

改良版 RIC2S 分解前処理つき CG 法の収束性評価

井上 明彦 * 柿原 正伸 * 藤野 清次 ** 吉田 正浩 *

* 九州大学大学院システム情報科学府 ** 九州大学情報基盤センター

共役勾配法は収束性を向上させるために前処理と共に用いられることが多く、その前処理として不完全コレスキー分解が広く用いられてきた。しかし、この不完全分解は行列によっては分解の途中で対角項の値が負になり分解の破綻が起こることがわかってきた。そこで、近年ロバストな分解として、RIC 分解や RIC2S 分解が提案された。RIC 分解においては、分解中に行う対角項に対する修正量を緩和するという処理を行うことで収束性を向上させた対角緩和付き準 RIC 分解が共著者らにより提案され、数値実験でその有効性が示されている。そこで、本論文では、RIC2S 分解において要素を棄却するときの修正量に対して緩和処理を適用し、収束性の向上を図る。そして、数値実験により、その有効性を検証する。

Evaluation of Convergence Rate of Modified RIC2S Factorization Preconditioning for CG method

Akihiko INOUE * Masanobu KAKIHARA * Seiji FUJINO ** Masahiro YOSHIDA *

* Graduate School of Information Science and Electrical Engineering, Kyushu University

** Computing and Communications Center, Kyushu University

Incomplete Cholesky factorization preconditioning has been used with CG method to improve convergence rate. However, the incomplete factorization can fail for a general symmetric positive definite matrix due to the occurrence of non-positive pivots. RIC factorization and RIC2S factorization have been suggested to overcome this problem. For RIC factorization, revising relaxation on amount of modification for diagonal entries was effective strategy. In this paper, we adopt the diagonal relaxation in RIC2S factorization for improving convergence rate. Numerical experiments shows the validity of proposed preconditioning.

1 はじめに

偏微分方程式の境界値問題を有限要素法や有限差分法で解く際に生ずる対称正定値な係数行列を持つ連立一次方程式の解法として共役勾配 (Conjugate Gradient, 以下 CG と略す) 法がある。CG 法は前処理と共に用いられることが多く、従来からある前処理として不完全コレスキー (Incomplete Cholesky, 以下 IC と略す) 分解がある。IC 分解は、行列によっては分解の途中に対角項の値が負になり分解の破綻が起こることが知られている。そこで、近年様々なロバストな分解が提案されてきた [3][5][6]。その一つに Ajiz らによって提案された RIC (Robust IC) 分解がある [1]。この分解は、分解過程で生まれた非零要素がある棄却許容値 (drop tolerance value) τ よりも小さいときその要素を棄却し、要素が棄却される度に対角項の修正を行い、対角項の値が負になることを防いだ分解法である。

一方、Tismenetsky によって提案された RIC2 分解は、棄却許容値よりも小さな要素も保存し、分解に反映させることでより厳密な分解を行い、CG 法が収束するまでの反復回数を少なくした分解法である [11]。しかし、RIC2 分解は棄却する要素も保存するためメモリを多く必要とする。また、完全コレスキー分解に近い計算なので前処理の計算時間が多くかかる。そこで、計算のコストを抑えるため、分解中に生じた棄却許容値 τ の 2 次のオーダー $O(\tau^2)$ の要素は保存せず棄却する RIC2S 分解が提案された [8]。RIC2S 分解では、RIC 分解と同様に要素が棄却される度に対角項の修正を行う。

RIC 分解においては、分解中に行う対角項に対する修正量を緩和することで収束性を向上させた対角緩和付き準 RIC 分解が共著者らにより提案され、数値実験でその有効性が示されている [7]。そこで、本論文では、RIC2S 分解においてオーダー $O(\tau^2)$ の要素を棄却するときの修正量に対して緩和処理を適用し、その有効性を検証する。数値実験では、薄膜磁気記録ヘッドの静磁場解析、立方体型磁性体の静磁界解析と、カスプコイル励磁型単磁極ヘッドの静磁界解析の問題に対して各前処理を適用したときの収束性を比較し、新しい前処理の有効性を検証する。

