

適応的リスタートを用いた非定常反復法の 収束性について (Part II)

津野直人[†] 野寺 隆[†]

一般共役残差法の算法の1つとして知られる ORTHOMIN(k) 法に新しい適応的リスタート技法を導入することで、安定して高速な収束が得られることを、分散メモリ型の並列計算機 AP1000 を用いたいくつかの数値実験とともに報告する。

Convergence Properties for an Adaptive Restarted Procedure of Nonstationary Iterative Methods (Part II)

TSUNO NAOTO[†] and NODERA TAKASHI[†]

In this paper, applying the new adaptive restarted procedure, which based on monitoring the norm of residuals, to ORTHOMIN(k) method, we are able to obtain the stable and fast convergence properties. Some numerical experiments are given for using a distributed-memory parallel machine AP1000.

1. はじめに

流体の運動などの連続的な物理現象の解析の多くはその現象を表す偏微分方程式によって行なわれる。とくに時間的に変化しない現象(定常現象)は偏微分方程式の境界値問題という形で表される。これを数値的に解く場合、与えられた方程式を何らかの方法で離散化すると、大規模で疎(sparse)な $n \times n$ の係数行列 A を持つ連立1次方程式

$Ax = b$
が得られる。

連立1次方程式の数値解法は大きく直接法と反復法に分けられるが、係数行列が大規模で疎のときには解を求めるのに必要な計算時間とメモリ容量の節約のために反復法が用いられることが多い。係数行列が対称な連立1次方程式の場合には、Hestenes と Stiefel¹⁾ により提案された共役勾配法が有効な算法であることはいうまでもない^{2),5)}。しかし、係数行列が非対称な問題に対しては、共役勾配法と同じ理論に基づく算法は成立しないので⁴⁾、いまだ決定的な解法が提案されておらず、現在もさまざまな解法の研究、開発が行なわれている^{2),5)}。

リスタートによる非定常反復法の加速は、行列の前処理に代り注目を集めつつある⁷⁾。その理由の一つは、行列の前処理はかなりアート性が強いので、なか

か適切なものが見つからないからである。しかし、反復の過程でリスタートを行う時点を決定することは、それほど簡単なことではない。近年、稲津・野寺⁸⁾ によって提案された適応的リスタート技法は疑似残差法に対して有効に作用し、その収束性をほぼ最適なものとすものである。この技法を別の算法に対しても利用することは可能であり、野寺・津野^{9),10)} にその有効性について述べられている。

本稿では非対称系の問題に対する一般共役残差法^{2),3)} のクラスに属する ORTHOMIN(k) 法に、疑似残差法で用いた ψ の変動率を計算し、それをモニターすることによってリスタートの時点を決定するのではなく、残差ベクトルのノルムの収束性の減少率をモニターすることで疑似残差法とは別の観点に立つ適応的リスタート技法を導入する。このようにすると算法の安定した良い収束性を得ることができることを報告する。

2. 一般共役残差法

非対称問題に対する反復法の1つに、一般共役残差法 (generalized conjugate residual method, GCR 法とも呼ばれる) がある。この解法の算法を図1に示す。

この方法はクリロフ部分空間法 (Krylov subspace method) と呼ばれるクラスに分類されており、各反復によって計算される解ベクトルの列 $\{x_i\}$ とそれに対応する残差ベクトルの列 $\{r_i\}$ が

$$x_i \in x_0 + \mathcal{K}_i,$$

[†] 慶應義塾大学理工学部

Faculty of Science and Technology, Keio University

$$A^T \mathbf{r}_i \perp \mathcal{K}_i,$$

という条件を満たすように構成することになる。ただし、 \mathcal{K}_i は次のようなクリロフ部分空間とする。

$$\mathcal{K}_i = \text{span}\{\mathbf{r}_0, A\mathbf{r}_0, A^2\mathbf{r}_0, \dots, A^{i-1}\mathbf{r}_0\}$$

