Vol.2010-HPC-126 No.37
2010/8/4

定常反復法型前処理つき GCR(k) 法の性能評価

関 本 幹[†1] 藤 野 清 次[†2]

大規模な連立一次方程式を Krylov 部分空間法を用いて解く場合，前処理を用いることにより解法の収束性は向上する．特に，ILU(0) 前処理はフィルインを考慮しないので非対称行列を持つ連立一次方程式の前処理としてよく利用される．しかし，ILU(0) 前処理は前処理行列生成コストが大きく，さらに，前進・後退代入計算が余分に必要になる．本論文では，前処理生成コストが極めて小さくかつ余分な前進・後退代入計算を必要としない定常反復法型前処理を GCR(k) 法に適用し，数値実験を通してその有効性を示す．

Convergence Estimation of GCR(k) Method with Stationary Type of Preconditioning

Takashi Sekimoto[†1] and Seiji Fujino[†2]

Preconditioning techniques improve convergence rate of Krylov subspace methods when we solve a large non-singular linear system of equations $A\boldsymbol{x} = \boldsymbol{b}$. ILU(0) decomposition preconditioning without extra fill-in is well known as a technique for linear systems with nonsymmetric coefficient matrix. However, this preconditioning doesn't sometimes work well due to high cost to build preconditioner matrix and extra cost of forward and backward substitutions. In this paper, we propose restarted GCR(k) method with stationary type of preconditioning. Through numerical experiments, we make clear that this preconditioning has excellent convergence rate.

†1 九州大学大学院システム情報科学府
Graduate School of Information Science and Electrical Engineering, Kyushu University
†2 九州大学情報基盤研究開発センター
Research Institute for Information Technology, Kyushu University

1. はじめに

解くべき連立一次方程式を

$$A\boldsymbol{x} = \boldsymbol{b} \tag{1}$$

とする．ただし，係数行列 A は大きさ $n \times n$ の非対称行列，$\boldsymbol{x}$ と $\boldsymbol{b}$ は次数 n の解ベクトルと右辺項ベクトルとする．このとき，初期残差ベクトルを $\boldsymbol{r}_0(= \boldsymbol{b} - A\boldsymbol{x})$ とし，m 次元の Krylov 部分空間

$$K_k(A; \boldsymbol{r}_0) = \mathrm{Span}\{\boldsymbol{r}_0, A\boldsymbol{r}_0, \ldots, A^{k-1}\boldsymbol{r}_0\} \tag{2}$$

において，正規直交基底を生成して近似解 $\boldsymbol{x}_k$ を求める手法を Krylov 部分空間法と呼ぶ．Krylov 部分空間法において，正規直交基底の計算に Arnoldi 原理を適用し，残差ベクトルに最小条件を課した解法に GCR(k) 法 (Generalized Conjugate Residual method，一般化共役残差法，k はリスタート周期を意味する) がある[2)]．

一方，前処理技法は Krylov 部分空間法の収束性を向上させる技法である．特に，ILU(0) 前処理は非対称行列を係数に持つ連立一次方程式における Krylov 部分空間法の収束性を向上させる前処理として広く知られている．しかし，ILU(0) 前処理では反復過程に入る準備段階での前処理行列生成コストが大きく，さらに，係数行列 A とベクトル $\boldsymbol{v}$ の積 $A\boldsymbol{v}$ と同程度の計算コストを必要とする前進・後退代入計算が余分に必要になる．その課題に対し，2010 年に尾上らにより，前処理行列生成コストを大幅に減らしかつ余分な前進・後退代入計算を必要としない定常反復法型前処理が提案された[1)4)5)]．本論文では，定常反復法型前処理を GCR(k) 法に適用し，その収束性を前処理なしの GCR(k) 法および ILU(0) 前処理つき GCR(k) 法と比較評価する．

本論文の構成は以下の通りである．第 2 節で，GCR(k) 法の概要を述べる．第 3 節で，定常反復法前処理つき GCR(k) 法の算法について述べ，さらに，計算量を削減した定常反復法型前処理つき GCR(k) 法の算法ついて述べる．第 3 節で，変種定常反復法型前処理つき GCR(k) 法の算法を記述する．第 5 節で，数値実験を通して定常および変種定常反復法型前処理つき GCR(k) 法の有効性を示す．第 6 節で，まとめと今後の課題について記述する．

