

SOR 法を用いた可変的前処理付き一般化共役残差法

阿部邦美* 張 紹良** 長谷川秀彦† 姫野龍太郎*

*理化学研究所情報環境室 **東京大学大学院工学系研究科 †図書館情報大学

A SOR-base Variable Preconditioned GCR Method

Kuniyoshi Abe*, Shao-Liang Zhang**, Hidehiko Hasegawa† and Ryutaro Himeno*

* The Institute of Physical and Chemical Research ** University of Tokyo

†University of Library and Information Science

Abstract. We propose a new preconditioning strategy for the Krylov subspace method for solving a large sparse system $Ax = b$. The basic idea is to use an approximation to $A^{-1}v$ for $K^{-1}v$: the benefit is substantial when the preconditioner K sufficiently approximates A . The preconditioning is performed by approximately solving $Az = v$ by some iterative method. Then different types of preconditioner can be applied at each iterative step. In the present paper, we combine the SOR method as a preconditioner and the GCR(m) method as a solver, and show that our preconditioning has lower iterative counts and shorter computation time than ILU(0).

1 はじめに

$n \times n$ の正則な大規模疎行列 A を係数行列, n 次元ベクトル b を右辺項とする連立一次方程式

$$(1.1) \quad Ax = b$$

を Krylov 部分空間解法によって近似的に解くことを考える.

Krylov 部分空間解法の端緒となった Conjugate Gradient method (共役勾配法, CG 法) [7] は対称正定値問題に対する解法として 1952 年に発表された後, 収束性の良さを示す反面丸め誤差に敏感なため, しばらくの間は顧みられなかった. しかし, 1970 年代に前処理の併用とともに再び脚光を浴びるようになった. この前処理とは, 数学的同値性が保たれるような変換を行ない, Krylov 部分空間解法の収束性に好ましいように係数行列の固有値分布を改善する方法である. 前処理との併用によって, Krylov 部分空間解法の収束性が大幅に改善されることが報告されている.

一般の非対称行列に対する前処理付き Krylov 部分空間解法のアルゴリズムでは, まず係数行列 A に近似的に等しく, $K^{-1}v$ の計算が容易にできるような前処理行列 K を構築する. 行列 K については, これまでに多くの構築方法が提案されてきたが, 代表的な方法として, 不完全 LU 分解 (Incomplete LU factorization, ILU)[2, 8, 10] が知られている. この方法では, $A \approx LU$ と分解して前処理行列 $K = LU$ を構築し, $K^{-1}v$ は直接法によって $Kz = v$ を解いて求める.

近年では, Generalized Minimal RESidual method (GMRES 法) [11] の変形として FGMRES 法 [9], GMRESR 法 [12] が開発されている. FGMRES 法は, 反復毎に前処理行列を変え (K_k), 反復過程で計算された $K_k^{-1}v$ を保存することによって近似解を構成する. GMRESR 法の場合, 行列分離を利用した GMRES 法の導出において前処理行列の可変性を見出し, 適当な $A^{-1}v$ の近似を求めることによって前処理を行なう. 実際, これら 2 つの方法においては, GMRES 法のアルゴリズムで現れる $K^{-1}v$ を計算する代わりに, 方程式 $Az = v$ を解くことによって前処理が行なわれている. このとき, 方程式 $Az = v$ は主に GMRES 法を用いて固定した回数反復で解かれる. これは, 新たな前処理の概念, すなわち反復毎に前処理を変えているとみなせる.

一方, われわれは前処理行列が満たす基本的性質に着眼し, FGMRES 法や GMRESR 法とは異なったアプローチの新しい前処理を提案する. その方法は, 前処理行列 K が係数行列 A の近似であること, および反復過程で現れる $K^{-1}v$ の計算に着目する. このとき, 前処理の効果という観点から, $K^{-1}v$ は $A^{-1}v$ を十分に近似することが望ましい. それゆえ, $K^{-1}v$ を求める代わりに $A^{-1}v$ の近似を求める. 言い換えれば, 反復過程で解かなければならない連立一次方程式 $Kz = v$ の代わりに, 方程式 $Az = v$ を任意の反復解法によって近似的に解く. ただし, FGMRES 法や GMRESR 法のように一定回数反復によるのではなく, 精度と反復回数に関する停止条件を設定する. すると, 各反復で反復回数を変化させることができるため, 反復毎に異なった前処理を適用したとみなせる. それゆえ, われわれはこのような前処理を可変的前処理 (Variable preconditioning) と呼ぶ. 可変的前処理は, GMRES 法に限定することなく, 様々な Krylov 部分空間解法に適用できる可能性を持つこと, また新たな停止条件を設定することによって FGMRES 法や GMRESR 法とは異なった可変性を実現できることが特長である.

本研究では, この前処理を Generalized Conjugate Residual method (一般化共役残差法, GCR 法) [3] に適用する. GCR 法の反復過程では, 前処理として方程式 $Az = r$ を解く必要がある. 演算量の少ない解法で, 反復の初期段階に必要な精度を満たせば効率的に前処理される. そこで, そのような特性の解法を選択するために, 予めいくつかの解法を方程式 (1.1) に適用し, それらの収束の振舞いを考察する. 結果としてもっとも効率的と考えられた Successive Over-Relaxation method (SOR 法) [6, 13] を適用する.

本論文 2 節では, 前処理付き GCR(m) 法のアルゴリズムの概略について記す. 3.1 小節では, 提案する前処理の理論的根拠, およびその構築方法, さらに 3.2 小節では, 可変的前処理を GCR(m) 法に実装する方法を述べる. 4.1 小節では前処理としての方程式 $Az = r$ に SOR 法を適用した理由を述べ, 4.2 小節の数値実験では Helmholtz 方程式を離散化して得られた問題を取り上げ, 可変的前処理が効果的であること, また各反復で前処理が変化することを示す. 最後に, 5 節でまとめを行なう.

