完全軸選択による行列の LU 分解の応用
– ベクトル列の加速法 –

2001MM066 太田 裕文
指導教員　鳥居 達生

1　はじめに

完全軸選択の LU 分解を用いてベクトル列の加速を行う研究をする。

数列の加速は $Aitken$ 加速がよく知られている。また、ベクトル列の加速法については最小二乗法に基づく $Aitken$ 加速の拡張が知られている。この研究ではベクトル列の加速を一般的に用いられる最小 2 乗法を使わずに、完全軸選択の LU 分解を用いて行い、その有効性を検証する [1][3]。

2　完全軸選択を用いた LU 分解

2.1　完全軸選択（完全ピボット選択法）

m 行 n 列の行列 A を完全軸選択を用いると、行列

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1k} & \cdots & a_{1q} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2k} & \cdots & a_{2q} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kk} & \cdots & a_{kq} & \cdots & a_{kn} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ a_{p1} & a_{p2} & \cdots & a_{pk} & \cdots & a_{pq} & \cdots & a_{pn} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mk} & \cdots & a_{mq} & \cdots & a_{mn} \end{pmatrix}$$

の全体から絶対値最大の要素を探す。
$\max_{k\le i,j\le n}|a_{ij}^{(k-1)}| = |a_{pq}^{(k-1)}|$ をピボットとして、A の第 k 行と第 p 行、第 k 列と第 q 列を入れ替える。完全軸選択では、列の交換も行うので、(q,k) に対応する置換行列 $Q^{(k-1)}(q = k$ の時は $Q^{(k-1)} = I)$ を記憶しておく。

$k-1$ 段から k 段への移行の算法。

$$\begin{aligned} a_{ij}^{(k)} &= a_{ij}^{(k-1)} + m_i^{(k)} a_{kj}^{(k-1)} \\ b_i^{(k)} &= b_i^{(k-1)} + m_i^{(k)} b_k^{(k-1)} \\ m_i^{(k)} &= -a_{ik}^{(k-1)}/a_{kk}^{(k-1)} \qquad i = k+1, \cdots, n \end{aligned}$$

$a_{kk}^{(k-1)} \neq 0$ でも絶対値がかなり小ならば、$m_i^{(k)}$ の計算にオーバーフローを生ずる危険がある。たとえそれが生じなくても、小さい数での割り算の結果、精度は一般に悪くなる [3]。

2.2　完全軸選択 LU 分解

完全軸選択の LU 分解をすることで A は $PAQ = LU$ となる。P, Q を移項して $A = P^tLUQ^t$ とする。これを $Ax = b$ に代入すると $P^tLUQ^tx = b$ となる。これを解くとき部分ピボット選択の場合と異なる点は逆代入後並べ替えが必要であることであって

$$LUQ^tx = Pb = c$$

$$\begin{aligned} UQ^tx &= L^{-1}c = d \\ Q^tx &= U^{-1}d \\ x &= QU^{-1}d \end{aligned}$$

となり解 x を求めることができる。ここで L としては、U と同じ大きさの下三角部分行列を利用する。それに伴い、c も変更を受ける。

完全ピボット選択法は行の交換に加えて列の交換も行うのでプログラムの計算量は多くなるが、部分ピボット選択法よりも丸め誤差の増大を抑制することができる。

3　ベクトル列の加速法

3.1　$Aitken$ 法による数列の加速法

$\{x_\nu\}$ を α に線形収束する任意の数列とすれば、$0 < |\kappa| < 1$ をみたす定数 κ と 0 に収束する数列 $\{\varepsilon_\nu\}$ が存在して

$$x_{\nu+1} - \alpha = (\kappa + \varepsilon_\nu)(x_\nu - \alpha) \tag{1}$$
$$x_{\nu+2} - \alpha = (\kappa + \varepsilon_{\nu+1})(x_{\nu+1} - \alpha) \tag{2}$$

(1),(2) より $\varepsilon_\nu, \varepsilon_{\nu+1}$ を無視して κ を消去すれば

$$\alpha = x_\nu - \frac{(x_{\nu+1} - x_\nu)^2}{x_{\nu+2} - 2x_{\nu+1} + x_\nu}$$