また、 $\mathbf{r}_i = \mathbf{b} - A\mathbf{x}_i$ は近似解ベクトル \mathbf{x}_i に対する残差ベクトルであるとする。したがって、前述の式より計算誤差の影響をまったく受けなければ、最大 n 回の反復で真の解を計算するということが保証されている。しかしそのためには、共役性を保つためにそれまでの解の探索に使われた方向ベクトル（図1の算法中のベクトル \mathbf{p}_i ）をすべて記憶しておく必要があり、計算時間とメモリ容量の問題から GCR 法を厳密に用いることは事実上不可能である。そこで通常は以下にあげる GCR 法の変形版が用いられる。いずれの解法も最大 n 回の反復で収束するという保証はなくなるが、現実的な計算時間とメモリ容量で実行することができ、十分な結果が得られることが多い。また、GCR 法の特徴の1つである、残差ノルムが単調減少するという性質もそのまま受け継がれる。

2.1 GCR(r) 法

GCR(r) 法は GCR 法の反復を r 回繰り返したあとにリスタート、すなわち、そのときの解ベクトルを新たに初期解ベクトルとして最初から GCR 法を始める解法である。 r の値が大きければ、必要な計算時間とメモリ容量も大きくなるので、一般にあまり大きな値は用いられない。

2.2 ORTHOMIN(k) 法

ORTHOMIN(k) 法は GCR 法ではすべて記憶しておくべき方向ベクトルを直前の k 本だけに限り、その他の部分では GCR 法とまったく同じように反復を進めていく解法である。この方法は、図1の算法において添字 j, h の範囲を変えることで実現できる。GCR(r) 法よりも多くの情報を参照しながら計算を行なうのでより速い収束が期待されるが、リスタートを行わないために計算誤差が蓄積する可能性もある。 k の値は GCR(r) 法の r の値と同じ理由により、あまり大きな値は用いられない。

2.3 ORTHOMIN-R(k, r) 法

ORTHOMIN(k, r) 法は ORTHOMIN(k) 法の反復を r 回繰り返したあとにリスタートを行なう解法である。ORTHOMIN(k) 法の収束の速さに加え、周期的に行なわれるリスタートが計算誤差の蓄積を取り除く効果をもたらす。 k の値は前述 2 つの方法と同様、あまり大きな値は用いられないが、 r の値はメモリ容量にはまったく影響せず、計算時間にもほとんど影響しないので任意の値を用いることができる。

3. 適応的なリスタート技法

ORTHOMIN-R(k, r) 法は r 回の反復を行なったあと強制的にリスタートを実行する。そのため、OR-

```

1. Choose  $\mathbf{x}_0$ .
2. Compute  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$ .
3. Set  $\mathbf{p}_0 = \mathbf{r}_0$ .
4. for  $i = 1, 2, \dots$ 
    4.1  $\mathbf{q}_{i-1} = A\mathbf{p}_{i-1}$ 
    4.2  $\psi_{i-1} = (\mathbf{q}_{i-1}, \mathbf{q}_{i-1})$ 
    4.3  $\alpha_{i-1} = (\mathbf{r}_{i-1}, \mathbf{q}_{i-1})/\psi_{i-1}$ 
    4.4  $\mathbf{x}_i = \mathbf{x}_{i-1} + \alpha_{i-1}\mathbf{p}_{i-1}$ 
    4.5  $\mathbf{r}_i = \mathbf{r}_{i-1} - \alpha_{i-1}\mathbf{q}_{i-1}$ 
    4.6 for  $j = 1, 2, \dots, i-1$ 
        4.6.1  $\beta_i^{(j)} = -\left\{ (A\mathbf{r}_i, \mathbf{q}_j) + \sum_{h=1}^{j-1} \beta_i^{(h)}(\mathbf{q}_h, \mathbf{q}_j) \right\} / \psi_j$ 
    endfor
    4.7  $\mathbf{p}_i = \mathbf{r}_i + \sum_{h=1}^{i-1} \beta_i^{(h)}\mathbf{p}_h$ 
    4.8 If converge, escape the loop.
endifor