2 従来の前処理

本節においては, 前処理付き GCR(m) 法のアルゴリズム, 不完全 LU 分解による前処理行列の構築方法, さらに $K^{-1}v$ の計算方法について記す.

2.1 前処理付き GCR(m) 法

$n \times n$ の正則行列 K を A の前処理行列とすると、連立一次方程式 (1.1) を数学的に同値な

$$(AK^{-1})(Kx) = b$$

と変換する。Krylov 部分空間解法では、 AK^{-1} が単位行列の近似、すなわち係数行列 A の近似となるように前処理行列 K を選ぶことが良いとされている。なぜなら、 AK^{-1} の固有値は 1 の近くに密集するため、Krylov 部分空間解法の収束が速くなると期待できるからである。そして、変換された連立一次方程式

$$(2.1) \quad \tilde{A}\tilde{x} = b$$

に Krylov 部分空間解法を適用する。ただし、 $\tilde{A} = AK^{-1}$ 、 $\tilde{x} = Kx$ である。

ここでは、Krylov 部分空間解法として残差最小性に基づく解法の 1 つである GCR 法に演算量や記憶容量を考慮した GCR(m) 法 [3, 5] を用いる。ただし、 m はリスタート周期を表す。方程式 (2.1) に GCR(m) 法を適用し、元の A 、 x の系に戻せば以下の前処理付き GCR(m) 法のアルゴリズムが得られる [3, 5]。

前処理付き GCR(m) 法のアルゴリズム：

Let \mathbf{x}_0 be an initial guess.

repeat

set $\mathbf{r}_0 = b - A\mathbf{x}_0$, $\mathbf{p}_0 = K^{-1}\mathbf{r}_0$, $\mathbf{q}_0 = A\mathbf{p}_0$

for $k = 0, 1, \dots, m - 1$

$$\alpha_k = \frac{(\mathbf{r}_k, \mathbf{q}_k)}{(\mathbf{q}_k, \mathbf{q}_k)}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{q}_k$$

if $\|\mathbf{r}_{k+1}\|_2 \leq \varepsilon_{\text{TOL}} \cdot \|\mathbf{r}_0\|_2$ then exit

$$\beta_{k,i} = -\frac{(AK^{-1}\mathbf{r}_{k+1}, \mathbf{q}_i)}{(\mathbf{q}_i, \mathbf{q}_i)}, \quad i \leq k$$

$$\mathbf{p}_{k+1} = K^{-1}\mathbf{r}_{k+1} + \sum_{i=0}^k \beta_{k,i} \mathbf{p}_i$$

$$\mathbf{q}_{k+1} = AK^{-1}\mathbf{r}_{k+1} + \sum_{i=0}^k \beta_{k,i} \mathbf{q}_i$$

end for

$$\mathbf{x}_0 = \mathbf{x}_m$$

end repeat

2.2 不完全 LU 分解

一般に、疎行列 A を LU 分解すると L, U は疎でなくなる。ただし、 L は下三角行列、 U は上三角行列を表す。そこで、次のような点に考慮して、近似的な LU 分解を行なうのが不完全 LU 分解である。

- LU 分解が簡単に計算できる
- L, U は係数行列 A と同程度に疎である

実際に、不完全 LU 分解では、

$$A = LU + R$$

と表される。ここで、 R は係数行列 A と行列 LU との差である。一般の LU 分解では $R = O$ である。このとき得られた行列 LU を前処理行列 K として用いる。

前処理としてよく使用されている ILU(0)[2, 8, 10] は、不完全 LU 分解で L, U の非零要素を A の非零要素と一致させたものである。

次に、アルゴリズムで現れる

$$(2.2) \quad K^{-1}v$$

を計算するために、 K を係数にもつ連立一次方程式

$$Kz = v$$

を解く。 $K = LU$ であるから、式 (2.3) は前進代入、式 (2.4) は後退代入を用いて解ける。

$$(2.3) \quad Ly = v,$$

$$(2.4) \quad Uz = y.$$

3 新たな前処理の提案

本節では、提案する前処理の概念、および GCR(m) 法への実装について述べる。

3.1 提案する前処理の概念

従来の前処理では、まず K を定め、反復過程で $K^{-1}v$ を計算する。われわれの提案する方法では、前処理行列 K が係数行列 A と近似的に等しいという性質に着目して、 $K^{-1}v$ を計算する代わりに、適当な反復解法を用いて $A^{-1}v$ の近似を求めることによって前処理を行なう。

2.1小節からわかるように、前処理行列 K は係数行列 A の近似となるように構築される。

$$K \approx A.$$

この性質に着目すれば，式 (2.2) は次のように近似できる．

$$K^{-1}v \approx A^{-1}v.$$

このとき， K が A を十分に良く近似していると，言い換えれば $K^{-1}v$ が $A^{-1}v$ を十分に良く近似していると，前処理の効果は大きいことが期待できる．したがって， $A^{-1}v$ を求めることが理想的であるが，一般には計算量の面で困難があり，それゆえ従来は計算がより簡単に行なえるような K を定め $K^{-1}v$ を求めざるを得なかった．

そこで，本研究では以下のように計算量が少ない方法で $A^{-1}v$ の近似を求めることを考える．すなわち，式 (2.2) の代わりに，反復解法を用いて適当な精度を満たすように方程式 (3.1) を解くことによって $A^{-1}v$ の近似を求める．

$$(3.1) \quad Az = v.$$

ここでの反復解法は任意の方法（定常反復解法，あるいは Krylov 部分空間解法）でよい．その上， $A^{-1}v$ の近似を直接的な手段によって求めるため，前処理行列 K を構築する初期段階の手間を省くことができる．

