

準疎行列係数連立一次方程式に対する  
前処理法の研究

山崎 育朗

システム情報工学研究科

筑波大学

2013年11月

## 論文要旨

本論文では、非零要素数が多く、絶対値の小さい要素を多く含む行列（準疎行列）を係数行列にもつ連立一次方程式（準疎行列係数連立一次方程式）に対し、Krylov 部分空間反復解法の収束性を向上させる方法である前処理法に関する研究結果を示す。準疎行列係数連立一次方程式には、疎行列化手法である Cutoff を用いた前処理法が有効である。そのため、この Cutoff を用いた前処理法に対する 2 点の研究結果を示す。

まず、高い計算並列性をもつ従来の前処理法に着目し、同前処理法を準疎行列係数連立一次方程式に適したものに改良することを提案する。次に、Cutoff を適用するにあたり Cutoff パラメータを設定する必要がある、このパラメータの自動推定法を提案する。本論文では、それらの手法を様々な分野で現れる準疎行列に対して適用し、結果を示す。

# 目次

<b>第 1 章 序論</b>	<b>1</b>
1.1 反復解法と前処理法	2
1.2 準疎行列係数連立一次方程式に対する前処理法に関する研究成果	2
1.3 本論文の構成	3
<b>第 2 章 連立一次方程式の反復解法と前処理法</b>	<b>4</b>
2.1 Krylov 部分空間反復解法	4
2.1.1 GMRES 法	5
2.1.2 リスタート付き GMRES 法	6
2.1.3 GMRES-IR 法	6
2.1.4 GCR 法・リスタート付き GCR 法	7
2.2 前処理法	9
2.2.1 係数行列 $A$ を近似する前処理法	10
Fill-in を考慮しない不完全分解	10
閾値付き不完全分解	11
2.2.2 逆行列 $A^{-1}$ を近似する前処理法	11
近似逆行列前処理法	11
<b>第 3 章 準疎行列係数連立一次方程式に対する高い計算並列性をもつ前処理法</b>	<b>18</b>
3.1 緒言	18
3.2 ブロック近似逆行列前処理法	19
3.3 Cutoff 付きブロック近似逆行列前処理法	20
3.4 数値実験	21
3.4.1 数値実験 1 (提案法と従来法の比較実験)	21
数値例 1	22
数値例 2	23
数値例 3	24
3.4.2 数値実験 2 (BSAIC における収束性悪化の解析とその対策)	25

数值例 1 . . . . .	25
数值例 2 . . . . .	26
数值例 3 . . . . .	27
3.5 結言 . . . . .	27
<b>第 4 章 近似係数行列に対する完全分解を用いた前処理法の Cutoff パラメータ自動推定</b>	<b>44</b>
4.1 緒言 . . . . .	44
4.2 近似係数行列に対する完全分解を用いた前処理法 . . . . .	45
4.3 Cutoff パラメータの自動推定法 . . . . .	46
4.3.1 残差が収束する Cutoff パラメータの推定 . . . . .	46
4.3.2 最適な Cutoff パラメータの推定 . . . . .	48
4.3.3 自動選択アルゴリズム . . . . .	50
4.4 数値実験 . . . . .	51
4.4.1 数值例 1 . . . . .	51
疎行列用直接解法を用いた前処理法の性能 . . . . .	52
提案法の有効性の検証 . . . . .	52
4.4.2 数值例 2 . . . . .	53
疎行列用直接解法を用いた前処理法の性能 . . . . .	53
提案法の有効性の検証 . . . . .	53
4.5 結言 . . . . .	54
<b>第 5 章 結論</b>	<b>63</b>
参考文献	65
謝辞	68
研究業績一覧	70

# 目次

2.1	Algorithm of Arnoldi method. . . . .	13
2.2	Algorithm of GMRES method. . . . .	13
2.3	Algorithm of GMRES( $m$ ) method. . . . .	14
2.4	Algorithm of IRA method. . . . .	14
2.5	Algorithm of GMRES-IR( $m, k$ ) method. . . . .	15
2.6	Algorithm of GCR method. . . . .	15
2.7	Algorithm of GCR( $m$ ) method. . . . .	16
2.8	Algorithm of preconditioned GMRES( $m$ ) method. . . . .	16
2.9	Algorithm of preconditioned GMRES-IR( $m, k$ ) method. . . . .	17
2.10	Algorithm of preconditioned GCR( $m$ ) method. . . . .	17
3.1	The features of $A$ for raefsky2. . . . .	29
3.2	The computation time of GMRES(50) preconditioned BSAIC corresponding to $l$ for raefsky2 ( $\omega = 0.0$ ). . . . .	30
3.3	The computation time of GMRES(50) preconditioned BSAIC corresponding to $\theta$ for raefsky2 ( $\omega = 0.0$ ). . . . .	30
3.4	The computation time of GMRES(50) preconditioned BSAIC corresponding to $l$ for raefsky2 ( $\omega = 0.1042$ ). . . . .	31
3.5	The computation time of GMRES(50) preconditioned BSAIC corresponding to $\theta$ for raefsky2 ( $\omega = 0.1042$ ). . . . .	31
3.6	The features of $A$ for fp. . . . .	32
3.7	The computation time of GMRES(50) preconditioned BSAIC corresponding to $l$ for fp ( $\omega = 9.0$ ). . . . .	33
3.8	The computation time of GMRES(50) preconditioned BSAIC corresponding to $\theta$ for fp ( $\omega = 9.0$ ). . . . .	33
3.9	The computation time of GMRES(50) preconditioned BSAIC corresponding to $l$ for fp ( $\omega = 0.0$ ). . . . .	34

3.10	The computation time of GMRES(50) preconditioned BSAIC corresponding to $\theta$ for fp ( $\omega = 0.0$ ). . . . .	34
3.11	The features of $A$ for EGF. . . . .	35
3.12	The computation time of GMRES(50) preconditioned BSAIC corresponding to $l$ for EGF ( $\omega = -0.09$ ). . . . .	36
3.13	The computation time of GMRES(50) preconditioned BSAIC corresponding to $\theta$ for EGF ( $\omega = -0.09$ ). . . . .	36
3.14	The computation time of GMRES(50) preconditioned BSAIC corresponding to $l$ for EGF ( $\omega = -0.2$ ). . . . .	37
3.15	The computation time of GMRES(50) preconditioned BSAIC corresponding to $\theta$ for EGF ( $\omega = -0.2$ ). . . . .	37
3.16	Eigenvalue distributions of $AM$ for raefsky2. . . . .	38
3.17	Eigenvalue distributions of $AM$ for fp. . . . .	40
3.18	Eigenvalue distributions of $AM$ for EGF. . . . .	42
4.1	Algorithm of our proposed method. . . . .	55
4.2	The features of $A$ for fem_filter. . . . .	56
4.3	Total time corresponding to $\theta$ for fem_filter. . . . .	57
4.4	The features of $A$ for EGFR2. . . . .	59
4.5	Total time corresponding to $\theta$ for EGFR2. . . . .	60
4.6	Breakdown of Profiling time for EGFR2. . . . .	62

# 第1章 序論

近年、科学技術計算において自然現象や工学現象の解析は重要な課題の一つであり、高性能なコンピュータや大規模並列計算環境上での数値シミュレーションは必要不可欠なものとなっている。このような物理現象や自然科学などをモデル化し離散化することで連立一次方程式

$$Ax = b, A \in \mathbb{C}^{n \times n}, x, b \in \mathbb{C}^n, \quad (1.1)$$

に帰着され、同方程式を解くことで自然現象や工学現象の解析が計算機上で可能になる。一般に、連立一次方程式の求解部分が計算時間の大部分を占めるため、同方程式を高速に解くことが全体の計算時間の大幅な減少につながる。

従来、連立一次方程式 (1.1) の係数行列  $A$  は大規模な疎行列である場合が多かったが、近年、ナノシミュレーションなど様々な分野で以下のような特徴をもつ行列（以下、準疎行列と呼ぶ）が現れている：

$$\left\{ \begin{array}{l} \cdot 1 \text{ 行あたりの非零要素数が多い} \\ \cdot \text{ 行列要素の値が広く分布し、絶対値が小さい要素が多い.} \end{array} \right.$$

本稿では、このような準疎行列を係数行列にもつ連立一次方程式（以下、準疎行列係数連立一次方程式と呼ぶ）を対象とする。

準疎行列係数連立一次方程式は、例えば分子軌道計算に代表されるナノシミュレーションの分野などで現れる固有値問題

$$Fv = \lambda Sv, S, F \in \mathbb{C}^{n \times n}, v \in \mathbb{C}^n \setminus \{0\}, \lambda \in \mathbb{C}, \quad (1.2)$$

を shift-invert Arnoldi 法 [4] や Sakurai-Sugiura 法 [27] などの固有値解法で解く際等に現れる。なお、式 (1.2) は一般化固有値問題を表し、 $S = I$  ( $I$  は単位行列) の場合は標準固有値問題となる。上記の固有値解法で現れる連立一次方程式の係数行列  $A$  は一般化固有値問題の場合

$$A = \omega S - F, S, F \in \mathbb{C}^{n \times n}, \omega \in \mathbb{C}, \quad (1.3)$$

標準固有値問題の場合

$$A = \omega I - F, \quad I, F \in \mathbb{C}^{n \times n}, \quad \omega \in \mathbb{C}, \quad (1.4)$$

で表される。パラメータ  $\omega$  は、求めたい固有値付近に設定される。

## 1.1 反復解法と前処理法

式 (1.3) や式 (1.4) のようにパラメータ  $\omega$  をもつ準疎行列係数連立一次方程式を解くことを考える。一般に、連立一次方程式を解く方法は直接解法と反復解法に大別できる。大規模な行列に対し直接解法で解くことは実用上困難であるため、疎行列向きの解法の一種である Krylov 部分空間反復解法 [6] を用いる。Krylov 部分空間反復解法は大規模な疎行列を係数行列にもつ連立一次方程式に対しては直接解法を用いるより計算量が少なく、高速に解くことのできる可能性がある解法である。

しかしながら、同法のみを用いると求解できない場合が多々あるため、収束性を改善するために同法は前処理法と組み合わせて用いる必要がある。前処理法とは、連立一次方程式を前もって反復解法で容易に解ける形に同値変換することにより、Krylov 部分空間反復解法の収束性を向上させる方法である。前処理法には様々な方法があるが、従来の前処理法は疎行列向きであるため、準疎行列には有効でない場合がある。

## 1.2 準疎行列係数連立一次方程式に対する前処理法に関する研究成果

準疎行列に対して Cutoff と呼ばれる疎行列化手法が有効であることが知られている [20]。本稿では、準疎行列係数連立一次方程式に対する Cutoff を用いた前処理法に関して行った研究結果を 2 点示す。

1 点目は高い並列性能をもつ準疎行列係数連立一次方程式に対する前処理法の提案である。マルチコア CPU や GPU クラスタなどの普及に伴い、並列計算が重視されている。そこで、既存の計算並列性が高い前処理法を準疎行列向きに改良した前処理法を提案し、提案法の有効性をいくつかの数値実験で検証する。また、提案法の問題点についても調査を行い、改善策を示す。

2 点目は前処理法に用いるパラメータの自動推定法の提案である。先行研究で提案されている準疎行列係数連立一次方程式に対して有効な前処理法 [20] は、提案法よりも設定するパラメータが少なく、前処理法の性能が Cutoff パラメータというパラメータにのみ左右されることに着目し、同前処理法における Cutoff パラメータの自動推定法を提案する。



### 1.3 本論文の構成

本稿の構成は以下のようになっている。まず、第 2 章では連立一次方程式の求解方法として Krylov 部分空間反復解法について述べ、反復解法に対し収束性を向上させるために適用する前処理法についても述べる。第 3 章では高い計算並列性をもつ準疎行列係数連立一次方程式に対する前処理法として Cutoff 付きブロック近似逆行列前処理法に関して述べる。さらに、第 4 章では先行研究の前処理法 [20] における Cutoff パラメータを自動で推定する方法について述べる。最後に、第 5 章で本稿において得られた結論と課題をまとめる。

なお、本論文を通して内積は  $(x, y) \equiv x^H y$  で定義し、記号  $^H$  は共役転置を表す。

## 第2章 連立一次方程式の反復解法と前処理法

本章では、Krylov 部分空間反復解法、非 Hermite 行列向き解法の一つである GMRES 法、リスタート付き GMRES 法、GMRES-IR 法、GCR 法、リスタート付き GCR 法について述べる。また、Krylov 部分空間反復解法の収束性を向上させるための前処理法についても述べる。

なお、以下の記述では  $W$  ノルムとして  $\|\mathbf{x}\|_W = \mathbf{x}^H W \mathbf{x}$ 、 $W$  内積として  $(\mathbf{x}, \mathbf{y})_W = \mathbf{x}^H W \mathbf{y}$  をそれぞれ用いる。ここで、記号  $^H$  は共役転置を表す。

### 2.1 Krylov 部分空間反復解法

$n \times n$  行列  $A$  を係数行列にもつ連立一次方程式  $A\mathbf{x} = \mathbf{b}$  を Krylov 部分空間反復解法で解くことを考える。ここで、 $\mathbf{x}$  を解ベクトル、 $\mathbf{b}$  を右辺ベクトルとする。同方程式の数値解を効率よくかつ精度よく求めることは科学技術計算において重要な位置を占めるテーマの1つである。連立一次方程式の数値解法は、直接解法と反復解法に大別できる。直接解法は全体の演算量が  $O(n^3/3)$  に比例し、メモリや計算時間を膨大に必要とするため、小規模な密行列や帯行列などに使用される。また、Krylov 部分空間反復解法は一回あたりの反復の演算量が  $O(n^2)$  に比例し、非零要素数が少なくなるに従い計算時間も短くなるため、大規模な疎行列に用いられることが多い。本研究では、大規模疎行列向き解法である Krylov 部分空間反復解法に着目する。

Krylov 部分空間反復解法では、 $k$  番目の近似解  $\mathbf{x}_k$  は

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(A; \mathbf{r}_0)$$

を満たすように計算される。ここで、 $\mathbf{x}_0$  は初期解、 $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$  は初期残差を表す。また、 $\mathcal{K}_k(A; \mathbf{r}_0)$  は行列  $A$  と初期残差  $\mathbf{r}_0$  によって生成される Krylov 部分空間

$$\text{span}(\mathbf{r}_0, A\mathbf{r}_0, \dots, A^{k-1}\mathbf{r}_0)$$

である。このとき、対応する残差  $\mathbf{r}_k$  は

$$\mathbf{r}_k \in \mathcal{K}_{k+1}(A; \mathbf{r}_0)$$

を満たす。しかしながら、この条件だけでは近似解は一意に定まらないため、残差  $\mathbf{r}_k$  に対してある条件を課すことで近似解が一意に定められる。

### 2.1.1 GMRES 法

本節では、係数行列  $A$  を非 Hermite 行列とし、Krylov 部分空間反復解法に属する代表的な解法である GMRES 法（一般化最小残差法：Generalized Minimal RESidual method）[24] に関して述べる。

GMRES 法とは、 $k$  回目の反復において、Krylov 部分空間  $\text{span}(\mathbf{r}_0, A\mathbf{r}_0, \dots, A^{k-1}\mathbf{r}_0)$  に属する全てのベクトル  $\mathbf{x}_k$  から残差の 2 ノルム  $\|\mathbf{r}_k\|_2 = \|\mathbf{b} - A\mathbf{x}_k\|_2$  を最小化するような近似解  $\mathbf{x}_k$  を見つけ出す方法である。つまり、

$$\begin{aligned} \mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(A; \mathbf{r}_0) \quad \text{s.t.} \quad \|\mathbf{b} - A\mathbf{x}_k\|_2 \rightarrow \min \\ \mathcal{K}_k(A; \mathbf{r}_0) = \text{span}(\mathbf{r}_0, A\mathbf{r}_0, \dots, A^{k-1}\mathbf{r}_0) \end{aligned}$$

である。

GMRES 法は Arnoldi 法 [3] を用いて、Krylov 部分空間  $\mathcal{K}_k(A; \mathbf{r}_0)$  の正規直交基底  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  を作成する。Fig. 2.1 に Arnoldi 法のアルゴリズムを示す。なお、正規直交基底  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  を作成するにあたり Fig. 2.1 において  $W = I$  としている。Arnoldi 法における行列ベクトル積が GMRES 法の主要演算となる。 $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  は Krylov 部分空間  $\mathcal{K}_k(A; \mathbf{r}_0)$  の基底であるため、 $\mathbf{x}_0 + \mathcal{K}_k(A; \mathbf{r}_0)$  に属するベクトル  $\mathbf{x}_k$  は以下のように表すことができる。

$$\mathbf{x}_k = \mathbf{x}_0 + V_k \mathbf{y}_k.$$

$V_k$  は Krylov 部分空間  $\mathcal{K}_k(A; \mathbf{r}_0)$  の正規直交基底  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  を並べた行列を表し、 $\mathbf{y}_k$  は  $k$  次ベクトルである。また、以下のような関係が得られる。

$$\begin{aligned} AV_k &= V_k H_k + h_{k+1,k} \mathbf{v}_{k+1} \mathbf{e}_k^T \\ &= V_{k+1} \bar{H}_k \end{aligned} \tag{2.1}$$

ただし、 $\bar{H}_k$  は  $(k+1) \times k$  の上 Hessenberg 行列であり、 $H_k$  は  $\bar{H}_k$  の最後の行を取り除いた上

Hessenberg 行列である。以上を用いて、 $\mathbf{x}_k$  に対する残差  $\mathbf{r}_k$  を計算する。

$$\begin{aligned}
 \mathbf{r}_k &= \mathbf{b} - A\mathbf{x}_k \\
 &= \mathbf{b} - A(\mathbf{x}_0 + V_k\mathbf{y}_k) \\
 &= \mathbf{r}_0 - V_{k+1}\overline{H}_k\mathbf{y}_k \\
 &= \beta\mathbf{v}_1 - V_{k+1}\overline{H}_k\mathbf{y}_k \\
 &= V_{k+1}(\beta\mathbf{e}_1 - \overline{H}_k\mathbf{y}_k)
 \end{aligned} \tag{2.2}$$

ただし、 $\beta = \|\mathbf{r}_0\|_2$ 、 $\mathbf{e}_1^T = (1, 0, \dots, 0)$  であり、 $\mathbf{v}_1 = \mathbf{r}_0 / \|\mathbf{r}_0\|_2$  であることから  $\mathbf{r}_0 = \beta\mathbf{v}_1$  となる。これより、残差ベクトルの2ノルム  $\|\mathbf{r}_k\|_2$  を最小にするには次の最小二乗問題を解けばよい。

$$\operatorname{argmin}_{\mathbf{y}_k} \|\beta\mathbf{e}_1 - \overline{H}_k\mathbf{y}_k\|_2$$

以上の計算を基に構築された GMRES 法のアルゴリズムを Fig. 2.2 に示す。

### 2.1.2 リスタート付き GMRES 法

GMRES 法は残差ノルムが小さくなるまで反復を繰り返す方法であり、正規直交基底のベクトル列  $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$  も収束するまで増え続ける。従って、反復回数が増えるに従い  $V_k$  と  $\overline{H}_k$  を保存していく必要がある。これは計算量及びメモリ使用量の点から実用的ではない。そのため、リスタートを行うのが一般的である。これをリスタート付き GMRES 法 (GMRES( $m$ ) 法) [24] という。GMRES( $m$ ) 法は  $m$  回反復時の解を初期解とし、再び GMRES( $m$ ) 法を適用し残差が収束判定条件を満たした時に反復を終了させる方法である。GMRES( $m$ ) 法のアルゴリズムを Fig. 2.3 に示す。

