

## ORTHOMIN( $k$ ) 法に対する適応的リスタート

津野直人<sup>†</sup> 森屋健太郎<sup>†</sup> 野寺隆<sup>†</sup>

一般共役残差法の打ち切り版である ORTHOMIN( $k$ ) 法は、大規模で疎な非対称行列を係数とする連立1次方程式  $Ax = b$  の反復解法として広く利用されている。ORTHOMIN( $k$ ) 法の収束を速めるためには、リスタート技法を用いるのだが、そのタイミングを見つけるのはそれほど簡単なことではない。本稿では、この算法のリスタートのタイミングを自動的に決定する適応的リスタート技法を提案する。最後に、これらの算法を分散メモリ型の並列計算機 AP1000 に実装し、その数値実験の結果について述べる。

### The Adaptive Restarted Procedure for ORTHOMIN( $k$ ) Method

NAOTO TSUNO,<sup>†</sup> KENTARO MORIYA<sup>†</sup> and TAKASHI NODERA<sup>†</sup>

The ORTHOMIN( $k$ ) method, a truncated version of GCR (generalized conjugate residual) method, has been widely used for solving large and sparse nonsymmetric linear systems of equations  $Ax = b$ . In order to accelerate the convergence of ORTHOMIN( $k$ ) method, we usually use the restarting technique. But, it is not so easy to find out the restarting timing of its algorithm. In this paper, we will propose the adaptive restarted procedure which will detect the restarting timing of the ORTHOMIN( $k$ ) automatically. At last, numerical experiments are given to show the effectiveness of the adaptive restarted procedure combined with ORTHOMIN( $k$ ) method on a distributed memory parallel machine AP1000.

#### 1. はじめに

偏微分方程式の境界値問題を有限差分法や有限要素法で離散化した場合など、大規模で疎 (sparse) な  $n \times n$  次の係数行列  $A$  をもつ連立1次方程式

$$Ax = b \quad (1)$$

は理工学のあらゆる分野に現れる。

連立1次方程式 (1) に対する反復法は、任意の初期近似解から始めて厳密解へと近づく近似解の列を構成し、適切な時点で十分に精度の良い近似解を計算する方法である。代表的な非定常反復法として、特に式 (1) の係数行列が対称の場合には、CG法<sup>1)</sup> (conjugate gradient method, 共役勾配法ともいう) を挙げることができる。一方、係数行列が非対称な場合には、未だ決定的な方法が提案されておらず、さまざまな算法の研究開発が行われている。また、係数行列の前処理という既存の算法の収束性を加速させる技法についても活発な研究が行われている<sup>4)</sup>。算法の収束性を加速するもう1つの手法は、反復のある時点で新しい残差ベクトルを計算しなおし、算法をリスタートすることである<sup>5)</sup>。リスタートのタイミングを決定することはそれほど簡単なことではないが、近年、疑似残差法に

関してリスタートのタイミングを自動的に決定する手法が提案されている<sup>6)</sup>。

本稿では、係数行列が非対称の問題に対する反復法として ORTHOMIN( $k$ ) 法を取りあげる。ORTHOMIN( $k$ ) 法は非対称係数行列の問題を解く GCR法<sup>3)</sup> (generalized conjugate residual method, 一般共役残差法ともいう) のクラスに属する算法である。本稿では、この ORTHOMIN( $k$ ) 法の算法のある状況においてリスタートさせることにより、その収束を加速させることを考える。

#### 2. ORTHOMIN( $k$ ) 法

正則で非対称な係数行列をもつ連立1次方程式 (1) を解く非定常反復法の1つに GCR法<sup>3)</sup> がある。通常この方法は Krylov 部分空間法と呼ばれる反復法のクラスに分類されている。つまり、行列  $M$  とベクトル  $v$  による第  $i$  次の Krylov 部分空間を

$$\mathcal{K}_i(M; v) := \text{span}\{v, Mv, M^2v, \dots, M^{i-1}v\}$$

で表すことにすると、GCR法が生成する近似解の列  $\{x_i\}$  は

$$x_i \in x_0 + \mathcal{K}_i(A; r_0)$$

を満たすように構成されることになる。ここで  $r_0$  は初期近似解  $x_0$  に対応する残差ベクトル ( $r_i := b - Ax_i$ )