```

図1 GCR法の算法

Fig. 1 The Algorithm of GCR method

THOMIN(k) 法が良い収束性を示しているときにも強制的に行なわれるリスタートが悪影響を及ぼし、かえって収束が遅くなってしまふことが考えられる。このような強制的なリスタートによる悪影響を解消するため、ORTHOMIN(k) 法の収束性が悪くなったときに限りリスタートを行なう適応的なリスタート技法を導入する。以下、この技法を ORTHOMIN(k) 法に適用した方法を ORTHOMIN(k)-AR 法と書くことにする。

適応的なリスタート技法は、もともと稲津・野寺⁸⁾により ORTHORES 法に導入された技法である。この技法を導入するにあたり 1 番のポイントになるのは、どのタイミングでリスタートを実行するか、すなわち残差ノルムの収束性をどのようにして判断するか、という点にある。また、残差ノルムの収束性を判断するときに行列とベクトルの積やベクトルの内積といった大きな計算を必要とすると、仮に反復回数が少なくなるとしても計算時間は増えてしまい満足な結果は得られない。そのため、いかに少ない計算量で残差ノルムの収束性を判断できるか、ということも求められる。そこで残差ノルムの収束性の判断について考える。

3.1 残差ノルムの収束性の判断

前回の報告¹⁰⁾ では、数値実験の結果をもとに算法中で使われるスカラと残差ノルムの間の関係を予想し、そのスカラの振る舞いから間接的に残差ノルムの収束性を判断した。具体的には図1の算法中のスカラ ψ に注目し、スカラ ψ の振る舞いが停滞しているときには残差ノルムも停滞しているという予想をたててリスタートを実行するタイミングを判断した。このとき、スカラ ψ の変動係数を計算することで停滞しているかどうかを判断した。

しかし、ORTHOMIN(k)法では通常算法の停止を判断するために各反復後に相対残差ノルムを計算する。この相対残差ノルムを利用する方が算法中のスカラを通すよりも正確に収束性の判断を与えることができる。そこで今回はこの相対残差ノルムの値をもとにして、直接リスタートの実行を判断することにする。

先に述べた通りORTHOMIN(k)法には残差ノルムが単調減少するという性質がある。この性質も考慮し、以下のようにしてリスタートを実行するかどうかを決定する。まずORTHOMIN(k)法の反復を r 回行なう間に、1回でも与えた基準以上の残差ノルムの減少がみられたなら、ORTHOMIN(k)法は収束の方向に向かってしていると判断しリスタートは実行せずにさらに反復を r 回続ける。逆に r 回の反復すべてにおいて残差ノルムの減少が基準に達しないなら、ORTHOMIN(k)法は停滞していると判断しリスタートを実行する。残差ノルム減少の基準およびリスタート判定の周期については、後ほど数値実験を参考に決定する。

残差ノルムの収束性の判断に必要な計算とメモリ容量は明らかに問題を生じるものではない。また分散メモリ型の並列計算機に実装することを考えても通信を必要としないためボトルネックにはならない。

以上のような適応的なリスタート技法を導入することで、ORTHOMIN-R(k, r)法で考えられる強制的なリスタートの及ぼす悪影響が解消され、種々の問題に対してより良い収束がみられることが期待される。

4. 数 値 例

矩形領域 $\Omega = [0, 1] \times [0, 1]$ 上で、以下にあげる2つの楕円型偏微分方程式の境界値問題について数値実験を行なった⁶⁾。

問題 1

$$\begin{aligned} -u_{xx} - u_{yy} + Du_x &= G(x, y) \\ u(x, y)|_{\partial\Omega} &= 1 + xy \end{aligned}$$

問題 2

$$\begin{aligned} -u_{xx} - u_{yy} + D \left\{ \left(y - \frac{1}{2} \right) u_x \right. \\ \left. + \left(x - \frac{1}{3} \right) \left(x - \frac{2}{3} \right) u_y \right\} &= G(x, y) \\ u(x, y)|_{\partial\Omega} &= 1 + xy \end{aligned}$$

