpha q$

$v = [LDU]^{-1}Ar$

$\beta = (v, q)/\mu$

$p = r - \beta p, \quad q = v - \beta q$

(注)
(r, q)はベクトルの内積を示す。

図2.1 ILUCR法の計算手順

行列Aを不完全LDU分解する。

$\mathbf{r} = \mathbf{r}^* = [LDU]^{-1}(\mathbf{b} - A\mathbf{x})$

$\mathbf{p} = \mathbf{p}^* = \mathbf{r}$, $\mu_1 = (\mathbf{r}, \mathbf{r}^*)$

収束まで以下を反復する

$\mathbf{q}^* = A^T[LDU]^{-T}\mathbf{p}^*$

$\mathbf{q} = [LDU]^{-1}A\mathbf{p}$

$\alpha = \mu_1/(\mathbf{q}, \mathbf{p}^*)$

$\mathbf{x} = \mathbf{x} + \alpha\mathbf{p}$

$\mathbf{r} = \mathbf{r} - \alpha\mathbf{q}$, $\mathbf{r}^* = \mathbf{r} - \alpha\mathbf{q}^*$

$\mu_2 = \mu_1$, $\mu_1 = (\mathbf{r}, \mathbf{r}^*)$

$\beta = \mu_1/\mu_2$

$\mathbf{p} = \mathbf{r} + \beta\mathbf{p}$, $\mathbf{p}^* = \mathbf{r}^* + \beta\mathbf{p}^*$

図2.2 ILUBCG法の計算手順

図2.1及び図2.2はいずれも非対称な連立一次方程式$A\mathbf{x} = \mathbf{b}$の解$\mathbf{x}$を求めるものである。行列Aが規則的疎行列であろうと不規則的疎行列であろうと図2.1及び図2.2の手順は同じである。異なるのは不完全三角分解，$A\mathbf{P}$の計算及び$[LDU]^{-1}\mathbf{P}$などの具体的計算である。

規則的疎行列の例として三次元直方体領域における移流拡散方程式を7点差分公式で離散化した行列を使用する。

この場合の行列Aの非ゼロ要素の形を図2.3に示す。不完全三角分解は元の行列Aの非ゼロ要素より非ゼロ要素の数を増加させないこと及び対角行列Dを使用してLDUの形に分解する方針とする。この方針で不完全三角分解すると分解後の下三角行列Lは元の行列Aの下三角部分と対角要素を除き値まで一致する。上三角行列Uは行列Aの上三角部分と対角要素を除き一致する。図2.4に不完全三角分解後の下三角行列Lの形を，図2.5に対角行列Dの形を，図2.6に上三角行列Uの形を示す。

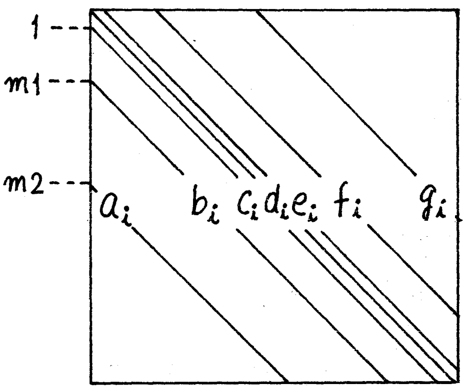


図2.3 行列Aの形

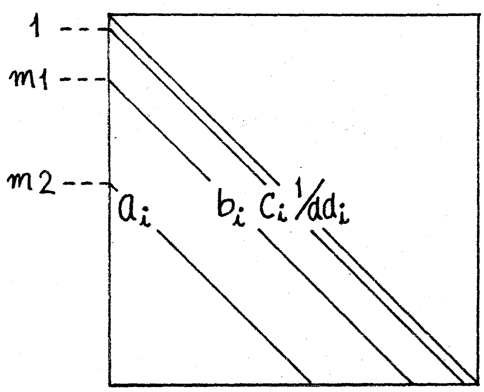


図2.4 分解後の行列Lの形

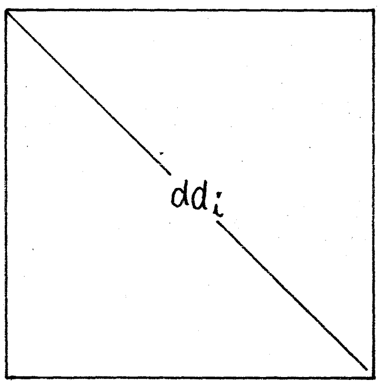


図2.5 分解後の行列Dの形

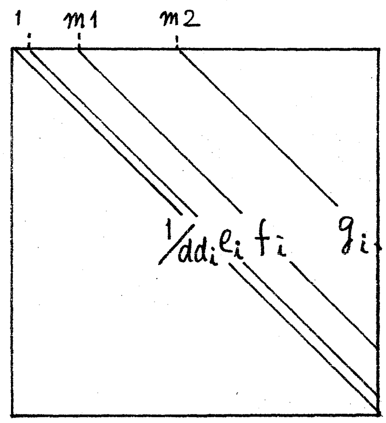


図2.6 分解後の行列Uの形

次に不完全三角分解と$[LDU]^{-1}\mathbf{r}$及び$[LDU]^{-T}\mathbf{r}$の計算式を示す。不完全三角分解，$[LDU]^{-1}\mathbf{r}$の計算はILUCR法とILUBCG法に共通である。$[LDU]^{-T}\mathbf{r}$の計算はILUBCG法にだけ使用される。次元数はnとする。