<sup>†</sup> 慶應義塾大学理工学部

Faculty of Science and Technology, Keio University

```

1: Choose  $x_0$ .
2:  $r_0 = b - Ax_0$ 
3: for  $i = 0, 1, 2, \dots$ 
  3.1: if  $i = 0$  then
    3.1.1:  $p_0 = r_0$ 
    else
    3.1.2: for  $j = \sigma, \sigma + 1, \dots, i - 1$ 
      3.1.2.1:  $\beta_i^{(j)} = -\frac{(Ar_i, Ap_j)}{(Ap_j, Ap_j)}$ 
    endfor
    3.1.3:  $p_i = r_i + \sum_{j=\sigma}^{i-1} \beta_i^{(j)} p_j$ 
  endif
  3.2:  $\alpha_i = \frac{(r_i, Ap_i)}{(Ap_i, Ap_i)}$ 
  3.3:  $x_{i+1} = x_i + \alpha_i p_i$ 
  3.4:  $r_{i+1} = r_i - \alpha_i Ap_i$ 
  3.5: If converge, escape the loop.
endifor
※ ただし  $\sigma = \max\{0, i - k\}$ 

```

図1 ORTHOMIN(k)法の算法

である。さらに、GCR法の方向ベクトルと残差ベクトルは

$$\begin{aligned}
 p_i A^T A p_j &= 0 \quad (i \neq j) \\
 \|r_{i+1}\| &= \min_{\alpha} \|b - A(x_i + \alpha p_i)\|
 \end{aligned}$$

を常に満たすように計算される。

GCR法を厳密に実行することを考えると、方向ベクトルの計算でそれ以前のすべての方向ベクトルとの $A^T A$ -直交化を行う必要がある。このことから大規模な問題を扱うとき、GCR法の厳密な実行は計算時間とメモリ容量の点から事実上不可能となる。そこでGCR法の変形版として、新しい方向ベクトルを直前の $k$ 個の方向ベクトルとのみ $A^T A$ -直交させるORTHOMIN(k)法<sup>2)</sup>が提案されている。図1にORTHOMIN(k)法の算法を示す。

### 3. ORTHOMIN(k)法の収束に与えるリストार्टの影響

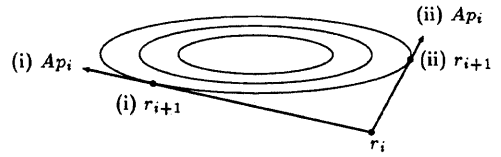
ORTHOMIN(k)法の残差ベクトルを計算する漸化式は

$$\begin{aligned}
 r_{i+1} &= r_i - \alpha_i Ap_i \\
 &= r_i - \alpha_i' \frac{Ap_i}{\|Ap_i\|}
 \end{aligned} \quad (2)$$

で与えられている。この節では、残差ノルムの挙動と漸化式に現れるスカラー $\alpha_i'$ の関係、および収束に与えるリストार्टの影響について考える。

#### 3.1 ORTHOMIN(k)法の方向ベクトル

残差ノルムが大きく減少するとき $|\alpha_i'|$ の値も大きくなるのは当然である。一方、残差ノルムの減少がわず



(i)  $|\alpha_i'|$ の値が大きい場合  
(ii)  $|\alpha_i'|$ の値が小さい場合  
図2 方向ベクトルに沿って進む距離の違い

かな場合には、図2に示すように $|\alpha_i'|$ が大きな値をとるか小さな値をとるかは定まらない。図2において同心状に描かれている楕円は残差ノルムの等しい点をつないだものである。例えば(i)の方向では(ii)の方向に沿って進む距離の約2倍の距離を進んでいる。つまり、残差ノルムの減少がわずかな場合でも、 $|\alpha_i'|$ の値によって状況は異なることになる。