問題 1, 問題 2 ともに解を $u(x, y) = 1 + xy$ と設定し、右辺 $G(x, y)$ を決定する。また D は実数定数である。この D の値を変えることで、離散化した結果得られる係数行列が変化する。以上の問題を5点中心差分法により離散化し、連立1次方程式を得る。ここで、離散化によるメッシュ幅を h で表すことにする。

各実験において、算法に与える初期解ベクトルは零ベクトルとし、収束条件は相対残差ノルムで $\|r_i\|/\|r_0\| \leq 1.0 \times 10^{-12}$ とした。すべての計算は倍

精度で行ない、算法中の行列とベクトルに関する演算を分散メモリ型の高並列計算機 AP1000 (富士通)で64個のプロセッサを用いて並列に処理した。

実験 1 [各解法の収束性の比較]

問題 1 および問題 2 に対して次のような条件で各解法がどのような収束を示すか観測する。

- メッシュサイズ: 256×256
- D の値: Dh が決められた値となるように決定
- 解法:
 - (1) GCR(5)
 - (2) ORTHOMIN-R(5,10)
 - (3) ORTHOMIN-R(5,20)
 - (4) ORTHOMIN-R(5,50)
 - (5) ORTHOMIN-R(5,100)
 - (6) ORTHOMIN(5)

問題 1 に対する結果を表 1 に、問題 2 に対する結果を表 2 に示す。表 1 および表 2 の中で太字はその解法が最も速く収束していることを示している。またそれぞれの問題に対し、 $Dh = 2^2$ の場合の残差ノルムの収束する様子を図 2 に示す。

問題 1 に対しては GCR(5)法が最も速い収束を示している。また、ORTHOMIN-R(5, r)法をみると、 r の値が小さいほど収束性が良くなっている。つまり、この問題に対してはリスタートが非常に効果的に作用し、リスタートを行なう回数が多いほど収束性が良くなるという傾向がみられる。一方、問題 2 に対しては問題 1 と逆の傾向がみられ、リスタートを多く行なう GCR(5)法は非常に悪い収束性を示している。しかし ORTHOMIN-R(5,20)法や ORTHOMIN-R(5,50)法の結果から分かるように、リスタートを適度に行なうことで ORTHOMIN(5)法よりも速い収束を得ることが可能である。

ORTHOMIN-R(5,20)法はリスタートが収束性に良い影響を与える問題、悪い影響を与える問題のどちらに対しても安定した速い収束性を示している。したがって、この解法をもとに適応的なリスタート技法を導入する。すなわち、残差ノルム減少の基準を $\log_{10}(1.0 \times 10^{-12})/1000$ として、これより残差ノルムの減少が少ない反復が20回連続したときにリスタートを実行するように定める。基準の値は1000回の反復(約100秒)で収束条件が満たされることを期待して定めた。

実験 2 [適応的なリスタート技法]

問題 1 および問題 2 に対して、次のような条件で適応的なリスタート技法を導入した ORTHOMIN(k)-AR 法の収束を観測する。

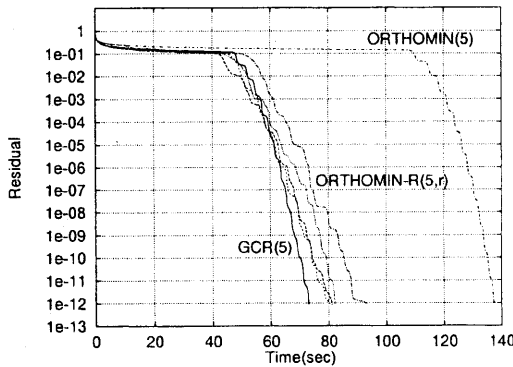
- メッシュサイズ: 256×256
- D の値: 実験 1 と同じ
- 解法: ORTHOMIN(5)-AR

表1 問題1に対する各解法の実行時間(sec)
Table 1 The executive time (sec) of various methods for problem 1

