クリロフ部分空間法のための前処理法の比較について

宮野, 駿 / MIYANO, Shun

(出版者 / Publisher)
法政大学大学院理工学・工学研究科
(雑誌名 / Journal or Publication Title)
法政大学大学院紀要. 理工学・工学研究科編 / 法政大学大学院紀要. 理工学・工学研究科編
(巻 / Volume)
57
(開始ページ / Start Page)
1
(終了ページ / End Page)
4
(発行年 / Year)
2016-03-24
(URL)
https://doi.org/10.15002/00013058

法政大学大学院理工学・工学研究科紀要　Vol.57(2016 年 3 月)　法政大学

クリロフ部分空間法のための前処理法の比較について

THE COMPARISON OF PRECONDITIONED SYSTEM
FOR KRYLOV SUBSPACE METHODS

宮野駿
Shun MIYANO
指導教員　堀端　康善

法政大学大学院理工学研究科電気電子工学専攻修士課程

The problem of large-scale systems of linear equations, we describe the comparison of preconditioned system for Krylov subspace methods. We used the preconditioned system in Krylov subspace methods, and we compared various preconditioned system.

***Key Words** : Preconditioned System, Krylov Subspace Methods,*

１．はじめに

複雑な形状の計算領域を持つ楕円型偏微分方程式を有限差分法を用いて解く場合，方程式を離散化しようとすると境界の処理が複雑かつ困難であるため，一般座標変換を用いて長方形などの簡単な領域に写像し，その上で有限差分法を用いるのが有効である．その場合，離散化して得られた連立 1 次方程式

$$A\boldsymbol{x} = \boldsymbol{b} \qquad (1)$$

の係数行列 A は非対称行列で，直交座標で離散化したときに比べ，比零要素の対角線の本数が多くなる．このような方程式を解く方法として，クリロフ部分空間法を用いる．また，この計算を高速化する技術の 1 つに前処理があげられる．(1)式の両辺に係数行列 A に近い前処理行列 C^{-1} を用意し，

$$C^{-1}A\boldsymbol{x} = C^{-1}\boldsymbol{b} \qquad (2)$$

として計算することで，収束性の向上を期待することが出来る．その前処理に用いられる行列 C^{-1} にもいくつかの種類がある．例えば，Jacobi 前処理，Gauss-Seidel 前処理，SOR 前処理，Symmetric Gauss-Seidel 前処理，Symmetric SOR 前処理などがあげられる[1].

本稿では，これらの前処理法をクリロフ部分空間法に実装し，得られた結果を比較・検討することを目的とする．

２．計算領域と境界条件

本研究で実際に解く境界値問題として，図 1 のように極座標で与えられている扇形の計算領域を長方形の計算領域に一般座標変換し，ラプラス方程式

$$\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} = 0 \qquad (3)$$

を図 2，図 3，図 4 のように r_x=1.0，2.0，3.0 と計算領域を歪ませていく 3 ケースについてディリクレ型境界条件の元で解く．なお，以下のようなディリクレ型境界条件を用いる[2].