(1) 不完全三角分解

対角行列の要素dd_i ($i=1,2,\cdots,n$) だけ計算すればよい。

$$dd_i = \frac{1}{d_i - c_i \times e_{i-1} \times dd_{i-1} - b_i \times f_{i-m_1} \times dd_{i-m_1} - a_i \times g_{i-m_2} \times dd_{i-m_2}} \qquad (2.1)$$

$$(i=1,2,\cdots,n)$$

(2) $\mathbf{q} = [LDU]^{-1}\mathbf{r}$の計算

$L\mathbf{z} = \mathbf{r}$, $DU\mathbf{q} = \mathbf{z}$と置いて，$\mathbf{z}$を経由し$\mathbf{q}$を求める。

$$z_i = dd_i \times (r_i - c_i \times z_{i-1} - b_i \times x_{i-m_1} - a_i \times x_{i-m_2}) \qquad (2.2)$$

$$(i=1,2,\cdots,n)$$

$$q_i = z_i - dd_i \times (e_i \times q_{i+1} + f_i \times q_{i+m_1} + g_i \times q_{i+m_2}) \qquad (2.3)$$

$$(i=n,n-1,\cdots,1)$$

(3) $\mathbf{q} = [LDU]^{-T}\mathbf{r}$の計算

$U^TD^T\mathbf{z} = \mathbf{r}$, $L^T\mathbf{q} = \mathbf{z}$と置いて，$\mathbf{z}$を経由して$\mathbf{q}$を求める。

$$z_i = r_i - e_{i-1} \times dd_{i-1} \times z_{i-1} - f_{i-m_1} \times dd_{i-m_1} \times z_{i-m_1} - g_{i-m_2} \times dd_{i-m_2} \times z_{i-m_2} \qquad (2.4)$$

$$(i=1,2,\cdots,n)$$

$$q_i = dd_i \times (z_i - c_{i+1} \times q_{i+1} - b_{i+m_1} \times q_{i+m_1} - a_{i+m_1} \times q_{i+m_2}) \qquad (2.5)$$

$$(i=n,n-1,\cdots,1)$$

2.2 MILUCR法とMILUBCG法の計算方法

ILUCR法及びILUBCG法を改良したMILUCG法(Modified ILUCR法)及びMILUBCG法(Modified ILUBCG法)の計算手順も同じく図2.1及び図2.2に従う．異なるのは不完全三角分解の方法だけである。不完全三角分解はMILUCR法もMILUBCG法も同一である。

不完全三角分解の補正方法として次の二つを考えた。

(1) 方式1 ---加速係数 α

(2.1)式の不完全三角分解を次のように変更する。

$$\left.\begin{aligned} w_i &= d_i - c_i \times dd_{i-1} \times \{ e_{i-1} + \alpha \times \underline{(f_{i-1} + g_{i-1})} \} \\ &\quad - b_i \times dd_{i-m_1} \times \{ f_{i-m_1} + \alpha \times \underline{(g_{i-m_1} + e_{i-m_1})} \} \\ &\quad - a_i \times dd_{i-m_2} \times \{ g_{i-m_2} + \alpha \times \underline{(e_{i-m_2} + f_{i-m_2})} \} \\ dd_i &= 1/w_i \end{aligned}\right\} \quad (2.6)$$

(2) 方式2 ---加速係数 σ

(2.1)式の不完全三角分解を次のように変更する．

$$\left.\begin{aligned} w_i &= (1+\sigma) \times d_i - c_i \times dd_{i-1} \times (e_{i-1} + \underline{f_{i-1} + G_{i-1}}) \\ &\quad - b_i \times dd_{i-m_1} \times (f_{i-m_1} + \underline{g_{i-m_1} + e_{i-m_1}}) \\ &\quad - a_i \times dd_{i-m_2} \times (g_{i-m_2} + \underline{e_{i-m_2} + f_{i-m_2}}) \\ dd_i &= 1/w_i \end{aligned}\right\} \quad (2.7)$$

方式2はGusstafsson流[5]の補正をほどこしたものである。これは〜〜〜のものを追加しただけでは，もとの行列がせっかく

M行列であっても，不完全三角分解の安定性や正則分離性が保証されないから，対角項 d_i に $(1+\sigma)$，$\sigma>0$ を掛けることによって補償するのである。方式1は同じ目的で対角項を増加する代りに追加する項に1以下の定数 α を掛けるものである。

今回の数値実験では方式1を採用した。さらに汎用化を考えて $\alpha=0.95, 0.9, 0.75, 0.5, 0.0, -1.0$ を用意しておきその中の一つを自動選定する方法を採用した。選定はまず0.95から順に始めて，$i=1,2,\cdots,n$ のすべてで $w_i > 0.1\times d_i$ となればそれを加速係数とした。MILUCR法とMILUBCG法の加速係数の選定方法は対称疎行列に対するMICCG法（Modified Incomplete Cholesky Conjugate Gradient Method）のように一定値に固定することができず微妙なところがある。本件については村田教授の「CR(k)法の加速について」[6]等を参考にされたい。