2. GCR(k) 法

以下に，GCR(k) 法の算法を示す．ここで，k はリスタート周期を意味する．

Vol.2010-HPC-126 No.37
2010/8/4

GCR(k) 法の算法

$$
\begin{aligned}
&\text{Let } \boldsymbol{x}_0 \text{ be an initial guess,} && (3)\\
&\text{repeat,} && (4)\\
&\quad \text{set } \ \boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0, \boldsymbol{p}_0 = \boldsymbol{r}_0, \boldsymbol{q}_0 = A\boldsymbol{p}_0, && (5)\\
&\quad \text{for } \ m = 0, 1, \ldots, k-1, && (6)\\
&\qquad \alpha_m = \frac{(\boldsymbol{r}_m, \boldsymbol{q}_m)}{(\boldsymbol{q}_m, \boldsymbol{q}_m)}, && (7)\\
&\qquad \boldsymbol{x}_{m+1} = \boldsymbol{x}_m + \alpha_m \boldsymbol{p}_m, && (8)\\
&\qquad \boldsymbol{r}_{m+1} = \boldsymbol{r}_m - \alpha_m \boldsymbol{q}_m, && (9)\\
&\qquad \text{if } \ ||\boldsymbol{r}_{m+1}||_2 / ||\boldsymbol{r}_0||_2 \le \epsilon \text{ then stop,} && (10)\\
&\qquad \beta_{m,i} = -\frac{(A\boldsymbol{r}_{m+1}, \boldsymbol{q}_i)}{(\boldsymbol{q}_i, \boldsymbol{q}_i)}, \qquad i \le m, && (11)\\
&\qquad \boldsymbol{p}_{m+1} = \boldsymbol{r}_{m+1} + \sum_{i=0}^{m} \beta_{m,i} \boldsymbol{p}_i, && (12)\\
&\qquad \boldsymbol{q}_{m+1} = A\boldsymbol{r}_{m+1} + \sum_{i=0}^{m} \beta_{m,i} \boldsymbol{q}_i, && (13)\\
&\quad \text{end for,} && (14)\\
&\quad \boldsymbol{x}_0 = \boldsymbol{x}_k, && (15)\\
&\text{end repeat.} && (16)
\end{aligned}
$$

3. 定常反復法型前処理つき GCR(k) 法

3.1 素朴な定常反復法型前処理つき GCR(k) 法

本節では，素朴な定常反復法型の前処理について記述する[1)]．まず，係数行列 A を $A = L + D + U$ と分解する．ただし，L，D，U は狭義下三角行列，対角行列，狭義上三角行列を各々意味する．ここで，連立一次方程式 (1) を前処理を用いて解くことを考える．前処理行列を $K = K_1K_2$ とすると，式 (1) は

$$(K_1^{-1}AK_2^{-1})(K_2\boldsymbol{x}) = K_1^{-1}\boldsymbol{b}, \tag{17}$$

$$\tilde{A}\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{b}}. \tag{18}$$

と変形される．さらに，$K_1 = I$，$K_2 = K_2$ とおき，$K_2 = L + D$ とすると，$\tilde{A}$，$\tilde{\boldsymbol{x}}$，$\tilde{\boldsymbol{b}}$ は次のように表される．

$$
\begin{aligned}
\tilde{A} &= AK_2^{-1} = (L + D + U)(L + D)^{-1}\\
&= I - U(L + D)^{-1},
\end{aligned} \tag{19}
$$

表 1　前処理行列 M，N の設定
Table 1 Determination of preconditioner M, N.

method	M	N
Jacobi	D	$-L-U$
GS	$L+D$	$-U$
SOR	$L+\frac{1}{\omega}D$	$(\frac{1}{\omega}-1)D-U$

$$\tilde{\boldsymbol{x}} = K_2\boldsymbol{x} = (L + D)\boldsymbol{x}, \tag{20}$$

$$\tilde{\boldsymbol{b}} = \boldsymbol{b}. \tag{21}$$

また，式 (2) における反復 k 回目の残差ベクトルを $\tilde{\boldsymbol{r}}_k = \tilde{\boldsymbol{b}} - \tilde{A}\tilde{\boldsymbol{x}}_k$ と定義すると，

$$\tilde{\boldsymbol{r}}_k = \boldsymbol{r}_k \tag{22}$$

と表される．式 (18)∼(21) において，$M = (L + D)$，$N = -U$ とすると $A = M - N$ であることがわかる．すなわち，M，N は定常反復法である Jacobi 法，GS(Gauss-Seidel) 法，SOR(Successive Over Relaxation) 法の行列分離方式を用いることができる．**表 1** に前処理行列 M，N の設定を示す．ただし，**表 1** 中の ω は加速係数を意味する．