方程式 (3.1) を解く場合，一定回数の反復を行なうのではなく，精度と反復回数の両方を停止条件として設定する．このとき，結果的に各反復における反復回数が変化するので，各反復での前処理は異なってくる．

以上，可変的前処理の特徴は次のようにまとめられる．

- 前処理行列 K を構築する必要がない
- $K^{-1}v$ を求める代わりに，反復解法を用いて $Az = v$ の近似解を求める
- 各反復において，反復回数が増えることにより，異なった前処理が適用できる

3.2 GCR(m) 法への実装

可変的前処理を GCR(m) 法へ施したアルゴリズムは，2.1小節で述べた前処理付き GCR(m) 法における $K^{-1}r$ の計算を $A^{-1}r$ の近似を求めるように書き直せばよい．そのようにして得られる以下のアルゴリズムを，Variable preconditioned GCR(m) method(可変的前処理付き GCR(m) 法，略して VPGCR(m) 法) と名付ける．

可変的前処理付き GCR(m) 法 (VPGCR(m)) のアルゴリズム：

Let x_0 be an initial guess.

repeat

set $r_0 = b - Ax_0$

approximately solve $Ap = r_0$ using some iterative method to get p_0

set $q_0 = Ap_0$

for $k = 0, 1, \dots, m - 1$

$$\alpha_k = \frac{(\mathbf{r}_k, \mathbf{q}_k)}{(\mathbf{q}_k, \mathbf{q}_k)}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{q}_k$$

if $\|\mathbf{r}_{k+1}\|_2 \leq \varepsilon_{\text{TOL}} \cdot \|\mathbf{r}_0\|_2$ then exit

approximately solve $Az = \mathbf{r}_{k+1}$ using some iterative method to get \mathbf{z}_{k+1}

$$\beta_{k,i} = -\frac{(Az_{k+1}, \mathbf{q}_i)}{(\mathbf{q}_i, \mathbf{q}_i)}, \quad i \leq k$$

$$\mathbf{p}_{k+1} = \mathbf{z}_{k+1} + \sum_{i=0}^k \beta_{k,i} \mathbf{p}_i$$

$$\mathbf{q}_{k+1} = Az_{k+1} + \sum_{i=0}^k \beta_{k,i} \mathbf{q}_i$$

end for

$$\mathbf{x}_0 = \mathbf{x}_m$$

end repeat

方程式 $Ax = b$ を解く際の反復を外部反復， $Az = \mathbf{r}_{k+1}$ を反復解法で解く際の反復を内部反復と呼ぶ．

次に，外部反復の各反復では方程式 $Az = \mathbf{r}_{k+1}$ を反復解法で近似的に解かなければならない．このとき，定常反復解法や Krylov 部分空間解法など様々な解法の使用が考えられるが，次のような点に考慮する必要がある．

- 内部反復の総反復回数は多くなるため，計算時間，演算量の少ない解法を選ぶ
- 少ない反復回数で必要な精度（十分な精度は要求されない）を満たすことが望ましい．したがって，反復の初期段階において残差ノルムが振動，停滞せず急速に減少する解法が好ましい
- 十分な精度を要求しないため，頑健な解法を用いる必要性はない

以上のような点を考慮して，内部反復に使用する解法を実験によって決める（4.1小節）．

外部反復の残差，すなわち内部反復の右辺項は反復毎に変化するため，予め反復回数を固定して解く場合，近似解の精度が不十分で前処理の効果が得られない可能性がある．また，方程式（3.1）の相対残差ノルム（ $\|\mathbf{r}_{k+1} - Az\|_2 / \|\mathbf{r}_{k+1}\|_2$ ）が1より小さくなるような近似解が求まれば前処理の効果が期待できる．それゆえ，残差に基づいた収束判定が重要である．さらに，必要な精度を満たすまでに多くの反復回数を要すると効率が悪くなるため，最大反復回数を設ける．

これらの理由から，次のような停止条件を設定した．ただし， $k+1$ 回目の外部反復における内部反復の l 回目に求められた近似解を $z_{k+1}^{(l)}$ と表す．また，内部反復に Krylov 部分空間を用いた場合は停止条件 1(A)，定常反復解法を用いた場合は停止条件 1(B) を使用する．

内部反復に使用する解法の停止条件：

条件 1, 2 のいずれか一方の条件を満たした場合に 内部反復を停止する .

1. (A) $\|r_{k+1} - Az_{k+1}^{(l)}\|_2 / \|r_{k+1}\|_2 \leq \delta$.
(B) $\|z_{k+1}^{(l)} - z_{k+1}^{(l-1)}\|_\infty / \|z_{k+1}^{(l)}\|_\infty \leq \delta$.
2. (内部反復の反復回数 l) = N_{\max} .

4 数値実験

正方領域 $\Omega = (0, \pi) \times (0, \pi)$ において, 次の Helmholtz 方程式の離散近似解を求める .

$$(4.1) \quad \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \sigma^2 u = 0, \quad 0 < x, y < \pi.$$

境界条件は以下の通りとする .

$$\begin{aligned} u_x |_{x=0} &= i\sqrt{\sigma^2 - \frac{1}{4}} \cos \frac{y}{2}, && \text{Neumann 条件,} \\ u_x - i\sqrt{\sigma^2 - \frac{1}{4}} u |_{x=\pi} &= 0, && \text{放射条件,} \\ u_y |_{y=0} &= 0, && \text{Neumann 条件,} \\ u |_{y=\pi} &= 0, && \text{Dirichlet 条件.} \end{aligned}$$

$i^2 = -1$ である . また, 連立一次方程式 (1.1) の右辺項は式 (4.1) の厳密解 $u(x, y) = e^{i\sqrt{\sigma^2 - \frac{1}{4}}x} \cos \frac{y}{2}$ を与えて計算する [1] .