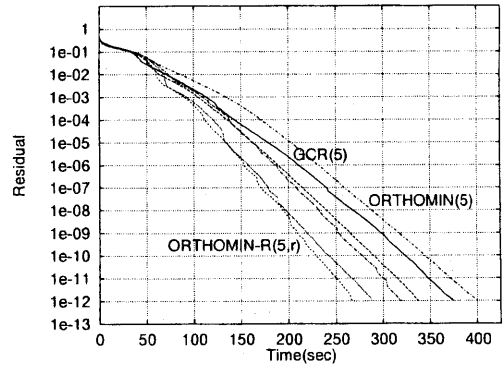
算法	Dh の値				
	2 ⁻²	2 ⁻¹	2 ⁰	2 ¹	2 ²
GCR(5)	138.12	70.14	73.16	70.23	73.18
ORTHOMIN-R(5,10)	107.92	82.98	75.77	73.39	81.07
ORTHOMIN-R(5,20)	91.67	86.97	83.51	79.36	79.96
ORTHOMIN-R(5,50)	119.87	86.91	81.21	79.11	82.21
ORTHOMIN-R(5,100)	122.40	81.59	79.88	83.65	93.10
ORTHOMIN(5)	200.13	84.73	91.44	100.14	137.44

表2 問題2に対する各解法の実行時間(sec)
Table 2 The executive time (sec) of various methods for problem 2

算法	Dh の値				
	2 ⁻²	2 ⁻¹	2 ⁰	2 ¹	2 ²
GCR(5)	430.50	258.74	256.04	295.54	375.22
ORTHOMIN-R(5,10)	208.80	142.96	181.67	241.46	337.36
ORTHOMIN-R(5,20)	139.12	131.13	142.56	169.40	267.13
ORTHOMIN-R(5,50)	124.13	119.71	137.00	182.81	287.49
ORTHOMIN-R(5,100)	139.35	122.19	132.18	185.51	319.21
ORTHOMIN(5)	180.00	119.89	145.39	207.69	399.94



(a) 問題 1



(b) 問題 2

図2 Dh = 2²の場合の残差ノルムの収束の様子
Fig. 2 The convergence behavior of residual norm for Dh = 2²

- * 残差ノルム減少の基準: -12/1000
- * リスタート判定の周期: 20回

結果を表3に示す。表3の中で太字はORTHOMIN-R(5,20)法よりも収束が速いことを示している。また、リスタートを実行した回数をORTHOMIN-R(5,20)法のものに合わせて、問題1に対するものを表4に、

問題2に対するものを表5に示す。

ORTHOMIN(5)-AR法は問題1に対してはORTHOMIN-R(5,20)法とほとんど変わらないが、若干遅い収束を示している。これは先に述べた通り、この問題に対してリスタートが多いほど収束が速くなる傾向があるためと考えられる。一方、問題2に対してはほぼすべての場合でORTHOMIN-R(5,20)法よりも

表 3 ORTHOMIN(5)-AR 法の実行時間 (sec)

Table 3 The executive time(sec) of ORTHOMIN(5)-AR method

	Dh の値				
	2 ⁻²	2 ⁻¹	2 ⁰	2 ¹	2 ²
問題 1	110.82	83.16	81.30	79.82	90.07
問題 2	138.02	120.19	131.81	179.77	249.81

表 4 問題 1 に対するリスタートの回数

Table 4 The restarting counts for problem 1

算 法	Dh の値				
	2 ⁻²	2 ⁻¹	2 ⁰	2 ¹	2 ²
ORTHOMIN-R(5,20)	44	42	40	39	39
ORTHOMIN(5)-AR	31	22	21	22	24

表 5 問題 2 に対するリスタートの回数

Table 5 The restarting counts for problem 2

算 法	Dh の値				
	2 ⁻²	2 ⁻¹	2 ⁰	2 ¹	2 ²
ORTHOMIN-R(5,20)	67	62	67	81	130
ORTHOMIN(5)-AR	40	28	31	58	106

