

応用科学における偏微分方程式と数値解析  
京都大学数理解析研究所

## 偏微分方程式に対する PCG 法について

慶應義塾大学理工学部 高橋 秀俊  
野寺 隆

### 1. はじめに

近年、高速電子計算機のめざましい利用により、工学問題から生ずる偏微分方程式の数値解法に新たな関心をもたれてきている。特に、有限要素法 (FEM) の急速な発展によつて、偏微分方程式の境界値問題の離散近似から生ずる大規模な連立一次方程式の新たな解法として、Gauss の消去法とともに、共役勾配法 (Conjugate Gradient Method) が再び脚光をあびてきた。

共役勾配法 (CG 法) は、1952 年に、M. R. Hestenes と E. Stiefel [4] によつて開発された解法で、反復解法でありながら、有限回 (高々  $n$  回) のステップで厳密解に到達する性質をもっている。特に、偏微分方程式の境界値問題の有限差分 (FDM) 近似や FEM 解析から生ずる大型疎行列をもつ連立一次方程式の解法として、非常に手軽で有効であることはよく知られている事実である [3], [7], [8]。けれども、時として行列の条件数がかなり悪い場合であるが、いささかデリケートでくせが強く、多少、その収束性にむらがあり、他の解法と比較して十分に速いとはいえない場合もあることが知られている [9], [10]。

今回、我々は、最近報告された行列の条件数を著く改善する行列の preconditioning を利用することにより、従来用いられてきた古典的な CG 法の

収束性よりも格段に速い収束性を持つPCG (Preconditioned Conjugate Gradient) アルゴリズムを開発したので、それについて述べ、さらに、Laplace方程式および重調和方程式の差分近似によって得られる大規模な線型方程式にそのアルゴリズムを適用した場合の詳細な数値実験とともに報告する。

## 2. PCGアルゴリズム

行列のpreconditioningの原理は、O. Axelsson [1] によって提案され、CG法と結合することにより、悪い条件数を持つ線型方程式を解く問題に対して、その行列の固有値の分布を改良することにより、反復解法としてのCG法の収束性を高める方法として導入された。

今、偏微分方程式の境界値問題の離散近似から得られる連立一次方程式

$$A x = b, \quad (2.1)$$

を解く問題を考える。ここで行列Aは、 $n \times n$ の大型疎行列であり、かつ対称な正定値行列であるものとする。

方程式 (2.1) を解く古典的なCG法のアルゴリズムには、HestenesとStiefel (H-S) 版のほかにいくつかの変種があるが、ここでは、次のものを使用する。

(ALGORITHM A)

(1) 任意の初期値  $x_1$  を選び、残差ベクトル

$$r_1 = b - A x_1,$$

および、方向ベクトル

$$p_1 = r_1,$$

を計算する。

(2) 次の手順をくりかえす。( $k = 1, 2, 3, \dots$ )

$$x_{k+1} = x_k + \alpha_k p_k, \quad (2.2a)$$

$$r_{k+1} = r_k - a_k A p_k, \quad (2.2b)$$

$$p_{k+1} = p_k + b_k r_{k+1}, \quad (2.2c)$$

$$a_k = 1 / (p_k, A p_k), \quad (2.2d)$$

$$b_k = 1 / (r_{k+1}, r_{k+1}). \quad (2.2e)$$

この算法は、戸川氏によって高橋版といわれているものであり、方向ベクトル  $p_k$  のスケールが通常のものとは異なっている。詳細は、文献〔9〕を見よ。

ここで、このアルゴリズムを一般化することを考える。今、方程式 (2.1) と同値なつぎのような変換行列  $B$  を掛けた方程式

$$B A x = B b, \quad (2.3)$$

を解くことを考える。行列  $B$  は、任意の  $n \times n$  の対称な正定値行列であるものとする。仮定より、行列  $A$  および  $B$  がともに正定値行列なので、 $B^{1/2}$  が存在して、方程式 (2.3) は、

$$(B^{1/2} A B^{1/2}) (B^{-1/2} x) = B^{1/2} b, \quad (2.4)$$

と変形できる。すなわち、方程式 (2.1) を解くことは、変換行列を方程式 (2.1) に掛けあわせた方程式 (2.4) を解くことに同値なので、ALGORITHM A を、それに適用することにより、次のアルゴリズムが得られる。これを一般CG (Generalized Conjugate Gradient) アルゴリズムと呼ぶことにする。