いま第 $i$ 回目の反復を考える。このとき記憶している過去に用いた $k$ 個の方向ベクトルを $p_{i-1}, p_{i-2}, \dots, p_{i-k}$ とし、残差ベクトルを

$$r_i = v + \gamma_{i-1} p_{i-1} + \gamma_{i-2} p_{i-2} + \dots + \gamma_{i-k} p_{i-k} \quad (3)$$

と書くことにする。ただし、 $v$ は $p_{i-1}, p_{i-2}, \dots, p_{i-k}$ と $A^T A$ -直交するベクトルである。このとき、新しい方向ベクトルは式(3)を記憶している方向ベクトルと $A^T A$ -直交させて得られるので

$p_i = v$ となる。よって第 $i+1$ 回目の残差ベクトルは漸化式(2)により

$$\begin{aligned}
 r_{i+1} &= r_i - \alpha_i' \frac{Ap_i}{\|Ap_i\|} \\
 &= (v + \gamma_{i-1} p_{i-1} + \gamma_{i-2} p_{i-2} + \dots \\
 &\quad \dots + \gamma_{i-k} p_{i-k}) \\
 &\quad - \alpha_i' (v' + \gamma_i' v + \gamma_{i-1}' p_{i-1} \\
 &\quad + \gamma_{i-2}' p_{i-2} + \dots + \gamma_{i-k}' p_{i-k})
 \end{aligned} \quad (4)$$

と計算される。ただし、 $v'$ は $v, p_{i-1}, p_{i-2}, \dots, p_{i-k}$ と $A^T A$ -直交するベクトルである。さらに次の方向ベクトルを考えると、今度は式(4)と $p_i, p_{i-1}, \dots, p_{i-k+1}$ との $A^T A$ -直交化から

$p_{i+1} = \gamma_{i-k} p_{i-k} - \alpha_i' (v' + \gamma_{i-k}' p_{i-k})$ となる。したがって、 $|\alpha_i'|$ の値が小さいときには、新しい方向の中で過去に最小化した方向 $p_{i-k}$ が大きな割合を占めることになる。

新しく進む方向の中で過去に最小化した方向が大きな割合を占めるということは、その方向に進む距離が引続き小さいままであると考えられる。つまり、これ以後の反復においても同様に、選ばれる方向ベクトルの中で過去に選んだ方向ベクトルが大きな割合を占めるといった現象が続くことになり、残差ノルムの停滞がしばらく続くことと予想される。一方、 $|\alpha_i'|$ の値がある程度大きいときには、過去に用いた方向ベクトルと $A^T A$ -直交する方向 $v'$ が新しい方向ベクトルの中で

それなりの割合を占めることになる。したがって、たとえ残差ノルムの減少がわずかな場合でも過去に選んだ方向ベクトルとの  $A^T A$ -直交性は高くなる。つまり、これ以後の反復で残差ノルムが減少する可能性に期待することができる。

### 3.2 リスタートの影響

ここで、第  $i$  回目の反復の前にリスタートした場合を考える。このときの方向ベクトルは  $A^T A$ -直交化が行われないため、残差ベクトルをそのまま用いて

$$p_i = r_i$$

である。この方向の中で過去に最小化した方向は  $p_{i-1}, p_{i-2}, \dots, p_0$  である。リスタートしない場合と比べると、 $A^T A$ -直交化をしない分、これらの方向が  $p_i$  に占める割合は大きくなる。したがって  $|\alpha'_i|$  の値はリスタートしない場合よりも小さくなる。さらに次の反復を考えると、残差ベクトルは漸化式 (2) により

$$\begin{aligned} r_{i+1} &= r_i - \alpha'_i \frac{A p_i}{\|A p_i\|} \\ &= r_i - \alpha'_i (v' + \gamma_i r_i) \end{aligned} \quad (5)$$