この境界値問題に対して, Ω を x, y 方向ともに M 等分した正方格子を考え, 式 (4.1) を 5 点中心差分で離散化する . 得られた複素係数行列 A は $n \times n$ ($n = (M + 1) \times M$) の Non-Hermitian 行列となる .

以下の数値実験では, 定数 $\sigma = 1.5$ (波数 $\sigma^2 = 2.25$), 3.5 (波数 $\sigma^2 = 12.25$), また $M = 100$ にとり, 101×100 個の未知数をもつ連立一次方程式 (1.1) を解く . 定数 $\sigma = 1.5$ の場合には GCR(9) 法 (反復 9 回ごとにリスタート), 定数 $\sigma = 3.5$ の場合には GCR(20) 法 (反復 20 回ごとにリスタート) を適用した . また, すべての数値実験は PC-AT 互換機 (Pentium III 800MHz) において富士通 Fortran コンパイラの倍精度演算によって実行された . さらに, 外部反復において, 初期ベクトル $x_0 = 0$, 収束判定条件 $\epsilon_{\text{TOL}} = 10^{-12}$ を採用した .

4.1 内部解法の選択

3.2 小節で述べたように, 反復の初期段階で急速に残差ノルムが減少し, かつ演算量の少ない解法を内部反復に用いるとき, 前処理が効率的に行なえる . そこで, 予め連立一次方程式 (1.1) をいくつかの反復解法で解いたときの収束振舞いを調べ, 望ましい振舞いをもつ解法を内部反復に用いる .

連立一次方程式 (1.1) を SOR 法, GCR(m) 法, ILU(0) 付き GCR(m) 法によって解く. ここでは, Bi-Conjugate Gradient Method (双共役勾配法, Bi-CG 法) [4] 系統の解法は残差ノルムが単調減少しないため, 内部反復に用いる解法として適当でない判断し, 選択しない.

定数 $\sigma = 1.5, 3.5$ に関する SOR 法, GCR (m) 法, ILU(0) 付き GCR(m) 法の収束振舞いを Fig. 1, Fig. 2に示す. グラフにおける “SOR(1.1)” , “SOR(1.9)” は加速係数がそれぞれ $\omega = 1.1, 1.9$ の場合の SOR 法, “GCR” は GCR(m) 法, “ILU(0)-GCR” は ILU(0) 付き GCR(m) 法の収束振舞いである. また, グラフの横軸は反復回数, 縦軸は相対残差ノルム (点線), および SOR 法の相対誤差 (実線, $\log_{10}(\| \mathbf{x}_k - \mathbf{x}_{k-1} \|_{\infty} / \| \mathbf{x}_k \|_{\infty})$) を表す. ただし, 点線で表された GCR(m) 法, ILU(0) 付き GCR(m) 法の相対残差ノルムは $\log_{10}(\| \mathbf{r}_k \|_2 / \| \mathbf{r}_0 \|_2)$, SOR 法の場合は $\log_{10}(\| \mathbf{b} - A\mathbf{x}_k \|_2 / \| \mathbf{b} - A\mathbf{x}_0 \|_2)$ で計算されている. また, $\mathbf{r}_k, \mathbf{x}_k$ はアルゴリズムによって漸化的に計算された残差, 近似解である.

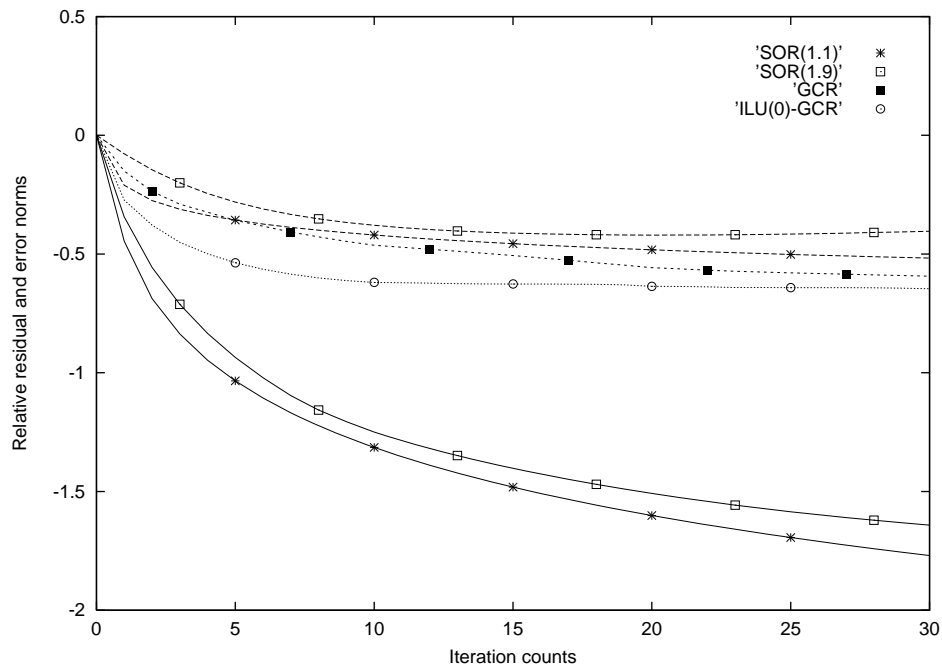


Fig. 1. Convergence history of SOR and GCR methods ($\sigma = 1.5$).