[ALGORITHM GA]

(1) 任意の初期値  $x_1$  を選び、残差ベクトル

$$r_1 = b - A x_1,$$

および、方向ベクトル

$$p_1 = b_0 B r_1,$$

を計算する。ここで、行列  $B$  は、任意の対称な正定値行列である。

(2) 次の手順をくりかえす。 ( $k = 1, 2, 3, \dots$ )

$$x_{k+1} = x_k + a_k p_k, \quad (2.5a)$$

$$r_{k+1} = r_k - a_k A p_k, \quad (2.5b)$$

$$p_{k+1} = p_k + b_k B r_{k+1}, \quad (2.5c)$$

$$a_k = 1 / (p_k, A p_k), \quad (2.5d)$$

$$b_k = 1 / (r_{k+1}, B r_{k+1}). \quad (2.5e)$$

この算法は、行列  $B$  が、 $p_k$  の漸化式 (2.5 c) と (2.5 e) 式で表される係数  $b_k$  の計算式の内積計算の中に現れることにより、ALGORITHM A の算法よりも、より対称性のあることがわかる。また、 $B = I$  (単位行列) と置くことにより、この算法は、ALGORITHM A となる。

ここで、上記の一般CG法の特性を明らかにするために、この算法から生成される生成多項式について考える。

漸化式 (2.5 b) および (2.5 c) から、残差多項式  $R_1(z), R_2(z), \dots$  および方向多項式  $P_1(z), P_2(z), \dots$  は、初期値を  $R_1(z) = I, P_1(z) = b_0 I$  にとることにより、次の漸化式で定義することができる。

$$R_{k+1}(z) = R_k(z) - a_k z P_k(z), \quad (2.6a)$$

$$P_{k+1}(z) = P_k(z) + b_k R_{k+1}(z). \quad (2.6b)$$

一般に、これらの生成多項式を利用して、残差ベクトルおよび方向ベクトルを

$$r_k = R_k(A B) r_1, \quad (2.7a)$$

$$p_k = B P_k(A B) r_1. \quad (2.7b)$$

と表すことができる。

残差多項式 (2.6 a) から, 帰納法によって

$$R_{k+1}(z) = R_1(z) - z \sum_{i=1}^k a_i P_i(z),$$

となる。ここで,

$$Q_k(z) = \sum_{i=1}^k a_i P_i(z), \quad (2.8)$$

と置くことにより,

$$R_{k+1}(z) = R_1(z) - z Q_k(z), \quad (2.9)$$

となり, よって

$$Q_k(z) = (I - R_{k+1}(z)) / z, \quad (2.10)$$

となる。(2.10) 式および漸化式 (2.5 a) から, 方程式 (2.1) の近似解は

$$x_k = x_1 + B Q_{k-1}(AB) r_1, \quad (2.11)$$

と表すことができる。ここで,  $Q_{k-1}(AB)$  は, 誤差関数

$$E_A(x_k) = \frac{1}{2} (\bar{x} - x_k, A(\bar{x} - x_k)), \quad (2.12)$$

を最小にする行列  $AB$  に対する多項式である。ここで,  $\bar{x}$  は真の解であるものとする。今, 行列  $AB$  が高々  $p$  ( $< n$ ) 個の異なる固有値を持つものとするれば,

$$R_p(AB) = 0, \quad (2.13)$$

となる最小多項式が存在して, (2.10) 式より

$$Q_{p-1}(AB) = (AB)^{-1}, \quad (2.14)$$

となる。よって

$$E_A(x_p) = 0, \quad (2.15)$$

となる。故に、一般CG法は、高々  $p$  ステップで終了することになる。さらにここで、もし行列  $AB$  が近似単位行列であるならば、一般CG法は、非常に速く収束する可能性がある。

これらの事柄から、一般CG法がその効力を発揮するためには、preconditioning 行列  $B$  が、次の二つの条件をいかにうまく満たすように構成できるかに従属することになる。