GMRES( $m$ ) 法は GMRES 法よりも計算量及びメモリ使用量が少ないが、収束性が悪化する。この原因はこれまでに求めてきた正規直交基底を破棄したため、精度の良い近似解を得られなくなるためである。他の場合として、解くべき行列 (係数行列  $A$ ) に絶対値が小さい固有値が存在する場合にも収束性が悪化することがある。この問題点を改善した方法として Morgan によって提案された Implicitly Restarted GMRES (GMRES-IR) 法 [16, 18] がある。

### 2.1.3 GMRES-IR 法

小さな値の固有値を取り除く (デフレーションする) ために、GMRES( $m$ ) 法を改良した GMRES-IR( $m, k$ ) 法を適用することを考える。 $k$  は取り除く固有値の数である。2.1.2 節で述べ

たように解くべき行列  $A$  に小さな値の固有値が存在することが収束性の悪化に繋がる。収束性を改善するには、この小さな値の固有値を取り除けば良い。しかしながら、 $A$  は大規模行列であるため固有値を求めることは困難である。そこで、式 (2.1) から  $A$  の近似固有値（調和 Ritz 値, Harmonic Ritz value）[21] を QR 法 [10] で算出する [14]。計算した調和 Ritz 値の絶対値が大きい  $p (= m - k)$  個をシフトに選び、固有値の一部を求める方法である Implicitly restarted Arnoldi (IRA)[29] 法により絶対値の小さい調和 Ritz 値に対応する調和 Ritz ベクトルが張る  $k$  次元の Krylov 部分空間を求める。この部分空間を次に生成する探索空間に含ませてリスタートする。IRA 法及び GMRES-IR( $m, k$ ) 法のアルゴリズムを Fig. 2.4, Fig. 2.5 にそれぞれ示す。

### 2.1.4 GCR 法・リスタート付き GCR 法

本節では、GCR 法（一般化共役残差法：Generalized Conjugate Residual method）[9, 11] に関して述べる。

GCR 法は 1983 年に Eisenstat らによって提案された非 Hermite 連立一次方程式向けの Krylov 部分空間反復解法である。同解法は、理論上は 2.1.1 節で述べた GMRES 法と同様に同じ残差を各反復毎に生成するが、近似解の計算を漸化式で行うため GMRES 法のように最小二乗問題を解く必要がない。

GCR 法は GMRES 法と同様 Fig. 2.1 を用い  $W = A^H A$  とし、 $A^H A$  正規直交系を作成する。2.1.1 節同様に正規直交基底  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  を並べた行列  $V_k$  を用いると  $\mathbf{x}_k = \mathbf{x}_0 + V_k \mathbf{y}_k$  となり、残差は式 (2.2) から

$$\mathbf{r}_k = \mathbf{r}_0 - AV_k \mathbf{y}_k = \mathbf{r}_0 - \sum_{i=1}^k y_i^{(k)} A\mathbf{v}_i \quad (2.3)$$

と表せる。ここで、 $y_i^{(k)}$  はベクトル  $\mathbf{y}_k$  の  $i$  番目の要素を表す。

$y_i^{(k)}$  は  $\mathbf{r}_k$  のノルムを最小化するよう決められ、 $\{A\mathbf{v}_1, A\mathbf{v}_2, \dots, A\mathbf{v}_k\}$  は正規直交系であるため

$$y_i^{(k)} = (A\mathbf{v}_i, \mathbf{r}_0)$$

になる。また、 $y_i^{(n)}$  が  $n$  に依存しないことに着目すると、式 (2.3) から次の漸化式が得られる。

$$\mathbf{r}_k = \mathbf{r}_{k-1} - y_k A\mathbf{v}_k \quad (2.4)$$

ここで、 $\mathbf{p}_k = y_k A \mathbf{w}_{k+1}$  と定義し、Fig. 2.1 の4~7行目を考えると

$$\mathbf{p}_k = y_k A \mathbf{v}_k - \sum_{i=1}^k (\mathbf{v}_i, y_k A \mathbf{v}_k)_{A^H A} \mathbf{v}_i$$

式(2.4)を上式に代入すると

$$\mathbf{p}_k = \mathbf{r}_k - \sum_{i=1}^k (\mathbf{v}_i, \mathbf{r}_k)_{A^H A} \mathbf{v}_i - (\mathbf{r}_{k-1} - \sum_{i=1}^k (\mathbf{v}_i, \mathbf{r}_{k-1})_{A^H A} \mathbf{v}_i)$$

が得られる。残差  $\mathbf{r}_{m-1}$  は Krylov 部分空間  $K_m$  のベクトルであるため、 $K_m$  の  $A^H A$  正規直交系  $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m\}$  で展開すると、

$$\mathbf{r}_{k-1} = \sum_{i=1}^k (\mathbf{v}_i, \mathbf{r}_{k-1})_{A^H A} \mathbf{v}_i$$

となるため、補助ベクトル  $\mathbf{p}_k$  は

$$\mathbf{p}_k = \mathbf{r}_k - \sum_{i=1}^k (\mathbf{v}_i, \mathbf{r}_k)_{A^H A} \mathbf{v}_i$$

で表せる。また、 $\mathbf{p}_k$  の定義から、 $\mathbf{p}_k = \pm \|A \mathbf{p}_k\| \mathbf{v}_{k+1}$  と書け、次式が成り立つ。

$$\begin{aligned} (\mathbf{v}_i, \mathbf{r}_k)_{A^H A} \mathbf{v}_i &= \frac{(A \mathbf{p}_i, A \mathbf{r}_k)}{(A \mathbf{p}_i, A \mathbf{p}_i)} \mathbf{p}_i \\ y_k A \mathbf{v}_k = (\mathbf{r}_0, A \mathbf{v}_k) A \mathbf{v}_k &= \frac{(A \mathbf{p}_{k-1}, \mathbf{r}_0)}{(A \mathbf{p}_{k-1}, A \mathbf{p}_{k-1})} A \mathbf{p}_{k-1} \end{aligned}$$

従って、

$$\begin{aligned} \mathbf{p}_k &= \mathbf{r}_k - \sum_{i=0}^{k-1} \frac{(A \mathbf{p}_i, A \mathbf{r}_k)}{(A \mathbf{p}_i, A \mathbf{p}_i)} \mathbf{p}_i \\ \mathbf{r}_k &= \mathbf{r}_{k-1} - \frac{(A \mathbf{p}_{k-1}, \mathbf{r}_0)}{(A \mathbf{p}_{k-1}, A \mathbf{p}_{k-1})} A \mathbf{p}_{k-1} \end{aligned}$$

が成り立つ。一方、

$$\mathbf{r}_{k-1} = \mathbf{r}_0 - \sum_{i=1}^{k-1} y_i A \mathbf{v}_i$$

から,  $\mathbf{p}_{k-1}$  は  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{k-1}$  と  $A^H A$  直交であるため,

$$(A\mathbf{p}_{k-1}, \mathbf{r}_{k-1}) = (A\mathbf{p}_{k-1}, \mathbf{r}_0)$$

となり, ここで

$$\alpha_k = \frac{(A\mathbf{p}_k, \mathbf{r}_k)}{(A\mathbf{p}_k, A\mathbf{p}_k)}, \quad \beta_{i,k} = -\frac{(A\mathbf{p}_i, A\mathbf{r}_{k+1})}{(A\mathbf{p}_i, A\mathbf{p}_i)}$$

とおくと

$$\begin{aligned} \mathbf{p}_k &= \mathbf{r}_k + \sum_{i=0}^{k-1} \beta_{i,k-1} \mathbf{p}_i \\ \mathbf{r}_k &= \mathbf{r}_{k-1} - \alpha_{k-1} A\mathbf{p}_{k-1} \end{aligned}$$

が得られ, 近似解  $\mathbf{x}_k$  に関する漸化式も

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

のように得られる.

GCR 法及びそのリスタート版である GCR( $m$ ) 法のアルゴリズムを Fig. 2.6, Fig. 2.7 にそれぞれ示す.

## 2.2 前処理法

本節では, Krylov 部分空間反復解法と組み合わせて用いることで反復解法の収束性を向上させる前処理法に関して述べる.

$n \times n$  行列  $A$  を係数行列にもつ連立一次方程式  $A\mathbf{x} = \mathbf{b}$  を Krylov 部分空間反復解法で解くことを考える. ここで,  $\mathbf{x}$  は解ベクトル,  $\mathbf{b}$  は右辺ベクトルである. 前節における Krylov 部分空間反復解法のみで連立一次方程式  $A\mathbf{x} = \mathbf{b}$  を解く場合, 残差が収束せず求解が不可能であったり, 求解は可能だが反復回数が多くなり求解に要する時間が膨大になることがある. このような場合, 収束性を向上させ, 求解を可能にし, また反復回数を減少させるため前処理法を用いることが一般的である.

前処理法とは, 連立一次方程式の係数行列が単位行列に近いとき真の解へ高速に収束することに着目し, 係数行列  $A$  を単位行列  $I$  に近似させるような前処理行列  $K$  を生成し, 連立一次方程式  $A\mathbf{x} = \mathbf{b}$  を同値変換することである. 前処理法を適用した GMRES( $m$ ) 法, GMRES-IR( $m, k$ ) 法及び GCR( $m$ ) 法のアルゴリズムをそれぞれ Fig. 2.8, Fig. 2.9, Fig. 2.10 に示す.

前処理法の種類として、係数行列  $A$  を近似する前処理法と逆行列  $A^{-1}$  を近似する前処理法に大別できる。

### 2.2.1 係数行列 $A$ を近似する前処理法

係数行列  $A$  を近似する前処理法は、係数行列  $A$  を

$$A \approx M_1 M_2 (= K) \iff M_1^{-1} A M_2^{-1} \approx I$$

のように不完全に分解し、連立一次方程式  $Ax = b$  を

$$\underbrace{(M_1^{-1} A M_2^{-1})}_{\tilde{A}} \underbrace{(M_2 x)}_{\tilde{x}} = \underbrace{M_1^{-1} b}_{\tilde{b}} \quad (2.5)$$

と同値変換する方法である。なお、式 (2.5) の場合には両側前処理、 $M_1 = I, M_2 \approx A$  とした場合には右前処理、 $M_1 \approx A, M_2 = I$  とした場合には左前処理と呼ばれる。

$\tilde{A}$  は単位行列に近いほど、解くことが容易である。実際の前処理行列  $K$  としては、 $M_1$  と  $M_2$  の逆行列が少ない手間で計算が可能な方法が要求される。

係数行列  $A$  を近似する前処理法として従来良く用いられる方法は  $A$  を

$$A \approx LU$$

のように不完全分解する方法である。ここで、 $L$  は下三角行列、 $U$  は上三角行列である。係数行列  $A$  を近似する前処理法の代表例として、Fill-in を考慮しない不完全分解及び閾値付き不完全分解 [17, 25, 26] に関して述べる。

#### Fill-in を考慮しない不完全分解

Fill-in を考慮しない不完全分解 (Incomplete LU factorization with 0 level of Fill-in, ILU(0)) は以下のように定義される。係数行列  $A$  が非 Hermite 行列とした場合、 $A$  を

$$A = LU + R \quad (a_{ij} = 0 \Rightarrow l_{ij}, u_{ij} = 0)$$

と分解を行う。ここで  $L$  は下三角行列、 $U$  は上三角行列、 $R$  は行列  $A$  の不完全さを表す行列であり、係数行列  $A$  の  $i, j$  要素  $a_{ij}$  が 0 であるならば  $L, U$  の要素は計算せずに  $l_{ij} = 0, u_{ij} = 0$  とする。



### 閾値付き不完全分解

閾値付き不完全分解 (Incomplete LU factorization with threshold dropping, ILUT) について述べる. この方法では, 棄却則として係数行列  $A$  の要素  $a_{ij}$  が 0 でも  $l_{ij}, u_{ij}$  を計算し,  $l_{ij}, u_{ij}$  の絶対値が閾値  $\delta$  を超えない場合,  $l_{ij} = 0, u_{ij} = 0$  とする. すなわち,

$$A = LU + R \quad ( |l_{ij}| \leq \delta \Rightarrow l_{ij} = 0, |u_{ij}| \leq \delta \Rightarrow u_{ij} = 0 )$$

である. なお,  $\delta = 0$  の場合は完全分解となり,  $R = O$  である.

### 2.2.2 逆行列 $A^{-1}$ を近似する前処理法

係数行列  $A$  の逆行列  $A^{-1}$  を近似する前処理法は,

$$A^{-1} \approx M \iff AM \approx I$$

のように  $A^{-1}$  を近似するような前処理行列  $M$  を作成し, 連立一次方程式  $Ax = b$  を

$$\begin{aligned} \underbrace{(AM)}_{\tilde{A}} \underbrace{(M^{-1}x)}_{\tilde{x}} &= b \\ \tilde{A} \tilde{x} &= b \end{aligned}$$

と同値変換する方法である.

$M^{-1}$  が  $A^{-1}$  を良く近似する, つまり  $\tilde{A}$  が単位行列に近づくと Krylov 部分空間反復解法の収束性が向上する.

逆行列  $A^{-1}$  を近似する前処理法の代表例として, 近似逆行列前処理法 (Sparse Approximate Inverse) [7] がある. 2.2.2 節及び 3.2 節では近似逆行列前処理法に関して述べる.

### 近似逆行列前処理法

本節では, 近似逆行列前処理法 (Sparse Approximate Inverse, SAI) [7] に関して述べる. SAI は  $A$  を係数行列とするとき逆行列  $A^{-1}$  を近似する前処理行列  $M$  を生成する前処理法であり, 以下に示すようなフロベニウスノルム最小化に基づく手法 [8, 13, 15, 22] である.

$$\min_M \|AM - I\|_F^2, \tag{2.6}$$

ただし、 $I$ は単位行列を表し、 $M$ は疎行列である。上記のフロベニウスノルム  $\|AM - I\|_F^2$  は

$$\|AM - I\|_F^2 = \sum_{k=1}^n \|Am_k - e_k\|_2^2, \quad (2.7)$$

のように表すことができる。ここで、 $m_k$ は前処理行列  $M$ の第  $k$ 番目の列ベクトルであり、 $e_k$ は単位行列  $I$ の第  $k$ 番目の列ベクトルである。そのため、前処理行列  $M = [m_1, m_2, \dots, m_n] \approx A^{-1}$ は以下の  $n$ 個の最小二乗問題を解くことで作成される。

$$\min_{m_k} \|Am_k - e_k\|_2^2, \quad k = 1, 2, \dots, n.$$

この  $n$ 個の最小二乗問題は互いに独立であるため、マルチコア CPU やクラスタなどの計算機環境で並列処理が可能となる。また、前処理行列  $M$ の非零構造は任意に設定することができ、係数行列  $A$ の非零構造を採用する場合は

$$\text{spy}(M) = \text{spy}(A),$$

となる。ここで "spy" は行列の非零構造を表している。

**Algorithm** Arnoldi method

- 1: Choose a vector  $\mathbf{v}_1$  such that  $\|\mathbf{v}_1\|_W = 1$
- 2: **For**  $j = 1, 2, \dots, m$  do:
- 3:   **Compute**  $\mathbf{w}_j = A\mathbf{v}_j$
- 4:   **For**  $i = 1, 2, \dots, j$  do:
- 5:     **Compute**  $h_{i,j} = (\mathbf{v}_i, \mathbf{w}_j)_W$
- 6:     **Compute**  $\mathbf{w}_j = \mathbf{w}_j - h_{i,j}\mathbf{v}_i$
- 7:   **End**
- 8:   **Compute**  $h_{j+1,j} = \|\mathbf{w}_j\|_W$ . **If**  $h_{j+1,j} = 0$  **Stop**
- 9:   **Compute**  $\mathbf{v}_{j+1} = \mathbf{w}_j/h_{j+1,j}$
- 10: **End**

Fig. 2.1: Algorithm of Arnoldi method.

**Algorithm** Generalized Minimal RESidual (GMRES) method

- 1:  $\mathbf{x}_0$  is an initial guess, **Compute**  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$
- 2: **Compute**  $\beta = \|\mathbf{r}_0\|_2$  and  $\mathbf{v}_1 = \mathbf{r}_0/\beta$
- 3: **Compute**  $V_k, \overline{H}_k$  with Arnoldi method
- 4: **Define** the  $(k+1) \times k$  Hessenberg matrix  $\overline{H}_k = \{h_{ij}\}_{1 \leq i \leq k+1, 1 \leq j \leq k}$
- 5: **Compute**  $\mathbf{y}$ , the minimizer of  $\|\beta\mathbf{e}_1 - \overline{H}_k\mathbf{y}\|_2$ , and  $\mathbf{x}_k = \mathbf{x}_0 + V_k\mathbf{y}$

Fig. 2.2: Algorithm of GMRES method.

**Algorithm** GMRES( $m$ ) method

- 1:  $\mathbf{x}_0$  is an initial guess, **Compute**  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$
- 2: **Compute**  $\beta = \|\mathbf{r}_0\|_2$  and  $\mathbf{v}_1 = \mathbf{r}_0/\beta$
- 3: **Compute**  $V_m, \overline{H}_m$  with Arnoldi method
- 4: **Compute**  $\mathbf{y}$ , the minimizer of  $\|\beta\mathbf{e}_1 - \overline{H}_m\mathbf{y}\|_2$ , and  $\mathbf{x}_m = \mathbf{x}_0 + V_m\mathbf{y}$
- 5: **If** satisfied **Stop**, **else Set**  $\mathbf{x}_0 = \mathbf{x}_m$  and go to 1

Fig. 2.3: Algorithm of GMRES( $m$ ) method.

**Algorithm** Implicitly Restarted Arnoldi (IRA) method

- 1: **Input**  $m, k$  ( $m > k$ )
- 2: **Set**  $p = m - k$
- 3: **Input** Hessenberg matrix  $H_m$  and shift  $\mu_1, \mu_2, \dots, \mu_p$
- 4: **Set**  $Q = I_m$
- 5: **For**  $j = 1, 2, \dots, p$  **do**:
- 6:   **Factor**  $Q^{(j)}R^{(j)} = H_m - \mu_j I_m$
- 7:   **Compute**  $H_m = \mu_j I_m + R^{(j)}Q^{(j)}$
- 8:   **Compute**  $Q = QQ^{(j)}$
- 9: **End**
- 10: **Compute**  $\mathbf{v}_{k+1} = \mathbf{v}_{m+1}h_{m+1,m}\mathbf{e}_m^\top Q\mathbf{e}_k + V_m Q\mathbf{e}_{k+1}h_{k+1,k}$
- 11: **Compute**  $V_k = V_m Q(\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_k)$
- 12: **Compute**  $h_{k+1,k} = \|\mathbf{v}_{k+1}\|_2$
- 13: **Compute**  $\mathbf{v}_{k+1} = \mathbf{v}_{k+1}/h_{k+1,k}$

Fig. 2.4: Algorithm of IRA method.