[結果]

ILU(0) 付き GCR(m) 法は $\sigma = 1.5$ のとき 16979 回, $\sigma = 3.5$ のとき 13394 回で収束したが, GCR(m) 法, SOR 法は共に 30000 回までの反復で収束しなかった. また, SOR 法の相対誤差は停滞してしまうが, 反復の初期段階において残差より小さい値となることがわかる.

[考察]

反復の初期段階における SOR 法, GCR(m) 法, ILU(0) 付き GCR(m) 法の相対残差は, 同じように振舞う. 一方, SOR 法の相対誤差は相対残差に比べ反復の初期段階で急速に減少している. 反復 1 回当たりの演算量は SOR 法, GCR(m) 法, ILU(0) 付き GCR(m) 法の順に少ない. この考察から, 3.2 小節で述べた条件を満足し, 前処理が効率的に行なわれる

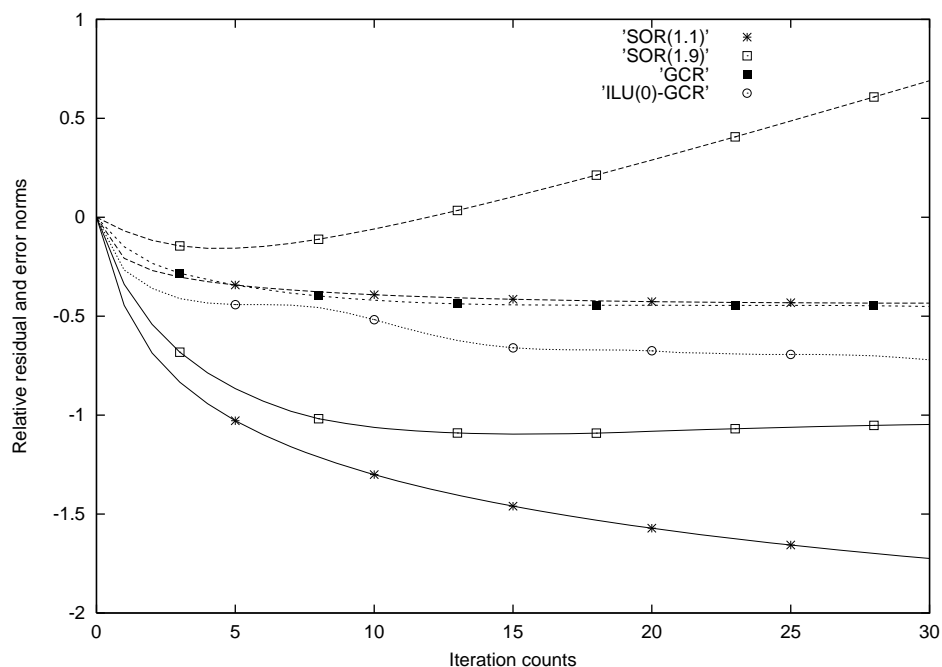


Fig. 2. Convergence history of SOR and GCR methods ($\sigma = 3.5$).

のは相対誤差を停止条件とした SOR 法とみなせる．そこで，以下では内部反復に SOR 法を用いた．

4.2 実験結果

本小節では，内部反復に SOR 法を用いた可変的前処理付き GCR(m) 法，および ILU(0) 付き GCR(m) 法の収束性，計算時間を比較し，提案した可変的前処理が有効であることを示す．さらに，内部反復の反復回数が各反復で変化し，異なった前処理が適用されたことを示す．

内部反復における相対誤差の収束判定条件，および最大反復回数は次の値を採用した．

$\sigma = 1.5$ の場合：

- 相対誤差の収束判定条件は $\delta = 10^{-1.5} \simeq 0.031$
- 最大反復回数は $N_{\max} = 50$

$\sigma = 3.5$ の場合：

- 相対誤差の収束判定条件は $\delta = 10^{-1.25} \simeq 0.056$
- 最大反復回数は $N_{\max} = 70$

内部反復においては，初期ベクトル $z_{k+1}^{(0)} = \mathbf{0}$ とし，SOR 法の加速係数は $\omega = 1.1, 1.3, 1.5, 1.7, 1.9$ と変化させた．

このような停止条件のもとで SOR 法を用いた可変的前処理付き GCR(m) 法と ILU(0) を前処理とした GCR(m) 法の反復回数及び計算時間を Table 1に示す .

Table 1. Iteration counts and computation time for VPGCR(m) and ILU(0)-GCR(m).

Preconditioning	$\sigma = 1.5$		$\sigma = 3.5$	
	Iteration counts	Computation time	Iteration counts	Computation time
ILU(0)	16979	652.4 sec	13394	1083.2 sec
SOR($\omega = 1.9$)	40	16.7 sec	42	27.8 sec
SOR($\omega = 1.7$)	213	84.6 sec	1000	302.5 sec
SOR($\omega = 1.5$)	546	216.8 sec	1403	414.1 sec
SOR($\omega = 1.3$)	838	330.6 sec	2199	637.9 sec
SOR($\omega = 1.1$)	1114	440.6 sec	2734	790.9 sec

さらに , SOR 法を用いた可変的前処理付き GCR(m) 法および ILU(0) を前処理とした GCR(m) 法の収束特性をそれぞれ Fig. 3 , Fig. 4に示す . グラフの横軸は外部反復回数 , 縦軸は相対残差ノルム ($\log_{10} (\| r_k \|_2 / \| r_0 \|_2)$) を表す .