2.3 今回提案のベクトル計算機向き方式

CR法及びBCG法はベクトル計算機向き解法であるが，不完全三角分解と組み合わせたILUCR法，ILUBCG法，MILUCR法及びMILUBCG法はそのままではベクトル計算に不適になる。

ILUCR法及びMILUCR法では通常下記二つの計算がベクトル計算に不適となる。

(a)　行列の不完全三角分解。（LDUと表す）

(b)　不完全三角分解行列を使用した前進及び後退代入。

($q = [LDU]^{-1} r$ と表す)

さらにILUBCG法及びMILUBCG法では一般に次の計算がベクトル計算不適となる。

(C) 不完全三角分解の転置行列を使用した前進及び後退代入

($q = [LDU]^{-T} r$ と表す)

上記三つの計算は(2.1)式から(2.7)式で示すように、一つ前で計算した値を使用するというデータの依存関係があり、ベクトル計算に適さない。即ち三次元の移流拡散方程式を差分法で離散化した行列では図2.7のような関係がある。ここで nx, ny, nz はそれぞれ x, y, z 方向の分割数で、n は行列の次元数で $n=nx \times ny \times nz$ とする。未知ベクトル q は三次元表示で $q(i, j, k)$ と表す。

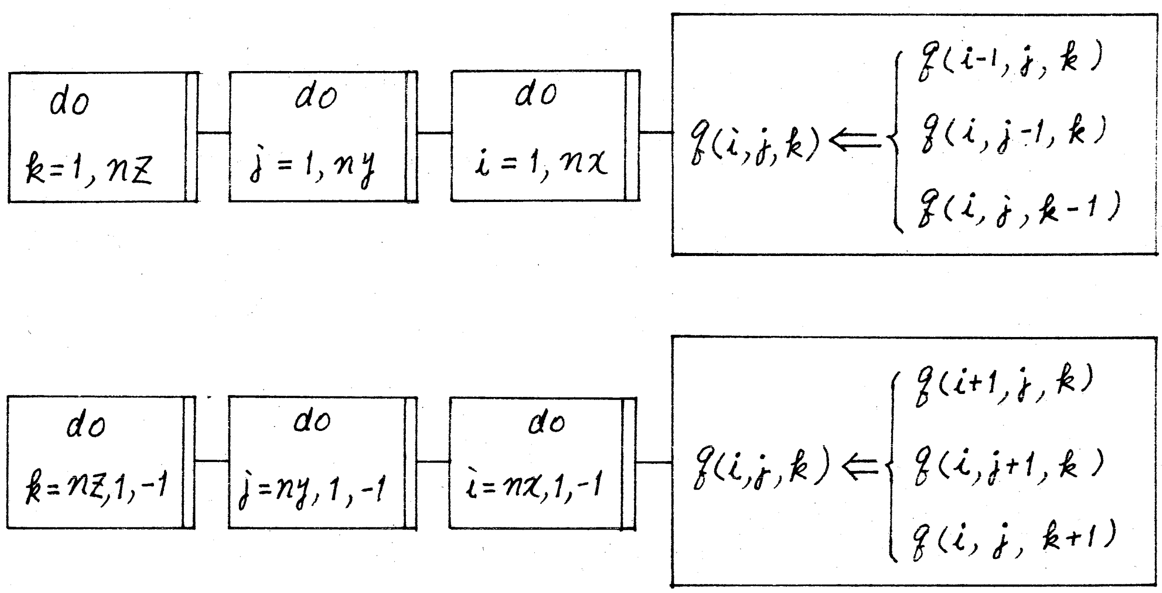


図2.7　項番(a),(b),(C)の計算におけるデータの依存関係

しかし，$i+j+k$ の値が3から $nx+ny+nz$ まで又はその逆に $nx+ny+nz$ から3まで，$i+j+k$ の値が一定となるものをまとめて計算する方法を採用すると次のことが分かる。 また，$i+j+k$ の値が一定のものをまとめて計算するにはリストベクトルが有効となる。

(a) $g(i,j,k)$ を計算するための $g(i-1,j,k)$，$g(i,j-1,k)$ 及び $g(i,j,k-1)$ は既に計算済みの値となりデータの依存関係がなくなる。このためベクトル計算に適する。

(b) $g(i-1,j,k)$，$g(i,j-1,k)$ 及び $g(i,j,k-1)$ の値の計算が完了後 $g(i,j,k)$ を計算するという関係は保持しているため反復回数当りの収束速度は変化しない。

図2.8に(2.6)式で示す不完全三角分解の計算を本方法でベクトル計算機向きに変えた場合の計算方法を示す。図2.9にILUCR法，ILUBCG法，MILUCR法及びMILUBCG法に共通なベクトル計算機向き $g=[LDU]^{-1}r$ の計算方法を示す。