(1) 行列  $B$  は、元の行列  $A$  の近似逆行列であり、かつ  $A$  と同程度に疎でなければならない。

(2)  $Bz$  (preconditioning 行列  $\times$  ベクトル) は、できるだけ簡単に評価できなければならない。

ここで、行列  $B$  の構成方法としてもっとも簡単に思い付くことは、実際に、行列  $A$  の完全Cholesky分解を利用して真の逆行列を計算して利用することである。けれども、偏微分方程式の境界値問題の離散近似から得られる大次元の疎行列  $A$  の真の逆行列を構成することは、計算機の利用コストの面からも、また使用領域の面においても、あまり芳しい事とは思われない。だがしかし、行列  $A$  の真の逆行列としてではなく、その行列の単なる近似逆行列を実際に構成することは、それほど困難なことではない。

我々は、次のようにして行列  $B$  を構成した。まず、第二の条件を満足するような行列  $B$  は、二つの互いに疎な三角行列を利用して、

$$B = (C C^T)^{-1}, \quad (2.16)$$

と選ぶことができる。即ち、 $Bz$  の評価は、三角行列を係数として持つ二つの方程式を逐次に解くことにより、非常に簡単に計算できる。

問題になるのは行列  $C$  の構成方法であるが、ここでは、元の行列  $A$  の分離 (splitting) を利用して、次のようにした。

行列Aを

$$A = L + D + L^T, \quad (2.17)$$

と表すものとすれば、一般性を失うことなく

$$\begin{aligned} A' &= D^{-1/2} A D^{-1/2}, \\ &= L' + I + (L')^T, \end{aligned} \quad (2.18)$$

と表すことができる。ここで、Iは単位行列である。今、行列Cを

$$C = (I + \omega L'), \quad (2.19)$$

( $\omega$ : adjustable parameter)

と置くことにする。前に述べたBz (=u)の評価は、次の二つの方程式

$$C v = z, \quad (2.20a)$$

$$C^T u = v, \quad (2.20b)$$

を逐次解くことにより計算できることになる。ここで、 $\omega$ は、SSORの加速係数に類似したパラメーターであり、 $0 \leq \omega < 2$ の範囲の値を取ることができる。このようにして選んだBは、真の逆行列 $A^{-1}$ とは程遠いけれども、行列の固有値分布（即ち条件数）を著しく改善する性質を持っている。

最後に、この算法を計算機に実現する場合の作業領域と計算量について考える。

即ち、この算法は、四つのベクトルの作業領域（ $r_k$ ,  $p_k$ ,  $x_k$  および  $Ap_k$  と  $Br_{k+1}$  を計算するために必要な一つの補助ベクトルの領域）が必要である。 $Ap_k$  および  $Br_{k+1}$  の作業領域は、あたかも二つ必要であるかのように思われるけれども、一つのベクトル領域をお互いに利用することができる。また、その計算量においては、行列Aの非零要素総数を $nz(A)$ とするならば、一回の反復にたいして、 $5n + 2 + 2nz(A)$ 回の掛算と、 $5n + 2nz(A)$ 回の足算を必要とする。これは、古典的なCG法（ALGORITHM A）の約2倍の計

算量に匹敵する。この算法の効果的な実現については、文献〔11〕を見よ。

### 3. 数値例

数値実験例として、二つの偏微分方程式の境界値問題を取り扱うものとする。

(例1) (Laplace方程式のDirichlet問題)

境界  $\partial\Gamma$  を持つ菱形領域  $\Gamma$  (図3.1) における Laplace 方程式の境界値問題

$$\Delta u = \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) u = 0, \quad (x, y) \in \Gamma, \quad (3.1)$$

$$u(x, y) = g_1(x, y) \in \partial\Gamma, \quad (3.2a)$$

(第一種境界条件)

$$\frac{\partial u}{\partial n} = g_2(x, y) \in \partial\Gamma, \quad (3.2b)$$

(第二種境界条件)

の差分近似解  $u_1 = u(x_1, y_1)$ ,  $u_2, \dots, u_n$  を求めることを考える。境界条件は、図3.1に示したように取るものとする。今、菱形領域上に、一辺  $h = 1/100$  の一様な大きさの正三角形格子を設定して、七点差分近似式を利用して、この領域を離散化する。この場合、得られた方程式は、9801個の未知数を持つことになる。