- Algorithm GMRES-IR( $m, k$ ) method**
- 1: **Compute**  $p = m - k$  and  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$
  - 2: **Compute**  $\beta = \|\mathbf{r}_0\|_2$  and  $\mathbf{v}_1 = \mathbf{r}_0/\beta$
  - 3: **Compute**  $V_{m+1}, \overline{H}_m$  with Arnoldi method
  - 4: **Compute**  $\mathbf{y}$ , the minimizer of  $\|V_{m+1}^T \mathbf{r}_0 - \overline{H}_m \mathbf{y}\|_2$  and  $\mathbf{x}_m = \mathbf{x}_0 + V_m \mathbf{y}$
  - 5: **If** satisfied **Stop**, **else proceed**
  - 6: **Compute** the harmonic Ritz values  $\tilde{\theta}_1, \tilde{\theta}_2, \dots, \tilde{\theta}_m$
  - 7: **Sort**  $|\tilde{\theta}_1| \geq |\tilde{\theta}_2| \geq \dots \geq |\tilde{\theta}_m|$
  - 8: **Set** shift  $\tilde{\theta}_1, \tilde{\theta}_2, \dots, \tilde{\theta}_p$
  - 9: **Update**  $V_{k+1}, \overline{H}_k$  with IRA method
  - 10: **Go to** 3, and resume the Arnoldi method from step  $k + 1$

Fig. 2.5: Algorithm of GMRES-IR( $m, k$ ) method.

- Algorithm GCR method**
- 1: **Compute**  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$
  - 2: **Set**  $\mathbf{p}_0 = \mathbf{r}_0, \mathbf{q}_0 = A\mathbf{p}_0$
  - 3: **For**  $k = 0, 1, \dots$ , **until**  $\|\mathbf{r}_k\|_2 \leq \varepsilon \|\mathbf{b}\|_2$  **do:**
  - 4:      $\alpha_k = (\mathbf{q}_k, \mathbf{r}_k) / (\mathbf{q}_k, \mathbf{q}_k)$
  - 5:      $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$
  - 6:      $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{q}_k$
  - 7:      $\beta_{i,k} = -(\mathbf{q}_i, A\mathbf{r}_{k+1}) / (\mathbf{q}_i, \mathbf{q}_i)$
  - 8:      $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \sum_{i=0}^k \beta_{i,k} \mathbf{p}_i$
  - 9:      $\mathbf{q}_{k+1} = A\mathbf{r}_{k+1} + \sum_{i=0}^k \beta_{i,k} \mathbf{q}_i$
  - 10: **End**

Fig. 2.6: Algorithm of GCR method.

**Algorithm GCR( $m$ ) method**

- 1: **Compute**  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$
- 2: **Set**  $\mathbf{p}_0 = \mathbf{r}_0, \mathbf{q}_0 = A\mathbf{p}_0$
- 3: **For**  $k = 0, 1, \dots, m - 1$  until  $\|\mathbf{r}_k\|_2 \leq \varepsilon\|\mathbf{b}\|_2$  do:
- 4:    $\alpha_k = (\mathbf{q}_k, \mathbf{r}_k) / (\mathbf{q}_k, \mathbf{q}_k)$
- 5:    $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$
- 6:    $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{q}_k$
- 7:    $\beta_{i,k} = -(\mathbf{q}_i, A\mathbf{r}_{k+1}) / (\mathbf{q}_i, \mathbf{q}_i)$
- 8:    $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \sum_{i=0}^k \beta_{i,k} \mathbf{p}_i$
- 9:    $\mathbf{q}_{k+1} = A\mathbf{r}_{k+1} + \sum_{i=0}^k \beta_{i,k} \mathbf{q}_i$
- 10: **End**
- 11: **Set**  $\mathbf{x}_0 = \mathbf{x}_m$  and go to 1

Fig. 2.7: Algorithm of GCR( $m$ ) method.

**Algorithm Preconditioned GMRES( $m$ ) method**

- 1: **Compute**  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0, \beta = \|\mathbf{r}_0\|_2$  and  $\mathbf{v}_1 = \mathbf{r}_0/\beta$
- 2: **For**  $j = 1, 2, \dots, m$  do:
- 3:   **Compute**  $\mathbf{w} = AK^{-1}\mathbf{v}_j$
- 4:   **For**  $i = 1, 2, \dots, j$  do:
- 5:     **Compute**  $h_{i,j} = (\mathbf{v}_i, \mathbf{w})$
- 6:     **Compute**  $\mathbf{w} = \mathbf{w} - h_{i,j}\mathbf{v}_i$
- 7:   **End**
- 8:   **Compute**  $h_{j+1,j} = \|\mathbf{w}\|_2$  and  $\mathbf{v}_{j+1} = \mathbf{w}/h_{j+1,j}$
- 9:   **Define**  $V_m = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m], \overline{H}_m = \{h_{i,j}\}_{1 \leq i \leq m+1, 1 \leq j \leq m}$
- 10: **End**
- 11: **Compute**  $\mathbf{y} = \operatorname{argmin}_{\mathbf{y}} \|\beta \mathbf{e}_1 - \overline{H}_m \mathbf{y}\|_2$ , and  $\mathbf{x}_m = \mathbf{x}_0 + K^{-1}V_m \mathbf{y}$
- 12: **If** satisfied **Stop**, **else Set**  $\mathbf{x}_0 = \mathbf{x}_m$  and go to 1

Fig. 2.8: Algorithm of preconditioned GMRES( $m$ ) method.

**Algorithm** Preconditioned GMRES-IR( $m, k$ ) method

- 1: **Compute**  $p = m - k$  and  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$
- 2: **Compute**  $\beta = \|\mathbf{r}_0\|_2$  and  $\mathbf{v}_1 = \mathbf{r}_0/\beta$
- 3: **Compute**  $V_{m+1}, \overline{H}_m$  with Arnoldi method ( $\mathbf{w} = AK^{-1}\mathbf{v}$ )
- 4: **Compute**  $\mathbf{y} = \operatorname{argmin}_{\mathbf{y}} \|\beta\mathbf{e}_1 - \overline{H}_m\mathbf{y}\|_2$  and  $\mathbf{x}_m = \mathbf{x}_0 + K^{-1}V_m\mathbf{y}$
- 5: **If** satisfied **Stop**, **else proceed**
- 6: **Compute** the harmonic Ritz values  $\tilde{\theta}_1, \tilde{\theta}_2, \dots, \tilde{\theta}_m$
- 7: **Sort**  $|\tilde{\theta}_1| \geq |\tilde{\theta}_2| \geq \dots \geq |\tilde{\theta}_m|$
- 8: **Set** shift  $\tilde{\theta}_1, \tilde{\theta}_2, \dots, \tilde{\theta}_p$
- 9: **Update**  $V_{k+1}, \overline{H}_k$  with IRA method
- 10: **Go to** 3, and resume the Arnoldi method from step  $k + 1$

Fig. 2.9: Algorithm of preconditioned GMRES-IR( $m, k$ ) method.

**Algorithm** preconditioned GCR( $m$ ) method

- 1: **Compute**  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$
- 2: **Set**  $\mathbf{p}_0 = K^{-1}\mathbf{r}_0, \mathbf{q}_0 = A\mathbf{p}_0$
- 3: **For**  $k = 0, 1, \dots$ , until  $\|\mathbf{r}_k\|_2 \leq \varepsilon\|\mathbf{b}\|_2$  **do**:
- 4:    $\alpha_k = (\mathbf{q}_k, \mathbf{r}_k)/(\mathbf{q}_k, \mathbf{q}_k)$
- 5:    $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k\mathbf{p}_k$
- 6:    $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k\mathbf{q}_k$
- 7:    $\mathbf{z}_{k+1} = K^{-1}\mathbf{r}_{k+1}$
- 8:    $\beta_{i,k} = -(\mathbf{q}_i, A\mathbf{z}_{k+1})/(\mathbf{q}_i, \mathbf{q}_i)$
- 9:    $\mathbf{p}_{k+1} = \mathbf{z}_{k+1} + \sum_{i=0}^k \beta_{i,k}\mathbf{p}_i$
- 10:    $\mathbf{q}_{k+1} = A\mathbf{z}_{k+1} + \sum_{i=0}^k \beta_{i,k}\mathbf{q}_i$
- 11: **End**

Fig. 2.10: Algorithm of preconditioned GCR( $m$ ) method.

# 第3章 準疎行列係数連立一次方程式に対する高い計算並列性をもつ前処理法

本章では、Cutoff 付きブロック近似逆行列前処理法 (Block Sparse Approximate Inverse with Cutoff, BSAIC) [33] に関して述べる。

BSAIC は並列性が高い前処理法である SAI を準疎行列係数連立一次方程式向きに改良した前処理法である。BSAIC の導出を述べ、従来法である SAI 及び Block SAI との比較実験を行い、性能を評価する。また、BSAIC の問題点やその改善策についても議論する [35]。

## 3.1 緒言

連立一次方程式

$$Ax = b, A \in \mathbb{C}^{n \times n}, x, b \in \mathbb{C}^n,$$

は様々な分野で現れ、それらの行列は大規模で疎な行列である場合が多い。このような大規模疎行列を係数行列にもつ連立一次方程式の解法として、前処理法付き Krylov 部分空間反復解法が用いられてきた。

しかしながら、近年、分子軌道計算に代表されるナノシミュレーションの分野などでは比較的密な行列 (準疎行列) が現れている。この準疎行列を係数行列にもつ連立一次方程式 (準疎行列係数連立一次方程式) に対し、従来の前処理法は主に疎行列向きの手法であるため有効でない場合が多々ある。

準疎行列係数連立一次方程式に対する前処理法として、近似係数行列に対する完全分解を用いた前処理法 [20] が提案されている。しかしながら、同前処理法は並列化処理が困難であり、マルチコア CPU や GPU などの計算機環境で高い並列性を発揮することが難しい。

本章では、高い計算並列性をもつ SAI に着目し、準疎行列係数連立一次方程式に対して有効かつ並列処理に優れた前処理法を提案する。その際、Cutoff と呼ばれる疎行列化技法を用いて近似係数行列を作成し、同行列に対し SAI の前処理性能を向上させたブロック近似逆行列前処理法 (Block SAI) [5] を適用し、準疎行列係数連立一次方程式に対する前処理法とする。



本章の構成は次のようになっている。次節では、SAIの前処理性能を向上させるために開発されたBlock SAIについて述べる。その後、提案法であるCutoff付きブロック近似逆行列前処理法に関して述べる。次に、複数の準疎行列に対し、提案法と従来法の性能を反復回数や計算時間の観点から比較する数値実験を行う。その後、Cutoffパラメータと収束性の関係を固有値分布の観点から調査し、より大きなCutoffパラメータを適用できる方法を提示する。最後に、本章をまとめる。

### 3.2 ブロック近似逆行列前処理法

2.2.2節ではSAIに関して述べたが、本節ではブロック近似逆行列前処理法 (Block SAI) [5, 30] について述べる。Block SAIは前処理行列の精度を向上させるためにBarnardとGroteにより提案された前処理法である。Block SAIでは、式(2.6)のフロベニウスノルムが式(2.7)と同様に

$$\|AM - I\|_F^2 = \sum_{k=1}^L \|AM_k - E_k\|_F^2,$$

と表すことができる。ただし、 $L = \lceil n/l \rceil$  ( $l$ : Block size) であり、 $E_k$ は $I = [E_1, E_2, \dots, E_L]$ となる単位行列 $I$ の部分行列である。また、 $l = 1$ つまり $L = n$ のときはSAIとなる。以上より、SAIと同様に前処理行列 $M = [M_1, M_2, \dots, M_L]$ は以下の $L$ 個の最小二乗問題を解くことで作成される。

$$\min_{M_k} \|AM_k - E_k\|_F^2, \quad k = 1, 2, \dots, L.$$

SAIにおいて、 $M$ の非零構造として $A$ の非零構造を採用する場合、 $A$ と $M$ の非零構造は一致する。しかしながら、Block SAIの場合、Block化によって $A$ の $i, j$ 要素 $a_{ij}$ が零の場合でも $M$ の $i, j$ 要素 $m_{ij}$ に非零要素が入り込むFill-inが発生するため、 $M$ の非零構造は以下のようになる。

$$\text{spy}(M) = \text{spy}(\tilde{A}).$$

ここで、 $\tilde{A}$ はBlock SAIによるFill-inを考慮した行列を表している。

### 3.3 Cutoff 付きブロック近似逆行列前処理法

本節では、非零要素数を削減するための方法である Cutoff を述べ、前節で述べた Block SAI と Cutoff を組み合わせた提案法に関して述べる。また、Cutoff と Block 化で起こりうる影響を考察する。

Cutoff とは係数行列  $A = [a_{ij}]$  の非零要素数を削減する方法であり、以下のように定義される。

$$A(\theta) = [\tilde{a}_{ij}], \quad \tilde{a}_{ij} = \begin{cases} a_{ij}, & (|a_{ij}| > \theta \times \max_{i,j} |a_{ij}| \text{ or } i = j) \\ 0, & \text{otherwise} \end{cases}. \quad (3.1)$$

ここで、 $0 \leq \theta \leq 1$  であり、 $A(\theta)$  は非零要素数を削減するため係数行列  $A$  に対し Cutoff を適用して生成された近似係数行列である。また、 $A(0) = A$  及び  $A(1) = \text{diag}(A)$  を満たす。 $\text{diag}(A)$  は  $A$  の対角成分のみの対角行列を表す。

式 (3.1) に定義される Cutoff により作成された近似係数行列  $A(\theta)$  に対して SAI を適用した前処理法

$$\min_{m_k} \|A(\theta)m_k - e_k\|_2^2, \quad k = 1, 2, \dots, n. \quad (3.2)$$

を Cutoff 付き近似逆行列前処理法 (SAI with Cutoff, SAIC) と名付ける。SAIC において、前処理行列  $M$  の非零構造は  $A(\theta)$  の非零構造を利用している。SAI に Cutoff を適用したことにより前処理行列の作成コストの減少が予想できるが SAI の性能低下が考えられる。つまり、Cutoff パラメータ  $\theta$  の上昇に伴い早期に解が求まらなくなる可能性がある。そこで、Cutoff と Fill-in を考慮した Block SAI を組み合わせた前処理法

$$\min_{M_k} \|A(\theta)M_k - E_k\|_F^2, \quad k = 1, 2, \dots, L \quad (3.3)$$

を提案する。同前処理法を Cutoff 付きブロック近似逆行列前処理法 (Block SAI with Cutoff, BSAIC) [33] と名付ける。Block 化することで Fill-in が入り込み、前処理法がよりロバストになり、また、SAIC に比べより大きな Cutoff パラメータ  $\theta$  を用いることが可能になると考えられる。

次に、BSAIC における Cutoff と Block 化で起こりうる影響を考察する。最初に Cutoff の影響について述べる。Cutoff を適用することで係数行列  $A$  の非零要素数が削減され、前処理行列の作成に要する計算コストも削減される。しかしながら、大きな  $\theta$  は前処理行列の性能を低下させ、反復回数が増大することが考えられる。また、小さな  $\theta$  は反復回数の増加を抑

えられるが、前処理行列作成の計算コストが増大することが予想できる。そのため、適切な  $\theta$  を選択する必要がある。

また、Block 化では以下の二つの影響が考えられる。一つ目は先ほど述べた Fill-in による前処理行列の性能改善である。この Fill-in により反復回数の減少が期待できる。二つ目は最小二乗問題の解く回数が減少し前処理行列作成に要する時間が減少する。しかしながら、最小二乗問題一つあたりの計算時間は増大するため Block size  $l$  を適切に選択する必要がある。

### 3.4 数値実験

本節では、複数の準疎行列係数連立一次方程式に対し数値実験を行った。3.4.1 節では、同連立一次方程式を、SAI, Block SAI, SAI with Cutoff (SAIC), そして提案法である Block SAI with Cutoff (BSAIC) を Krylov 部分空間反復解法に適用して解き、行列ベクトル積の回数、前処理時間、Krylov 部分空間反復解法及び全体の計算時間などを比較した。3.4.2 節では、Cutoff パラメータ  $\theta$  を増大させていくと求解不可能となる原因を考察し、BSAIC に対し改善策を適用した場合の行列ベクトル積の回数、前処理時間などを述べた。

数値実験に用いた計算機は CPU: Intel Xeon Processor X5550 (2.66GHz) 4-Core  $\times$  2, Memory: 48GB (4GB $\times$ 12), OS: CentOS 5.4 であり、MATLAB 7.12 を用いた。

Krylov 部分空間反復解法は2章で述べた、非 Hermite 行列向き解法である GMRES(50) 法を用いた。また、 $k$  反復目の残差ベクトル  $\mathbf{r}_k$  が  $\|\mathbf{r}_k\|_2/\|\mathbf{b}\|_2 \leq 10^{-10}$  を満たした場合に反復を停止した。最大行列ベクトル積回数は 5,000 回、初期解ベクトル  $\mathbf{x}_0$  の成分は全て 0、右辺ベクトル  $\mathbf{b}$  の成分はランダムとした。表中における“Preconditioner”は前処理法の種類、“The number of Mat-Vec”は行列ベクトル積の回数、“Cutoff”は Cutoff に要した時間、“Preconditioning”は前処理法に要した時間（前処理時間），“Iteration”は行列ベクトル積に要した時間（反復時間），“Total”は合計時間、“Krylov”は Krylov 部分空間反復解法の種類を表す。また、 $\dagger$  は反復解法における残差が収束条件を満たさず、最大行列ベクトル積回数に達したことを示す。

#### 3.4.1 数値実験 1（提案法と従来法の比較実験）

数値実験 1 ではそれぞれ異なる分野から現れる 3 つの準疎行列係数連立一次方程式に対し、SAI, Block SAI, SAIC 及び提案法である BSAIC を適用した GMRES(50) 法を用いて実験を行った。各々の行列ベクトル積回数、前処理時間、反復時間、合計時間などを示した。

なお、SAI 及び Block SAI においては前処理行列  $M$  の初期非零構造として  $A$  の非零構造を採用し、SAIC 及び BSAIC においては  $A(\theta)$  の非零構造を採用している。

### 数値例 1

数値例 1 では, The University of Florida Sparse Matrix Collection [31] の計算流体力学分野における圧縮駆動管内の非圧縮性流から現れる行列 “raefsky2” を対象とした. 同行列  $A$  は式 (1.4) のように  $A = \omega I - F$  と表される. なお, 行列  $F$  は “raefsky2” であり, 非対称行列である. 係数行列  $A$  の次元数は 3,242, 非零要素数は 293,551 (一行あたりの平均非零要素数は約 90) である. 行列  $A$  の非零構造図及び係数行列  $A$  の行列要素の絶対値の対数による分布をそれぞれ Fig. 3.1 (a) 及び (b) に示す,

シフトパラメータ  $\omega$  を 0.0, 0.1042 としたときの実験結果を Table 3.1 及び Table 3.2 に示す. また, Block size  $l$  は 30 とし, Cutoff パラメータ  $\theta$  は  $10^{-1}, 10^{-2}, 10^{-3}$  としている.

$\omega = 0.0$  における Table 3.1 より, SAI と Block SAI を比較すると Block 化によって行列ベクトル積回数が減少し, 前処理時間も減少していることが分かる. しかしながら, 反復時間が増大している. これは, Block 化によって前処理行列に Fill-in が入り込み非零要素数が増えたことによる一回あたりの行列ベクトル積時間増加が原因だと考えられる. SAIC や BSAIC においては, SAI 及び Block SAI の場合よりも行列ベクトル積回数が減少している場合がある. これは Cutoff により SAIC 及び BSAIC で生成される前処理行列の性質が良くなったためと考えられる.