2 前処理つき CG 法

対称正定値行列を係数行列 $A = (a_{ij})$ に持つ連立一次方程式

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

を前処理つき CG (Preconditioned CG, 以下 PCG と略す) 法で解くことを考える。ここで、 A は大きさ $n \times n$ の正平方行列、 x , b は大きさ n の解ベクトルおよび右辺ベクトルとする。まず、係数行列 A を近似分解して、以下のように前処理行列 M を構成する。

$$M = U^T U (\simeq A). \quad (2)$$

ここで、 $U = (u_{ij})$ は上三角行列、上付き添字 T は転置を表す。この U および U^T を用いて、方程式 (1) を

$$(U^{-T} A U^{-1}) (Ux) = U^{-T} b \quad (3)$$

のように変換する．この操作は前処理と呼ばれる．前処理によって，式(3)は， $U^{-T}AU^{-1}$ ， Ux ， $U^{-T}b$ をそれぞれ係数行列，解ベクトル，右辺ベクトルとする新たな方程式となる．

3 スペクトル条件数

スペクトル条件数とは，連立一次方程式 $Ax = b$ の解の精度に影響を与える指標であり，計算誤差(丸め誤差や打ち切り誤差)の増幅率を表す[10]．したがって，係数行列 $U^{-T}AU^{-1}$ のスペクトル条件数が大きな値であると，計算誤差が解の精度に悪影響を与え，CG法における解の収束が遅くなる．

3.1 スペクトル条件数の定義

はじめに，スペクトル条件数を定義する上で必要となるベクトルと行列のノルムを定義する．本論文では，ベクトル $x = (x_1, x_2, \dots, x_n)^T \in R^n$ のノルムとして2-ノルムを用いる．その定義は

$$\|x\|_2 = \left(\sum_{i=1}^n |x_i|^2 \right)^{\frac{1}{2}} \quad (4)$$

である．また，ベクトル x に対する行列 A の一般的なノルムは

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} \quad (5)$$

と定義される．ここで，ベクトル x のノルムが2-ノルムであるとき，行列 A のノルムをスペクトルノルムと呼ぶ．行列 A が対称正定値行列のときのスペクトルノルムは

$$\|A\|_2 = \lambda_{\max}(A) \quad (6)$$

と表される．ただし， $\lambda_{\max}(A)$ は行列 A の絶対値最大固有値を表す．

最後に，条件数の定義を行う．一般的な条件数は次のように定義される．

$$\text{cond}(A) = \|A\| \cdot \|A^{-1}\| \quad (7)$$

ここで， $\|A\|$ ， $\|A^{-1}\|$ が対称正定値行列 A のスペクトルノルムならば，行列 A の条件数は

$$\text{cond}(A) = \lambda_{\max}(A) \lambda_{\max}(A^{-1}) = \left| \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} \right| \quad (8)$$

と表すことができる．ただし， $\lambda_{\min}(A)$ は行列 A の絶対値最小固有値である．この条件数のことを A のスペクトル条件数と呼び， $C(A)$ と表すことにする．

4 RIC 分解

RIC 分解は，次元数 n の係数行列 A を

$$A = U^T U - R - R^T - D \quad (9)$$

のように近似分解する方法である[1]．ここで， U は上三角行列， $R = (r_{ij})$ は狭義上三角行列， $D = (d_i)$ は対角行列，上付き添え字 T は行列の転置を各々表す．行列 R と行列 D は分解の様子を表す形式的な行列であり，いずれも PCG 法の中では使用されない．狭義上三角行列 R は行列 U のスパース性を保持するために，分解中に U の要素を棄却することを表す．すなわち， U の要素がある棄却許容値 τ ($0 \leq \tau \leq 1$) よりも小さいときその要素を棄却す

る．RIC 分解では， U の要素が棄却されたとき対角項の値が負にならないように対角項に修正量を加え，理論的にロバスト(robust)性が保証される．

しかし，RIC 分解では，行列積 $U^T U$ のスペクトル条件数の上限は，棄却許容値 τ に関する1次のオーダー $O(\tau)$ で見積もられる．具体的には， c_1 を定数として，次の式で表される[8]．

$$C(U^{-T}AU^{-1}) \leq 2 + \frac{2c_1\tau}{\lambda_{\min}(A)} \quad (10)$$

スペクトル条件数の上限が大きいため，場合によってはCG法がなかなか収束せず反復計算に時間が掛かる場合がある．そこで，次の節では，不完全分解 $U^T U$ のスペクトル条件数の上限が棄却許容値 τ に関する2次のオーダー $O(\tau^2)$ で抑えられる RIC2 分解について述べる．