$$\text{on } WX,\ \phi = 0 \qquad (4)$$

$$\text{on } XY,\ \phi = \frac{\sin\theta}{r_{XY}} \qquad (5)$$

$$\text{on } YZ,\ \phi = \frac{1}{r_{YZ}} \qquad (6)$$

$$\text{on } ZW,\ \phi = \frac{\sin\theta}{r_{ZW}} \qquad (7)$$

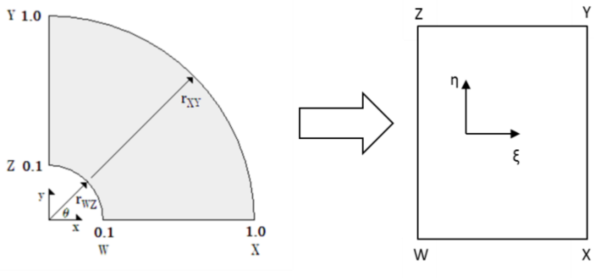

図１：計算領域と一般座標変換

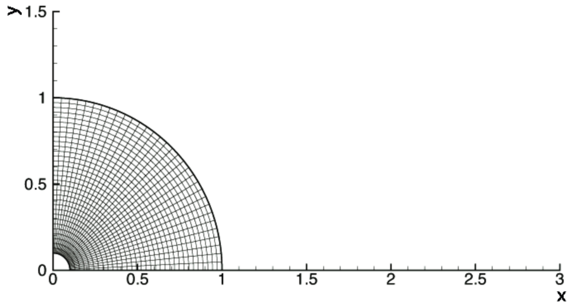

図2：r_x=1.0 としたときの格子

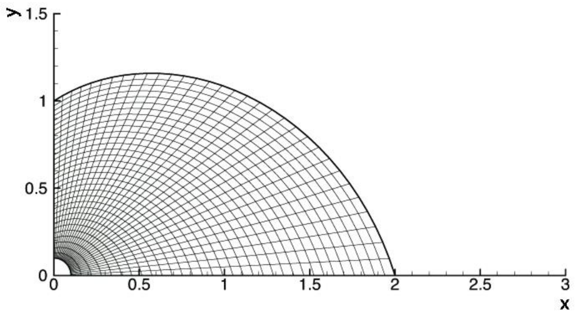

図3：r_x=2.0 としたときの格子

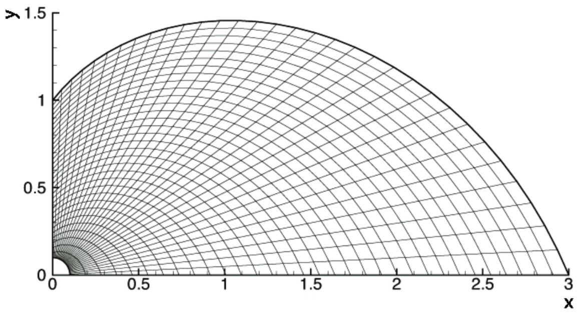

図4：r_x=3.0 としたときの格子

３． 前処理法

本研究で用いた前処理法について述べていく．

(1)ILU 分解

反復法の多くの計算で用いられている方法で，今回は係数行列 A の 9 本の対角要素のみを用いて，不完全 LU 分解をして，

$$A = LD^{-1}U \tag{8}$$

と近似する方法で，この 9 本の対角要素で構成された $LD^{-1}U$ を前処理行列 C とする．

(2)Jacobi 前処理

前処理法の中で，もっとも簡単な前処理で，係数行列 A の対角成分だけで構成される対角行列 D を前処理行列 C とする．

(3)Gauss-Seidel 前処理

係数行列 A の対角成分を含む下三角行列 E を前処理行列 C とする．

(4)SOR 前処理

係数行列 A の対角成分を含む下三角行列 E を用いるが，対角成分を除く下三角行列の要素にのみ加速係数 ω をかけたものを前処理行列 C とする．

(5)SGS 前処理

係数行列 A の対角成分を含む下三角行列 E と係数行列 A の対角成分の逆行列 D^{-1}，係数行列 A の対角成分を含む上三角行列 F を用いた $ED^{-1}F$ を前処理行列 C とする．

(6)SSOR 前処理

SGS 前処理の E と F の対角成分を除く要素に加速係数 ω をかけたものを前処理行列 C とする．[1]

４． 数値実験

離散化して得られた大規模な連立 1 次方程式に前処理を施したクリロフ部分空間法で解く際，計算領域を r_x=1.0，r_x=2.0，r_x=3.0 としたときについて，それぞれ格子数 257×257，513×513，769×769 にしたときの数値実験を行った．

クリロフ部分空間法として，GMRES(k)法，また，CG 系統の解法から BCG 法，CGS 法，BCGSTAB 法，BCGSTAB2 法，GPBCG 法，BCGSafe 法を用い，また，CR 系統の解法から BCR 法,CRS 法,BCRSTAB 法 BCRSTAB2 法,GPBCR 法,BCRSafe 法を用いた．各種解法にそれぞれ 3.前処理法で示した前処理法を施し，CPU 時間について比較したものを表２に，反復回数について比較したものを表３に示す．今回は数値実験で得られたデータが膨大なため，CGS 法において，格子数 769×769，前処理なし，Jacobi 前処理，SGS 前処理，SSOR 前処理，ILU 分解を用いて得られた結果のみを示す．ただし，SSOR 前処理の加速係数 ω は 1.9 を用いる．

また，数値実験の際に使用した計算機環境を以下の表１に示す．

表１：数値実験に使用した計算機環境

計算機	HP Z800 Workstation
CPU	Quad-core Intel Xeon X5677 3.46GHz
OS	Red Hat Enterprise Linux5
Compiler	Intel Fortran Compiler 12.0.0
メモリ	48GB
精度	倍精度
収束判定条件	$\|\|r_k\|\|/\|\|b\|\| < 10^{-12}$

表 2：各種前処理法を施した CGS 法での CPU 時間[s]

前処理法	r_x	反復回数
	1.0	73.0
前処理なし	2.0	収束せず
	3.0	収束せず
	1.0	56.0
Jacobi	2.0	61.2
	3.0	57.3
	1.0	15.7
ILU	2.0	12.9
	3.0	11.4
	1.0	32.7
SGS	2.0	36.1
	3.0	40.0
	1.0	8.3
SSOR(ω=1.9)	2.0	9.8
	3.0	10.3

表 3：各種前処理法を施した CGS 法での反復回数

前処理法	r_x	反復回数
	1.0	3777
前処理なし	2.0	収束せず
	3.0	収束せず
	1.0	2794
Jacobi	2.0	2707
	3.0	2835
	1.0	382
ILU	2.0	315
	3.0	276
	1.0	801
SGS	2.0	877
	3.0	982
	1.0	203
SSOR(ω=1.9)	2.0	239
	3.0	251

５． 結論

前処理法を変えて得られた結果を比較すると，前処理を施すだけで高速化されることが分かる．比較のために表 2，3 にはまとめたが，Jacobi 法は係数行列 A の対角成分しか使用していないため，高速化されているとはいえ，今回のような大規模な連立 1 次方程式に用いるには難がある．Gauss-Seidel 前処理，SOR 前処理に関しても同じことが言える．しかしながら，ILU 分解，SGS 前処理，SSOR 前処理は表 2，3 を見ても分かるように格段に高速化が出来ている．また，SSOR 前処理を施した場合が CPU 時間，反復回数ともに最も少なく，非常に有効な前処理法であることが分かった．SSOR 前処理は 3.前処理法で述べたように，前処理行列のアルゴリズムがとても簡便で，ILU 分解のような複雑な計算を必要としないため，今後，利用頻度は高くなっていくと言える．

参考文献

1) Yousef Saad : Iterative Methods for Sparse Linear Systems SECOND EDITION, Society for Industrial and Applied Mathematics, 2003

2) 浅井雄太：クリロフ部分空間法と多重格子法の数値実験による比較，法政大学, 2013