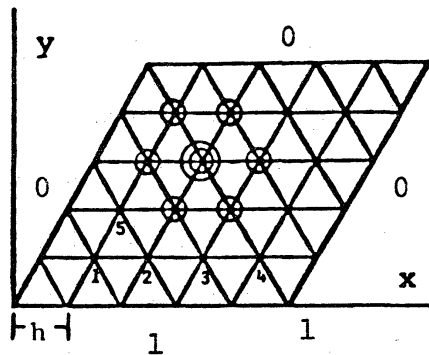


図3.1 菱形領域  $\Gamma$  と境界条件

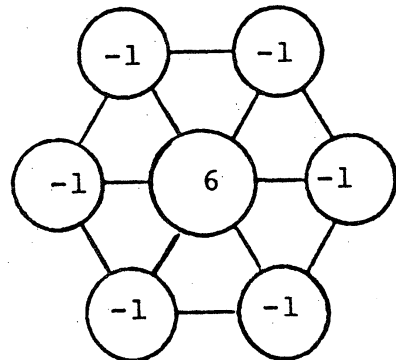


図3.2 七点差分近似式



図4.3に、CG法を利用してこの方程式を解いた場合の残差の収束状況をしめした。この図から、preconditioningを伴う一般CG法 (ALGORITHM GA) の収束性は、古典的なCG法 (ALGORITHM A) のそれよりもはるかによいことがわかる。だがしかし、古典的なCG法の収束状況も、かなり良好である。すなわち、この場合、収束に必要な反復回数は、格子点の総数でなく、一次元的な菱形領域の直径に従属しているからである。また、菱形領域の対称性と正三角形格子の特殊性によって、元の行列Aに重複固有値が多く、収束をより有利にしているからである。

図3.4に、preconditioning変数 $\omega$ と収束に必要な反復回数の相関を示した。この図から、最適な $\omega$ を利用した一般CG法が収束するのに必要な反復回数は、単なる古典的なCG法を使用した場合に必要反復回数の凡そ二分の一乗であることがわかる。

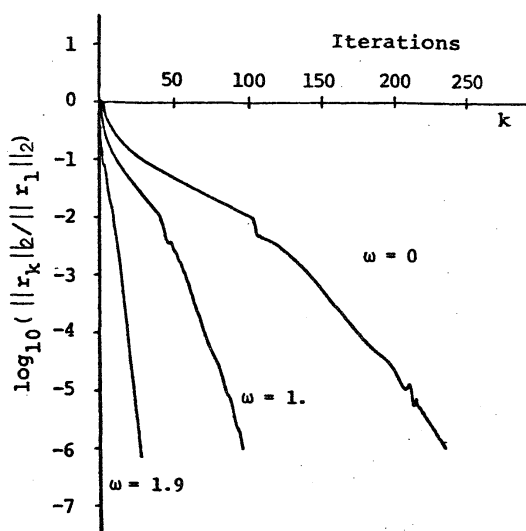


図3.3 残差の収束状況

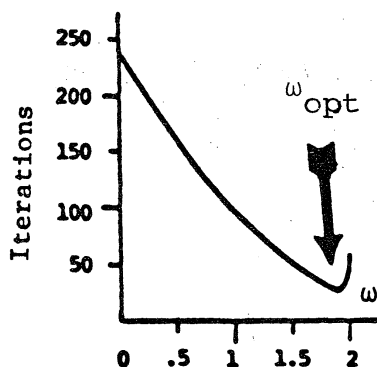


図3.4  $\omega$ と反復回数の相関

#### (例2) (重調和方程式のDirichlet問題)

次に、境界  $\partial\Gamma$  を持つ単位正方形領域 $\Gamma$  (図3.5) における重調和方程式の境界値問題

$$\Delta^2 u = \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right)^2 u = 0, \quad (x, y) \in \Gamma, \quad (3.3)$$

$$u(x, y) = g_1(x, y) \in \partial\Gamma, \quad (3.4a)$$

(第一種境界条件)

$$\frac{\partial u}{\partial n} = g_2(x, y) \in \partial\Gamma, \quad (3.4b)$$

(第二種境界条件)