図2.8と図2.9に使用するリストベクトルは三次元問題で $nx=ny=nz=3$ とした場合の具体例を示すと下記のようになる。このとき $m1=3$, $m2=9$, $n=27$ である。

$n0+1$

$n0=nx+ny+nz-2=7$

LN;

1	2	3	4	5	6	7	8
0	1	4	10	17	23	26	27

L;

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	2	4	10	3	5	7	11	13	19	6	8	12	14	16	20	22	9	15

do $j = 1, no$

do $k = LN(j)+1, LN(j+1)$

$i = L(k)$

$w_1 = c(i)*dd(i-1)*\{e(i-1)+\alpha*[f(i-1)+g(i-1)]\}$

$w_2 = b(i)*dd(i-m1)*\{f(i-m1)+\alpha*[e(i-m1)+g(i-m1)]\}$

$w_3 = a(i)*dd(i-m2)*\{g(i-m2)+\alpha*[f(i-m2)+e(i-m2)]\}$

$dd(i) = 1.0/[d(i)-w_1-w_2-w_3]$

図2.8 ベクトル計算機向き不完全三角分解

do $j = 1, no$

do $k = LN(j)+1, LN(j+1)$

$i = L(k)$

$q(i) = dd(i)*[r(i) - c(i)*q(i-1)$

$\qquad - b(i)*q(i-m1) - a(i)*q(i-m2)]$

do $j = no, 1, -1$

do $k = LN(j)+1, LN(j+1)$

$i = L(K)$

$q(i) = q(i) - dd(i)*[e(i)*q(i+1)$

$\qquad + f(i)*q(i+m1) + g(i)*q(i+m2)]$

図2.9 ベクトル計算機向き $q = [LDU]^{-1}\, \mathbf{r}$ の計算

3. 規則的疎行列における数値実験結果

ここでの数値実験の目的は今回提案したベクトル計算機用解法について次の二点を確認することである。

(a) ILUCR法, MILUCR法, ILUBCG法及びMILUBCG法のすべてにおいて反復回数当りの収束速度はスカラ用に作成されたものと同一である。

(b) スカラの処理速度がほぼ同一の汎用計算機HITAC M-280Hに比較して, 今回提案したプログラムはベクトル計算機HITAC S-810で何倍高速化できるか。

項番(a)については, すべての数値実験結果でベクトル用に改良したものとスカラ用のもので一致した。

計算速度の比較には(3.1)式に示す三次元移流拡散方程式を7点中央差分公式及び風上差分公式で離散化した行列を使用した。計算対象領域は直方体領域で, 表三面は0に固定した境界条件(Γ_1)で, 裏三面は拡散による移動のない境界(Γ_2)とする。流れはx軸に平行な流れだけあり, その速度は$y=0$の面でv_0とし, 一般に$v_0(1-(y/y_{max})^5)$とした。

$$-div(k \cdot grad\phi) + \boldsymbol{v} \cdot grad\phi = f \qquad (3.1)$$

$$\left.\begin{array}{ll} \phi = 0 & on\ \Gamma_1\ (表三面) \\ (k \cdot grad\phi) \cdot \boldsymbol{n} = 0 & on\ \Gamma_2\ (裏三面) \end{array}\right\} 境界条件 \qquad (3.2)$$

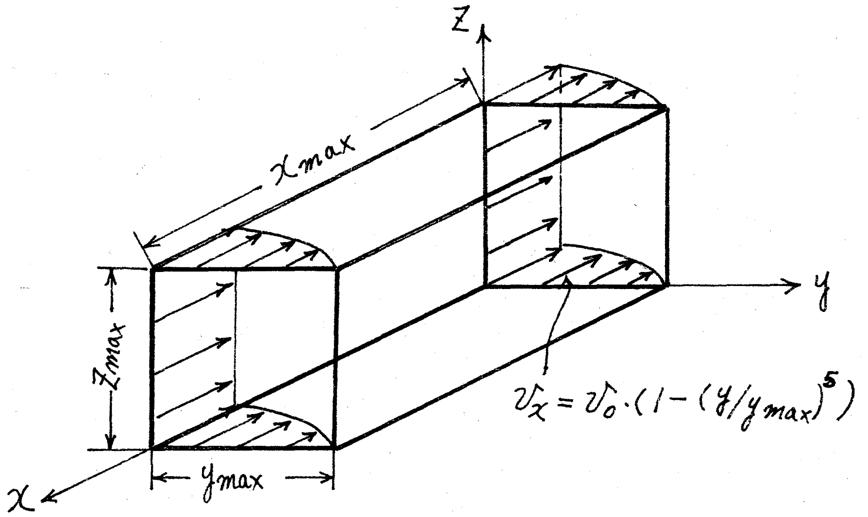


図 3.1　数値実験モデル

数値実験モデルを図3.1に示す。今回の数値実験では $x_{max}=5$, $y_{max}=z_{max}=2$, $K=1.0$ とした。

数値実験は 40×20×20 に等分割し，中央差分公式又は風上差分公式を使用して 16000次元の次のような連立一次方程式を作成する。v_0 は 0, 0.1, 1, 10, 100 とした。

$$A\mathbf{x} = \mathbf{b} \tag{3.3}$$

未知数となる節点の番号は x 方向，y 方向，z 方向の順に付けた。$\mathbf{x}^{\nu}$ を ν 回反復計算時の解ベクトルとするとき解の収束を判定する基準として次式で表現される二乗ノルムの相対残差を使用した。

$$\text{相対残差} = \frac{\|A\mathbf{x}^{\nu} - \mathbf{b}\|_2}{\|\mathbf{b}\|_2} \tag{3.4}$$