と計算される。ただし、 $v'$  は  $r_i$  と  $A^T A$ -直交するベクトルである。このときの方向ベクトル  $p_{i+1}$  は式 (5) を  $p_i$  と  $A^T A$ -直交化させて得られるので

$$p_{i+1} = -\alpha'_i v'$$

となる。もちろんこの方向には過去に最小化した方向も含まれる。しかし、その割合は行列  $A$  とそのときの方向ベクトル (つまり残差ベクトル)  $p_i$  に依存する。つまり、リスタートを行うことで方向ベクトルに予測のつかない変化を与えることになる。

以上の事柄に基づき、収束に与えるリスタートの影響について考えてみる。残差ノルムが収束に向かっているときにリスタートを行うと、 $A^T A$ -直交の情報とは関係なく新しい方向を決定するので、場合によっては収束を鈍らせることも起こりうる。この場合、リスタートは行わずに  $A^T A$ -直交の情報を利用した方が確実である。一方、収束が停滞しているときには、前述のように新しい方向ベクトルの中で過去に選んだ方向ベクトルが大きな割合を占めるといった現象が起こる。このような状況においてはリスタートを行うことで選ばれる方向ベクトルに変化を与え、これが収束を速めるきっかけを作る可能性がある。

リスタートの効果的な使い方を考えると、「収束が停滞しているときにのみリスタートの実行を考慮する」というのが 1 つの理想といえる。この考えは後述の適応的リスタートと呼ばれる技法の基本概念と一致する。

## 4. ORTHOMIN( $k$ ) 法のための適応的リスタート

適応的リスタート (adaptive restarted procedure) はもともと稲津・野寺<sup>6)</sup>により ORTHORES 法に導入された技法である。ORTHOMIN( $k$ ) 法は、残差ノル

ムを最小化するという ORTHORES 法にはない特徴をもち、近似解や残差を構成する漸化式も ORTHORES 法とは異なる。そこで我々は ORTHORES 法とは別の観点にたち、ORTHOMIN( $k$ ) 法における適切なリスタートの実行について考えることにする。

前節で述べたように、ORTHOMIN( $k$ ) 法の収束が進んでいるかどうかは、実際の残差ノルムの減少量ではなく方向ベクトルに沿って進んだ距離によって判断することが適切である。そこで、この距離に基づいてリスタートのタイミングを考えることにする。反復全体を通して考えるため、残差ノルムに対する相対量を取り、さらに算法中でベクトルが正規化されていないことも考慮して次のようなタイミングでリスタートの実行を考えることにする。

### (1) リスタートのタイミングに関するルール

$\|\alpha_i A p_i\| / \|r_i\|$  の値が、連続する  $k$  回の反復を通じてあらかじめ与えられたパラメータ  $\epsilon$  よりも小さいときに、大幅な収束は期待できないと判断し、リスタートの実行を考える。

パラメータ  $\epsilon$  の値は、我々の数値実験から得られた経験では 0.1 前後が有効である。

リスタートに対して期待することは、その後の反復で方向ベクトルに沿って進む距離が長くなることである。もしリスタートを行ってもこの距離が短いままなら、収束が速まる可能性は相変わらず低く、さらに  $A^T A$ -直交性も失われてしまうことからリスタートは行わない方がよい。しかしこの距離をあらかじめ知るためには、それに相当する反復を行うのと同じ計算コストがかかり効率的とは言いがたい。そこで、1 回目の適応的リスタートの判断では無条件にリスタートを行い、2 回目以後の適応的リスタートの判断では、それ以前のリスタートの結果を参考にしてリスタートを行うか行わないかを決定することにする。