ゆえに、列 $\{y_\nu\}$ を

$$\begin{aligned} y_\nu &= x_\nu - \frac{(x_{\nu+1} - x_\nu)^2}{x_{\nu+2} - 2x_{\nu+1} + x_\nu} \\ &= x_{\nu+2} - \frac{(x_{\nu+2} - x_{\nu+1})^2}{x_{\nu+2} - 2x_{\nu+1} + x_\nu} \end{aligned} \tag{3}$$

により定義すれば、$\{y_\nu\}$ は $\{x_\nu\}$ より速く α に近づくと予想される。事実この推測は正しく、後で示すように

$$\lim_{\nu\to\infty} \frac{y_\nu - \alpha}{x_{\nu+2} - \alpha} = 0 \tag{4}$$

が成り立つのである。すなわち $y_\nu - \alpha = o(x_{\nu+2} - \alpha)$。(3) を $Aitlen$(エイトケン) の加速法という。x_ν の収束が遅い場合に適用される。一般に、収束列 x_ν から、より速い収束列 y_ν を作る方法を加速法という [3]。

3.2　ベクトル列の加速

反復法によって次の n 次元連立 1 次方程式を解く。方程式は反復法であるから

$$x = Bx + c \tag{5}$$

の形で与えられるものとする。これを一定回数反復して得られるベクトル列を加速する。

x_0 は初期値。

$$x_{k+1} = Bx_k + c \qquad k = 0, 1, \cdots, h$$
$$x = x_k + \text{補正ベクトル}$$

ここで、h は内部反復回数である。$\Delta x_{k-1} = x_k - x_{k-1}$ とおく、このとき $\Delta X_0 = (\Delta x_{h-1}, \Delta x_{h-2}, ..., \Delta x_0)$ は、$n \times h$ の長方形行列となる。ここで $\alpha = (\alpha_0, \alpha_1, ..., \alpha_{h-1})^t$ とおいて、補正ベクトルを$\Delta X_0 \alpha$ と仮定する。$x = x_k + \Delta X_0 \alpha$ を (5) に代入する。

$$\begin{aligned} x_k + \Delta X_0 \alpha &= B(x_k + \Delta X_0 \alpha) + c \\ &= Bx_k + B\Delta X_0 \alpha + c \\ &= (Bx_k + c) + (B\Delta X_0)\alpha \\ &= x_{k+1} + \Delta X_1 \alpha \end{aligned}$$

ここで $\Delta X_1 = (\Delta x_h, \Delta x_{h-1}, ..., \Delta x_1)$ と定義した。最後の等式は、次の関係式を用いて導いた。

$$\begin{array}{rcl} -\left\{ \begin{array}{rcl} x_{k+1} & = & Bx_k + c \\ x_k & = & Bx_{k-1} + c \end{array} \right. & & \\ \hline x_{k+1} - x_k & = & B(x_k - x_{k-1}) \\ \Delta x_k & = & B\Delta x_{k-1} \end{array}$$

これをまとめると、

$$\begin{aligned} x_h + \Delta X_0 \alpha &= x_{h+1} + \Delta X_1 \alpha \\ x_h - x_{h+1} &= (\Delta X_1 - \Delta X_0)\alpha \\ -\Delta x_h &= \Delta^2 X_0 \alpha \end{aligned}$$

ここで $\Delta^2 X_0 = \Delta X_1 - \Delta X_0$ と置いた。ΔX_0^2 の大きさは $n \times h$ の行列である。α に関する過剰条件の連立 1 次方程式 $\Delta^2 X_0 \alpha = -\Delta x_h$ を、α について解く。これを補正して近似解（ベクトル）を求める。$\varepsilon = 1.0 \times 10^{-6}$ を与え、収束判定 $\|\Delta x_k\| < \varepsilon$ を満たせばそれを解とする。また、収束判定を満たさなければ求めた解を初期値 x_0 として収束判定を満たすまで反復し続ける。