以下に，素朴な定常反復法型前処理つき GCR(k) 法の算法を示す．算法中の下線は GCR(k) 法と異なる更新式を意味する．

素朴な定常反復法型前処理つき GCR(k) 法の算法

$$
\begin{aligned}
&\text{Let } \boldsymbol{x}_0 \text{ be an initial guess,} && (23)\\
&\text{repeat,} && (24)\\
&\quad \text{set } \ \boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0, \ \ \boldsymbol{p}_0 = \boldsymbol{r}_0, && (25)\\
&\quad \text{Compute } M^{-1}\boldsymbol{p}_0 && (26)\\
&\quad \text{Compute } NM^{-1}\boldsymbol{p}_0 && (27)\\
&\quad \underline{\boldsymbol{q}_0 = \boldsymbol{p}_0 - NM^{-1}\boldsymbol{p}_0}, && (28)\\
&\quad \text{for } \ m = 0, 1, \ldots, k-1, && (29)\\
&\qquad \alpha_m = \frac{(\boldsymbol{r}_m, \boldsymbol{q}_m)}{(\boldsymbol{q}_m, \boldsymbol{q}_m)}, && (30)\\
&\qquad \text{Compute } M^{-1}\boldsymbol{p}_m && (31)\\
&\qquad \underline{\boldsymbol{x}_{m+1} = \boldsymbol{x}_m + \alpha_m M^{-1}\boldsymbol{p}_m}, && (32)\\
&\qquad \boldsymbol{r}_{m+1} = \boldsymbol{r}_m - \alpha_m \boldsymbol{q}_m, && (33)\\
&\qquad \text{if } \ ||\boldsymbol{r}_{m+1}||_2 / ||\boldsymbol{r}_0||_2 \le \epsilon \text{ then stop,} && (34)\\
&\qquad \text{Compute } M^{-1}\boldsymbol{r}_{m+1} && (35)\\
&\qquad \text{Compute } NM^{-1}\boldsymbol{r}_{m+1} && (36)
\end{aligned}
$$

Vol.2010-HPC-126 No.37
2010/8/4

$$\underline{\boldsymbol{v}_m = \boldsymbol{r}_{m+1} - NM^{-1}\boldsymbol{r}_{m+1}} \tag{37}$$

$$\beta_{m,i} = -\frac{(\boldsymbol{v}_m, \boldsymbol{q}_i)}{(\boldsymbol{q}_i, \boldsymbol{q}_i)}, \qquad i \le m, \tag{38}$$

$$\boldsymbol{p}_{m+1} = \boldsymbol{r}_{m+1} + \sum_{i=0}^{m} \beta_{m,i}\boldsymbol{p}_i, \tag{39}$$

$$\boldsymbol{q}_{m+1} = \boldsymbol{v}_m + \sum_{i=0}^{m} \beta_{m,i}\boldsymbol{q}_i, \tag{40}$$

end for, (41)

$\boldsymbol{x}_0 = \boldsymbol{x}_k$, (42)

end repeat. (43)

3.2 計算量を削減した定常反復法型前処理つき GCR(k) 法

定常反復法型前処理つき GCR(k) 法の計算量削減を考える．式 (31) は反復毎に計算されるが，$\tilde{\boldsymbol{x}} = M\boldsymbol{x}$ と置き換えることで削減することができる．ただし，リスタート毎に計算される初期残差ベクトル $\boldsymbol{r}_0$ は,

$$\begin{aligned} \boldsymbol{r}_0 &= \boldsymbol{b} - \tilde{A}\tilde{\boldsymbol{x}}_0 \\ &= \boldsymbol{b} - AM^{-1}\tilde{\boldsymbol{x}}_0 \\ &= \boldsymbol{b} - (M-N)M^{-1}\tilde{\boldsymbol{x}}_0 \\ &= \boldsymbol{b} - \tilde{\boldsymbol{x}}_0 + NM^{-1}\tilde{\boldsymbol{x}}_0 \end{aligned} \tag{44}$$

のように計算され，反復終了時には近似解ベクトルを $\boldsymbol{x} = M^{-1}\tilde{\boldsymbol{x}}$ と元に戻す必要がある．以下に，計算量を削減した定常反復法型前処理つき GCR(k) 法の算法を示す.