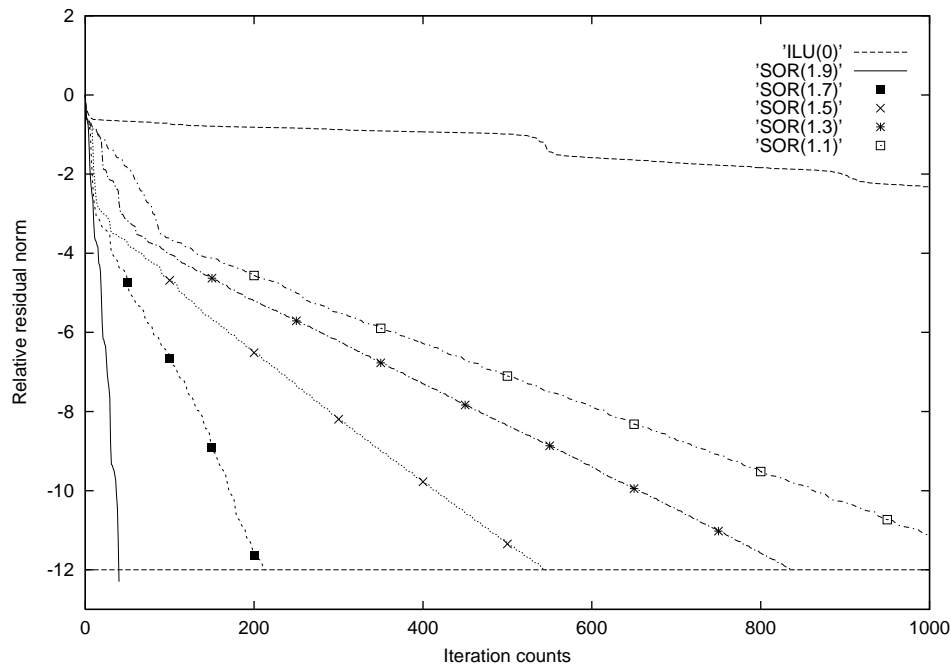


Fig. 3. Convergence history of VPGCR(9) and ILU(0)-GCR(9) methods ($\sigma = 1.5$).

[結果]

SOR 法 , GCR(m) 法を単独で適用した場合には収束しないにもかかわらず , 可変的前処理として SOR 法を用いた GCR(m) 法は収束している . また , Fig. 3 , Fig. 4からわかるように , ILU(0) を用いた場合はゆっくりと収束する一方 , SOR 法を用いた可変的前処理の場

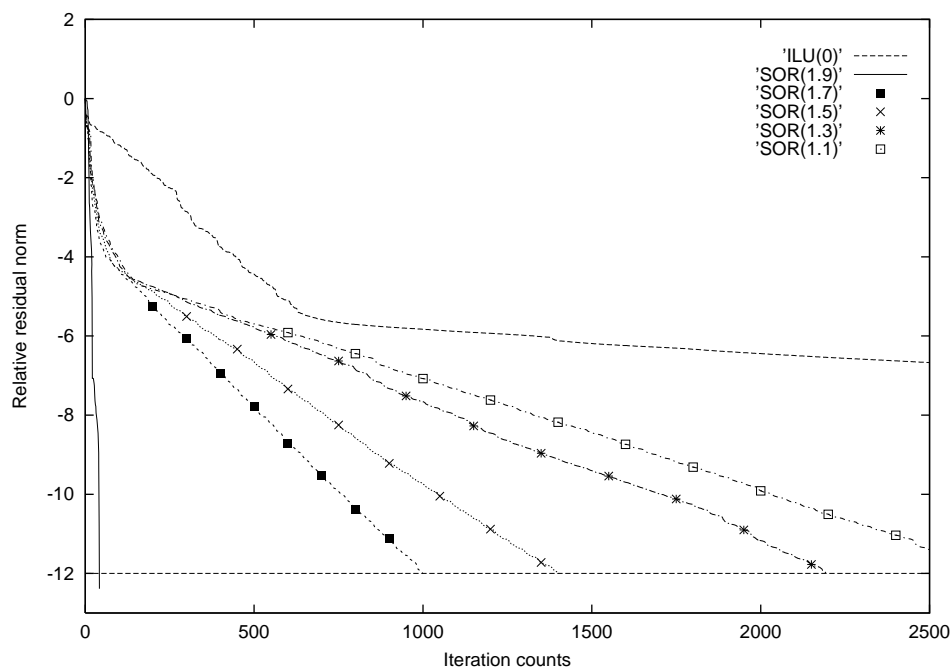


Fig. 4. Convergence history of VPGCR(20) and ILU(0)-GCR(20) methods ($\sigma = 3.5$).

合は急速に収束している。 $\sigma = 1.5$ のとき、ILU(0) を用いた場合 16979 回 (652.4sec)、SOR 法 ($\omega = 1.9$) を用いた可変的前処理は 40 回 (16.7sec)、SOR 法 ($\omega = 1.1$) の場合は 1114 回 (440.6sec) である。また、 $\sigma = 3.5$ のとき、ILU(0) を用いた場合 13394 回 (1083.2sec)、SOR 法 ($\omega = 1.9$) を用いた可変的前処理は 42 回 (27.8sec)、SOR 法 ($\omega = 1.1$) の場合は 2734 回 (790.9sec) である。ILU(0) を用いた場合と比較して、 $\sigma = 1.5$ の場合、SOR 法 ($\omega = 1.9$) を用いると外部反復回数はおおよそ 0.23%、計算時間はおおよそ 2.56%、SOR 法 ($\omega = 1.1$) を用いた場合はそれぞれおおよそ 6.56%、67.5% となる。また、 $\sigma = 3.5$ の場合、SOR 法 ($\omega = 1.9$) を用いると外部反復回数はおおよそ 0.31%、計算時間はおおよそ 2.56%、SOR 法 ($\omega = 1.1$) を用いた場合はそれぞれおおよそ 20.4%、73.0% となる。SOR 法の加速係数に依存して収束性は変化するが、反復回数、計算時間の両方で、SOR 法を用いた可変的前処理が ILU(0) を用いた前処理より効果的であった。