5 RIC2 分解と RIC2S 分解

5.1 RIC2 分解

RIC2 分解は係数行列 A を

$$A = U^T U + U^T R + R^T U \quad (11)$$

のように近似分解する方法である[11]．式(11)を要素ごとに書き下すと，

$$a_{ij} = \sum_{k=1}^i u_{ki}u_{kj} + u_{ki}r_{kj} + r_{ki}u_{kj} \quad (1 \leq i \leq j \leq n) \quad (12)$$

と表せる．したがって， $(U^T U + U^T R + R^T U)$ 分解における上三角行列 U の要素 u_{ij} は

$$u_{ii} = \left\{ a_{ii} - \sum_{k=1}^{i-1} (u_{ki}^2 + u_{ki}r_{ki} + r_{ki}u_{ki}) \right\}^{1/2}, \quad (13)$$

$$u_{ij} = \left\{ a_{ij} - \sum_{k=1}^{i-1} (u_{ki}u_{kj} + u_{ki}r_{kj} + r_{ki}u_{kj}) \right\} / u_{ii} \quad (14)$$

となる．分解の手順は以下ようになる．

RIC2 分解の分解手順

1. $i = 0$ (行カウンタを初期化)．
2. $i = i + 1$ (行カウンタを1増やす)．
3. 式(13)，(14)の計算を行い，第 i 行の要素を得る．
4. 以下の手順により3.で得られた要素を行列 U もしくは行列 R に格納する．
 - (a) $|u_{ij}| \geq \tau$ のとき， u_{ij} は行列 U の要素とする．
 - (b) $|u_{ij}| < \tau$ のとき， u_{ij} は行列 R の要素とする．
5. $i < n$ ならば2.へ戻る．そうでなければ，分解終了．

RIC2 分解では，以下の二つの条件を必要とする．

1. 誤差行列 R は，厳密な上三角行列である．つまり，次の式が成り立つ．ただし，“diag”は行列の対角部分を表す．

$$\text{diag}(R) = 0. \quad (15)$$

2. 行列 U と R のどの要素の積もすべて 0 である。すなわち, $u_{ij} \times r_{ij} = 0$ for all i, j (このとき行列 U と R は互いに構造的に直交すると呼ばれる [8]).

したがって, $U^T R$ の対角要素はすべて 0 である。すなわち,

$$\text{diag}(U^T R) = 0 \quad (16)$$

が成り立つ。これらの条件より, $(U^T U + U^T R + R^T U)$ 分解を一意的に決定することができる。

RIC2 分解は完全コレスキー分解 $(U + R)^T (U + R) = U^T U + U^T R + R^T U + R^T R$ の $R^T R$ の計算を省いたアルゴリズムである。誤差行列 R の要素の大きさはオーダー $O(\tau)$ なので, $R^T R$ の要素の大きさはオーダー $O(\tau^2)$ となる。したがって RIC2 分解はオーダー $O(\tau^2)$ の要素のみを棄却することになる。RIC2 分解においては, 行列積 $U^T U$ のスペクトル条件数の上限は, 次の式で評価される。

$$C(U^{-T} A U^{-1}) \leq \varphi_2(c_2 \tau / \lambda_{\min}(A)) \quad (17)$$

$$\varphi_2(t) = (\sqrt{t} + \sqrt{1+t})^4$$

すなわち, RIC2 分解においては, スペクトル条件数は棄却許容値 τ に関する 2 次のオーダー $O(\tau^2)$ で抑えられる。しかし, 誤差行列 R を保存するためメモリを多く必要とする。また, 誤差行列 R も分解計算に反映させるため前処理の計算時間が多くかかる。