計算量を削減した定常反復法型前処理つき GCR(k) 法の算法

Let $\boldsymbol{x}_0$ be an initial guess, $\underline{\tilde{\boldsymbol{x}}_0 = M\boldsymbol{x}}$, (45)

repeat, (46)

Compute $M^{-1}\tilde{\boldsymbol{x}}_0$ (47)

Compute $NM^{-1}\tilde{\boldsymbol{x}}_0$ (48)

set $\underline{\boldsymbol{r}_0 = \boldsymbol{b} - \tilde{\boldsymbol{x}}_0 + NM^{-1}\tilde{\boldsymbol{x}}_0}$, $\boldsymbol{p}_0 = \boldsymbol{r}_0$, (49)

Compute $M^{-1}\boldsymbol{p}_0$ (50)

Compute $NM^{-1}\boldsymbol{p}_0$ (51)

$\underline{\boldsymbol{q}_0 = \boldsymbol{p}_0 - NM^{-1}\boldsymbol{p}_0}$ (52)

for $m = 0, 1, \ldots, k-1$, (53)

$$\alpha_m = \frac{(\boldsymbol{r}_m, \boldsymbol{q}_m)}{(\boldsymbol{q}_m, \boldsymbol{q}_m)}, \tag{54}$$

$$\underline{\tilde{\boldsymbol{x}}_{m+1} = \tilde{\boldsymbol{x}}_m + \alpha_m \boldsymbol{p}_m}, \tag{55}$$

$$\boldsymbol{r}_{m+1} = \boldsymbol{r}_m - \alpha_m \boldsymbol{q}_m, \tag{56}$$

if $||\boldsymbol{r}_{m+1}||_2/||\boldsymbol{r}_0||_2 \le \epsilon$ then stop, (57)

Compute $M^{-1}\boldsymbol{r}_{m+1}$ (58)

Compute $NM^{-1}\boldsymbol{r}_{m+1}$ (59)

$$\underline{\boldsymbol{v}_m = \boldsymbol{r}_{m+1} - NM^{-1}\boldsymbol{r}_{m+1}}, \tag{60}$$

$$\beta_{m,i} = -\frac{(\boldsymbol{v}_m, \boldsymbol{q}_i)}{(\boldsymbol{q}_i, \boldsymbol{q}_i)}, \qquad i \le m, \tag{61}$$

$$\boldsymbol{p}_{m+1} = \boldsymbol{r}_{m+1} + \sum_{i=0}^{m} \beta_{m,i}\boldsymbol{p}_i, \tag{62}$$

$$\boldsymbol{q}_{m+1} = \boldsymbol{v}_m + \sum_{i=0}^{m} \beta_{m,i}\boldsymbol{q}_i, \tag{63}$$

end for, (64)

$\tilde{\boldsymbol{x}_0} = \tilde{\boldsymbol{x}_k}$, (65)

end repeat. (66)

4. 変種定常反復法型前処理付き GCR(k) 法

本節では，変種定常反復法型の前処理について記述する[1)]．変種定常反復法型の前処理では K_1，K_2 を $A = M - N$ の行列分離を用いて以下のようにおく.

$$K_1 = I, \tag{67}$$

$$K_2 = (N - \frac{1}{\delta}D)M. \tag{68}$$

ただし，δ は実数係数である．ここで，式 (17) より $\tilde{A}$，$\tilde{\boldsymbol{x}}$，$\tilde{\boldsymbol{b}}$，$\tilde{\boldsymbol{r}}$ は次のように表される.