また,  $\omega = 0.1042$  とした場合の Table 3.2 では, 従来法である SAI 及び Block SAI, そして SAIC や提案法である BSAIC のどの場合でも収束しなかった. この原因として, シフトパラメータによって係数行列  $A$  が悪条件になったことが考えられる.

次に, 上記の各  $\omega$  に対し, Cutoff パラメータ  $\theta$  を  $10^{-3}$  に固定し, Block size  $l$  を変化させたときの計算時間の推移を Fig. 3.2 及び Fig. 3.4 に, Block size  $l$  を 30 に固定し, Cutoff パラメータ  $\theta$  を変化させたときの計算時間の推移を Fig. 3.3 及び Fig. 3.5 にそれぞれ示す.

$\omega = 0.0$  における Fig. 3.2 に関して述べる.  $l$  を変化させたときの前処理時間は  $l = 10$  付近で最も短くなるが,  $l = 10$  より大きくすると増大する傾向となった. これは  $l$  が大きくなるにつれ最小二乗問題を解く回数が減少していくが, それに伴い最小二乗問題 1 つを解くコストが増大し, 最小二乗問題を解くコストの影響が上回ったためと考えられる. 反復時間においては  $l = 20$  のとき最も短くなるが,  $l = 20$  より大きい場合には増大していく傾向がある. これは  $l$  を増大させることで Fill-in が入り込み前処理行列の非零要素数が増えていき, それに伴い行列ベクトル積に時間を要しているためと考えられる. Fig. 3.3 においては,  $\theta$  を大きくすることで前処理時間が減少していき,  $\theta = 10^{-2}$  付近で急激に減少している. それに伴い, 反復時間も  $\theta = 10^{-2}$  付近から増大している. これは Cutoff により非零要素数が多く削減されたことで前処理行列の性能が悪化し, 収束により多くの反復を必要とするためと考えられる.

また,  $\omega = 0.1042$  においては, Fig. 3.4 及び Fig. 3.5 より, 前処理時間が図示されている

だけである。これは、どの場合の  $l$  や  $\theta$  を用いても残差が収束せず解が求まらなかったことを示している。このように、 $\omega$  によっては提案法である BSAIC を用いても収束しない場合もある。

## 数値例 2

数値例 2 では、The University of Florida Sparse Matrix Collection [31] の電磁気学分野における半導体製造プロセスなどに応用される 2 次元フォッカー・プランク方程式から現れる行列 “fp” を対象とした。同行列  $A$  は式 (1.4) のように  $A = \omega I - F$  と表される。なお、行列  $F$  は “fp” であり、非対称行列である。係数行列  $A$  の次元数は 7,548、非零要素数は 834,222 (一行あたりの平均非零要素数は約 111) である。行列  $A$  の非零構造図及び係数行列  $A$  の行列要素の絶対値の対数による分布をそれぞれ Fig. 3.6 (a) 及び (b) に示す、

シフトパラメータ  $\omega$  を 9.0, 0.0 としたときの実験結果を Table 3.3 及び Table 3.4 に示す。また、Block size  $l$  は 30 とし、Cutoff パラメータ  $\theta$  は  $10^{-9}$ ,  $10^{-10}$ ,  $10^{-11}$  としている。

$\omega = 9.0$  における Table 3.3 より、SAI は収束せず、Block SAI では収束しているが前処理に時間を要していることが分かる。SAIC にすることで、前処理時間が減少しているが最大行列ベクトル積回数以内に収束していない。BSAIC においては、いずれの  $\theta$  でも収束している。全体的に、 $\theta = 10^{-9}$  としたときの BSAIC を用いた場合が最速となった。

また、 $\omega = 0.0$  とした場合の Table 3.4 においては、 $\omega = 9.0$  の場合である Table 3.3 と比較すると Block SAI を用いても収束していないことが分かる。加えて、BSAIC における  $\theta = 10^{-9}$  でも収束せず、他の  $\theta$  の場合でも行列ベクトル積回数が増大している。合計時間としては、 $\theta = 10^{-10}$  とした BSAIC を用いた場合は最も速く収束した。

次に、上記の各  $\omega$  に対し、Cutoff パラメータ  $\theta$  を  $10^{-12}$  に固定し、Block size  $l$  を変化させたときの計算時間の推移を Fig. 3.7 及び Fig. 3.9 に、Block size  $l$  を 30 に固定し、Cutoff パラメータ  $\theta$  を変化させたときの計算時間の推移を Fig. 3.8 及び Fig. 3.10 にそれぞれ示す。

まず、 $\omega = 9.0$  とした場合の Fig. 3.7 及び Fig. 3.8 に関して述べる。Fig. 3.7 より、 $l$  を増大させるに従い反復時間が減少していくが  $l = 30$  付近から大きな変化が見られないことが分かる。また、前処理時間は  $l = 10$  付近を境に  $l$  の値を大きくするに従い増大していく。これは  $l$  が大きくなるにつれ最小二乗問題を解く回数が減少していくが、それに伴い最小二乗問題 1 つを解くコストが増大し、最小二乗問題を解くコストの影響が上回ったためと考えられる。Fig. 3.8 においては、 $\theta$  を増加させるに従い前処理時間は減少していくが、反復時間は  $\theta = 10^{-10}$  付近から増加していき、 $10^{-9}$  からは求解ができていないことが読み取れる。

また、 $\omega = 0.0$  においては、Fig. 3.9 より前処理時間は  $\omega = 9.0$  の Fig. 3.7 と同様に  $l = 10$  付近を境に徐々に増大していくことが分かる。しかしながら、反復時間が増減を繰り返して

いる。この理由として、 $l$ の値によって Fill-in の入る箇所などが変化し前処理行列の性質に影響を与えていることが考えられる。Fig. 3.10 からは、 $\theta$ が $10^{-16}$ のような小さな値や $10^{-9}$ といった大きな値を用いる場合には収束していないことが分かる。これは、元の係数行列  $A$  の性質が悪いため、 $10^{-16}$ のような小さな値の場合には  $A$  に近くなってしまい前処理行列の性質も悪くなってしまうことが原因の一つと考えられる。そして、 $10^{-9}$ などのより大きな値の場合も収束しないため  $\theta$ の設定には十分留意する必要がある。

### 数値例 3

数値例 3 では、EGF (Epidermal Growth Factor) と呼ばれる上皮成長因子の分子軌道計算から現れる行列を対象とした。分子軌道計算では一般化固有値問題 (1.2) が現れ、1 章で述べた Sakurai-Sugiura 法 [27] などの固有値解法を適用することで連立一次方程式を解くことになる。同行列  $A$  は式 (1.3) のように  $A = \omega S - F$  と表される。なお、行列  $F$  は実対称行列であり、 $S$  は正定値対称行列である。係数行列  $A$  の次元数は 4,505、非零要素数は 5,252,731 (一行あたりの平均非零要素数は約 1,166) である。行列  $A$  の非零構造図及び係数行列  $A$  の行列要素の絶対値の対数による分布をそれぞれ Fig. 3.6 (a) 及び (b) に示す、

シフトパラメータ  $\omega$  を  $-0.09$ ,  $-0.2$  としたときの実験結果を Table 3.5 及び Table 3.6 に示す。また、Block size  $l$  は 30 とし、Cutoff パラメータ  $\theta$  は  $10^{-5}$ ,  $10^{-6}$ ,  $10^{-7}$  としている。

$\omega = -0.09$  における Table 3.5 に関して述べる。SAI を適用した場合には、行列ベクトル積回数が 6 回で収束しているが前処理時間が約 13,000 秒と膨大である。Block SAI の場合は約 900 秒である。SAI に Cutoff を適用することで行列ベクトル積回数が増加しているが、前処理時間は減少している。BSAIC の場合には SAIC と比較して全体的に前処理時間及び行列ベクトル積回数が減少している。 $\theta = 10^{-5}$  とした BSAIC の場合が合計時間として最も速く収束した。

また、 $\omega = -0.2$  の場合である Table 3.6 に関して述べる。 $\omega = -0.09$  における Table 3.5 と比較すると、行列ベクトル積回数が増加し、SAIC における  $\theta = 10^{-5}$  のときには収束していない。合計時間では、 $\theta = 10^{-5}$  とした BSAIC が最も速かった。

次に、上記の各  $\omega$  に対し、Cutoff パラメータ  $\theta$  を固定し、Block size  $l$  を変化させたときの計算時間の推移を Fig. 3.12 及び Fig. 3.14 に、Block size  $l$  を固定し、Cutoff パラメータ  $\theta$  を変化させたときの計算時間の推移を Fig. 3.13 及び Fig. 3.15 にそれぞれ示す。

$\omega = -0.09$  における Fig. 3.12 より、 $l$  を増大させるに従い前処理時間は  $l = 20$  付近までは減少するが、その後はあまり変化が見られない。反復時間については  $l = 50$  付近から少々増大しているがあまり変化はない。Fig. 3.13 からは、 $\theta$  を増大させるに従い前処理時間が大幅に減少していく様子が分かる。反復時間については、 $\theta = 10^{-6}$  付近から増大している。

また、 $\omega = -0.2$  における Fig. 3.14 より  $l = 50$  付近から反復時間に大きな変化はなく、前処理時間は  $l = 20$  付近から大きな変化がないことが分かる。Fig. 3.15 では、 $\omega = -0.09$  における Fig. 3.13 と比べて大きな変化は見られなかった。

### 3.4.2 数値実験2 (BSAICにおける収束性悪化の解析とその対策)

数値実験1より、Cutoffパラメータ $\theta$ を上げていくと収束性が悪化していくが、前処理に要する時間は大幅に減少することが分かった。そのため、計算時間を考慮すると可能な限り大きな $\theta$ を用いることが望ましい。数値実験2では、 $\theta$ を上げていくと収束性が悪化する原因を固有値の観点から解析し、より大きな $\theta$ を用いても高速に収束する方法 [35] を考える。

BSAICにおけるBlock size  $l$ を固定し、 $\theta$ を変化させたときの係数行列  $A$  と前処理行列  $M$  の積  $AM$  の固有値分布を調べる。 $M$ は逆行列  $A^{-1}$  を近似しており、 $M$ が  $A^{-1}$  を良く近似しているならば  $AM$  は単位行列  $I$  に近くなる。つまり、 $M \approx A^{-1}$  のときには  $AM$  の固有値が1に密集していると考えられる。 $\theta$ を上げるに従い  $M$  は  $A^{-1}$  を良く近似できなくなると、 $AM$  がどのような振る舞いを行うかを数値実験1同様三つの行列に対し調査した。なお、 $\omega$  はそれぞれ raefsky2 の場合は 0.1042, fp の場合は 0.0, EGF の場合は  $-0.2$  とした。

#### 数値例1

3.4.1節の数値例1として用いた raefsky2 を対象に実験を行った。

#### AMの固有値分布の変化

Figs. 3.16 (a), (b), ..., (f) に、 $l = 30$  に固定し、 $\theta$  を  $1.0 \times 10^{-6}, 1.0 \times 10^{-5}, \dots, 1.0 \times 10^{-1}$  と変化させたときの  $AM$  の固有値分布をそれぞれ示し、 $\theta$  を変化させたときのある範囲に存在する  $AM$  の絶対固有値数を Table 3.7 に示す。なお、Figs. 3.16 (a), (b), ..., (f) の縦線は零固有値を表し、Table 3.7 における  $\#(|d| < value)$  は  $value$  未満の絶対固有値数を示している。

raefsky2 では、 $\theta$  を上昇させても固有値分布にほとんど変化は見られない。しかしながら、0 付近に固有値が存在していることが分かる。これは Table 3.7 から読み取れる。Table 3.2 より、SAI などの従来法及び提案法の BSAIC のどの前処理法を用いた場合も収束していないことが分かる。この求解が不可能な原因として、0 付近の固有値が存在していることが考えられる。そこで、 $AM$  の小さな値の固有値を取り除くことが収束性の改善に繋がると期待できる。小さな値の固有値を取り除く手法として 2.1.3 節で述べた GMRES-IR 法 [18] を適用することを考える。次節では、 $\theta = 1.0 \times 10^{-5}, 1.0 \times 10^{-3}$  のときに GMRES-IR 法を適用した結果について述べる。

### GMRES-IR 法の適用

Table 3.8 及び Table 3.9 に  $\theta = 1.0 \times 10^{-5}$ ,  $1.0 \times 10^{-3}$  としたときの GMRES-IR 法の実験結果をそれぞれ示す。ただし、Cutoff 及び前処理時間は同じ前処理法 (BSAIC) を用いているため不変である。

Table 3.8 より、GMRES(50) 法では収束せず、それに対し GMRES-IR 法においては取り除く固有値数  $k$  を 35, 40, 45 としたときには収束している。しかしながら、 $k = 5, \dots, 30$  としたときには収束していない。また、Table 3.9 における  $k = 15$  の場合のように解ける場合もあった。

### 数値例 2

3.4.1 節の数値例 2 として用いた fp を対象に実験を行った。

#### AM の固有値分布の変化

Figs. 3.17 (a), (b), ..., (f) に、 $l = 30$  に固定し、 $\theta$  を  $1.0 \times 10^{-13}$ ,  $1.0 \times 10^{-12}$ , ...,  $1.0 \times 10^{-8}$  と変化させたときの AM の固有値分布をそれぞれ示し、 $\theta$  を変化させたときのある範囲に存在する AM の絶対固有値数を Table 3.10 に示す。なお、Figs. 3.17 (a), (b), ..., (f) の縦線は零固有値を表し、Table 3.10 における  $\#(|d| < value)$  は  $value$  未満の絶対固有値数を示している。

Figs. 3.17 より、 $\theta$  を大きくするに従い AM の固有値分布が広がっていくことが分かる。それに伴い、Table 3.10 に示すように 0 付近の微小な固有値が増大していく。また、数値実験 1 より  $\theta$  を増大させると収束性が悪化することが分かった。これらの結果から、AM の固有値が 0 付近に密集していくと収束性が悪化すると考えられる。次節では、 $\theta = 1.0 \times 10^{-13}$ ,  $1.0 \times 10^{-10}$  のときに GMRES-IR 法を適用した結果について述べる。

### GMRES-IR 法の適用

Table 3.11 及び Table 3.12 に  $\theta = 1.0 \times 10^{-12}$ ,  $1.0 \times 10^{-10}$  としたときの GMRES-IR 法の実験結果をそれぞれ示す。ただし、Cutoff 及び前処理時間は同じ前処理法 (BSAIC) を用いているため不変である。

Table 3.11 及び Table 3.12 共に、GMRES(50) 法を用いた場合でも収束しているが行列ベクトル積回数が多く、反復時間に時間を要する。それに対し、GMRES-IR 法を適用することで行列ベクトル積回数が減少していることが分かる。しかしながら、Table 3.11 より取り除く固有値数  $k$  を 15, 25 とした場合には最大行列ベクトル積回数以内に収束していない。また、Table 3.12 においては、 $k$  を 15 以上とした場合には収束していない。これより、 $k$  は大きい値ほど良い収束性を示すものではないことが分かる。従って、 $\theta$  や  $l$  に加え、 $m$  及び  $k$  の選択も適切に行う必要がある。



### 数値例 3

3.4.1 節の数値例 3 として用いた EGF を対象に実験を行った。

#### AM の固有値分布の変化

Figs. 3.18 (a), (b), ..., (f) に,  $l = 30$  に固定し,  $\theta$  を  $1.0 \times 10^{-7}, 1.0 \times 10^{-6}, \dots, 1.0 \times 10^{-2}$  と変化させたときの AM の固有値分布をそれぞれ示し,  $\theta$  を変化させたときのある範囲に存在する AM の絶対固有値数を Table 3.13 に示す. なお, Figs. 3.18 (a), (b), ..., (f) の縦線は零固有値を表し, Table 3.13 における  $\#(|d| < \text{value})$  は  $\text{value}$  未満の絶対固有値数を示している.

数値例 2 の“fp” 同様に, Figs. 3.18 より,  $\theta$  を大きくするに従い AM の固有値分布が広がっていくことが分かる. それに伴い, Table 3.13 に示すように 0 付近の微小な固有値が増大していき,  $\theta = 10^{-4}$  においては  $10^{-1}$  未満の固有値が 100 近い数になっている. これらのことから, AM の固有値が 0 付近に密集していくと収束性が悪化すると考えられる. 次節では,  $\theta = 5.0 \times 10^{-5}, 1.0 \times 10^{-4}$  のときに GMRES-IR 法を適用した結果について述べる.

#### GMRES-IR 法の適用

Table 3.14 及び Table 3.15 に  $\theta = 5.0 \times 10^{-5}, 1.0 \times 10^{-4}$  としたときの GMRES-IR 法の実験結果をそれぞれ示す. ただし, Cutoff 及び前処理時間は同じ前処理法 (BSAIC) を用いているため不変である.

Table 3.14 より, GMRES(50) 法では収束していないのに対し, GMRES-IR 法を適用することで収束していることが分かる. また,  $k$  をある程度大きくすることで行列ベクトル積の回数が減少し反復時間も減少する傾向にある. 合計時間では GMRES-IR(50, 40) 法を適用した場合が最も速く収束した. また, Table 3.15 のように  $\theta = 1.0 \times 10^{-4}$  と, Cutoff パラメータを上昇させた場合は GMRES-IR 法を適用しても収束性が改善されないという結果になった. これは絶対値の小さな固有値が増え, デフレーションできない固有値が存在したためと考えられる.

## 3.5 結言

本章では, 高い計算並列性をもつ前処理法である SAI を準疎行列係数連立一次方程式に対する有効な前処理法へと改良した Cutoff 付きブロック近似逆行列前処理法 (Block SAI with Cutoff, BSAIC) を提案した. また, BSAIC の有効性を様々な分野から現れる準疎行列係数連立一次方程式に対する数値実験を通して示し, 収束しなくなる, または収束しない原因を固有値の観点から解析し改善策を提示した. その有効性も同様に数値実験より示した.

提案法の BSAIC と従来の前処理法である SAI, Block SAI を比較した場合, BSAIC は行列ベクトル積回数が増えるが前処理時間が減少し, 全体の計算時間も短くなった. BSAIC と Cutoff を付与した SAI (SAIC) を比較した場合には BSAIC は Block 化の影響で前処理時間及び行列ベクトル積回数が減少し, 合計時間も減少した. また, SAIC のときには収束しなかった Cutoff パラメータでも収束した. 以上より, 準疎行列係数連立一次方程式に対する BSAIC の適用でより高速に求解でき, より大きな Cutoff パラメータを適用することが可能になった.