5.2 RIC2S 分解

RIC2S 分解は係数行列 A を

$$A + \sigma \tau^2 I = U^T U + U^T R + R^T U - S \quad (18)$$

のように近似分解する方法である [8]。ここで, I は単位行列, S は対称非負定値な行列, σ は適当な定数である。後に示す数値実験では, 定数 σ と, 後述の定数 γ の値は, 論文に示されている $\sigma = 2, \gamma = 1$ をそれぞれ用いた [2]。

RIC2S 分解の手順は次のように表すことができる。

RIC2S 法の分解手順

1. $i = 0$ (行カウンタを初期化)。
2. $i = i + 1$ (行カウンタを 1 増やす)。
3. 棄却許容値 τ を与える。
4. 係数行列 A の対角項に $\sigma \tau^2$ を加え修正する。
5. $(U^T U + U^T R + R^T U)$ 分解を行い, 第 i 行の要素を得る。
6. 上記の 5. で得られた要素の中でオーダー $O(\tau^2)$ の大きさの要素, すなわち $\gamma \tau^2$ (γ は適当な定数) よりも値が小さい要素を棄却し, 対角要素の修正を行う。
7. RIC2 分解の分解手順中の第 4 項と同様に, 6. で棄却されなかった要素を U と R に各々格納する。
8. $i < n$ ならば 2. へ戻る。そうでなければ, 分解終了。

RIC2S 分解のアルゴリズムは次のようになる。

[RIC2S のアルゴリズム]

```
for  $i = 1, \dots, n$ 
   $d_i = (1 + \sigma \tau^2) a_{ii}$ 
```

```
end for
for  $i = 1, \dots, n$ 
  for  $j = i + 1, \dots, n$ 
     $v_j = a_{ij}$ 
  end for
  for  $k = 1, \dots, i - 1$ 
    for  $j = i + 1, \dots, n$ 
       $v_j = v_j - u_{ki} u_{kj} - u_{ki} r_{kj} - r_{ki} u_{kj}$ 
    end for
  end for
  for  $j = i + 1, \dots, n$ 
     $\xi = |v_j| / \sqrt{d_i d_j}$ 
    if  $\xi \leq \gamma \tau^2$  then
       $v_j = 0$ 
       $d_i = (1 + \xi) d_i, d_j = (1 + \xi) d_j$  (I)
    end if
  end for
   $u_{ii} = \sqrt{d_i}$ 
  for  $j = i + 1, \dots, n$ 
     $v_j = v_j / u_{ii}$ 
    if  $|v_j| \geq \tau$  then
       $u_{ij} = v_j$ 
    else if  $|v_j| > 0$  then
       $r_{ij} = v_j$ 
    end if
     $d_j = d_j - u_{ij}^2$ 
  end for
end for
```

RIC2S 分解は $(U^T U + U^T R + R^T U)$ 分解の途中で生じたオーダー $O(\tau^2)$ の要素を誤差行列 R に保存せずに棄却し, RIC 分解と同様に対角項の修正を行う。これにより, RIC2 分解で問題となった $(U^T U + U^T R + R^T U)$ 分解の計算量とメモリ量を軽減することが出来る。行列 $U^{-T} A U^{-1}$ のスペクトル条件数は, c_1, c_2 を定数として, 次の式で評価される。

$$C(U^{-T} A U^{-1}) \leq 2(1 + \frac{c_2^2}{\sigma}) \left(1 + (2c_2^2 + c_1 \gamma + \sigma) \frac{\tau^2}{\lambda_{\min}(A)} \right). \quad (19)$$

つまり, RIC2S 行列積 $U^T U$ のスペクトル条件数の上限は, RIC2 分解と同様に棄却許容値 τ に関する 2 次のオーダー $O(\tau^2)$ で抑えられる。すなわち, RIC2S 分解は RIC 分解と RIC2 分解の長所を合わせた分解方法であると言える。