$$\tilde{A} = (N - \frac{1}{\delta}D)^{-1} - NM^{-1}(N - \frac{1}{\delta}D)^{-1}, \tag{69}$$

$$\tilde{\boldsymbol{x}} = (N - \frac{1}{\delta}D)M\boldsymbol{x}, \tag{70}$$

$$\tilde{\boldsymbol{b}} = \boldsymbol{b}, \tag{71}$$

$$\tilde{\boldsymbol{r}} = \boldsymbol{r}. \tag{72}$$

さらに，第 3.2 節と同様に計算量を削減した変種定常反復法型前処理つき GCR(k) 法を考える．初期残差ベクトル $\boldsymbol{r}_0$ は,

$$\begin{aligned}\boldsymbol{r}_0 &= \boldsymbol{b} - \tilde{A}\tilde{\boldsymbol{x}}_0\\ &= \boldsymbol{b} - AM^{-1}(N - \frac{1}{\delta}D)^{-1}\tilde{\boldsymbol{x}}_0\\ &= \boldsymbol{b} - (M-N)M^{-1}(N - \frac{1}{\delta}D)^{-1}\tilde{\boldsymbol{x}}_0\\ &= \boldsymbol{b} - (N - \frac{1}{\delta}D)^{-1}\tilde{\boldsymbol{x}}_0 + NM^{-1}(N - \frac{1}{\delta}D)^{-1}\tilde{\boldsymbol{x}}_0 \end{aligned} \tag{73}$$

となり，反復終了時には近似解ベクトルを $\boldsymbol{x} = M^{-1}(N - \frac{1}{\delta}D)^{-1}\tilde{\boldsymbol{x}}$ と元に戻す．以下に，計算量を削減した変種定常反復法型前処理つき GCR(k) 法の算法を示す．

計算量を削減した変種定常反復法型前処理つき GCR(k) 法の算法

$$\begin{aligned}
&\text{Let } \boldsymbol{x}_0 \text{ be an initial guess, } \underline{\tilde{\boldsymbol{x}}_0 = (N - \tfrac{1}{\delta}D)M\boldsymbol{x},} && (74)\\
&\text{repeat,} && (75)\\
&\quad \text{Compute } (N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{x}_0 && (76)\\
&\quad \text{Compute } M^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{x}_0 && (77)\\
&\quad \text{Compute } NM^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{x}_0 && (78)\\
&\quad \text{set } \underline{\boldsymbol{r}_0 = \boldsymbol{b} - (N - \tfrac{1}{\delta}D)^{-1}\tilde{\boldsymbol{x}}_0 + NM^{-1}(N - \tfrac{1}{\delta}D)^{-1}\tilde{\boldsymbol{x}}_0,}\ \ \boldsymbol{p}_0 = \boldsymbol{r}_0, && (79)\\
&\quad \text{Compute } (N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{p}_0 && (80)\\
&\quad \text{Compute } M^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{p}_0 && (81)\\
&\quad \text{Compute } NM^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{p}_0 && (82)\\
&\quad \underline{\boldsymbol{q}_0 = (N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{p}_0 - NM^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{p}_0} && (83)\\
&\quad \text{for } \ m = 0, 1, \ldots, k-1, && (84)\\
&\qquad \alpha_m = \frac{(\boldsymbol{r}_m, \boldsymbol{q}_m)}{(\boldsymbol{q}_m, \boldsymbol{q}_m)}, && (85)\\
&\qquad \underline{\tilde{\boldsymbol{x}}_{m+1} = \tilde{\boldsymbol{x}}_m + \alpha_m \boldsymbol{p}_m,} && (86)\\
&\qquad \boldsymbol{r}_{m+1} = \boldsymbol{r}_m - \alpha_m \boldsymbol{q}_m, && (87)\\
&\qquad \text{if } \ ||\boldsymbol{r}_{m+1}||_2/||\boldsymbol{r}_0||_2 \le \epsilon \text{ then stop,} && (88)\\
&\qquad \text{Compute } (N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{r}_{m+1} && (89)\\
&\qquad \text{Compute } M^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{r}_{m+1} && (90)\\
&\qquad \text{Compute } NM^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{r}_{m+1} && (91)\\
&\qquad \underline{\boldsymbol{v}_m = (N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{r}_{m+1} - NM^{-1}(N - \tfrac{1}{\delta}D)^{-1}\boldsymbol{r}_{m+1}} && (92)\\
&\qquad \beta_{m,i} = -\frac{(\boldsymbol{v}_m, \boldsymbol{q}_i)}{(\boldsymbol{q}_i, \boldsymbol{q}_i)}, \qquad i \le m, && (93)\\
&\qquad \boldsymbol{p}_{m+1} = \boldsymbol{r}_{m+1} + \sum_{i=0}^{m} \beta_{m,i}\boldsymbol{p}_i, && (94)\\
&\qquad \boldsymbol{q}_{m+1} = \boldsymbol{v}_m + \sum_{i=0}^{m} \beta_{m,i}\boldsymbol{q}_i, && (95)\\
&\quad \text{end for,} && (96)\\
&\quad \tilde{\boldsymbol{x}}_0 = \tilde{\boldsymbol{x}}_k, && (97)\\
&\text{end repeat.} && (98)
\end{aligned}$$