の差分近似解を求めることを考える。この場合の境界条件は、図3.5に示したように、領域 $\Gamma$ の一边で、 $g_1(x, 0) = 1$ 、 $g_2(x, 0) = 0$ として、他の残りのすべての边上で、 $g_1(x, y) = 0$ 、 $g_2(x, y) = 0$ とする。今、この正方形領域上に一边  $h = 1/40$  の様な大きさの正方形格子を設定して重調和作用素  $\Delta^2$  に対する13点差分近似式  $\Delta_h^2$  (図3.6) を用いて、この領域を離散化するものとする。即ち、得られた方程式は、1521個の未知数を持つことになる。

この方程式を古典的なCG法で解いた場合に、その収束があまり芳しくないことは、よく知られている事実である(図3.7の $\omega = 0$ のグラフを見よ)。また、時として、理論的な終了反復回数を上回ることもある。その理由の一つは、丸め誤差による残差ベクトルおよび方向ベクトルの直交性の崩れであるけれども、さ

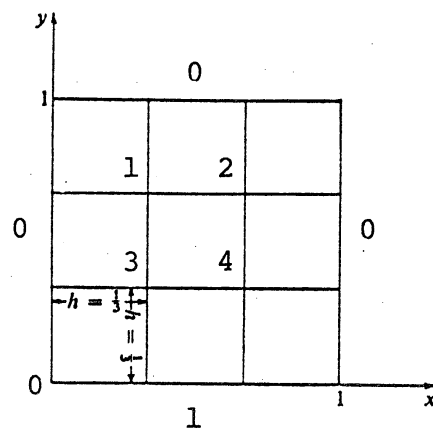


図3.5 正方形領域と境界条件

$$\Delta_h^2 u|_{i,j} = \frac{1}{h^4} \left( \begin{array}{c} \textcircled{1} \\ \textcircled{2} - \textcircled{-8} - \textcircled{2} \\ \textcircled{1} - \textcircled{-8} - \textcircled{20}_{i,j} - \textcircled{-8} - \textcircled{1} \\ \textcircled{2} - \textcircled{-8} - \textcircled{2} \\ \textcircled{1} \end{array} \right) + O(h^2)$$

図3.6 重調和作用素の13点  
差分近似式  $\Delta_h^2$

らに重要な要因は、離散化された方程式の係数行列の固有値に、極端に小さな値を持つものが存在して、この行列の条件数を著しく悪化させているからである。そこで、行列の固有値分布を大幅に改善するPCG法を用いて、この方程式の近似解を求めることになる。

我々は、この問題に対して、特に、次の二つのpreconditioningを試みた。

〔A方式〕

重調和作用素  $\Delta^2$  を直接近似した13点差分方程式の行列分離よりpreconditioning行列Bの要素となる三角行列Cおよび $C^T$ を構成する(図3.7)。

〔B方式〕

重調和作用素  $\Delta^2$  は、Laplace作用素  $\Delta$  を二度掛け合せたものと考えられるので、その近似逆行列として、Laplace作用素を五点差分近似した場合に用いるpreconditioning行列を2度掛け合せたものでCおよび $C^T$ を構成する(図3.8)。