### (2) リスタートの実行に関するルール

- (a) 1 回目のリスタートの判断では無条件にリスタートを行う。
- (b) リスタートを行った際は、その直前の  $k$  反復の中で進んだ距離の最大値と、リスタート直後の  $k$  反復の中で進んだ距離の最大値を比較し、リスタートの有効性を検証する。リスタートをした後に進んだ距離の最大値の方が大きいときには、「リスタートが有効」とあると考える。
- (c) 2 回目以後のリスタートの判断では、それ以前の結果から「リスタートが有効」とされているときにのみリスタートを行う。
- (d) (1) の条件が満たされなかったとき、方向ベクトルが大きく変化しており、それ以前のリスタートの有効性についての検証は意味をなさなくなる。したがって、次のリスタートの判断では無条件にリスタートを行う。

```

1: Choose  $\epsilon$ .
2:  $N = 0$ 
3: Set adapt_restart := on. ..... (2-a)
4: Do one iteration of ORTHOMIN( $k$ ) method.
5:  $\alpha'_i = \|\alpha_i A p_i\|$ 
6: if  $\alpha'_i / \|r_i\| < \epsilon$  then
    6.1:  $N = N + 1$ 
    else
    6.2:  $N = 0$ 
    6.3: Set adapt_restart := on. ..... (2-d)
    endif
7: if  $N = k$  then ..... (1)
    7.1: if adapt_restart = on then ..... (2-c)
        7.1.1: Do restart procedure.
        7.1.2:  $\alpha'_{\max} = \max_{i-k+1 \leq j \leq i} \alpha'_j$ 
        7.1.3: Set adapt_restart := off,
            but if  $\max_{i+1 \leq j \leq i+k} \alpha'_j > \alpha'_{\max}$ 
            then set adapt_restart := on.
            ..... (2-b)
        endif
    7.2:  $N = 0$ 
    endif
8: Goto step 4.

```

図3 ORTHOMIN( $k$ )法に対する適応的リスタートの算法

以上をまとめた適応的リスタートの算法を図3に示す。以下、ORTHOMIN( $k$ )法に適応的リスタートを組み合わせた方法をAR-ORTHOMIN( $k$ )法と書くことにする。

### 5. 数値例

ORTHOMIN( $k$ )法の収束に関して、Eisenstat, Elman, Schultz<sup>3)</sup>により係数行列の対称部分と歪対称部分の固有値分布が影響していることが証明されている。そこで数値例の1つは係数行列として3重対角行列を考え、その対称部分と歪対称部分の固有値分布を変化させて実験を行う。次に実際的な例として、偏微分方程式の離散化から生じる行列についての評価を行う。

各実験に共通する条件は特に断りのないかぎり以下のとおりである。

- 連立1次方程式の右辺: 係数行列  $A$  と厳密解  $x$  により  $b := Ax$  と定める
- 収束条件:  $\|r_i\| / \|r_0\| < 1.0 \times 10^{-12}$
- 初期近似解:  $x_0 = [0, 0, \dots, 0]^T$
- AR-ORTHOMIN( $k$ )法のパラメータ:  $\epsilon = 0.1$
- 計算機: 分散メモリ型並列計算機 AP1000 (富士通) で64台のプロセッサを使用
- 計算精度: 倍精度

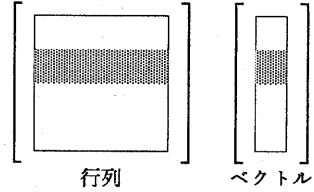


図4 あるプロセッサが担当する要素

● 時間計測: 1つの問題に対し10回の実験を行い、最良と最悪を除いた8回の実行時間の平均値  
 算法の並列化は行列、ベクトル、スカラの演算レベルで行った。各プロセッサではベクトルの一部とそれに対応する行列の行要素を記憶し(図4)、以下のよう  
 に演算を行った。

- ベクトルの加減算, スカラ倍  
 各プロセッサで独立に計算する。
- ベクトルの内積  
 各プロセッサで、担当する部分のみの内積(局所的な内積)を計算する。その後で、全プロセッサの局所内積の総和を計算する。
- 行列とベクトルの乗算  
 他のプロセッサが保持している演算に必要なベクトル要素をプロセッサ間通信により取得する。その後で、各プロセッサで独立に計算する。