5. 数値実験

5.1 計算機環境と計算条件

計算はすべて倍精度浮動小数点演算で行った．計算機は Nehalem（CPU：Intel Xenon X5570，クロック周波数：2.93GHz，メモリ：24Gbytes，OS：RedHat Enterprise Linux 5.2）を用いた．プログラムは Fortran90 を用いて実装し，最適化オプションは-O3 を使用した．右辺項ベクトル $\boldsymbol{b}$ は物理的条件から得られる値を用いた．収束判定条件は相対残差の 2 ノルム：$\| \boldsymbol{r}_{k+1} \|_2 / \| \boldsymbol{r}_0 \|_2 \le 10^{-8}$ とした．初期近似解 $\boldsymbol{x}_0$ はすべて 0 とした．行列は予め対角スケーリングによって対角項をすべて 1.0 に正規化した．最大反復回数は 10000 回とした．リスタート周期 k は 20，50，100，200，500，1000，2000，5000 の 8 通りとした．定常反復法型前処理の行列 M，N は SOR 法の行列分離を用いた．加速係数 ω は 0.5 から 1.9 までの 0.1 刻みで計 15 通りとした．変種定常反復法型前処理の行列 M，N は GS 法の行列分離を用い，V(Variant)-GS と表記する．実数係数 δ は 0.5 から 1.9 までの 0.1 刻みで計 15 通りとした．

5.2 テスト問題

表 2 に本実験で使用する 10 種類の非対称行列の特徴を示す．これらの行列はフロリダ大学の疎行列データベースから選出した[3]．

5.3 実験結果

表 3-4 に 3 種類の前処理を用いた GCR(k) 法の収束性を示す．**表 5** に前処理付き GCR(k)

表 2 10 種類の非対称行列の特徴
Table 2 Specifications of ten kinds of nonsymmetric coefficient matrices.

matrix	dimension	nnz	ave. nnz	analytical field
big	13,209	91,465	6.92	structural
bridge	64,461	4,373,817	67.85	
poisson3Da	13,514	352,762	26.10	
poisson3Db	85,623	2,374,949	27.74	
xenon1	48,600	1,181,120	24.30	
xenon2	157,464	3,866,688	24.56	
ex11	16,614	1,096,948	66.03	fluid-dynamics
ex19	16,614	1,096,948	66.03	
raefsky2	3,242	294,276	90.77	
raefsky3	21,200	1,488,768	70.22	

法と前処理無し GCR(k) 法の計算時間比較，**表 6** に定常反復法型，変種定常反復法型とILU(0) 前処理の計算時間比較を示す．

表中の時間の単位は，“ ave.itr-t ”ではミリ秒とし，それ以外はすべて秒とした．表中の定常反復法型前処理を用いた GCR(k) 法は $\omega = 1.0$ および最適な ω のみ示す．また，変種定常反復法型前処理を用いた GCR(k) 法は $\delta = 1.0$ および最適な δ のみ示す．表中の“ max ”は最大反復回数まで達したことを意味する．表中の“ k ”,“ ω ”,“ δ ”はリスタート周期，加速係数，実数係数を各々意味する．表中の“ itr.(iteration) ”,“ pre-t ”,“ itr-t ”,“ tot-t ”,“ ave.(average)itr-t ”は反復回数，前処理行列生成時間，反復計算に要した時間，合計時間，平均反復時間を各々意味する．表中の“ TRR(True Relative Residual) ”は真の相対残差の常用対数値である $\log_{10}\left(\frac{\|\boldsymbol{b}-A\boldsymbol{x}_{m+1}\|_2}{\|\boldsymbol{b}-A\boldsymbol{x}_0\|_2}\right)$ また，表中の“ ratio ”は前処理付き積型反復法の前処理無し GCR(k) 法に対する比を意味する．