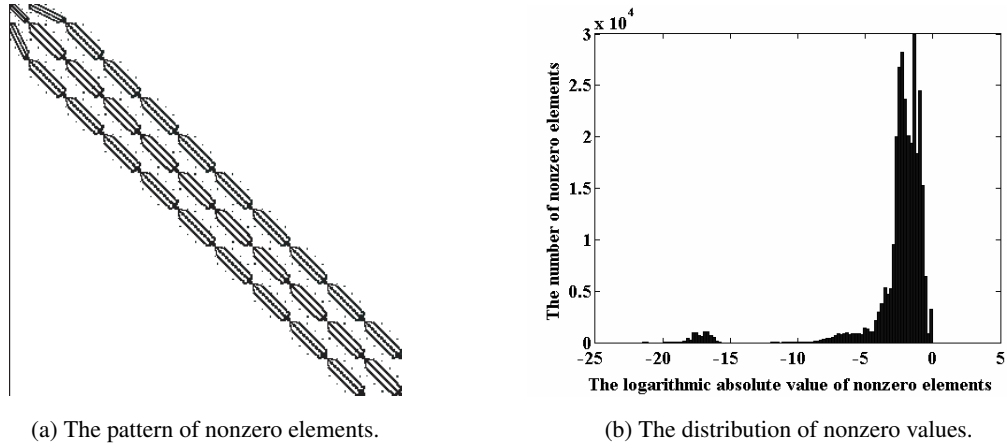

 Fig. 3.1: The features of  $A$  for raefsky2.

 Table 3.1: The results of  $\omega = 0.0$  for raefsky2.

Preconditioner	The number of Mat-Vec	Wall clock time [sec]				
		Cutoff	Preconditioning	Iteration	Total	
SAI	1,495	—	11.60	5.98	17.58	
Block SAI ( $l = 30$ )	1,198	—	4.30	7.42	11.72	
SAIC	$\theta = 10^{-1}$	1,791	0.01	1.03	5.64	6.68
	$\theta = 10^{-2}$	1,347	0.02	4.77	4.88	9.67
	$\theta = 10^{-3}$	1,481	0.02	9.98	5.70	15.70
BSAIC ( $l = 30$ )	$\theta = 10^{-1}$	1,540	0.01	0.99	7.58	8.58
	$\theta = 10^{-2}$	863	0.02	3.98	5.33	9.33
	$\theta = 10^{-3}$	980	0.02	4.24	5.99	10.25

 Table 3.2: The results of  $\omega = 0.1042$  for raefsky2.

Preconditioner	The number of Mat-Vec	Wall clock time [sec]				
		Cutoff	Preconditioning	Iteration	Total	
SAI	†	—	11.67	—	—	
Block SAI ( $l = 30$ )	†	—	4.54	—	—	
SAIC	$\theta = 10^{-1}$	†	0.02	1.19	—	—
	$\theta = 10^{-2}$	†	0.02	5.10	—	—
	$\theta = 10^{-3}$	†	0.04	9.83	—	—
BSAIC ( $l = 30$ )	$\theta = 10^{-1}$	†	0.01	1.27	—	—
	$\theta = 10^{-2}$	†	0.03	4.01	—	—
	$\theta = 10^{-3}$	†	0.03	4.41	—	—

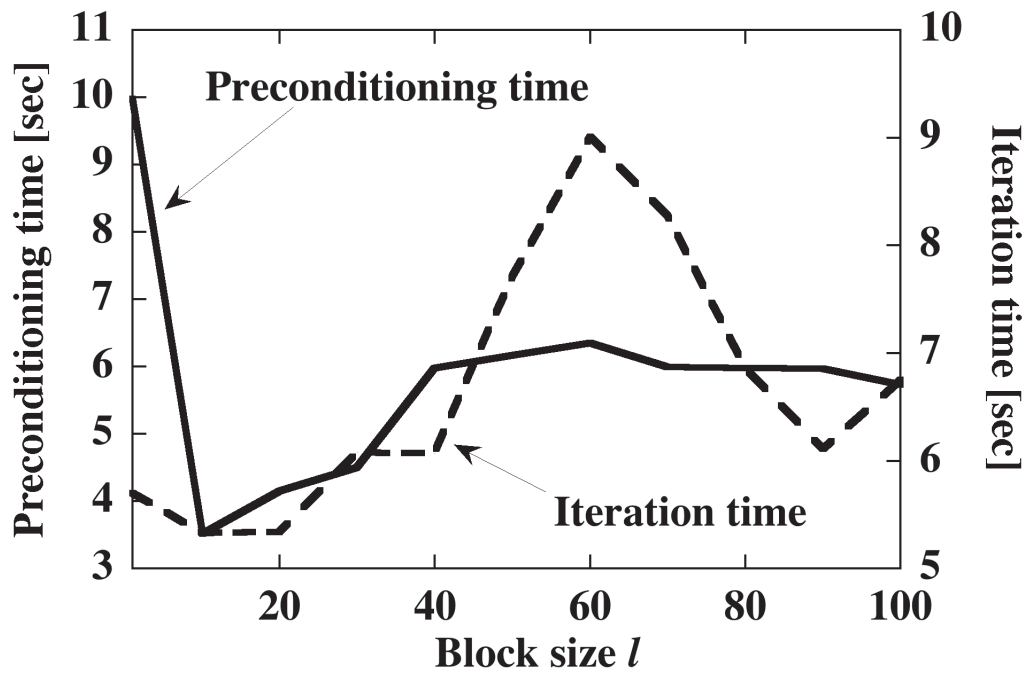


Fig. 3.2: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $l$  for raefsky2 ( $\omega = 0.0$ ).

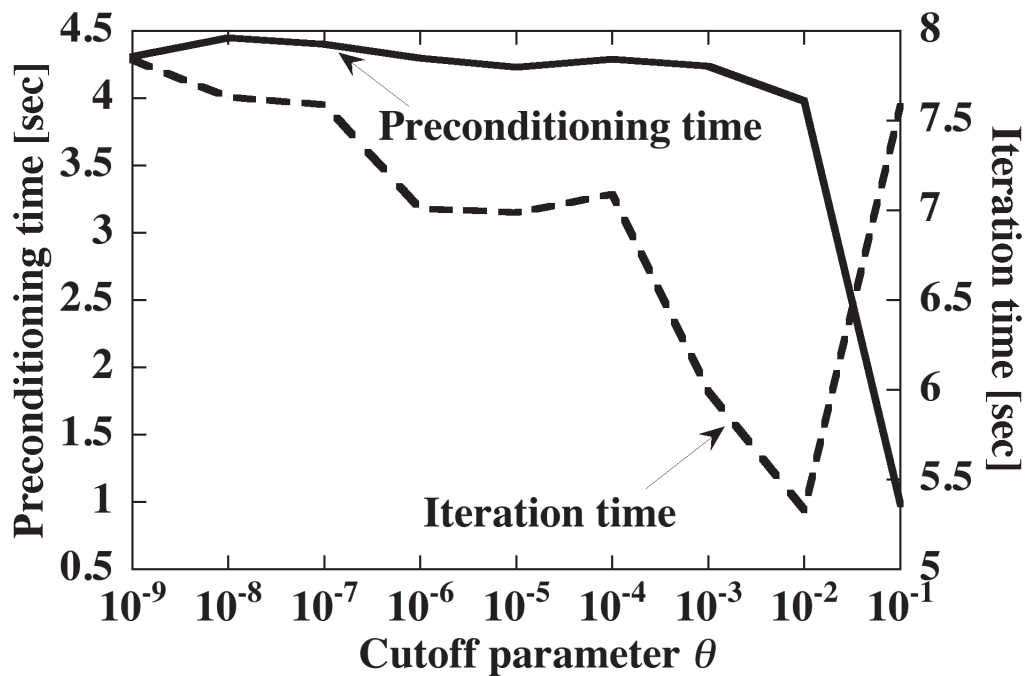


Fig. 3.3: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $\theta$  for raefsky2 ( $\omega = 0.0$ ).

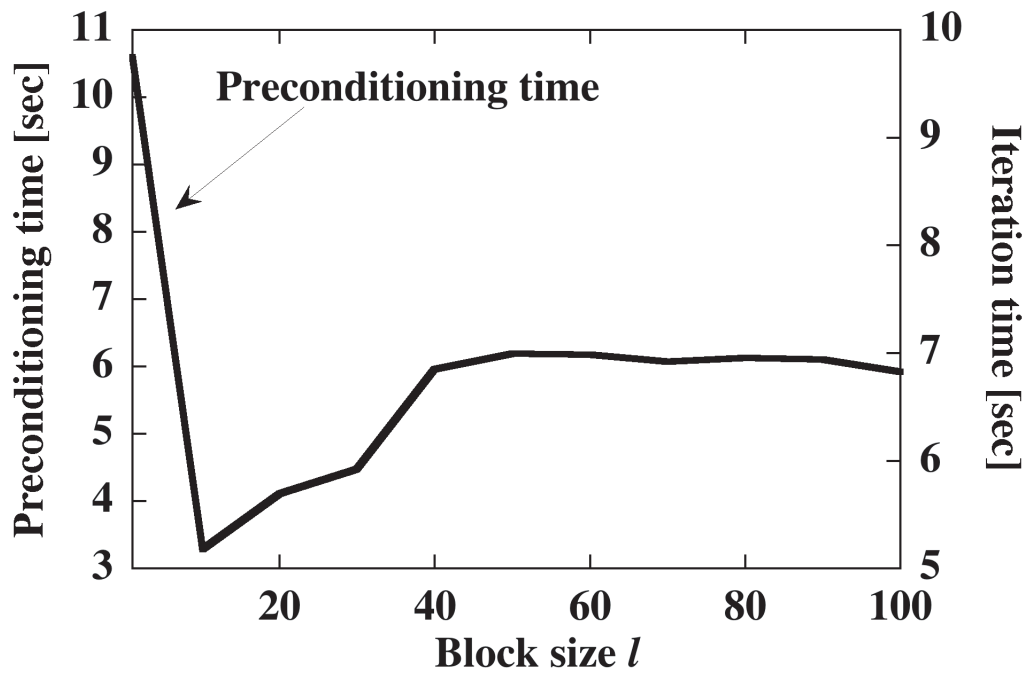


Fig. 3.4: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $l$  for raefsky2 ( $\omega = 0.1042$ ).

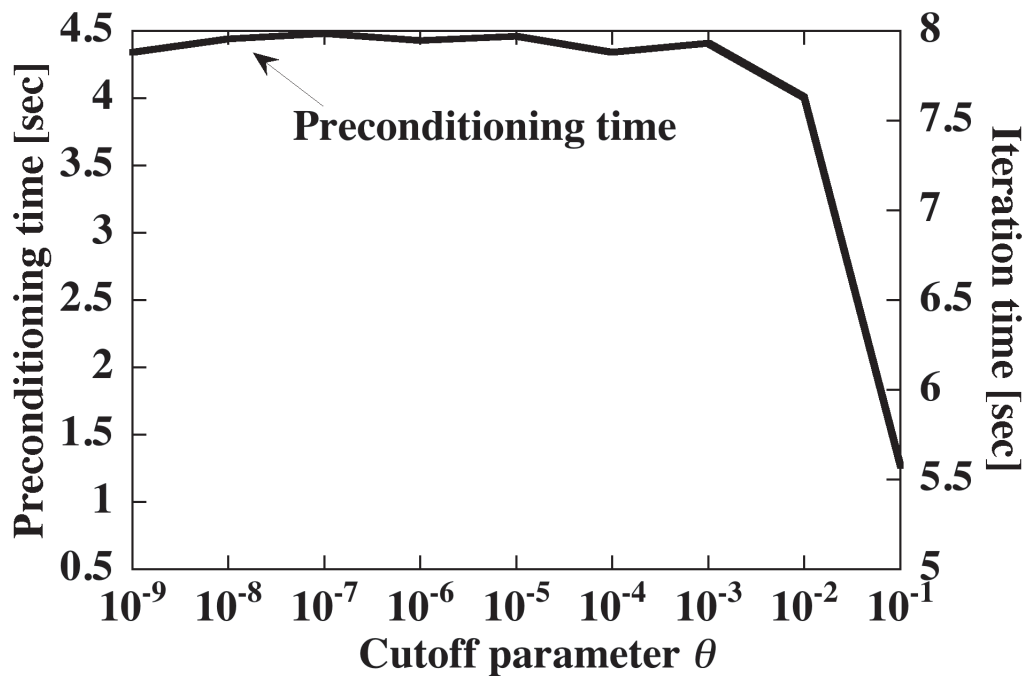


Fig. 3.5: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $\theta$  for raefsky2 ( $\omega = 0.1042$ ).

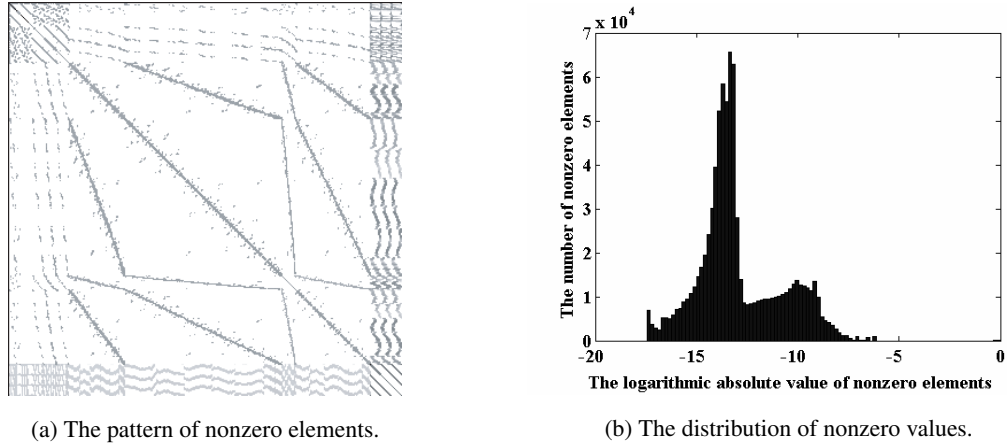

 Fig. 3.6: The features of  $A$  for fp.

 Table 3.3: The results of  $\omega = 9.0$  for fp.

Preconditioner		The number of Mat-Vec	Wall clock time [sec]			
			Cutoff	Preconditioning	Iteration	Total
SAI		†	—	192.81	—	—
Block SAI ( $l = 30$ )		42	—	209.37	0.74	210.11
SAIC	$\theta = 10^{-9}$	†	0.02	2.48	—	—
	$\theta = 10^{-10}$	†	0.03	5.55	—	—
	$\theta = 10^{-11}$	†	0.03	12.08	—	—
BSAIC ( $l = 30$ )	$\theta = 10^{-9}$	294	0.02	1.02	2.30	3.34
	$\theta = 10^{-10}$	45	0.03	4.43	0.52	4.98
	$\theta = 10^{-11}$	42	0.03	15.24	0.53	15.80

 Table 3.4: The results of  $\omega = 0.0$  for fp.

Preconditioner		The number of Mat-Vec	Wall clock time [sec]			
			Cutoff	Preconditioning	Iteration	Total
SAI		†	—	191.61	—	—
Block SAI ( $l = 30$ )		†	—	209.22	—	—
SAIC	$\theta = 10^{-9}$	†	0.02	2.65	—	—
	$\theta = 10^{-10}$	†	0.03	6.04	—	—
	$\theta = 10^{-11}$	†	0.03	11.14	—	—
BSAIC ( $l = 30$ )	$\theta = 10^{-9}$	†	0.02	1.03	—	—
	$\theta = 10^{-10}$	2,418	0.03	4.41	25.28	29.72
	$\theta = 10^{-11}$	1,359	0.04	15.04	15.89	30.97

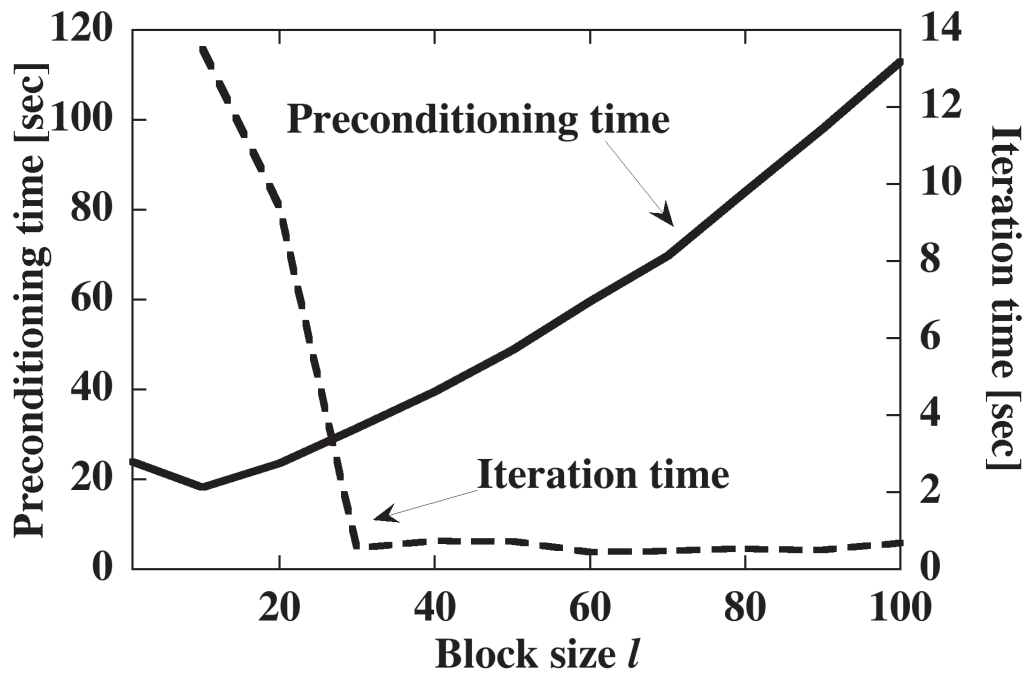


Fig. 3.7: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $l$  for fp ( $\omega = 9.0$ ).

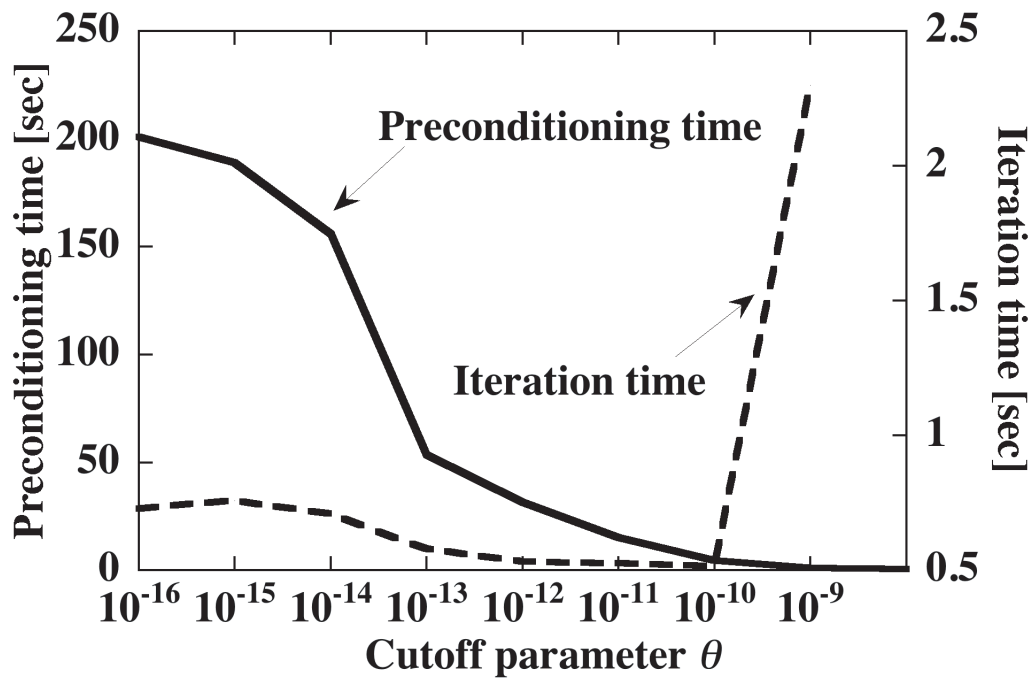


Fig. 3.8: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $\theta$  for fp ( $\omega = 9.0$ ).