反復計算の初期値は $(0,0,\cdots,0)^T$ なるゼロベクトルとした。相対残差 10^{-8} 以下になるまでの ILUCR法，MILUCR法，ILUBCG法

及び MILUBCG法の反復回数を図3.2及び図3.3に示す。図3.2は風上差分で離散化したものであり，図3.3は中央差分で離散化したものである。MILUCR法及びMILUBCG法の不完全分解は図2.8に示す計算式で行った。このとき加速係数αは6個の値 0.95, 0.9, 0.75, 0.5, 0.0及び−1.0を用意しておき，その中の一つを自動選定する方式を採用した。選定はまず0.95から順に始めて $i=1, 2, \cdots, n$ のすべてで次の式が成立すれば加速係数αとして採用した。

$$d(i)-w_1-w_2-w_3 \geq d(i)\times\varepsilon \qquad (3.5)$$

図3.2及び図3.3に示すものは $\varepsilon=0.1$ としたものである。

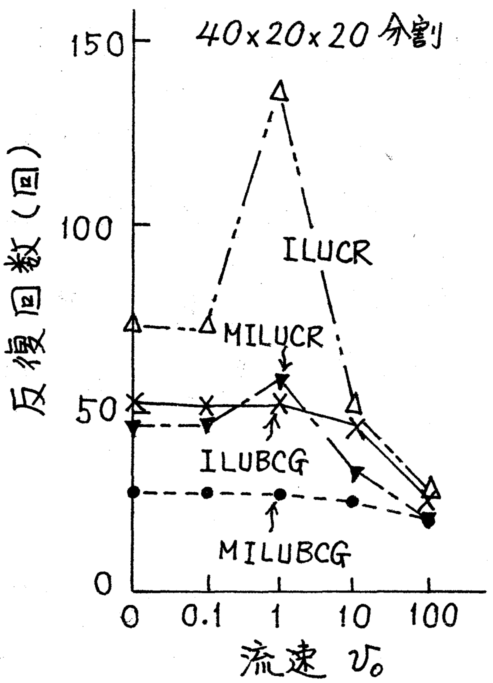


図3.2 風上差分での反復回数

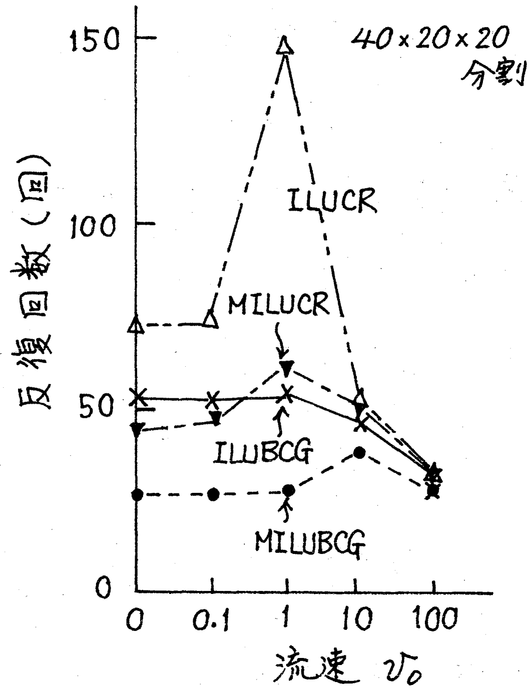


図3.3 中央差分での反復回数

MILUCR法とMILUBCG法の加速係数は同一である。風上差分の場合は流速v_0の変化にかかわらず一定で$\alpha=0.95$となった。図3.3の中央差分の場合はv_0が0, 0.1及び1のとき$\alpha=0.95$で，$v=10$のときは第3番目の値$\alpha=0.75$，$v=100$のときは第4番目の値$\alpha=0.0$を選定した。図3.2と図3.3を比較すると，風上差分と中央差分であまり差がないように見えるが実際は風上差分のとき収束は安定で，中央差分のとき不安定となる場合が多い。この例でも(3.5)式のεを10^{-2}とすると，中央差分で$v_0=10$のときαは0.90を選定し，MILUCRは収束しないという結果になった。これはスカラ用の計算方式でも同じである。

図3.4及び図3.5にHITAC M-280Hでの計算時間を示す。図3.4は風上差分で離散化したもので図3.2に対応するものである。図3.5は中央差分で離散化したもので図3.3に対応するものである。同じ計算をベクトル計算機HITAC S-810で実行したものを図3.6及び図3.7に示す。図3.6は風上差分で離散化したもの，図3.7は中央差分で離散化したものである。