数値実験の結果は反復回数と実行時間で示す。最大反復回数内で収束条件を満たさなかった場合には、最終的に得られた相対残差ノルムの値を記す。また、AR-ORTHOMIN( $k$ )法に関してはリスタートが行なわれた回数も記す。

#### 5.1 3重対角行列

例1. 係数行列  $A$  を

$$A := \begin{bmatrix} 1 & (2-\tau)\sigma & & 0 \\ \tau\sigma & \dots & \dots & \\ & \dots & \dots & (2-\tau)\sigma \\ 0 & & \tau\sigma & 1 \end{bmatrix} \in \mathbf{R}^{4096 \times 4096}, \quad (\sigma > 0, \tau > 1)$$

とし、厳密解を  $x = [1, 1, \dots, 1]^T$  とする。このとき係数行列  $A$  の対称部分  $M$  の最小固有値と最大固有値は

$$\lambda_{\min}(M) = 1 + 2\sigma \cos\left(\frac{4096}{4096+1}\pi\right)$$

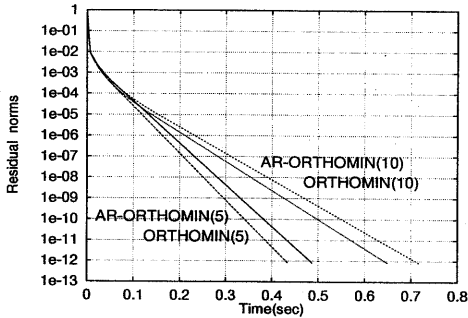
$$\lambda_{\max}(M) = 1 + 2\sigma \cos\left(\frac{1}{4096+1}\pi\right)$$

である。一方、係数行列  $A$  の歪対称部分  $R$  のスペクトル半径は Gerschgorin の定理から  $\rho(R) \leq 2\sigma(\tau-1)$  となる。 $\sigma = 0.1$  とし、 $\tau$  の値を変化させて実験を行った。表1にその結果を示す。

この問題に対する ORTHOMIN( $k$ )法の残差ノルムの振る舞いは、すべての場合で図5に示すように直線的な収束性を示した。収束が順調なことから、AR-

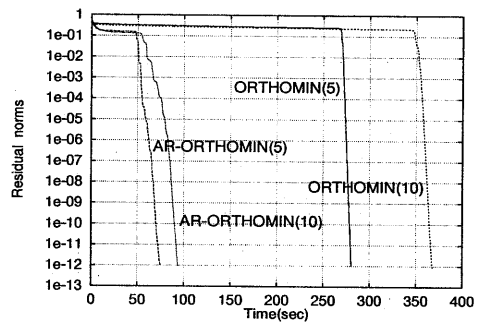
表 1 例 1 に対する実験結果 ( $\sigma = 0.1$ )  
Time: 実行時間 (秒) It.: 反復回数

$\rho(R)$	1.0		2.0		4.0		8.0		16.0	
	Time	It.	Time	It.	Time	It.	Time	It.	Time	It.
ORTHOMIN(5)	0.49	34	0.92	63	2.12	124	4.67	255	11.15	529
AR-ORTHOMIN(5)	0.43	34	0.82	63	1.85	124	4.71	255	9.99	529
ORTHOMIN(10)	0.72	34	1.54	63	3.77	123	8.28	250	16.76	509
AR-ORTHOMIN(10)	0.65	34	1.38	63	3.32	123	7.44	250	15.85	509
<< リスタート回数 >>										
AR-ORTHOMIN(5)	0		0		0		0		0	
AR-ORTHOMIN(10)	0		0		0		0		0	



$\sigma = 0.1, \rho(R) = 1.0$  の場合

図 5 例 1 に対する残差ノルムの収束する様子



$\rho(R) = 5.0, \sigma : \tau = 8 : 0$  の場合

図 6 例 2 に対する残差ノルムの収束する様子

ORTHOMIN( $k$ )法でリスタートが行われることはなかった。 $\sigma$ の値を  $0 < \sigma < 0.5$  の範囲で変化させた場合にも同様の結果が得られた。なお、反復回数が同じ場合でも実行時間に差が生じている。これは計算機の負荷などが原因の計測誤差と考えられる。このことは以下の実験結果についても同様である。