5.3 RIC2S 分解の改良

4 節で述べた RIC 分解において, 要素が棄却されたときの対角項に対する修正は, 理論的な意味でのロバスト性を保持するための処理である。しかし, この修正量は過剰であることが多く, 修正量を緩和することにより収束性を向上させた対角緩和付き RIC 分解が共著者らにより提案され, 数値実験でその有効性が示されている [7]。そこで, RIC2S 分解においてオーダー $O(\tau^2)$ の要素を棄却するときの修正量に対して緩和処理を適用することを考える。すなわち, RIC2S 分解において, 要素を棄却するときの対角項の修正量を $0 \leq \omega_m \leq 1$ なる乗算緩和係数 ω_m をかけることで小さくする。この分解を MRIC2S (Modified RIC2S) 分解と呼ぶ。MRIC2S 分解では, 前節の RIC2S 分解のアルゴリズム中の枠 (I) で囲んだ部分が次のように変わる。

$$d_i = (1 + \omega_m \xi) d_i, \quad d_j = (1 + \omega_m \xi) d_j.$$

MRIC2S 分解において、 $\omega_m = 1.0$ のときは RIC2S 分解に相当する。

6 数値実験

6.1 計算機環境と計算条件

数値実験は、HP workstation xw4100(CPU:Pentium4, クロック周波数: 3.2 GHz, 主記憶: 2 Gbytes, OS: Red Hat Linux ver. 7.3) 上で行った。コンパイルには Intel Fortran Compiler ver. 7.1 を用い、最適化オプションは-O3 を使用した。計算はすべて倍精度演算で行った。CG 法の収束判定条件は、相対残差の 2-ノルム: $\|r_{k+1}\|_2 / \|r_0\|_2$ の値が 10^{-8} 以下のときとした。方程式の右辺項は物理的な条件から得られる値をそのまま用いた。初期近似解 x_0 はすべて 0 とした。最大反復回数は行列の次元数の値とし、行列は予め対角項をすべて 1 に正規化の処理をした。

6.2 テスト行列

テスト行列には、電磁場解析の分野の問題から得られた 4 つの行列を使用した。表 1 にテスト行列の特徴を示す。行列“Thin_Head”は磁気記録の分野で検証用モデルとして検討された薄膜磁気記録ヘッドの静磁界解析で得られた行列である。図 1 に行列“Thin_Head”の解析モデル図を示す。行列“Cube_20”, 行列“Cube_30”は $1\text{m} \times 1\text{m} \times 1\text{m}$ の立方体型磁性体に一様磁場を付加したときの静磁界解析で得られた行列である。ただし、対称性を用いて $1/8$ 領域の解析を行っている。また、行列“Cusp_Head”はカスプコイル励磁型単磁極ヘッドの静磁界解析で得られた行列である [9]。

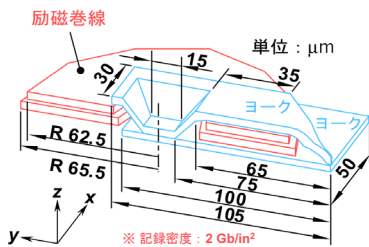


図 1: 行列“Thin_Head”の解析モデル図 [4]。

6.3 実験結果

表 2 にテスト行列に対する前処理つき CG 法の反復回数、計算時間とメモリ量の比較を示す。表中で“加速 IC(0)”は不完全コレスキー分解のときに係数行列 A の対角項に加速係数を乗じた加速係数つき IC(0) 分解 “緩和 RIC”は対角緩和つき準 RIC 分解を表す。表中のパラメータで、“ τ ”は棄却許容値、“加速係数”は加速係数つき IC(0) 分解における加速係数、“ ω_d ”は対角緩和係数、“ ω_m ”は乗算緩和係数を各々表す。反復回数で“max”となっている所は、最大反復回数までに収束しなかったことを表す。表中の計算時間の単位はすべて秒である。また、太字表記の数字は調べた前処理で最も計算時間が短かったものを表す。

加速係数つき IC(0) 分解の加速係数はまず 1.00 に設定し (これは IC(0) 分解に相当)、対角項の値が負になったら加速係数を 0.02 ずつ増加させ、対角項の値が正になるまで繰り返した。RIC 分解において棄却許容値 τ の値は 0.0005, 0.001, 0.005, 0.01, 0.05 の 5 通りを調べ、最も