各解法の特性をここで論じることはしない。図3.4，図3.5と図3.6，図3.7を比較することにより次のことが分かる。

40×20×20分割においてS-810のM-280Hに対する速度向上比率は

ILUCR及びMILUCR --- 約50倍

ILUBCG及びMILUBCG --- 約40倍 である。

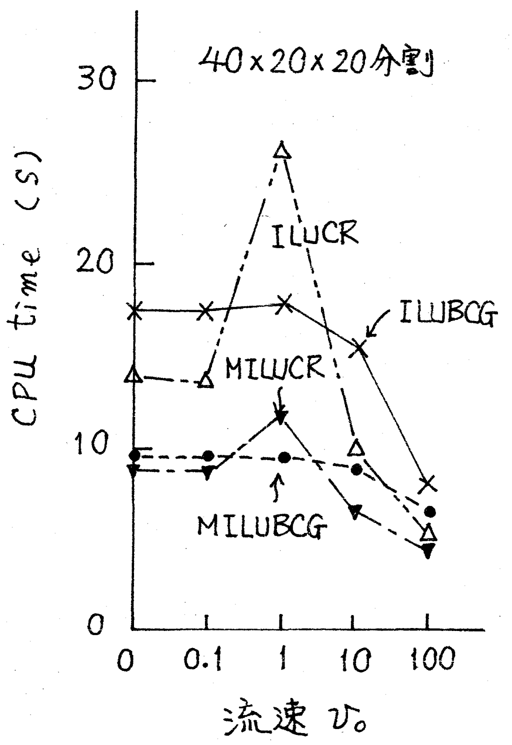


図3.4 M-280Hでの計算時間(風上差分)

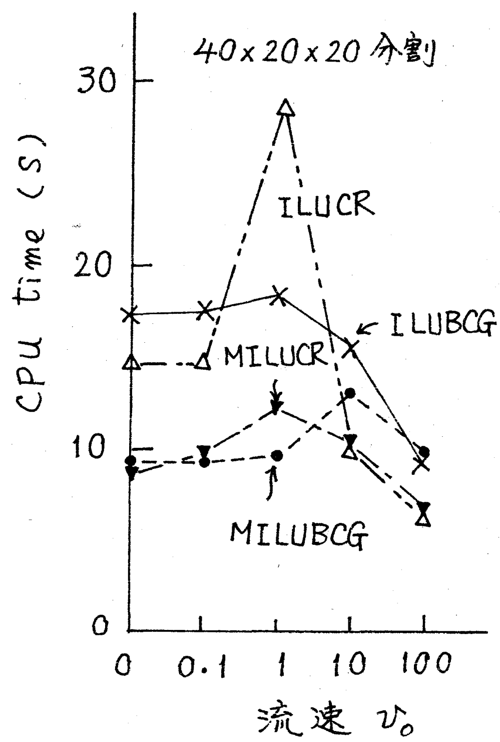


図3.5 M-280Hでの計算時間(中央差分)

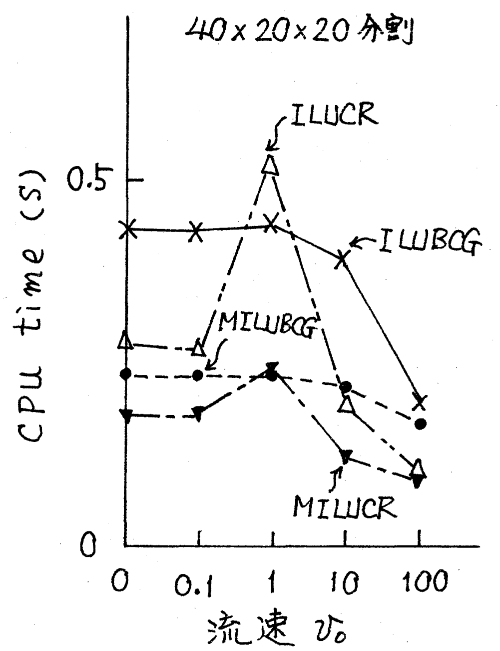


図3.6 S-810での計算時間(風上差分)

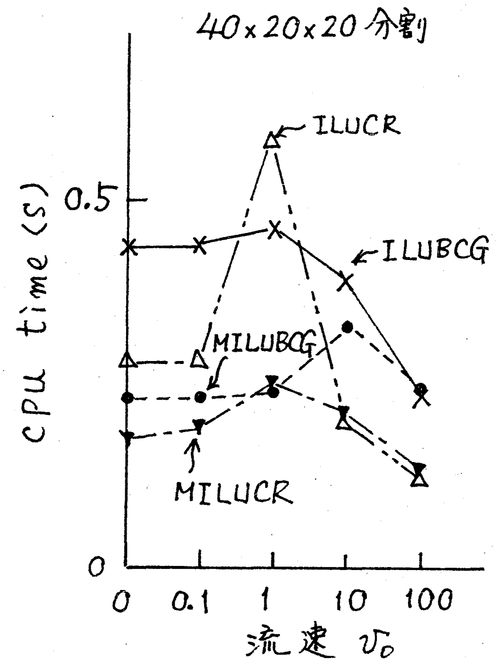


図3.7 S-810での計算時間(中央差分)