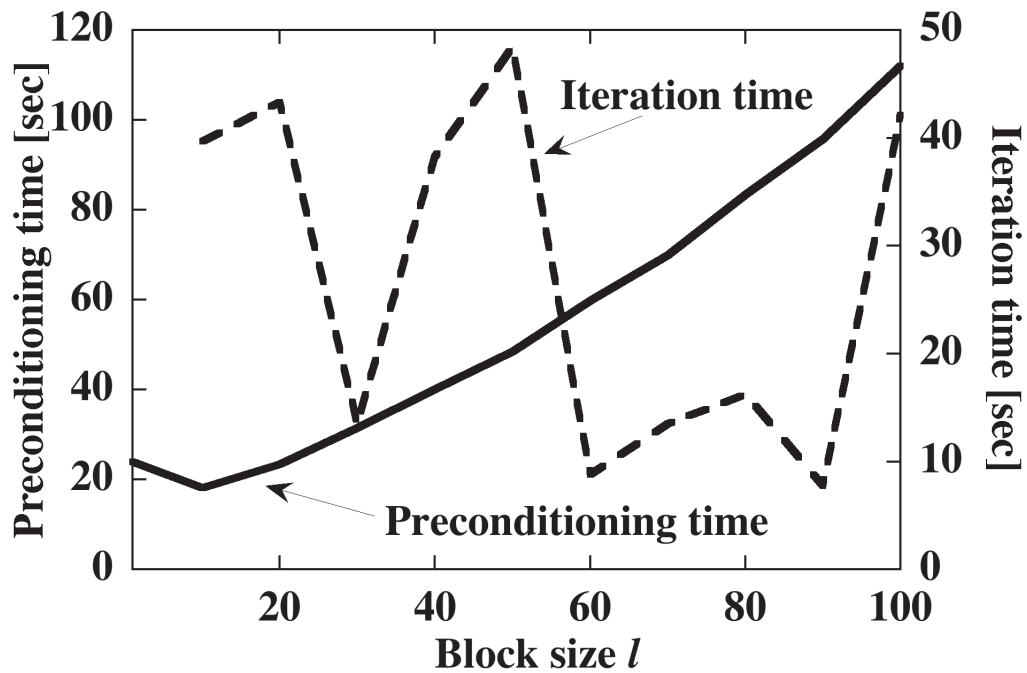


Fig. 3.9: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $l$  for fp ( $\omega = 0.0$ ).

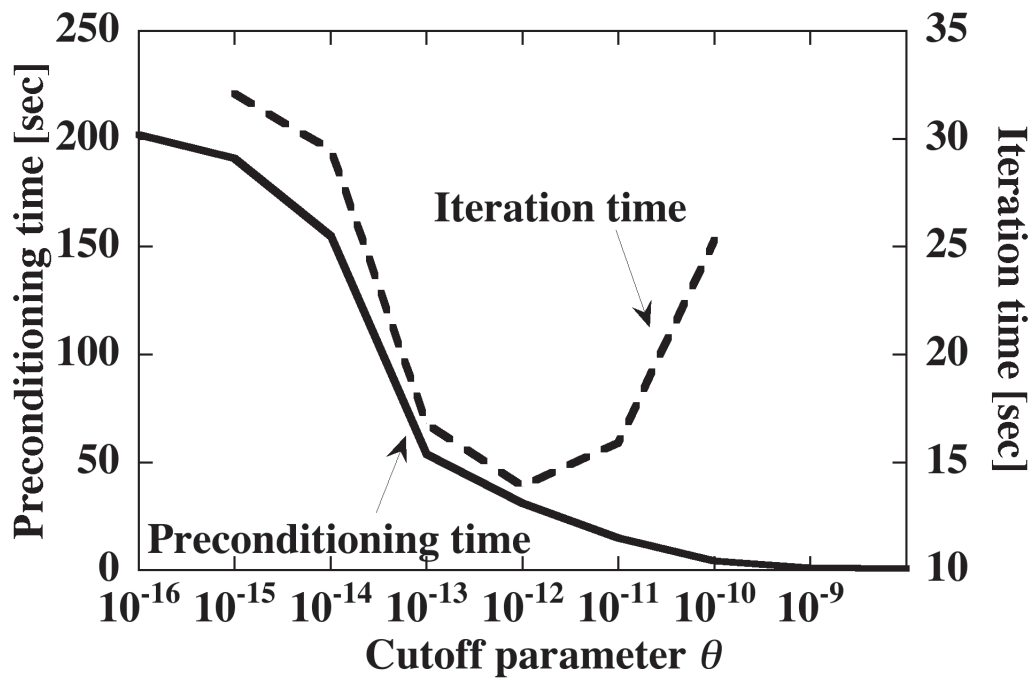


Fig. 3.10: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $\theta$  for fp ( $\omega = 0.0$ ).



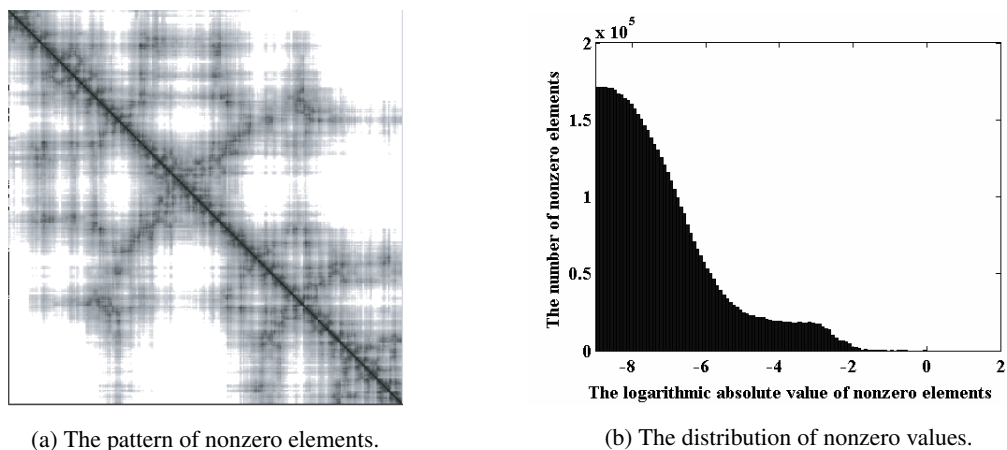

 Fig. 3.11: The features of  $A$  for EGF.

 Table 3.5: The results of  $\omega = -0.09$  for EGF.

Preconditioner	The number of Mat-Vec	Wall clock time [sec]				
		Cutoff	Preconditioning	Iteration	Total	
SAI	6	—	13,186.05	0.26	13,186.31	
Block SAI ( $l = 30$ )	4	—	934.84	0.22	935.06	
SAIC	$\theta = 10^{-5}$	39	0.16	49.68	0.80	50.64
	$\theta = 10^{-6}$	20	0.21	162.84	0.43	163.48
	$\theta = 10^{-7}$	13	0.28	1,046.35	0.32	1,046.95
BSAIC ( $l = 30$ )	$\theta = 10^{-5}$	24	0.19	9.37	0.54	10.10
	$\theta = 10^{-6}$	11	0.20	34.03	0.30	34.53
	$\theta = 10^{-7}$	8	0.29	164.80	0.27	165.36

 Table 3.6: The results of  $\omega = -0.2$  for EGF.

Preconditioner	The number of Mat-Vec	Wall clock time [sec]				
		Cutoff	Preconditioning	Iteration	Total	
SAI	14	—	13,166.72	0.49	13,167.21	
Block SAI ( $l = 30$ )	9	—	931.94	0.41	932.35	
SAIC	$\theta = 10^{-5}$	†	0.17	49.12	—	—
	$\theta = 10^{-6}$	165	0.21	163.30	3.45	166.96
	$\theta = 10^{-7}$	38	0.29	1,034.94	0.91	1,036.14
BSAIC ( $l = 30$ )	$\theta = 10^{-5}$	72	0.18	9.17	1.57	10.92
	$\theta = 10^{-6}$	32	0.21	33.73	0.80	34.74
	$\theta = 10^{-7}$	20	0.28	163.07	0.61	163.96

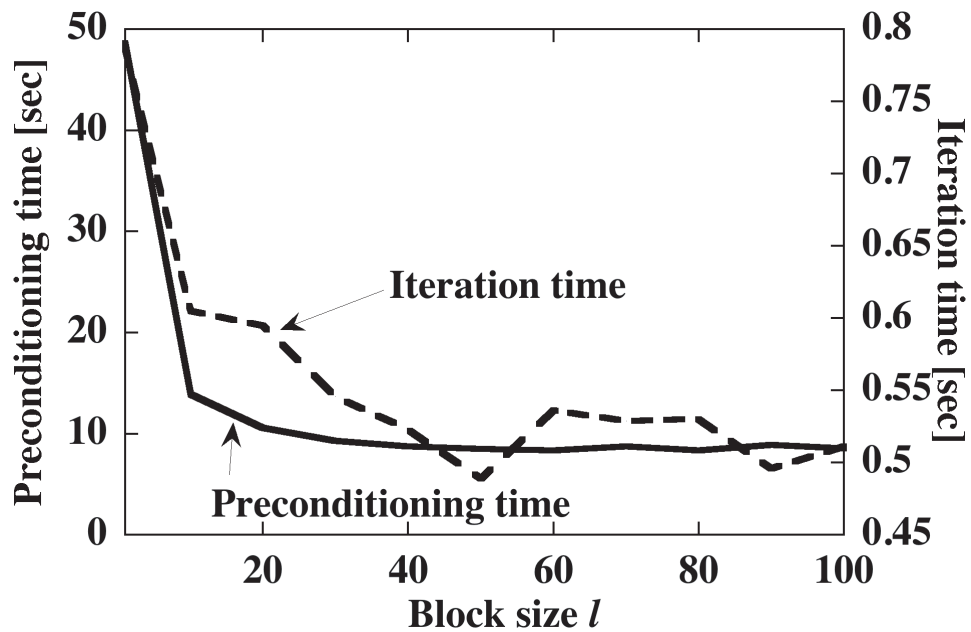


Fig. 3.12: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $l$  for EGF ( $\omega = -0.09$ ).

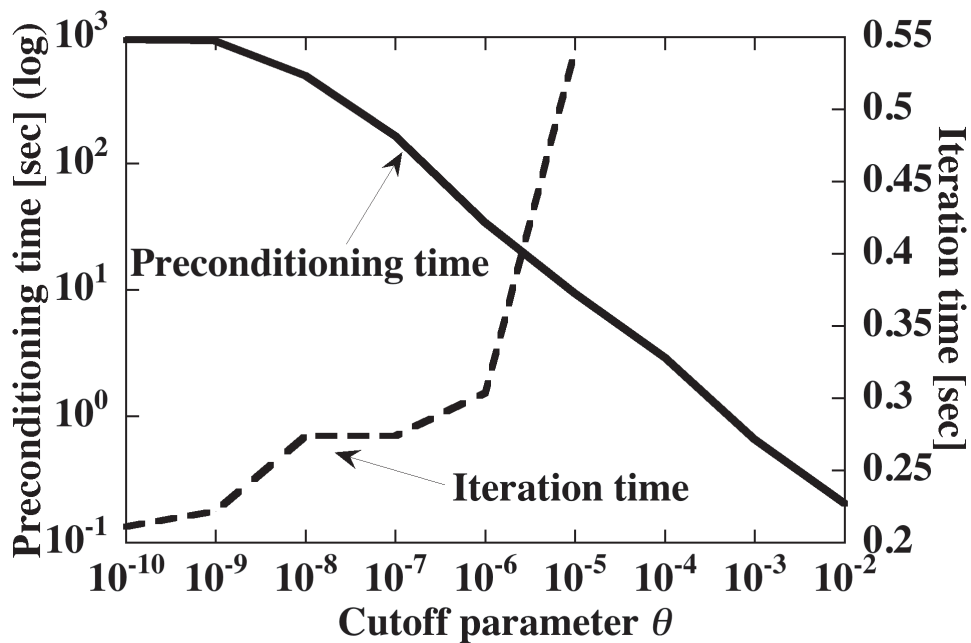


Fig. 3.13: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $\theta$  for EGF ( $\omega = -0.09$ ).

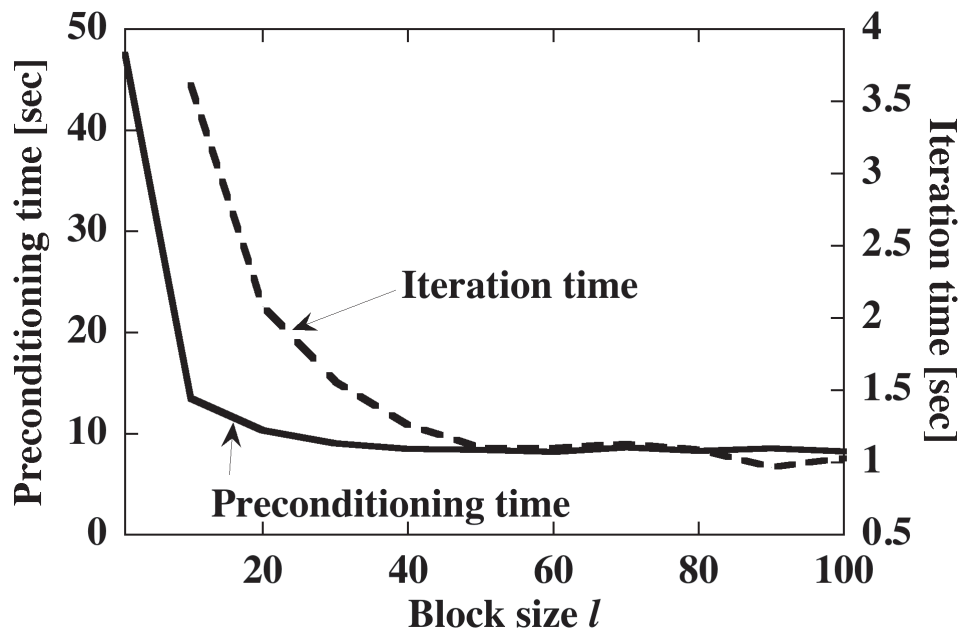


Fig. 3.14: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $l$  for EGF ( $\omega = -0.2$ ).

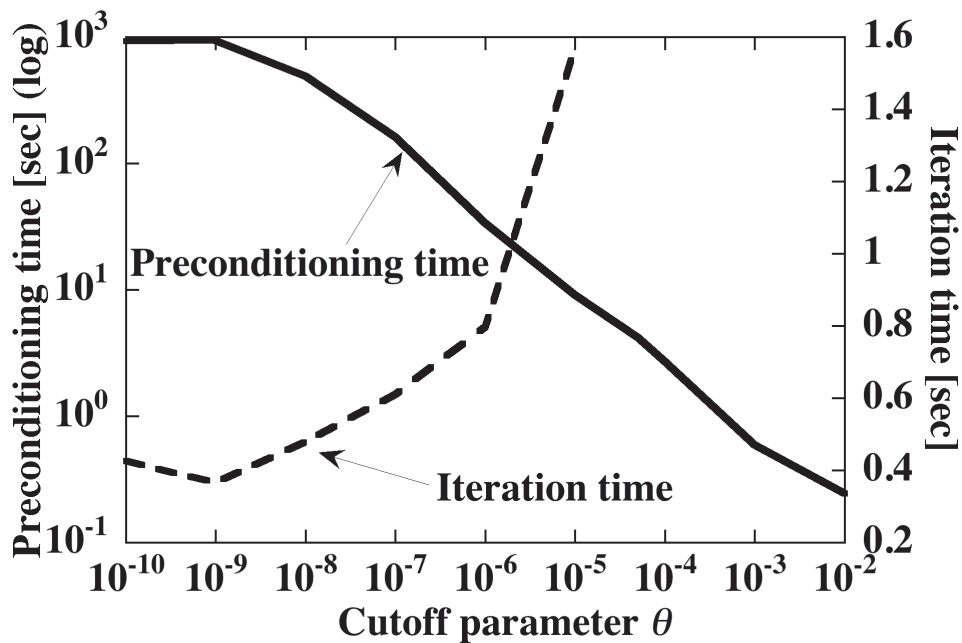


Fig. 3.15: The computation time of GMRES(50) preconditioned BSAIC corresponding to  $\theta$  for EGF ( $\omega = -0.2$ ).

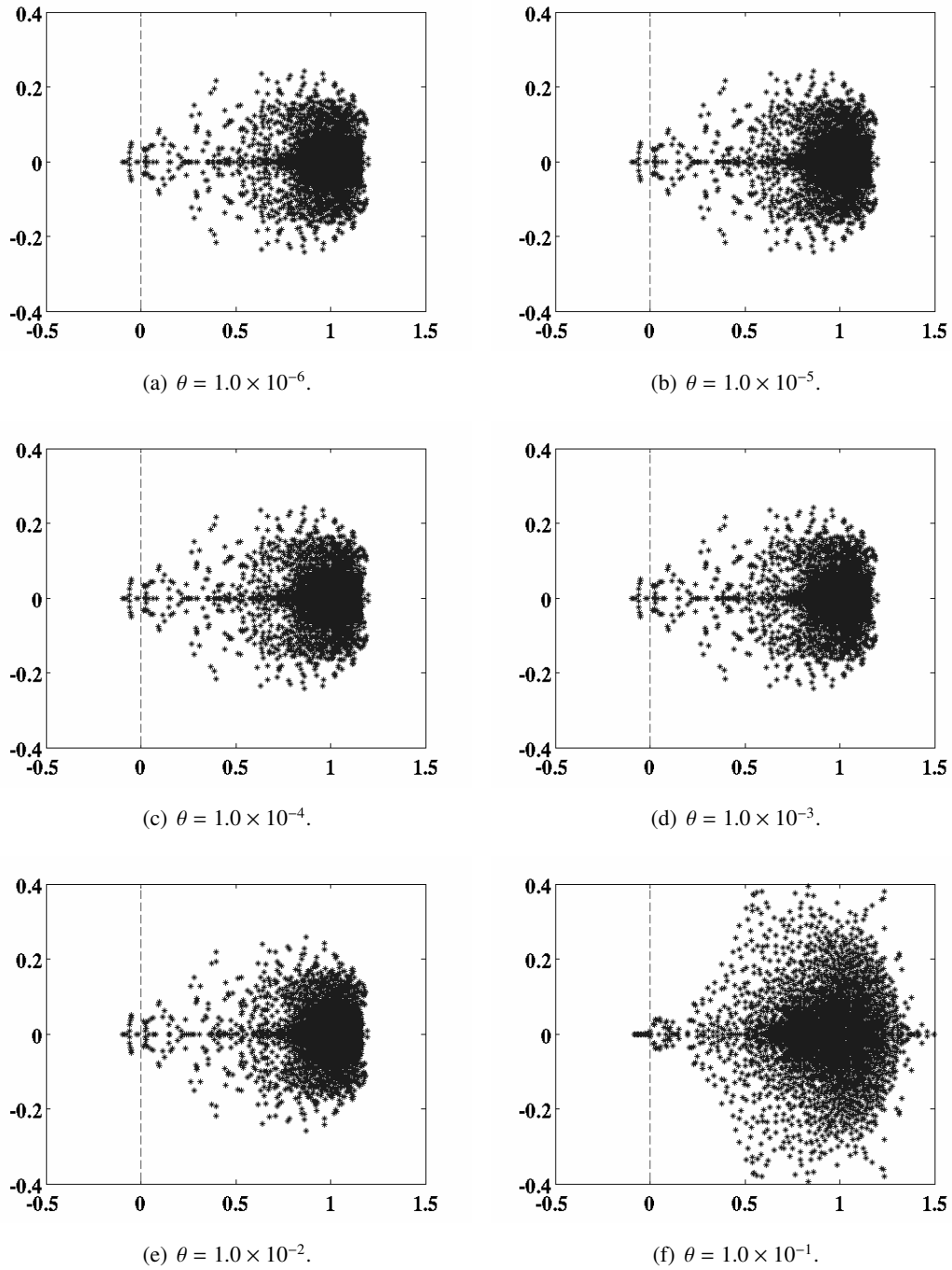


Fig. 3.16: Eigenvalue distributions of  $AM$  for raefsky2.