速い収束を示しており、適応的なリスタート技法を導入したことで収束性に悪い影響を与えるリスタートをうまく回避できたことを表している。

また、このときの残差ノルムの収束の様子をいくつか見てみると (図 3 および 図 4)、残差ノルムが収束の方向に向かうまでは ORTHOMIN(5)-AR 法も ORTHOMIN-R(5,20) 法もほとんど同じ振る舞いをしており、両者の差は収束が加速してから生じていることが分かる。このことから適応的なリスタート技法の基本的な考え方である、残差ノルムの収束性が悪くなったときにかぎりリスタートを行なう、といったことが結果に反映されていることが分かる。

適応的なリスタート技法を導入することで、必ずしもすべての問題に対して速い収束が得られるということではない。しかし、ORTHOMIN-R(k,r) 法で行なわれる強制的なリスタートが良い方向に影響するのか、悪い方向に影響するのかは問題に依存し、これをあらかじめ知ることは不可能である。適応的なリスタート技法を導入することで、種々の問題に対する収束の安定性は各段に向上しているといえる。

5. ま と め

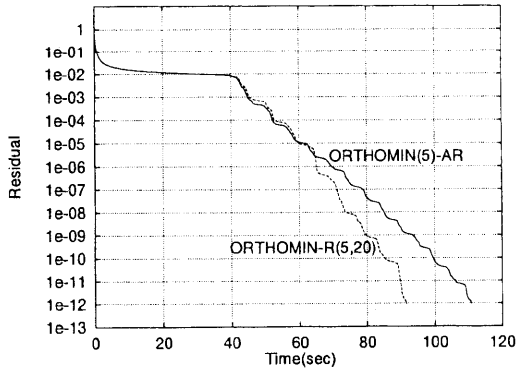
残差ベクトルのノルムの収束性をモニターする適応

的なリスタート技法を ORTHOMIN(k) 法に導入し、その残差ノルムの収束性について調べてきた。適応的なリスタート技法は非常に低コストに実行することができ、種々の問題に対して安定して速い収束を得られることが確かめられた。

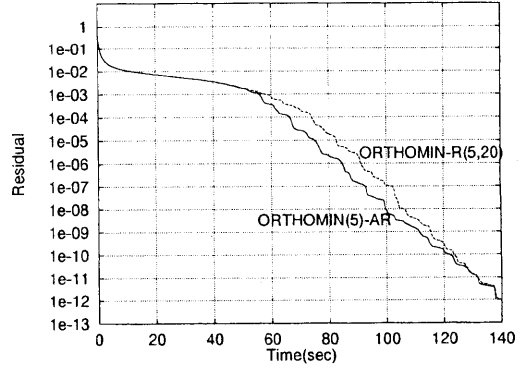
またこの技法は ORTHOMIN(k) 法や ORTHORES 法のみならず、その他多くの反復法にも導入することが可能である。その際には導入する方法に合ったリスタートの判断基準を考える必要がある。今回の ORTHOMIN(k) 法の場合も含めて、理論的な裏付けをもつ基準を与えることが今後の課題である。