次に、GCR(m) 法の各反復において異なる前処理が適用されていることを加速係数 $\omega = 1.9$ の場合で示す。Fig. 5, Fig. 6 における横軸は GCR (m) 法の反復回数、縦軸は GCR (m) 法の各反復での SOR 法の反復回数を表す。

$\sigma = 1.5$ のとき、内部反復は設定した最大反復回数 N_{\max} に達することなく相対誤差の収束判定条件 δ によって終了していることがわかる。 $\sigma = 3.5$ のとき、最大反復回数 N_{\max} 、または相対誤差の収束判定条件 δ によって終了していることがわかる。さらに、これらのグラフから、GCR (m) 法の各反復において内部反復に用いた SOR 法の反復回数は異なることがわかる。すなわち、右辺項の変化に応じて GCR (m) 法の各反復での前処理は異なり、可変的な前処理が実現できている。

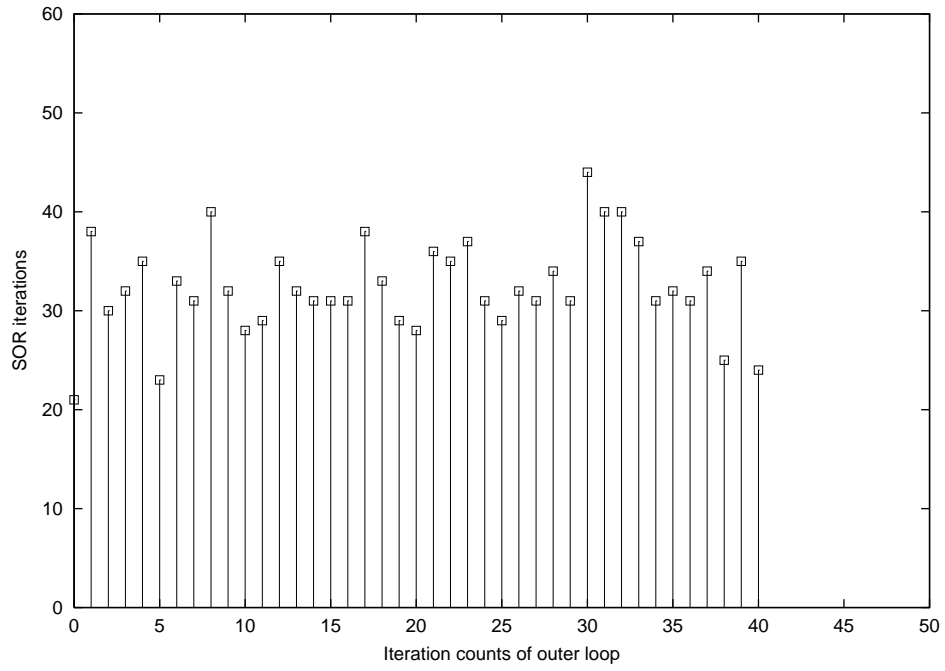


Fig. 5. The iteration counts of SOR method at each outer loop ($\sigma = 1.5$).

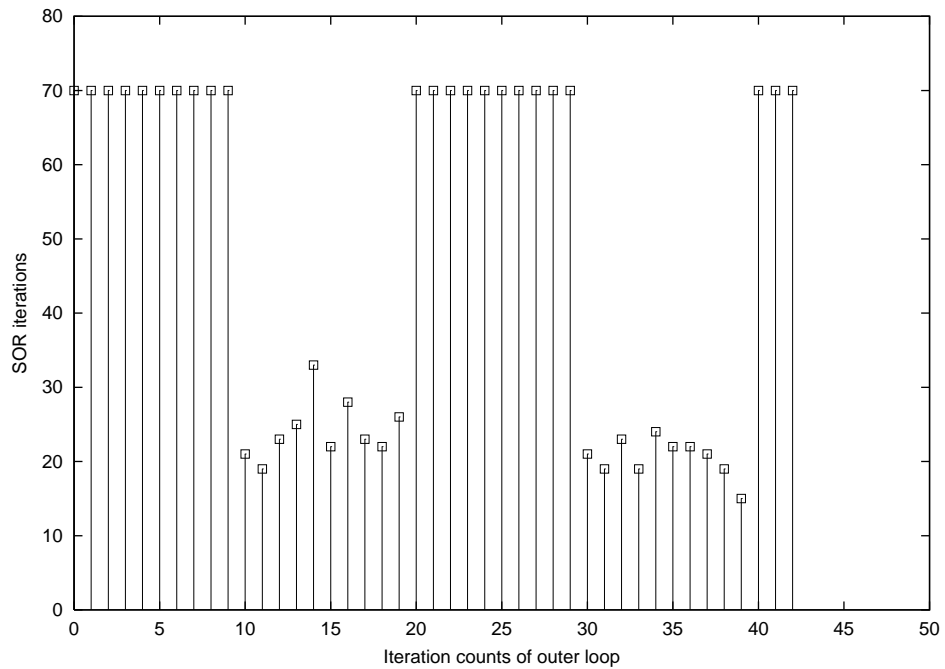


Fig. 6. The iteration counts of SOR method at each outer loop ($\sigma = 3.5$).