Table 3.7: The number of eigenvalues of  $AM$  around 0 ( $l = 30$ ) for raefsky2.

$\theta$	$\#( d  < 10^{-1})$	$\#( d  < 10^{-2})$	$\#( d  < 10^{-3})$
$1.0 \times 10^{-5}$	34	0	0
$1.0 \times 10^{-4}$	34	0	0
$1.0 \times 10^{-3}$	34	0	0
$1.0 \times 10^{-2}$	34	0	0
$1.0 \times 10^{-1}$	40	5	0

 Table 3.8: Results of GMRES(50) and GMRES-IR preconditioned with BSAIC ( $l = 30, \theta = 1.0 \times 10^{-5}$ ) for raefsky2.

Krylov	#MV <sub>s</sub>	Wall clock time [sec]			
		Cutoff	Precond.	Iter.	Total
GMRES(50)	†	0.02	4.46	—	—
IR(50, 5)	†	0.02	4.46	—	—
IR(50, 10)	†	0.02	4.46	—	—
IR(50, 15)	†	0.02	4.46	—	—
IR(50, 20)	†	0.02	4.46	—	—
IR(50, 25)	†	0.02	4.46	—	—
IR(50, 30)	†	0.02	4.46	—	—
IR(50, 35)	269	0.02	4.46	2.56	7.04
IR(50, 40)	265	0.02	4.46	2.42	6.90
IR(50, 45)	270	0.02	4.46	2.37	6.85

 Table 3.9: Results of GMRES(50) and GMRES-IR preconditioned with BSAIC ( $l = 30, \theta = 1.0 \times 10^{-3}$ ) for raefsky2.

Krylov	#MV <sub>s</sub>	Wall clock time [sec]			
		Cutoff	Precond.	Iter.	Total
GMRES(50)	†	0.03	4.41	—	—
IR(50, 5)	†	0.03	4.41	—	—
IR(50, 10)	†	0.03	4.41	—	—
IR(50, 15)	1,790	0.03	4.41	17.55	21.99
IR(50, 20)	†	0.03	4.41	—	—
IR(50, 25)	†	0.03	4.41	—	—
IR(50, 30)	†	0.03	4.41	—	—
IR(50, 35)	269	0.03	4.41	2.63	7.07
IR(50, 40)	264	0.03	4.41	2.39	6.83
IR(50, 45)	270	0.03	4.41	2.47	6.91

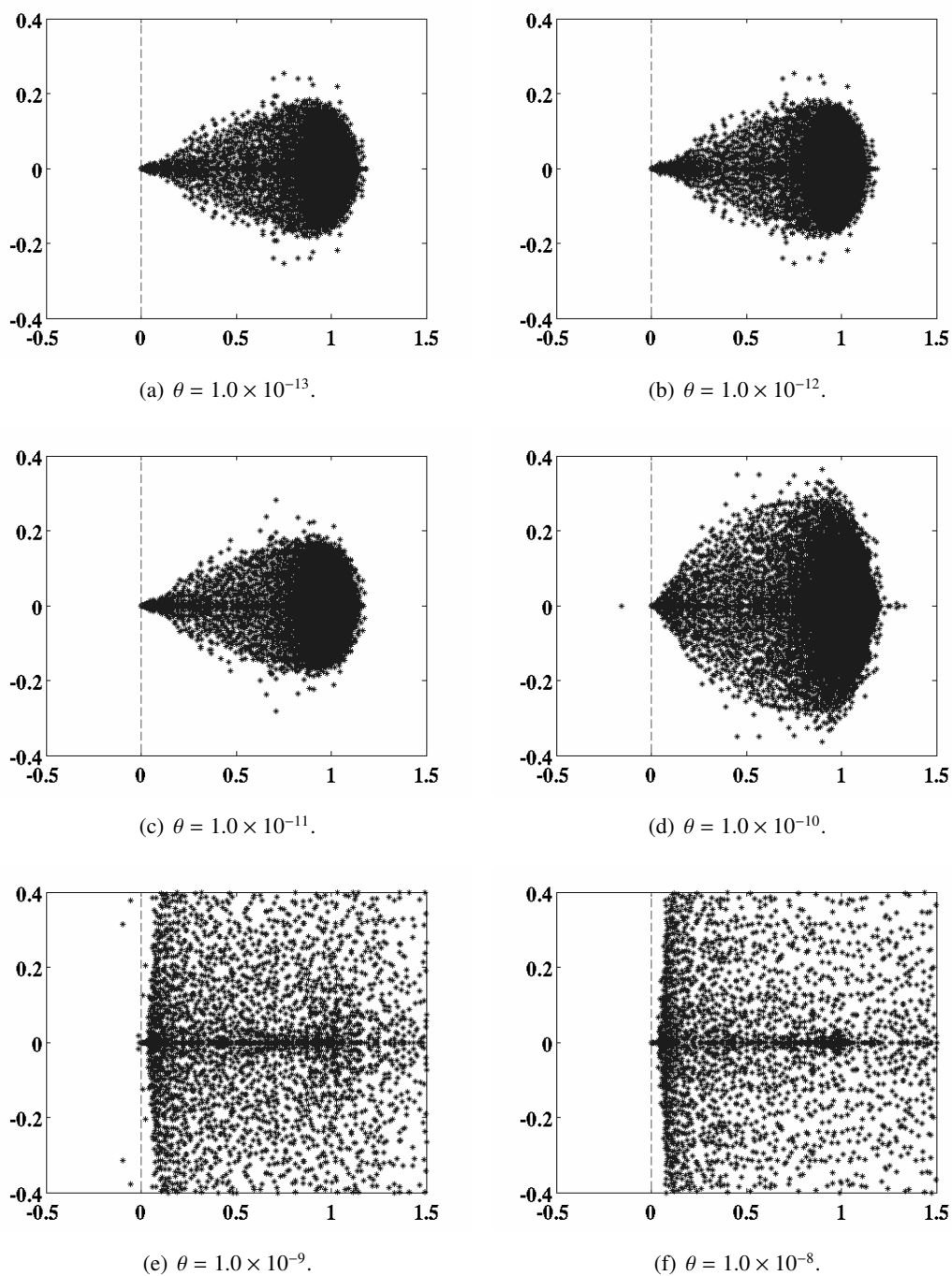


Fig. 3.17: Eigenvalue distributions of  $AM$  for  $fp$ .

Table 3.10: The number of eigenvalues of  $AM$  around 0 ( $l = 30$ ) for fp.

$\theta$	$\#( d  < 10^{-1})$	$\#( d  < 10^{-2})$	$\#( d  < 10^{-3})$
$1.0 \times 10^{-12}$	19	0	0
$1.0 \times 10^{-11}$	33	0	0
$1.0 \times 10^{-10}$	44	0	0
$1.0 \times 10^{-9}$	35	3	0
$1.0 \times 10^{-8}$	19	9	3

 Table 3.11: Results of GMRES(50) and GMRES-IR preconditioned with BSAIC ( $l = 30, \theta = 1.0 \times 10^{-12}$ ) for fp.

Krylov	#MVs	Wall clock time [sec]			
		Cutoff	Precond.	Iter.	Total
GMRES(50)	1,135	0.04	31.17	13.86	45.07
IR(50, 5)	417	0.04	31.17	8.36	39.57
IR(50, 10)	397	0.04	31.17	8.03	39.24
IR(50, 15)	†	0.04	31.17	—	—
IR(50, 20)	390	0.04	31.17	8.23	39.44
IR(50, 25)	†	0.04	31.17	—	—
IR(50, 30)	390	0.04	31.17	7.86	39.07
IR(50, 35)	393	0.04	31.17	7.88	39.09
IR(50, 40)	405	0.04	31.17	8.37	39.58
IR(50, 45)	432	0.04	31.17	8.75	39.96

 Table 3.12: Results of GMRES(50) and GMRES-IR preconditioned with BSAIC ( $l = 30, \theta = 1.0 \times 10^{-10}$ ) for fp.

Krylov	#MVs	Wall clock time [sec]			
		Cutoff	Precond.	Iter.	Total
GMRES(50)	2,418	0.03	4.41	25.28	29.72
IR(50, 5)	636	0.03	4.41	11.00	15.44
IR(50, 10)	626	0.03	4.41	11.09	15.53
IR(50, 15)	†	0.03	4.41	—	—
IR(50, 20)	†	0.03	4.41	—	—
IR(50, 25)	†	0.03	4.41	—	—
IR(50, 30)	†	0.03	4.41	—	—
IR(50, 35)	†	0.03	4.41	—	—
IR(50, 40)	†	0.03	4.41	—	—
IR(50, 45)	†	0.03	4.41	—	—

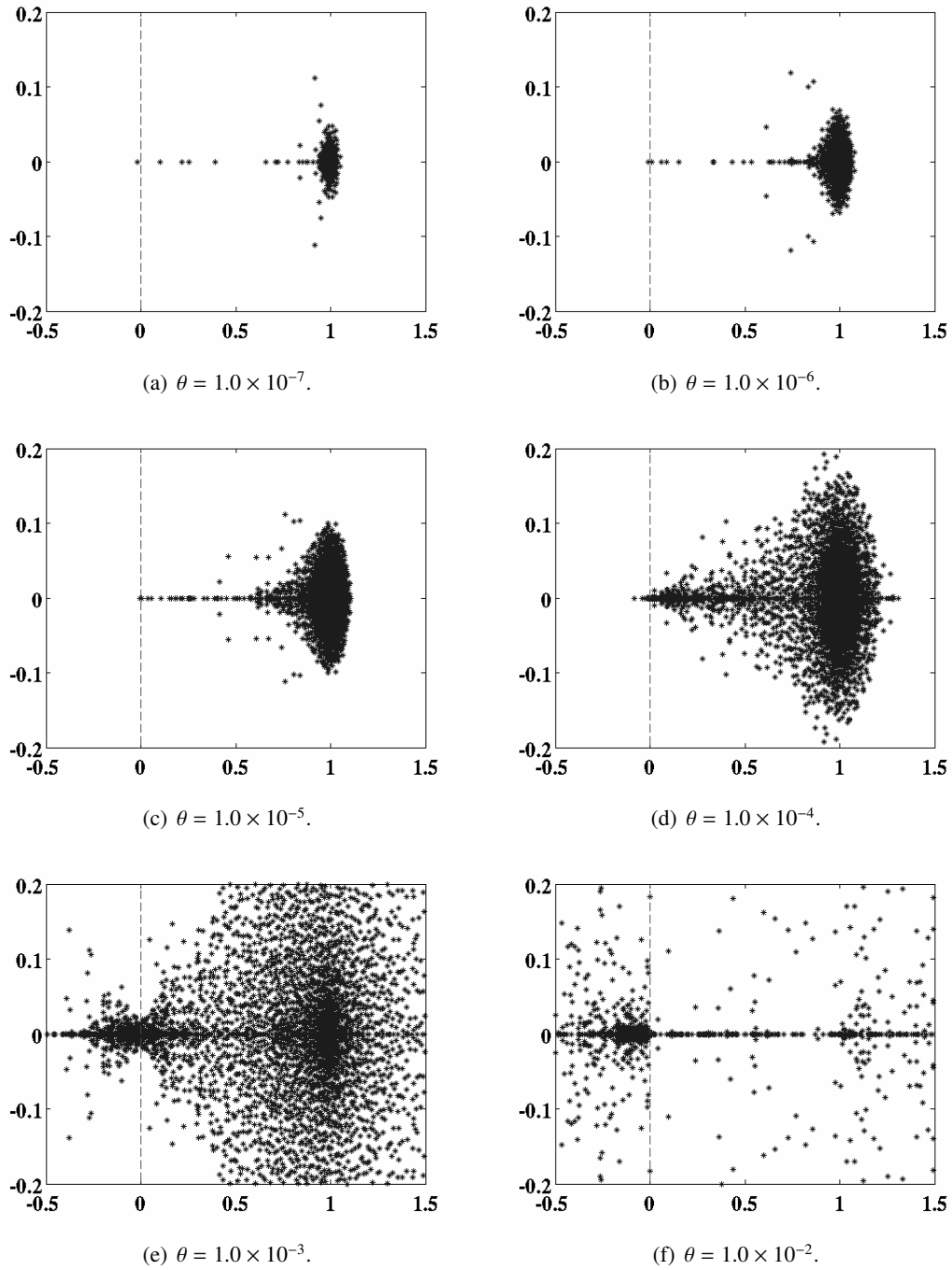


Fig. 3.18: Eigenvalue distributions of AM for EGF.



Table 3.13: The number of eigenvalues of  $AM$  around 0 ( $l = 30$ ) for EGF.

$\theta$	$\#( d  < 10^{-1})$	$\#( d  < 10^{-2})$	$\#( d  < 10^{-3})$
$1.0 \times 10^{-6}$	4	1	0
$1.0 \times 10^{-5}$	5	3	0
$1.0 \times 10^{-4}$	95	15	3
$1.0 \times 10^{-3}$	442	56	1
$1.0 \times 10^{-2}$	564	146	6

 Table 3.14: Results of GMRES(50) and GMRES-IR preconditioned with BSAIC ( $l = 30, \theta = 5.0 \times 10^{-5}$ ) for EGF.

Krylov	#MVs	Wall clock time [sec]			
		Cutoff	Precond.	Iter.	Total
GMRES(50)	†	0.16	4.22	—	—
IR(50, 5)	233	0.16	4.22	7.40	11.78
IR(50, 10)	199	0.16	4.22	6.38	10.76
IR(50, 15)	185	0.16	4.22	5.94	10.32
IR(50, 20)	177	0.16	4.22	5.68	10.06
IR(50, 25)	176	0.16	4.22	5.65	10.03
IR(50, 30)	175	0.16	4.22	5.71	10.09
IR(50, 35)	177	0.16	4.22	5.68	10.06
IR(50, 40)	181	0.16	4.22	5.55	9.93
IR(50, 45)	193	0.16	4.22	5.71	10.09

 Table 3.15: Results of GMRES(50) and GMRES-IR preconditioned with BSAIC ( $l = 30, \theta = 1.0 \times 10^{-4}$ ) for EGF.

Krylov	#MVs	Wall clock time [sec]			
		Cutoff	Precond.	Iter.	Total
GMRES(50)	†	0.14	2.72	—	—
IR(50, 5)	†	0.14	2.72	—	—
IR(50, 10)	†	0.14	2.72	—	—
IR(50, 15)	†	0.14	2.72	—	—
IR(50, 20)	†	0.14	2.72	—	—
IR(50, 25)	†	0.14	2.72	—	—
IR(50, 30)	†	0.14	2.72	—	—
IR(50, 35)	†	0.14	2.72	—	—
IR(50, 40)	†	0.14	2.72	—	—
IR(50, 45)	†	0.14	2.72	—	—

## 第4章 近似係数行列に対する完全分解を用いた前処理法の Cutoff パラメータ自動推定

準疎行列係数連立一次方程式に対する有効な前処理法として Cutoff を用いた前処理 [20, 33] が提案されている。これらの前処理法の性能は Cutoff パラメータ  $\theta$  に大きく影響するため、適切な  $\theta$  を選択することが重要である。

本章では、適切な Cutoff パラメータの見積もりについて述べ、推定法 [32] を提案する。なお、本稿では、Cutoff パラメータのみの設定で前処理行列を生成できる疎行列用直接解法を用いた前処理法 [20] を考える。

### 4.1 緒言

準疎行列  $A$  を係数行列にもつ連立一次方程式

$$Ax = b \tag{4.1}$$

を前処理法付き Krylov 部分空間反復解法で解くことを考える。同方程式に対する有効な前処理法として Cutoff を用いた前処理法である近似係数行列に対する疎行列用直接解法を用いた前処理法 [20] 及び提案法である BSAIC 前処理法 [33] がある。

両前処理法共に Cutoff パラメータが前処理時間や収束性に大きな影響を与える。例えば、大きすぎる Cutoff パラメータを用いた場合は前処理行列生成時間が短くなるが、Krylov 部分空間反復解法の収束性が悪化する。一方、小さすぎる Cutoff パラメータでは収束性は改善されるが、前処理行列の生成に要するコストが膨大になる。反復解法の収束性と前処理行列生成時間のトレードオフを踏まえた適切な Cutoff パラメータを選択することが重要である。

さらに、固有値問題 (1.2) から現れる連立一次方程式に Cutoff を用いた前処理法を適用する場合、 $\omega$  毎に係数行列の性質が異なるため  $\omega$  に応じた Cutoff パラメータの選択が必要となる。

本章では、適切な Cutoff パラメータの推定法を提案する。その際、近似係数行列に対する疎行列用直接解法を用いた前処理法 [20] に着目する。同前処理法は前処理行列作成に要するパラメータが Cutoff パラメータ 1 つのみであることから、第 3 章で述べた提案法の BSAIC 前

処理法に比べパラメータ設定が容易であり前処理法の性能が Cutoff パラメータにのみ左右されるためである。

近似係数行列に対する完全分解に要する演算量と前処理行列の非零要素数を低いコストで見積もることができるシンボリック分解，及び Krylov 部分空間反復解法における残差減少率などを基に，疎行列用直接解法を用いた前処理法において効果的な Cutoff パラメータの推定法を提案する。

本章の構成は次のようになっている。次節で，近似係数行列に対する完全分解を用いた前処理法に関して述べる。次に，適切な Cutoff パラメータを見積もる方法を述べる。その後，数値実験において，準疎行列係数連立一次方程式に対し提案法を適用し，その有効性を確かめる。最後に本章をまとめる。

## 4.2 近似係数行列に対する完全分解を用いた前処理法

2.2.1 及び 2.2.1 節で述べた不完全分解は本研究で扱うような準疎行列係数連立一次方程式に対して有効でない場合がある。Fill-in を考慮しない不完全分解の場合は残差が収束せず求解が不可能であったり，閾値付き不完全分解の場合では非零要素数が多い行列に対して分解を行うため演算量が膨大になることがあるためである [20]。

そこで，本節では，準疎行列係数連立一次方程式に対して有効な前処理法として提案されている Cutoff により作成した近似係数行列に対する疎行列用直接解法を用いた前処理法 [20] について述べる。