表 3-6 の観察から以下の知見を得られる．

- 定常反復法型前処理について，ω の値を増加，または減少させると収束性が優れるケースがある．
- 変種定常反復法型前処理について，δ の値を増加，または減少させると収束性が優れるケースがある．
- 10 種類のテスト行列において，最も速く収束したケースが多い前処理は変種定常反復法型の 7 ケースである．
- 前処理なし GCR(k) 法よりも 2 倍以上計算時間が短くなったケースは，ILU(0) 前処理で 4 ケース，定常反復法型前処理で 10 ケース，変種定常反復法型前処理で 10 ケースである．

6. まとめ

本論文では定常反復法型前処理および変種定常反復法型前処理を GCR(k) 法に適用し，さらに，計算量の削減を行った．その結果，前処理なし GCR(k) 法よりも収束性が優れ，ILU(0) 前処理つき GCR(k) 法では収束しなかった行列に対しても収束することがわかった．今後の課題は，定常反復法型前処理が有効である行列を事前に知ることができるような指標を考察することである．

参考文献

1) Y.Onoue, S.Fujino : A Proposal of GS(Gauss-Seidel)-based Preconditioner without Extra Forward and Backward Substitutions, 2nd International Kyoto-Forum on Krylov Subspace method, pp.29-36, 2010.
2) Stanley C. Eisenstat, Howard C. Elman, Martin H. Schultz : Variational Iterative Methods for Nonsymmetric Systems of Linear Equations, SIAM Journal on Numerical Analysis, Vol. 20, No.2, pp.345-357, 1983.
3) University of Florida Sparse Matrix Collection: http://www.cise.ufl.edu/research/sparse/matrices/index.html
4) Y.Saad : Iterative Methods for Sparse Linear Systems, 2nd Edition, SIAM, Philadelphia, 2003.
5) D.Young : Iterative methods for solving partial difference equations of ellipic type, Doctoral Thesis, Harvard University, 1950.

表 3 3 種類の前処理を用いた GCR(k) 法の収束性
Table 3 Convergence of three kinds of preconditioned GCR(k) methods.

matrix	pre-cond.	k	ω	δ	itr.	pre-t	itr-t	tot-t	ave. itr-t	TRR	ratio
big	-	20	-	-	max	-	-	-	-	-5.36	-
	ILU(0)	5000	-	-	253	0.01	1.36	**1.37**	5.38	-8.06	-
	SOR	500	1.0	-	413	0.00	3.36	3.36	8.14	-8.03	-
		1000	1.9	-	320	0.00	2.04	2.04	6.38	-8.04	-
	V-GS	2000		1.0	286	0.00	1.73	1.73	6.05	-8.08	-
		5000	-	1.7	272	0.00	1.54	1.54	5.67	-8.05	-
bridge	-	500	-	-	1926	0.00	104.07	104.07	54.03	-10.57	1.00
	ILU(0)	20	-	-	max	-	-	-	-	0.00	-
	SOR	100	1.0	-	2563	0.03	46.77	46.80	18.25	-10.62	0.45
		20	0.5	-	1310	0.03	14.59	14.62	11.14	-10.69	0.14
	V-GS	20		1.0	717	0.04	11.33	11.37	15.80	-10.73	0.11
		20	-	0.9	687	0.04	10.84	**10.88**	15.78	-10.50	0.10
poisson3Da	-	20	-	-	285	0.00	0.34	0.34	1.20	-8.08	1.00
	ILU(0)	20	-	-	57	0.05	0.13	0.18	2.30	-8.07	0.53
	SOR	20	1.0	-	99	0.00	0.13	0.13	1.31	-8.09	0.38
		20	1.1		89	0.00	0.12	**0.12**	1.35	-8.09	0.35
	V-GS	20		1.0	74	0.00	0.14	0.14	1.91	-8.12	0.41
		20	-	1.1	72	0.00	0.12	**0.12**	1.68	-8.10	0.35
poisson3Db	-	50	-	-	582	0.00	8.34	8.34	14.33	-8.05	1.00
	ILU(0)	100	-	-	110	0.56	3.12	3.68	28.35	-8.09	0.44
	SOR	20	1.0	-	204	0.03	2.24	2.27	10.98	-8.11	0.27
		20	0.9	-	202	0.03	2.20	2.23	10.90	-8.09	0.27
	V-GS	20		1.0	137	0.03	2.14	2.17	15.63	-8.12	0.26
		20	-	1.1	136	0.02	2.14	**2.16**	15.74	-8.15	0.26
xenon1	-	100	-	-	1956	0.00	19.05	19.05	9.74	-8.00	1.00
	ILU(0)	20	-	-	max	-	-	-	-	-2.24	-
	SOR	20	1.0	-	673	0.00	3.16	3.16	4.70	-8.00	0.17
		20	0.8	-	669	0.01	3.13	3.14	4.69	-	-
	V-GS	20	-	1.0	477	0.01	2.90	2.91	6.08	-8.02	0.15
		20	-	0.9	450	0.00	2.73	**2.73**	6.07	-8.01	0.14
xenon2	-	200	-	-	1514	0.00	81.46	81.46	53.81	-7.99	1.00
	ILU(0)	20	-	-	max	-	-	-	-	-2.22	-
	SOR	20	1.0	-	768	0.04	12.43	12.47	16.18	-8.01	0.15
		-	-	-	-	-	-	-	-	-	-
	V-GS	20	-	1.0	554	0.03	11.89	11.92	21.46	-8.01	0.15
		20	-	1.1	555	0.04	11.42	**11.46**	20.57	-8.01	0.14