図3.9と3.10において、相異なる $\omega$ に対して、それぞれのpreconditioningを用いた一般CG法の残

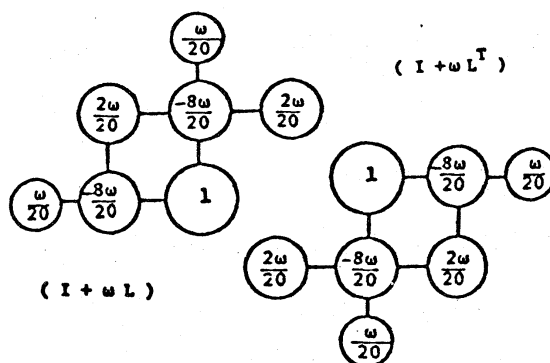


図3.7 行列B〔A方式〕

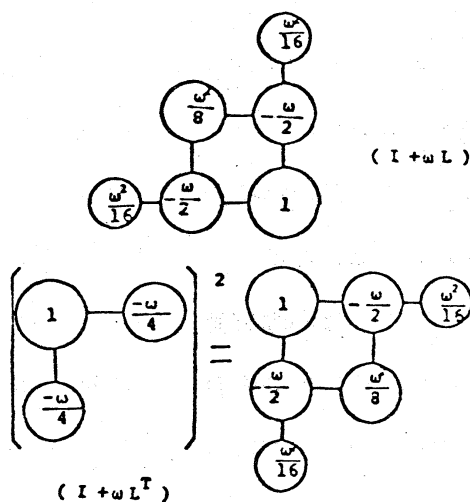


図3.8 行列B〔B方式〕

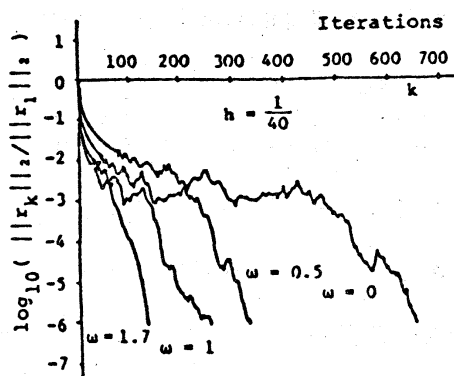


図3.9 残差の収束状況〔方式A〕

差の収束状況を示した。この図から、古典的なCG法の収束性があまり芳しくないにもかかわらず、PCG法は、かなりよい収束性を示していることがわかる。また、行列Bのpreconditioningは、方式Aのそれと比較して、一段とよい収束性を導いていることもわかるであろう。この事実を明確に把握する

ために、我々は、Lanczos法〔2〕を用いて、これらの行列の相異なる総ての固有値を計算した。この問題の場合、いくつかの重複固有値があり、相異なる固有値の数は、761個である。

図3.11に、最小固有値でスケーリングしたこれらの行列の固有値分布を示した。この図から明らかなように、方式Bを適用した行列の固有値分布が、他のものと比較して最もよく、分布の両端で疎で中程で密になっており、CG法が非常によい収束性を示す典型的な固有値分布の例になっている。

ちなみに、これら各々の行列の条件数を比較してみると、次のようになる。ここで、 $\omega$ は1.7の値で計算した。

$$\text{行列Aの条件数: } \text{Cond}(A) = 399374.65, \quad (3.5a)$$

$$\text{方式A: } \text{Cond}(C^{-1} A C^{-T}) = 1573.67, \quad (3.5b)$$

$$\text{方式B: } \text{Cond}(C^{-1} A C^{-T}) = 513.07. \quad (3.5c)$$

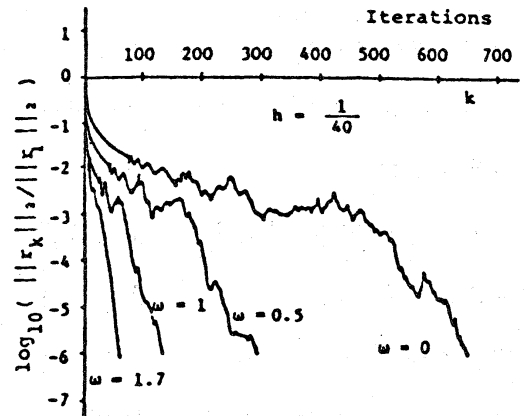


図3.10 残差の収束状況〔方式B〕

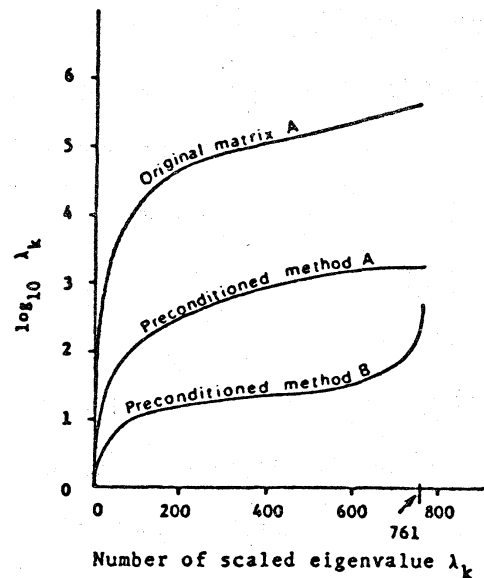


図3.11 行列の固有値分布

( $\omega=1.7$ )