4. 不規則的疎行列のILUCR法

汎用計算機向きILUCR法と収束が同一なベクトル計算機向き計算方法を示す。不完全LDU分解における行列LとUの非ゼロ要素の位置は元の行列の下三角部分及び上三角部分の非ゼロ要素の位置と同じものとする。ILUCR法のベクトル化においては不完全LDU分解と$[LDU]^{-1}r$の処理が問題となる。一般に行列L及びUの一行当りの非ゼロ要素数は少いため，一行当りの非ゼロ要素の計算をベクトル化しただけでは効果が小い。不規則的疎行列のILUCR法は偏微分方程式を有限要素法で離散化した場合に発生する連立一次方程式の解を求めるのが主目的である。この場合は未知数となる節点の番号付けを変更することにより，一定の範囲内でデータの独立性を確保することができ，ベクトル計算機で高速処理することができる。その原理は不規則的対称疎行列を係数とするICCG法と同一である。ICCG法の場合は既に報告[2)]しているが，ここではILUCR法の立場から再度同じ原理を説明する。

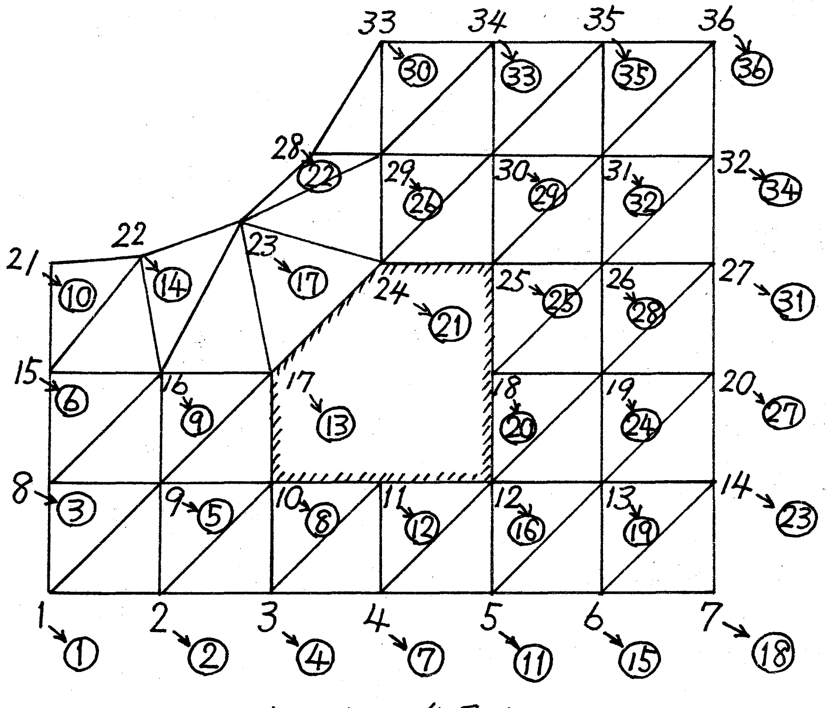
図4.1に不規則的疎行列を発生させる有限要素分割と未知数となる節点の番号付けを示す。オリジナル番号付けからベクトル計算機向き番号付けへの変換は，プログラムで次の条件のもとで実行する。

(a) 直接結合する節点の番号の大小関係はオリジナルな番

号とベクトル計算機向き番号で一致させる。

(b) 連続する番号の節点ができるだけ直接結合しないようにベクトル計算機向き番号を付ける。

項番(a)はILUCR法の収束率をスカラ計算の場合と同一に保つための条件であり，(b)はベクトル長をできるだけ長くするための条件である。



注： 1, 2, 3 --- オリジナル番号付け

①, ②, ③ --- ベクトル計算機向き番号付け

図4.1 有限要素分割と節点の番号付け

図4.2にオリジナルの番号付けで作成した疎行列の非ゼロ要素の列番号テーブルを示す。図4.3にベクトル計算機向きに作成した疎行列の非ゼロ要素の列番号テーブルを示す。

下三角(LL)			行番号	上三角(LU)		
			1	2	8	9
1			2	3	9	10
2			3	4	10	11
3			4	5	11	12
4			5	6	12	13
5			6	7	13	14
6			7	14		
1			8	9	15	16
1	2	8	9	10	16	17
2	3	9	10	11	17	
3	4	10	11	12		
4	5	11	12	13	18	19
5	6	12	13	14	19	20
6	7	13	14	20		
8			15	16	21	22
8	9	15	16	17	22	23
9	10	16	17	23	24	
12			18	19	25	26
12	13	18	19	20	26	27
13	14	19	20	27		
15			21	22		
15	16	21	22	23		
16	17	22	23	24	28	29
17	23		24	25	29	30
18	24		25	26	30	31
18	19	25	26	27	31	32
19	20	26	27	32		
23			28	29	33	
23	24	28	29	30	33	34
24	25	29	30	31	34	35
25	26	30	31	32	35	36
26	27	31	32	36		
28	29		33	34		
29	30	33	34	35		
30	31	34	35	36		
31	32	35	36			