3.3 プログラム

上で述べた $\Delta^2 X_0 \alpha = -\Delta x_h$ を完全軸選択の LU 分解を用いて α について解く。$\varepsilon = 1.0 \times 10^{-6}$ を与える。収束判定を行い、 $\|\Delta x_k\| < \varepsilon$ ならば反復終了。そうでなければ $x_{new} = x_k + \Delta X_0 \alpha$ とし、$x_0 = x_{new}$ とおいて反復する。

1000 行 1000 列の以下の非対称な帯行列の解を求め、収束の様子を調べる。

$$B = \begin{pmatrix} 0 & 0.4 & 0 & \cdots & 0 \\ 0.5 & 0 & 0.4 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0.4 \\ 0 & \cdots & 0 & 0.5 & 0 \end{pmatrix}$$

$$c = (0.6, 0.1, \cdots, 0.1, 0.5)^t$$

真の解は $(1, 1, \cdots, 1)^t$ である。

表 1：内部反復回数 $h = 2$

反復回数	x_1	x_2	$\cdots$	x_{1000}
1	0.93483614	0.83188996	$\cdots$	0.83205447
2	0.93078029	0.85009310	$\cdots$	0.87448901
3	0.99130420	0.96851503	$\cdots$	0.94698128
$\vdots$	$\vdots$	$\vdots$		$\vdots$
28	0.99999983	0.99999965	$\cdots$	0.99999735
29	0.99999995	0.99999986	$\cdots$	0.99999834
30	0.99999993	0.99999985	$\cdots$	0.99999874

表 2：内部反復回数 $h = 4$

反復回数	x_1	x_2	$\cdots$	x_{1000}
1	0.89693546	0.81794408	$\cdots$	0.80770771
2	0.96942468	0.87743217	$\cdots$	0.91353062
3	0.99007582	0.97909497	$\cdots$	0.96275675
$\vdots$	$\vdots$	$\vdots$		$\vdots$
15	1.00000041	1.00000021	$\cdots$	0.99999853
16	0.99999959	0.99999964	$\cdots$	0.99999891
17	1.00000023	1.00000017	$\cdots$	0.99999981

表 3：内部反復回数 $h = 6$

反復回数	x_1	x_2	$\cdots$	x_{1000}
1	0.83957536	0.77675311	$\cdots$	0.77826386
2	0.97767423	0.92836170	$\cdots$	0.93226421
3	0.98652306	0.98164703	$\cdots$	0.96779322
$\vdots$	$\vdots$	$\vdots$		$\vdots$
10	1.00000187	0.99999938	$\cdots$	0.99999690
11	1.00000005	0.99999986	$\cdots$	0.99999760
12	0.99999979	1.00000009	$\cdots$	0.99999857

ここで述べた $Aitken$ 加速法は、これらの実験例から有効であることがわかった。内部反復回数 h を増やせば $Aitken$ 法の反復回数は少なくなるが、内部反復の全回数と丸め誤差のことを考慮すれば 1000 次 のとき内部反復回数 h は 4 の場合が適当である。丸め誤差の影響は係数行列 $\Delta^2 X_0$ に顕著に現れる。

4 おわりに

ベクトル列の加速を完全軸選択の LU 分解による $Aitken$ 加速によって加速を行った。ヤコビ法と比べて少ない反復回数で収束判定を満たし、加速法が有効であることを確かめた。もっと実用的な大規模問題についても数値実験を行い、本方法の有効性を確かめることが今後の課題である。

参考文献

[1] 杉浦 洋：数値計算の基礎と応用, 数値解析学への入門, サイエンス社 (1997.11).

[2] 浦 昭二・原田賢一：C 入門、倍風館 (1994.7)

[3] 山本　哲朗：数値解析入門［増訂版］(2003.6)

[4] 仁木　滉・河野　敏行：楽しい反復法, 共立出版株式会社 (1998.1)

[5] 日置昌吾： 最小 2 乗法とその応用, ベクトル列の加速法, 南山大学数理情報学部数理科学科卒業論文 (2004)