参 考 文 献

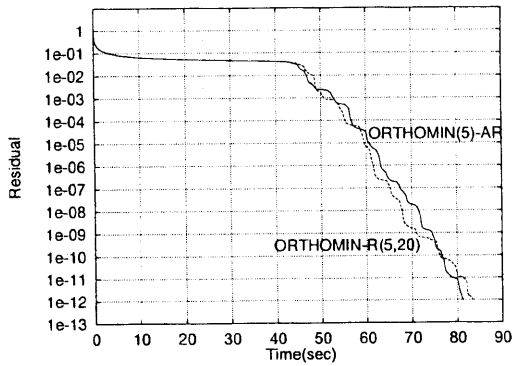
- 1) Stiefel, E. and Hestenes, M.: *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, Vol. 49, pp. 409-435 (1952).
- 2) Bruaset, A. M.: *A survey of preconditioned iterative methods*, Pitman Research Notes in Mathematics Series, Longman Scientific & Technical(1995).
- 3) Eisenstat, H. C., Elman, H. C. and Schultz, M. H.: *Variational iterative methods for non-symmetric systems of linear equations*, SIAM J. Numer. Anal., Vol. 20, No. 2, pp. 345-357(1983).
- 4) Faber, V. and Manteuffel, T. A.: *Necessary and sufficient conditions for existence of a conjugate gradient method*, SIAM J. Numer. Anal., Vol. 21, No. 2, pp. 352-362 (1984).
- 5) 名取, 野寺: 大型疎行列計算における反復解法, 情報処理, Vol. 28, No. 11, pp. 1452-1459 (1987).
- 6) 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).
- 7) 野寺, 稲津: リスタートを用いた疑似残差法の収束性について, 情報処理学会論文誌, Vol. 37, No. 6, pp. 1237-1240 (1996).
- 8) 稲津, 野寺: 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 9月号掲載予定(1996).
- 9) 野寺, 津野: 適応的なリスタートを用いた ORTHOMIN(k) 法の収束性について, 情報処理学会論文誌投稿中 (1996).
- 10) 野寺, 津野: 適応的なリスタートを用いた非定常反復法の収束性について, 情報処理学会研究報告, Vol. 96, No. 81, pp. 105-109(1996).



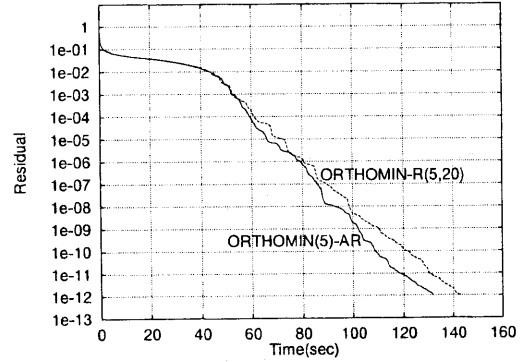
(a) $Dh = 2^{-2}$ の場合



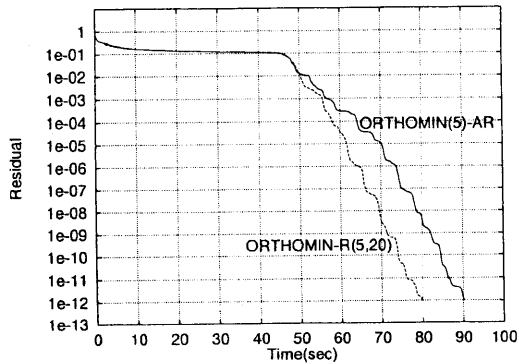
(a) $Dh = 2^{-2}$ の場合



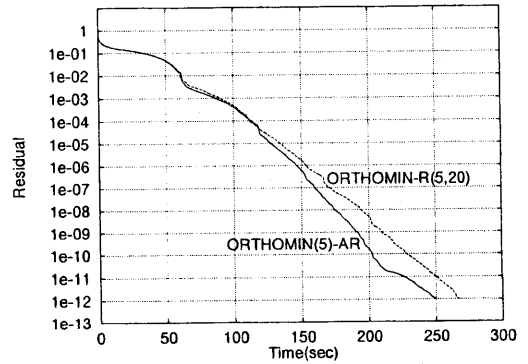
(b) $Dh = 2^0$ の場合



(b) $Dh = 2^0$ の場合



(c) $Dh = 2^2$ の場合



(c) $Dh = 2^2$ の場合

図 3 問題 1 に対する ORTHOMIN(5)-AR 法の残差ノルムの収束の様子

Fig. 3 The convergence behavior of residual norm of ORTHOMIN(5)-AR method for problem 1

図 4 問題 2 に対する ORTHOMIN(5)-AR 法の残差ノルムの収束の様子

Fig. 4 The convergence behavior of residual norm of ORTHOMIN(5)-AR method for problem 2