図4.2 オリジナル番号付け
非ゼロ要素列番号テーブル

下三角(LL)			ブロック番号	行番号	上三角(LU)		
			①	1	2	3	5
1			②	2	4	5	8
1				3	5	6	9
2			③	4	7	8	12
1	2	3		5	8	9	13
3				6	9	10	14
4			④	7	11	12	16
2	4	5		8	12	13	
3	5	6		9	13	14	17
6				10	14		
7			⑤	11	15	16	19
4	7	8		12	16		
5	8	9		13	17	21	
6	9	10		14	17		
11			⑥	15	18	19	23
7	11	12		16	19	20	24
9	13	14		17	21	22	26
15			⑦	18	23		
11	15	16		19	23	24	27
16				20	24	25	28
13	17			21	25	26	29
17				22	26	30	
15	18	19	⑧	23	27		
16	19	20		24	27	28	31
20	21			25	28	29	32
17	21	22		26	29	30	33
19	23	24	⑨	27	31		
20	24	25		28	31	32	34
21	25	26		29	32	33	35
22	26			30	33		
24	27	28	⑩	31	34		
25	28	29		32	34	35	36
26	29	30		33	35		
28	31	32	⑪	34	36		
29	32	33		35	36		
32	34	35	⑫	36			

図4.3 ベクトル計算機向き番号付け
非ゼロ要素列番号テーブル

図4.3のブロック番号⑦を見ると，下三角テーブルLLは11から17の番号で構成されていることが分かる。すなわち，ブロック⑦の行番号に対応する18番から22番の未知数の計算は既に計算が完了している11番から17番の値を使用して実行できる。この関係を利用して，不規則的非対称疎行列の不完全LDU分解と$[LDU]^{-1}\mathbf{r}$の計算がベクトル化できる。

図4.4に$[LDU]^{-1}\mathbf{r}$をベクトル計算する場合の計算方法を示す。ここでALは下三角テーブルLLに，AUはLUに対応して非ゼロ要素の値が入り，DDには分解後の対角要素が入る。またLN(i+1)はi番ブロックの最後の行番号を記憶させたものであり，NOはブロックの数である。行列LとUの一行当りの非ゼロ要素の最大数は共に3として示した。

```
do J = 1, NO
    IS = LN(J)+1 ;  IE = LN(J+1)
    do I = IS, IE
        q(I) = DD(I)*(R(I)-AL(I,1)*q(LL(I,1))
                          -AL(I,2)*q(LL(I,2))
                          -AL(I,3)*q(LL(I,3)))
do J = NO, 1, -1
    IS = LN(J)+1 ;  IE = LN(J+1)
    do I = IS, IE
        q(I) = q(I)-DD(I)*(AU(I,1)*q(LU(I,1))
                          +AU(I,2)*q(LU(I,2))
                          +AU(I,3)*q(LU(I,3)))
```

図4.4　$q = [LDU]^{-1}\cdot\mathbf{r}$ のベクトル計算の例

オリジナル番号付けからベクトル計算機向き番号付けへの変換方法は「ベクトル計算機向きICCG法[2)]」に記述してあるのと同一なのでここでは省略する。

ここで工夫した不規則的スパース行列のILUCR法をHITAC M-280H（スカラで計算）とHITAC S-810モデル20（ベクトルで計算）で測定した。その結果反復回数当りの収束は同一であり，S-810はM-280Hに比較して約30倍高速になった。測定に使用した方程式は移流拡散方程式で，有限要素法で離散化した次元数5000の連立一次方程式を使用した。ILUCR法の反復回数は100回で，行列の一行当りの非ゼロ要素数は平均約7である。

5. おわりに

汎用計算機向きILUCR法及びILUBCG法と収束が同一なベクトル計算機向きILUCR法及びILUBCG法について述べた。その原理は対称行列用のICCG法と同じである。ベクトル計算機向き方法では，40×20×20分割した三次元問題でベクトル計算機HITAC S-810モデル20を使用するとHITAC M-280HのILUCR法では約50倍，ILUBCG法では約40倍高速化できた。有限要素法で離散化した5000次元の不規則的疎行列に適用したILUCR法で約30倍高速化できた。

終りに御指導いただいた図書館情報大学村田健郎教授に感謝します。

参考文献

1) J.A.Meijerink, H.A.Van der Vorst; Iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix, Math.Comp. Vol 31, PP.148～162 (1977)

2) 後 保範; ベクトル計算機向きICCG法, 本講究録 514, PP. 110～134 (1984)

3) Fletcher; Conjugate gradient methods for indefinite systems, Proc. of the Dundee Biennial Conf. on Num. Anal. Springer-Verlag (1975)

4) Y. Saad; The Lanczos biorthogonalization Algorithm and other oblique Projection methods for solving large unsymmetric systems, SIAM J. Num. Anal., Vol. 19, PP. 485～506 (1982)

5) I.Gustafsson; A class of first order factorization methods, BIT. 18, PP. 142～156 (1978)

6) 村田 健郎; CR(k) 法の加速について, 本講究録 514, PP. 92～109 (1984)