### 5.2 偏微分方程式の離散化から生じる行列

例 2. 矩形領域  $\Omega = [0, 1] \times [0, 1]$  における楕円型偏微分方程式の境界値問題

$$-u_{xx} - u_{yy} + \sigma u_x + \tau u_y = f(x, y)$$

$$u(x, y)|_{\partial\Omega} = 1 + xy$$

を考える。ただし、右辺の関数  $f(x, y)$  は厳密解が  $u(x, y) = 1 + xy$  となるように定めるものとする。この問題を 5 点中心差分法を使ってメッシュサイズ  $256 \times 256$  で離散化し、連立 1 次方程式 (1) を作る。係数行列  $A$  の対称部分  $M$  は、 $\sigma, \tau$  の値にかかわらず Poisson 方程式の離散化から生じる行列と等しくなり、これは正定値であることが知られている。一方、係数行列  $A$  の歪対称部分  $R$  のスペクトル半径は  $\rho(R) \leq (\sigma + \tau)h/4$  となる。ただし、 $h$  はメッシュ幅を表し、 $h = 1/257$  である。表 2、表 3 にこの数値実験の結果を示す。

この問題に対する残差ノルムの振る舞いは  $\sigma, \tau$  の値に関らず、すべて図 6 のような収束性を示した。AR-ORTHOMIN( $k$ )法では ORTHOMIN( $k$ )法の残差ノルムが停滞しているところでリスタートが行われ、平

均で約 1.5 倍、最大で 4 倍近い加速 ( $\rho(R) = 5.0, \sigma : \tau = 8 : 0$  の場合) が得られた。

例 3. 矩形領域  $\Omega = [0, 1] \times [0, 1]$  における楕円型偏微分方程式の境界値問題

$$-u_{xx} - u_{yy} + \sigma \left\{ \left( y - \frac{1}{2} \right) u_x + \left( x - \frac{1}{3} \right) \left( x - \frac{2}{3} \right) u_y \right\} = f(x, y)$$

$$u(x, y)|_{\partial\Omega} = 1 + xy$$

を考える<sup>4)</sup>。右辺の関数  $f(x, y)$  は、例 2 と同様に厳密解が  $u(x, y) = 1 + xy$  となるように定めるものとする。この問題を例 2 と同様の方法で離散化し、連立 1 次方程式 (1) を作る。表 4 にこの数値実験の結果を示す。なお、最大反復回数は (行列の次数)  $\times 0.1$  回とした。

この問題に対する ORTHOMIN( $k$ )法の残差ノルムの振る舞いは、 $\sigma h$  の値に関らず、図 5 に良く似た傾きの緩い直線的な収束性を示した。収束が遅い事から AR-ORTHOMIN( $k$ )法ではリスタートによる収束性の改善が試みられている。リスタートはこの問題に対して効果的に作用せず大幅な加速を得ることはなかったが、ORTHOMIN( $k$ )法より極端に遅くなることもなかった。結果は ORTHOMIN( $k$ )法と同程度だった。

## 6. まとめ

第 5 節で示した数値実験から、AR-ORTHOMIN( $k$ )

表2 例2に対する実験結果 ( $\rho(R) = 0.5$ の場合)  
Time: 実行時間 (秒) It.: 反復回数

$\sigma : \tau$	8:0		7:1		6:2		5:3		4:4	
	Time	It.	Time	It.	Time	It.	Time	It.	Time	It.
ORTHOMIN(5)	64.01	1031	63.98	1030	55.20	891	47.76	772	45.37	732
AR-ORTHOMIN(5)	53.23	838	54.75	852	51.53	798	46.51	737	47.46	745
ORTHOMIN(10)	114.06	1258	103.27	1139	94.08	1039	82.08	906	83.55	920
AR-ORTHOMIN(10)	86.00	964	82.00	915	84.60	943	77.61	865	77.16	858
<< リスタート回数 >>										
AR-ORTHOMIN(5)	5		4		3		3		2	
AR-ORTHOMIN(10)	5		3		3		3		2	