計算時間が短いものを選んだ。対角緩和つき RIC 分解において棄却許容値 τ の値は RIC 分解で選んだ値を用い、対角緩和係数 ω_d の値は棄却許容値 τ の $1/1000, 1/500, 1/100, 1/50, 1/10$ 倍の 5 通りを調べ、最も計算時間の短いものを選んだ。RIC2S 分解において棄却許容値 τ の値は 0.01, 0.02, 0.05, 0.1, 0.2 の 5 通りを調べ、最も計算時間が短いものを選んだ。また、MRIC2S 分解において棄却許容値 τ の値は RIC2S 分解で選んだ値を用い、乗算緩和係数 ω_m ($0 \leq \omega_m \leq 1$) の値は 0.5, 0.4, 0.3, 0.2, 0.1, 0 の 6 通り調べ、最も計算時間が短いものを選んだ。

表 2 の結果から、以下のことがわかる。

- IC(0)CG 法ではどの行列でも収束しなかった。
- 行列“Thin_Head”, “Cube_20”, “Cube_30”に対して、MRIC2S 分解つき CG 法が最も計算時間が短く、対角スケーリングのみの CG 法に比べて計算時間が比で 0.22 ~ 0.31 に減少した。
- 行列“Cusp_Head”に対しては、RIC 分解つき CG 法では対角スケーリングのみの CG 法に比べて計算時間が比で 1.48 と逆に遅くなったが、RIC2S 分解つき CG 法、MRIC2S 分解つき CG 法では安定して収束し、MRIC2S 分解つき CG 法では計算時間が比で 0.50 に減少した。
- MRIC2S 分解つき CG 法は RIC2S 分解つき CG 法よりも速く収束し、行列“Cusp_Head”では RIC2S 分解つき CG 法の計算時間が 222.7 秒、MRIC2S 分解つき CG 法の計算時間が 156.1 秒となり、比で約 0.70 に減少した。

図 2 に、行列“Cusp_Head”における、前処理つき CG 法の相対残差の収束履歴を示す。横軸は計算時間、縦軸は相対残差 (常用対数目盛) である。図からわかるように、MRIC2S 分解つき CG 法の収束は他の前処理つき CG 法と比べて最も速い。

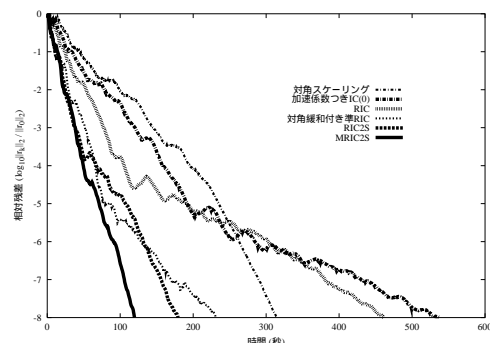


図 2: 対角スケーリング, 加速係数つき IC(0), RIC, 対角緩和つき準 RIC, RIC2S, MRIC2S 前処理付き CG 法の残差履歴 (行列“Cusp_Head”のとき)。

また、図 3 に、行列“Thin_Head”における RIC 分解, RIC2S 分解, MRIC2S 分解の棄却許容値 τ と反復回数の関係を示す。横軸は棄却許容値 τ , 縦軸は反復回数である。ただし、MRIC2S 分解において乗算緩和係数 ω_m の値はすべて 0.1 とした。図より、同じ棄却許容値 τ 同士で比較すると、RIC 分解に比べて RIC2S 分解は非常に速く収束していることがわかる。また、どの棄却許容値でも MRIC2S 分解は RIC2S 分解に比べて速く収束している。

表 1: テスト行列の特徴 .

行列	次元数	非零要素数	非零要素数 要素数/行	解析対象物
Thin_Head	174,395	2,846,205	16.3	薄膜磁気記録ヘッド
Cube_20	438,440	3,733,629	8.5	立方体型磁性体
Cube_30	1,490,460	12,790,849	8.6	立方体型磁性体
Cusp_Head	1,179,789	19,800,735	16.8	カスプコイル励磁型単磁極ヘッド

表 2: テスト行列に対する前処理つき CG 法の反復回数, 計算時間 [秒] とメモリ量の比較 .