表 4 3 種類の前処理を用いた GCR(k) 法の収束性（続き）
Table 4 Convergence of three kinds of preconditioned GCR(k) methods. (cont'd)

matrix	pre-cond.	k	ω	δ	itr.	pre-t	itr-t	tot-t	ave. itr-t	TRR	ratio
ex11	-	20	-	-	max	-	-	-	-	-7.08	-
	ILU(0)	20	-	-	max	-	-	-	-	0.00	-
	SOR	20	1.0	-	max	-	-	-	-	-7.22	-
		50	0.5	-	8299	0.01	28.93	**28.94**	3.49	-8.40	-
	V-GS	1000	-	1.0	max	-	-	-	-	-7.33	-
		20	-	0.5	8524	0.00	32.92	32.92	3.86	-8.46	-
ex19	-	20	-	-	max	-	-	-	-	-7.16	-
	ILU(0)	20	-	-	max	-	-	-	-	0.00	-
	SOR	20	1.0	-	max	-	-	-	-	-7.83	-
		200	0.8	-	7238	0.00	29.25	29.25	4.04	-8.15	-
	V-GS	20		1.0	10500	0.00	11.43	11.43	1.09	-7.93	-
		20	-	0.5	6253	0.00	6.81	**6.81**	1.09	-8.18	-
raefsky2	-	2000	-	-	337	0.00	0.61	0.61	1.81	-8.01	1.00
	ILU(0)	100	-	-	40	0.05	0.04	0.09	0.98	-8.60	0.15
	SOR	200	1.0	-	186	0.00	0.20	0.20	1.08	-8.12	0.33
		-	-	-	-	-	-	-	-	-	-
	V-GS	1000		1.0	70	0.00	0.06	**0.06**	0.84	-8.27	0.10
		-	-	-	-	-	-	-	-	-	-
raefsky3	-	50	-	-	8826	0.00	40.32	40.32	4.57	-9.10	1.00
	ILU(0)	5000	-	-	75	0.19	0.63	**0.82**	8.41	-8.85	0.02
	SOR	100	1.0	-	6239	0.01	38.43	38.44	6.16	-9.35	0.95
		20	0.6	-	5140	0.01	19.16	19.17	3.73	-9.42	0.48
	V-GS	200	-	1.0	2205	0.01	22.94	22.95	10.40	-9.35	0.57
		50	-	0.7	1893	0.02	11.42	11.44	6.03	-9.55	0.28

表 5 前処理付き GCR(k) 法と前処理無し GCR(k) 法の計算時間比較
Table 5 Comparison of total time of preconditioned GCR(k) methods to that of GCR(k) method.

ratio	∼0.25	0.25∼0.50	0.50∼0.75	0.75∼1.00	1.0∼	not conv.
ILU(0)	3	1	1	0	0	5
SOR	6	4	0	0	0	0
V-GS	7	3	0	0	0	0

表 6 定常反復法型，変種定常反復法型と ILU(0) 前処理の計算時間比較
Table 6 Comparison of total time of GCR(k) methods with stationary type and variant stationary type of preconditioning to that of ILU(0)-GCR(k) method.

ratio	∼0.25	0.25∼0.50	0.50∼0.75	0.75∼1.00	1.0∼	not conv.
SOR	5	0	2	0	3	0
V-GS	5	0	3	0	2	0