5 まとめ

われわれは，前処理行列 K が係数行列 A に近似的に等しいという性質に注目し， $K^{-1}v$ の代わりに $A^{-1}v$ の近似を用いる可変的前処理を提案した．この前処理では，方程式 $Az = v$ を任意の反復解法を用いて近似的に解く．このとき，精度と反復回数の両方を停止条件として設定する．すると，反復毎に反復回数は変わるため，反復毎に異なる前処理を適用できる．これは，一定回数の反復を行なう FGMRES 法や GMRESR 法とは異なったアプローチの可変的な前処理である．さらに，提案した可変的前処理では，外部反復および内部反復に様々な反復解法を組み合わせる利用できる．ここでは，内部反復に SOR 法を用いた可変的前処理を外部反復の GCR(m) 法に用いて，Helmholtz 方程式を離散化して得られた問題を解き，その有効性を示した．

数値実験から，SOR 法，GCR(m) 法を単独で適用した場合には収束しない，また ILU(0) 付き GCR(m) 法を適用した場合にはゆっくり収束する一方で，可変的前処理として SOR 法を用いた GCR(m) 法は急速に収束することがわかった．実際，SOR 法を用いる可変的前処理は ILU(0) を用いた場合と比べ，もっとも効果的な場合には外部反復回数が 1% 未満，計算時間が 3% になった．さらに，内部反復に用いた SOR 法の反復回数は各反復で異なっており，可変的な前処理が実現できている．

内部反復における停止条件，および SOR 法の加速係数が収束性に与える影響，外部反復，内部反復に用いる解法の組合せ，有効範囲の解析，さらに初期値，並列化の実装などについては今後の課題である．

謝辞

本論文を作成するにあたり有益な御助言を頂いた九州大学情報基盤センター 藤野清次 教授に心より御礼申し上げる．また，貴重な御指摘を頂いた査読者に深く感謝する．

参考文献

- [1] BAYLISS, A., GOLDSTEIN, C. and TURKEL, E., An Iterative Method for the Helmholtz Equations, *J. of Computational Physics*, **49** (1983), 443-457.
- [2] BRUASET, A. M., *A Survey of Preconditioned Iterative Methods*, Frontiers in Applied Mathematics 17, Longman Scientific and Technical, London, 1995.
- [3] EISENSTAT, S. C., ELMAN, H. C. and SCHULTZ, M. H., Variational iterative methods for nonsymmetric systems of linear equations, *SIAM J. Numer. Anal.*, **20** (1983), 345-357.
- [4] FLETCHER, R., Conjugate gradient methods for indefinite systems, in *Numerical Analysis Dundee 1975*, ed. by Watson, G., Lecture Notes in Mathematics, **506** (1976), Springer-Verlag, 73-89.
- [5] 藤野 清次, 張 紹良, 反復法の数理, 朝倉書店, 東京, 1996.

- [6] GOLUB, H. G. and VAN LOAN, F. C., *Matrix Computations*, Third ed., The Johns Hopkins University Press, Baltimore and London, 1996.
- [7] HESTENES, M. R. and STIEFEL, E., Methods of Conjugate Gradients for Solving Linear Systems, *J. Res. Nat. Bur. Standards*, **49** (1952), 409-435.
- [8] MEIJERINK, J. A. and VAN DER VORST, H. A., An Iterative Solution Method for Linear Systems of which the Coefficient Matrix is a Symmetric M-matrix, *Mathematics of Computation*, **31** (1977), 148-162.
- [9] SAAD, Y., A Flexible Inner-outer Preconditioned GMRES Algorithm, *SIAM J. Sci. Stat. Computing*, **14** (1993), 461-469.
- [10] SAAD, Y., *Iterative Methods for Sparse Linear Systems*, PWS, Boston, 1996.
- [11] SAAD, Y. and SCHULTZ, M. H., GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems, *SIAM J. Sci. Stat. Computing*, **7** (1986), 856-869.
- [12] VAN DER VORST, H. A. and VUIK, C., GMRESR: A family of Nested GMRES Methods, *Numer. Linear Algebra with Applica.*, **1** (1994), 369-386.
- [13] VARGA, R., *Matrix Iterative Analysis*, Prentice-Hall, New Jersey, 1962.

阿部 邦美 (正会員) 〒 351-0198 和光市広沢 2-1

1998年3月名古屋大学大学院人間情報学研究科博士後期課程満了。博士(学術)。国立阿南工業高等専門学校講師を経て、現在理化学研究所情報基盤研究部情報環境室協力研究員。線形計算に興味を持つ。情報処理学会, SIAM 各会員。

張 紹良 (正会員) 〒 113-8656 文京区本郷 7-3-1

1990年3月筑波大学大学院工学研究科博士課程修了。工学博士。東京大学大学院工学系研究科 助教授。大規模行列計算における反復解法の開発及び並列計算のアルゴリズムの研究に従事。情報処理学会, SIAM 各会員。

長谷川 秀彦 (正会員) 〒 305-8550 つくば市春日 1-2 (<http://www.ulis.ac.jp/~hasegawa/>)

1983年3月筑波大学大学院社会工学研究科中退。博士(工学)。図書館情報大学 助教授。線形数値計算とそのソフトウェアに興味を持つ。PHASE, LA 研究会 (<http://phase.etl.go.jp/>) を運営する。情報処理学会, SIAM, ACM 各会員。

姫野 龍太郎 (正会員) 〒 351-0198 和光市広沢 2-1

1979年3月京都大学大学院工学研究科博士前期課程修了。工学博士。日産自動車株式会社中央研究所, 総合研究所を経て、現在理化学研究所情報基盤研究部情報環境室 室長。流体の数値解析に従事。日本機械学会, 情報処理学会 各会員。