表3 例2に対する実験結果 ( $\rho(R) = 5.0$ の場合)  
Time: 実行時間 (秒) It.: 反復回数

$\sigma : \tau$	8:0		7:1		6:2		5:3		4:4	
	Time	It.	Time	It.	Time	It.	Time	It.	Time	It.
ORTHOMIN(5)	280.79	4220	194.31	2923	121.76	1834	99.70	1493	97.81	1474
AR-ORTHOMIN(5)	74.56	1148	79.43	1222	79.48	1217	57.29	883	62.61	956
ORTHOMIN(10)	367.84	4048	237.13	2610	162.97	1796	148.39	1633	147.19	1621
AR-ORTHOMIN(10)	93.56	1054	90.58	1016	100.88	1139	94.58	1062	98.52	1101
<< リスタート回数 >>										
AR-ORTHOMIN(5)	11		11		12		10		7	
AR-ORTHOMIN(10)	8		7		10		7		6	

表4 例3に対する実験結果  
Time: 実行時間 (秒) It.: 反復回数

$\sigma h$	$2^{-2}$		$2^{-1}$		$2^0$		$2^1$		$2^2$	
	Time	It.	Time	It.	Time	It.	Time	It.	Time	It.
ORTHOMIN(5)	213.55	3212	290.96	4369	344.28	5185	(4e-12)*		(1e-8)*	
AR-ORTHOMIN(5)	191.54	2896	217.57	3338	325.40	4945	(2e-10)*		(2e-8)*	
ORTHOMIN(10)	227.48	2499	309.03	3400	431.88	4751	(3e-12)*		(2e-8)*	
AR-ORTHOMIN(10)	243.68	2723	276.74	3111	482.00	5447	(3e-11)*		(2e-8)*	
<< リスタート回数 >>										
AR-ORTHOMIN(5)	18		63		58		43		35	
AR-ORTHOMIN(10)	15		24		53		52		28	

\* 反復を最大反復回数まで実行した後の相対残差ノルム

法は悪くても ORTHOMIN( $k$ ) 法と同程度の性能を保ちながら、問題によっては ORTHOMIN( $k$ ) 法の数倍の速さで収束するという結果が得られた。非対称行列を係数行列に持つ連立1次方程式の算法として ORTHOMIN( $k$ ) 法を利用する場合には、適応的なリスタート技法を有する AR-ORTHOMIN( $k$ ) 法を用いることが効果的である。

#### 参考文献

- 1) Hestnes, M. and Stiefel, E.: Methods of Conjugate Gradients for Solving Linear Systems, *J. Res. Nat. Bur. Standards*, Vol. 49, pp. 409-435 (1952).
- 2) Vinsome, P.: Orthomin, an iterative method for solving sparse sets of simultaneous linear equations, *In Proceedings of the Fourth Symposium on Reservoir Simulation*, Society of

Petroleum Engineering of AIME, pp. 149-159 (1976).

- 3) Eisenstat, S., Elman, H. and Schultz, M.: Variational Iterative Methods for Nonsymmetric Systems of Linear Equations, *SIAM J. Numer. Anal.*, Vol. 20, No. 2, pp. 345-357 (1983).
- 4) Joubert, W.: Lanczos Methods for the Solution of Nonsymmetric Systems of Linear Equations, *SIAM J. Matrix Anal. Appl.*, Vol. 13, No. 3, pp. 926-943 (1992).
- 5) 野寺隆, 稲津隆敏: リスタートによる疑似残差法の収束性の加速, *情報処理学会論文誌*, Vol. 37, No. 6, pp. 1237-1240 (1996).
- 6) 稲津隆敏, 野寺隆: 適応的なリスタートを用いた疑似残差法, *情報処理学会論文誌*, Vol. 37, No. 9, pp. 1637-1645 (1996).