行列	前処理	τ (加速係数)	$\omega_m(\omega_d)$	反復 回数	前処理	CG 反復	合計	比	メモリ [MB]
Thin_Head	対角 S	-	-	1415	-	43.3	43.3	1.00	41
	IC(0)	-	-	max	-	-	-	-	-
	加速 IC(0)	(2.38)	-	550	1.4	34.5	35.9	0.83	43
	RIC	0.01	-	258	3.7	21.6	25.2	0.58	118
	緩和 RIC	0.05	(1/10)	237	1.1	14.1	15.3	0.35	79
	RIC2S MRIC2S	0.05 0.1	- 0.1	156 224	7.2 2.5	8.9 11.1	16.1 13.6	0.37 0.31	148 105
Cube_20	対角 S	-	-	2104	-	84.1	84.1	1.00	64
	IC(0)	-	-	max	-	-	-	-	-
	加速 IC(0)	(1.86)	-	905	1.3	75.9	77.1	0.92	68
	RIC	0.01	-	258	2.9	28.3	31.2	0.37	174
	緩和 RIC	0.01	(1/10)	185	3.5	21.9	25.4	0.30	183
	RIC2S MRIC2S	0.05 0.05	- 0.0	268 200	6.0 6.6	21.7 16.2	27.7 22.7	0.33 0.27	237 242
Cube_30	対角 S	-	-	3012	-	566.7	566.7	1.00	220
	IC(0)	-	-	max	-	-	-	-	-
	加速 IC(0)	(1.86)	-	1294	4.4	438.2	442.6	0.78	232
	RIC	0.01	-	378	11.1	159.9	171.0	0.30	596
	緩和 RIC	0.01	(1/10)	297	13.5	134.3	147.8	0.26	629
	RIC2S MRIC2S	0.05 0.05	- 0.2	418 308	23.3 24.2	137.9 101.4	161.1 125.6	0.28 0.22	815 832
Cusp_Head	対角 S	-	-	1391	-	314.9	314.9	1.00	285
	IC(0)	-	-	max	-	-	-	-	-
	加速 IC(0)	(2.46)	-	1162	10.4	537.8	548.2	1.74	294
	RIC	0.05	-	1178	6.2	461.3	467.5	1.48	504
	緩和 RIC	0.05	(1/10)	535	7.4	232.8	240.2	0.76	532
	RIC2S MRIC2S	0.05 0.05	- 0.1	432 283	46.4 43.9	176.4 112.2	222.7 156.1	0.71 0.50	993 940

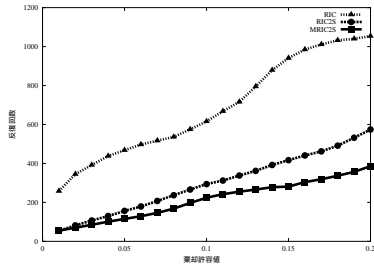


図 3: RIC 分解, RIC2S 分解, MRIC2S 分解の棄却許容値 τ と反復回数の関係 (行列 “Thin_Head” のとき).

6.4 乗算緩和係数 ω_m について

図 4 に行列 “Cusp_Head” に対する, MRIC2S 分解つき CG 法の乗算緩和係数 ω_m と反復回数の関係を, 図 5 に行列 “Cube_20” に対する, MRIC2S 分解つき CG 法の乗算緩和係数 ω_m と反復回数の関係各々示す, 図において, 横軸に乗算緩和係数 ω_m , 縦軸に合計の反復回数をとった. 図より, ω_m の値が 1.0 から小さくなるに従って単調に計算時間が減少しているという特徴が読み取れる. 行列 “Thin_Head”, 行列 “Cube_30” でも同様の特徴が見られた. したがって, 上の特徴的な性質を利用すると, 乗算緩和係数 ω_m の最適値は, 小さい値 (例えば, $\omega_m = 0$) から調べていくことで, 早く見つかる可能性が高いと考えられる.

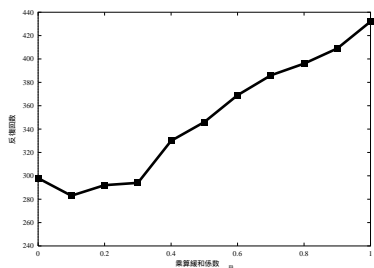


図 4: MRIC2S 分解つき CG 法の乗算緩和係数 ω_m と反復回数の関係 (行列 “Cusp_Head” のとき).