このことから、方式Bを適用した行列の条件数 (3.5 c) は、方式Aの条件数 (3.5 b) よりも一ランク、オーダーが下がっており、また、元の行列Aの条件数 (3.5 a) と比較してみると、 $10^3$  ほど減少しており、凡そ、(3.5 a) の二分の一乗のオーダーであることがわかる。詳細は、文献〔5〕を見よ。

#### 4. おわりに

偏微分方程式の境界値問題の離散近似によって得られる大型疎行列を持つ連立一次方程式の反復解法として、PCG法が、有力であることを示した。特に、高次の偏微分方程式の離散化から起る条件数が極端に悪い大型行列に対しても、preconditioningは、CG法の収束性をより高め、また、近似解の精度をより良いものにできる。

ここで、一番気にかかるのが、preconditioningによる計算量の増加であるが、CG法の反復回数的大幅な減少により全体の計算時間的大幅な短縮が期待できることを考え合せば、それほど問題にはならないであろう。さらに、文献〔11〕に示したように、アルゴリズムの計算量をより減少させるような実現を電子計算機に行えば、その有効性は、より有力なものとなる。特に、我々が提案した行列のpreconditioningは、それを実行するための余分な作業領域を必要としないので、より大次元の行列を取り扱う程、その有効性はさらに増大するであろう。また、非対称行列を持つような方程式を取り扱うPCG法については、文献〔10〕を見よ。

最後に、最も重要な事柄は、古典的なCG法の収束性は領域の格子点の番号づけに依存しないけれども、preconditioningを行うことにより、その収束性が格子点の番号づけに依存するようになることである。その理由は、元の行列Aをpreconditioning行列Bの構成要素である二つの三角行列Cおよび $C^T$ に分離すること自体が、格子点の番号づけに従属しているからである。

## 参考文献

- (1) O. Axelsson, Solution of linear system of equations, Lecture Note on Math. No. 572, Springer-Verlag, pp. 1-51 (1977).
- (2) J. Cullum, R. A. Willoughby, Lanczos and the computation in specified intervals of the spectrum of large, sparse real symmetric matrices, (ed. I. S. Duff and G. W. Stewart) SIAM, pp. 220-255 (1978).
- (3) P. Concus, G. H. Golub, D. P. O'Leary, A Generalized conjugate gradient method for the solution of elliptic partial differential equations, Sparse Matrix Comp. (ed. Bunch and Rose) Academic Press, pp. 309-332 (1976).
- (4) M. R. Hestenes, E. Stiefel, Method of conjugate gradients for solving linear systems, Nat Bur Standards J. Res. 49, pp. 409-436 (1952).
- (5) T. Nodera, H. Takahasi, Preconditioned conjugate gradient algorithm for solving biharmonic equation, Advance in Computer Method for P. D. E. IV, pp. 54-60 (1981).
- (6) T. Nodera, Variants of the PCG algorithm for nonsymmetric system of equations, Conference on Applied Linear Algebra, SIAM (1982).
- (7) J. K. Reid, On the method of conjugate gradients for the solution of large sparse systems of linear equations, in Large Sparse Sets of Linear Equations (ed. J. K. Reid), Academic Press, pp. 231-253 (1971).
- (8) 高橋, 野寺, 行列問題に対する Lanczos 法と共役勾配法, 京都大学数理解析研究所講究録 373, pp. 117-132 (1979).
- (9) H. Takahasi, T. Nodera, New variants of the conjugate gradient algorithm, Second International Congress on Numerical Methods for Engineering in Paris (ed. E. Absi et al), Dunod, pp. 209-219 (1980).

- (10) H. Takahasi, T. Nodera, Preconditioned conjugate gradient algorithms for nonsymmetric matrix, KEIO Math. Sem. Rep. No. 6, pp. 50-72 (1981). (10th IMACS World Congress on System Simulation and Scientific Computation, to appear).
- (11) 高橋, 野寺, PCGアルゴリズムの効果的な実現の一考察, 情報処理第24回(昭和57年前期)全国大会講演論文集.

(連絡住所)

〒223

神奈川県横浜市港北区日吉3-14-1

慶應義塾大学理工学部数理科学科

TEL (044) 61-1141 (代)