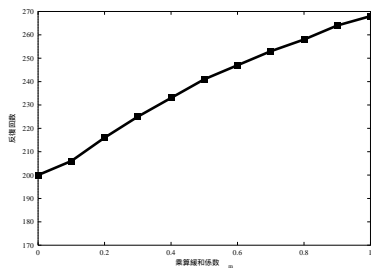


図 5: MRIC2S 分解つき CG 法の乗算緩和係数 ω_m と反復回数の関係 (行列 “Cube_20” のとき).

7 まとめ

本論文では, スペクトル条件数が棄却許容値の 2 次のオーダーで抑えられる RIC2S 分解において, 要素を棄却するときの対角項の修正量を緩和する処理を行うことで収束性を向上させた MRIC2S (Modified RIC2S) 分解前処理を提案した. そして, 数値実験により, その有効性を検証した. RIC 分解において収束性が悪かった行列 “Cusp_Head” に対して, RIC2S 分解, MRIC2S 分解は対角スケールリングのみの CG 法に比べて計算時間が比で 0.50 に減少した. また, MRIC2S 分解は, RIC2S 分解に比べ, 行列 “Cusp_Head” では計算時間が比で約 0.70 に減少し, 高い収束性を示した. これにより, MRIC2S 分解の有効性が確かめられた.

謝辞

貴重な行列データを提供していただいた岡山大学 藤原耕二助教授, サイエンスソリューションズ 亀有昭久氏, 佐賀大学の村松和弘助教授に心より感謝の意を表します.

参考文献

- [1] Ajiz, M. A., Jennings, A.: A robust incomplete Choleski-conjugate gradient algorithm, *Int. J. Numer. Methods Engrg.*, Vol. 20, pp. 949–966 (1984).
- [2] Axelsson, O., Kaporin, I., Konshin, I., Kucherov, A., Neytcheva, M., Polman, B., et al.: Comparison of algebraic solution methods on a set of benchmark problems in linear elasticity, *Tech. Report of Department of Mathematics*, University of Nijmegen, The Netherlands (2000).
- [3] Benzi, M., Tuma, M.: A robust incomplete factorization preconditioner for positive definite matrices, *Numer. Lin. Alg. Appl.*, Vol. 10, pp. 385–400 (2003).
- [4] Fujiwara, K., Ikeda, F., Kameari, A., Kanai, Y.: Thin film write head field analysis using a benchmark problem, *IEEE Trans. Magn.*, Vol. 36, pp. 1784–1787 (2000).
- [5] 池田優介, 藤野清次, 柿原正伸, 井上明彦: A-直交過程に基づく RIF 前処理の効率化について, 情報処理学会 コンピューティングシステム, Vol. 45, No. SIG6 (ACS6), pp. 95–104 (2004).
- [6] 井上明彦, 柿原正伸, 藤野清次, 南里豪志: フィルインを考慮しない RIC(0) 分解つき CG 法の収束性評価, 日本応用数理学会 2004 年度年会 講演予稿集, pp. 68–69 (2004).
- [7] 柿原正伸, 藤野清次: 構造解析で現れる線形方程式に対する対角緩和つき準ロバスト ICCG 法の収束性評価, 日本計算工学会論文集, No. 20040020 (2004).
<http://save.k.u-tokyo.ac.jp/jsces/trans/trans2004/No20040020.pdf>
- [8] Kaporin, I. E.: High quality preconditioning of a general symmetric positive definite matrix based on its $U^T U + U^T R + R^T U$ -decomposition, *Numer. Lin. Alg. Appl.*, Vol. 5, pp. 483–509 (1998).
- [9] 大竹雅哉, 高橋則雄, 品川公成: 三次元有限要素法を用いたカスプコイル励磁型単磁極ヘッドの記録・再生シミュレーション, 電子情報通信学会技術研究報告, MR2003-19 (2003).
- [10] 杉浦洋: 数値計算の基礎と応用 - 数値解析学への入門 -, サイエンス社 (2000).
- [11] Tismenetsky, M.: A new preconditioning technique for solving large sparse linear systems, *Lin. Alg. Appl.*, Vol. 154-156, pp. 331–353 (1991).