式(3.1)で定義された Cutoff について再度述べる。Cutoff は絶対値の小さい要素を削除し非零要素数を削減する方法であり，以下のように定義される。

$$A(\theta) = [\bar{a}_{ij}], \quad \bar{a}_{ij} = \begin{cases} a_{ij}, & (|a_{ij}| > \theta \times \max_{i,j} |a_{ij}| \text{ or } i = j) \\ 0, & \text{otherwise} \end{cases}.$$

ここで， $0 \leq \theta \leq 1$  であり， $A(\theta)$  は非零要素数を削減するため係数行列  $A$  に対し Cutoff を適用して生成された近似係数行列である。また， $A(0) = A$  及び  $A(1) = \text{diag}(A)$  を満たす。 $\text{diag}(A)$  は  $A$  の対角成分のみの対角行列を表す。

Cutoff 適用後，以下のように前処理行列を生成する。

$$A \approx A(\theta) = LU (= K),$$

ここで， $L, U$  は近似係数行列  $A(\theta)$  に対する完全分解によって生成される下三角行列及び上三角行列である。本稿では割愛するが， $L, U$  に対しても再度 Cutoff を施し， $L, U$  の非零要

素数を削減し前進後退代入を高速に行う前処理法 [34] が提案されている。

完全分解は MUMPS [19] や PARDISO [23] などの疎行列用直接解法パッケージを用いて行うことができる。また、Fill-in 数を減らし並列性を確保するため Nested Dissection (ND) オーダリング [12] を  $A(\theta)$  に適用する。上記の前処理法を Krylov 部分空間反復解法に適用し準疎行列係数連立一次方程式の解を得る。

Cutoff パラメータ  $\theta$  が小さくなるにつれ  $A(\theta)$  は  $A$  に近づき前処理行列の性能は向上するが、係数行列  $A$  の完全分解を行うことと等しくなり前処理時間が多くなる。逆に  $\theta$  が大きくなるに従い  $A(\theta)$  は  $A$  を近似できなくなり、前処理行列の性能が低下し収束に要する反復回数が増大することが考えられる。以上より、同前処理法の性能は Cutoff パラメータ  $\theta$  に大きく影響することが分かる。

### 4.3 Cutoff パラメータの自動推定法

推定法では、Cutoff パラメータの候補  $\Theta = \{\theta_0, \theta_1, \dots, \theta_{s-1}\}$  を設定し、 $\Theta$  から前処理行列の生成及び連立一次方程式の求解にかかる合計時間  $T(\theta)$  が最小となる  $\theta$  の自動選択を目指す。また、 $\theta_0 > \theta_1 > \dots > \theta_{s-1}$  とする。提案するアルゴリズムは以下のような4つの Phase から成る：

Phase I : Cutoff パラメータの候補  $\Theta$  を設定し、初期残差  $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$  を計算する。

Phase II : 残差が収束する  $\theta$  の推定を行う

Phase III : 合計時間が最小となる  $\theta$  を推定する

Phase IV : Phase III で推定した  $\theta$  を利用し連立一次方程式 (1.1) を解く

次節以降では、Phase II 及び Phase III について述べ、提案法のアルゴリズムを示す。

#### 4.3.1 残差が収束する Cutoff パラメータの推定

本節では、残差が0に収束するような Cutoff パラメータの範囲を推定する Phase II について述べる。本稿では、Krylov 部分空間反復解法の収束性の指標として

$$\tau(\theta) \equiv \lim_{k \rightarrow \infty} \left( \frac{\|\mathbf{r}_k\|_2}{\|\mathbf{r}_0\|_2} \right)^{1/k} \quad (4.2)$$

で定義される残差減少率  $\tau(\theta)$  を導入する。式 (4.2) における  $\mathbf{r}_0$ ,  $\mathbf{r}_k$  はそれぞれ初期残差ベクトル及び  $k$  反復目の残差ベクトルを表す。ここで、 $\tau(\theta) < 1$  のとき残差は収束する。Phase II では、 $\tau(\theta)$  を見積もり、 $\tau(\theta) < 1$  となる Cutoff パラメータを推定する。以下に Phase II の手順を示す。

Step 0.  $l = 0$  とする. 正の整数  $m (\leq s)$  を設定する.

Step 1. 基準となる Cutoff パラメータ  $\theta_l$  に対し完全分解を行い, 後述する  $\tau(\theta_l)$  の第 1 近似  $\bar{\tau}_1(\theta_l)$  を計算する.

Step 2.  $\cdot \bar{\tau}_1(\theta_l) < 1$  の場合は残差が 0 に収束すると思え, 合計時間  $T(\theta)$  を最小化する Phase III に移行する.

$\cdot \bar{\tau}_1(\theta_l) \geq 1$  の場合は残差が 0 に収束しないと見え,  $i = l + 1$  として Step 3 へ進む.

$\cdot l = s - 1$  の場合は設定した Cutoff パラメータ候補のうち,  $\theta_{s-1}$  が最適と判断し Phase IV に移行する.

Step 3. Cutoff パラメータ  $\theta_i$  に対し後述する  $\tau(\theta_i)$  の第 2 近似  $\bar{\tau}_2(\theta_i)$  を計算する.

Step 4.  $\cdot \bar{\tau}_2(\theta_i) < 1$  または  $i = l + m$  の場合は  $l = i$  とし, Step 1 へ進む.

$\cdot \bar{\tau}_2(\theta_i) \geq 1$  の場合は  $i = i + 1$  として Step 3 へ進む.

なお, Step 0 で設定した  $m$  は  $\bar{\tau}_2(\theta)$  を見積もる上限反復回数である.

以下では, 残差減少率  $\tau(\theta)$  の第 1 近似  $\bar{\tau}_1(\theta)$  及び第 2 近似  $\bar{\tau}_2(\theta)$  について説明する. Krylov 部分空間反復解法として 2.1.4 節で述べた GCR 法を用いると仮定すると, 不等式

$$\|\mathbf{r}_{k+1}\|_2 \leq \|\mathbf{r}_k - AK^{-1}\mathbf{r}_k\|_2,$$

が成り立つ [1, 2]. ここで,  $\mathbf{r}_k, \mathbf{r}_{k+1}$  はそれぞれ前処理法付き GCR 法の  $k$  番目,  $k + 1$  番目の残差ベクトルを表し,  $K^{-1}$  は前処理行列である. 本稿では,  $K^{-1} = A(\theta)^{-1}$  であるため

$$\|\mathbf{r}_{k+1}\|_2 \leq \|\mathbf{r}_k - AA(\theta)^{-1}\mathbf{r}_k\|_2, \quad (4.3)$$

となり,  $\|\mathbf{r}_i - AA(\theta)^{-1}\mathbf{r}_i\|_2 / \|\mathbf{r}_i\|_2$  が  $\mathbf{r}_i$  に大きく依存しないとし, 不等式 (4.3) を用いると式 (4.2) で定義される残差減少率  $\tau(\theta)$  は

$$\tau(\theta) \equiv \lim_{k \rightarrow \infty} \left( \frac{\|\mathbf{r}_k\|_2}{\|\mathbf{r}_0\|_2} \right)^{1/k} \leq \lim_{k \rightarrow \infty} \left( \prod_{i=0}^{k-1} \frac{\|\mathbf{r}_i - AA(\theta)^{-1}\mathbf{r}_i\|_2}{\|\mathbf{r}_i\|_2} \right)^{1/k} \approx \frac{\|\mathbf{r}_0 - AA(\theta)^{-1}\mathbf{r}_0\|_2}{\|\mathbf{r}_0\|_2},$$

と見積もることができる. 本稿では,  $\tau(\theta)$  の第 1 近似  $\bar{\tau}_1(\theta)$  として以下を用いる.

$$\bar{\tau}_1(\theta) \equiv \frac{\|\mathbf{r}_0 - AA(\theta)^{-1}\mathbf{r}_0\|_2}{\|\mathbf{r}_0\|_2}. \quad (4.4)$$

しかしながら, 式 (4.4) のように各 Cutoff パラメータ毎に  $A(\theta)^{-1}$  を計算することは実用上困難であるため, Phase II の Step 1 において  $\theta_l$  で生成した前処理行列を用いることで計算できる  $\bar{\tau}_1(\theta_l)$  を基準に各 Cutoff パラメータにおける  $\bar{\tau}_1(\theta)$  を見積もることを考える.

$\tilde{\tau}_1(\theta)$  を少ない計算量で見積もるため、式 (4.4) を

$$\begin{aligned} \tilde{\tau}_1(\theta) &= \frac{\|(A(\theta) - A)A(\theta)^{-1}\mathbf{r}_0\|_2}{\|\mathbf{r}_0\|_2} \leq \frac{\|A(\theta) - A\|_2 \|A(\theta)^{-1}\|_2 \|\mathbf{r}_0\|_2}{\|\mathbf{r}_0\|_2} \leq \frac{\|A(\theta) - A\|_2}{\|A(\theta)\|_2} \cdot \kappa(A(\theta)) \quad (4.5) \\ &\leq \frac{\|A(\theta) - A\|_F}{\|A(\theta)\|_F} \sqrt{n} \kappa(A(\theta)), \end{aligned}$$

のように式変形する。なお、 $\kappa(A(\theta))$  は  $A(\theta)$  の条件数を表し、 $\kappa(A(\theta)) = \|A(\theta)\|_2 \|A(\theta)^{-1}\|_2$  である。また、式変形には不等式  $\|A\|_2 \leq \|A\|_F \leq \sqrt{n}\|A\|_2$  を用いた。不等式 (4.5) における  $\|\cdot\|_F$  は Frobenius ノルムを表す。 $\|A(\theta) - A\|_F / \|A(\theta)\|_F$  は各 Cutoff パラメータにおいて計算コストが小さい。ここで、

$$\eta(\theta) = \frac{\|A(\theta) - A\|_F}{\|A(\theta)\|_F},$$

とおき、 $\tilde{\tau}_1(\theta) = \eta(\theta)\mu(\theta)$  とすると、

$$\tilde{\tau}_1(\theta) = \tilde{\tau}_1(\theta_l) \cdot \frac{\eta(\theta)\mu(\theta)}{\eta(\theta_l)\mu(\theta_l)},$$

となる。このとき、 $\mu(\theta)$  は  $\theta$  に大きく依存しないとすると、

$$\tilde{\tau}_1(\theta) \approx \tilde{\tau}_1(\theta_l) \cdot \frac{\eta(\theta)}{\eta(\theta_l)},$$

が成り立つ。本稿では、 $\tau(\theta)$  の第 2 近似  $\tilde{\tau}_2(\theta)$  として以下を用いる。

$$\tilde{\tau}_2(\theta) \equiv \tilde{\tau}_1(\theta_l) \cdot \frac{\eta(\theta)}{\eta(\theta_l)}. \quad (4.6)$$

式 (4.4) における残差減少率  $\tau(\theta)$  の第 1 近似  $\tilde{\tau}_1(\theta)$  は  $A(\theta)^{-1}$  の計算が必要となるため、計算量は大きいが良い精度で残差減少率  $\tau(\theta)$  を見積もることができると考えられる。一方、式 (4.6) における  $\tau(\theta)$  の第 2 近似  $\tilde{\tau}_2(\theta)$  は近似を重ねて導出しているため精度は低い可能性があるが、計算量は小さい。

### 4.3.2 最適な Cutoff パラメータの推定

本節では、合計時間が最小となるような Cutoff パラメータを推定する Phase III について述べる。連立一次方程式の求解に要する合計時間は前処理時間と反復時間から成る。疎行列用直接解法を用いた前処理法の場合、前処理に要する時間の多くは完全分解が占める。反復時間については、1 回反復において行列ベクトル積、前進後退代入や内積といった計算が反復回

数分実行される。また、係数行列  $A$  が準疎行列の場合、内積計算は非零要素数に比例する行列ベクトル積及び前進後退代入に比べ微小な計算量である。よって、合計時間  $T(\theta)$  を以下のようなモデル式で表す。

$$T(\theta) = T_{LU}(\theta) + c \cdot N(\theta)[T_{MV} + T_{sol}(\theta)]. \quad (4.7)$$

ここで、 $T_{LU}(\theta)$  は完全分解に要する時間、 $N(\theta)$  は反復回数、 $T_{MV}$  は行列ベクトル積1回の時間、 $T_{sol}(\theta)$  は前進後退代入1回の時間である。 $c$  は1反復あたりの行列ベクトル積の回数を表し、Arnoldi系の解法を用いる場合は  $c = 1$ 、Bi-Lanczos系の解法を用いる場合は  $c = 2$  である。 $T_{LU}(\theta)$ 、 $N(\theta)$ 、 $T_{sol}(\theta)$  は Cutoff パラメータ  $\theta$  に依存する。また、 $T_{MV}$  は  $\theta$  に依存しないため、実測値を用いる。

Phase III では、 $T_{LU}(\theta)$ 、 $N(\theta)$ 、 $T_{sol}(\theta)$  を見積もり、 $T(\theta)$  が最小となる Cutoff パラメータを推定する。以下に Phase III の手順を示す。

- Step 1. Phase II で推定された基準となる Cutoff パラメータ  $\theta_l$  に対する  $A(\theta_l)$  の完全分解から得られる  $T_{LU}(\theta_l)$ 、 $T_{sol}(\theta_l)$  及び後述する反復回数  $N(\theta_l)$  の第1近似  $\tilde{N}_1(\theta_l)$  を基に合計時間の第1近似  $\tilde{T}_1(\theta_l)$  を見積もる。また、 $j = l + 1$  とし Step 2 に進む。
- Step 2. 後述する  $T_{LU}(\theta_j)$ 、 $T_{sol}(\theta_j)$  の近似値  $\tilde{T}_{LU}(\theta_j)$ 、 $\tilde{T}_{sol}(\theta_j)$  及び  $N(\theta_j)$  の第2近似  $\tilde{N}_2(\theta_j)$  を基に、合計時間の第2近似  $\tilde{T}_2(\theta_j)$  を見積もる。
- Step 3.  $\tilde{T}_2(\theta_j) \leq \tilde{T}_2(\theta_{j-1})$  の場合は  $j = j + 1$  とし Step 2 へ進む。  
 $\tilde{T}_2(\theta_j) > \tilde{T}_2(\theta_{j-1})$  の場合は Step 4 へ進む。
- Step 4.  $\tilde{T}_2(\theta_{j-1}) \leq \tilde{T}_1(\theta_l) - T_{LU}(\theta_l)$  の場合は  $l = j - 1$  とし、 $A(\theta_l)$  の完全分解を行い、Step 1 へ進む。  
 $\tilde{T}_2(\theta_{j-1}) > \tilde{T}_1(\theta_l) - T_{LU}(\theta_l)$  の場合は  $\theta_l$  が最適なパラメータと判断し、連立一次方程式 (1.1) を解く Phase IV に移行する。

Phase III における Step 1 では、Phase II で推定された Cutoff パラメータ  $\theta_l$  に対する  $A(\theta_l)$  の完全分解から得られた  $T_{LU}(\theta_l)$  や  $T_{sol}(\theta_l)$  を使い、合計時間の第1近似  $\tilde{T}_1(\theta_l)$  を見積もる。Step 2 及び 3 において、 $\theta_{l+1}$  以降の合計時間の第2近似  $\tilde{T}_2(\theta)$  を見積もり、 $\tilde{T}_2(\theta_j) > \tilde{T}_2(\theta_{j-1})$  を満たした場合に Step 4 に移行する。Step 4 において、合計時間の第2近似  $\tilde{T}_2(\theta_{j-1})$  が  $\theta_l$  における反復時間 ( $\tilde{T}_1(\theta_l) - T_{LU}(\theta_l)$ ) よりも小さい、つまり  $A(\theta_{j-1})$  の完全分解を行っても  $\theta_l$  における計算時間よりも小さいならば Step 1 に移行する。逆に、 $\tilde{T}_2(\theta_{j-1})$  が  $\theta_l$  における反復時間よりも大きいならば  $\theta_l$  の計算結果を使い、Phase IV に移行する。

完全分解に要する時間  $T_{LU}(\theta)$ 、前進後退代入時間  $T_{sol}(\theta)$  及び反復回数  $N(\theta)$  の見積もりについて述べる。まずは、 $T_{LU}(\theta)$ 、 $T_{sol}(\theta)$  の近似  $\tilde{T}_{LU}(\theta)$ 、 $\tilde{T}_{sol}(\theta)$  について述べる。完全分解の演

算量は行列の非零構造で決まり、前進後退代入の演算量は前処理行列の非零要素数で決まる。このことから、Fill-in を特定し完全分解の演算量と前処理行列の非零要素数を見積もるシンボリック分解を利用する。シンボリック分解は完全分解に比べ少ない演算量で行えるため各 Cutoff パラメータにおいて適用が可能と考えられる。

基準とする時間として Cutoff パラメータ  $\theta_l$  における  $T_{\text{LU}}(\theta_l)$  及び  $T_{\text{sol}}(\theta_l)$  の計算時間の実測値を用いる。  $\theta_j$  ( $j = l+1, l+2, \dots$ ) においてシンボリック分解を行い、演算量  $f_j$  及び前処理行列の非零要素数  $nz_j$  を見積もる。計算時間が演算量や非零要素数に比例すると仮定すると、近似前処理時間  $\tilde{T}_{\text{LU}}(\theta_j)$  及び近似前進後退代入時間  $\tilde{T}_{\text{sol}}(\theta_j)$  は

$$\tilde{T}_{\text{LU}}(\theta_j) \equiv T_{\text{LU}}(\theta_l) \times \frac{f_j}{f_l}, \quad \tilde{T}_{\text{sol}}(\theta_j) \equiv T_{\text{sol}}(\theta_l) \times \frac{nz_j}{nz_l},$$

のように算出できる。ただし、  $T_{\text{LU}}(\theta) \approx \tilde{T}_{\text{LU}}(\theta)$ 、  $T_{\text{sol}}(\theta) \approx \tilde{T}_{\text{sol}}(\theta)$  である。

また、反復回数の第1近似  $\tilde{N}_1(\theta_l)$ 、第2近似  $\tilde{N}_2(\theta_j)$  は式 (4.4) 及び式 (4.6) で定義される残差減少率の第1近似  $\tilde{\tau}_1(\theta_l)$ 、第2近似  $\tilde{\tau}_2(\theta_j)$  を用い、それぞれ

$$\tilde{N}_1(\theta_l) \equiv \frac{\log \varepsilon}{\log \tilde{\tau}_1(\theta_l)}, \quad \tilde{N}_2(\theta_j) \equiv \frac{\log \varepsilon}{\log \tilde{\tau}_2(\theta_j)},$$

と表すことができる。ここで、  $\varepsilon$  は収束判定係数であり、  $N(\theta) \approx \tilde{N}_1(\theta) \approx \tilde{N}_2(\theta)$  である。

以上より、完全分解時間  $T_{\text{LU}}(\theta)$  及び前進後退代入時間  $T_{\text{sol}}(\theta)$  の近似は  $\tilde{T}_{\text{LU}}(\theta)$  及び  $\tilde{T}_{\text{sol}}(\theta)$  の各1種類である。また、反復回数  $N(\theta)$  の近似は  $\tilde{N}_1(\theta)$  及び  $\tilde{N}_2(\theta)$  の2種類である。よって、合計時間のモデル式 (4.7) は以下の2つの近似式となる。

$$\begin{aligned} \tilde{T}_1(\theta) &= T_{\text{LU}}(\theta) + c \cdot \tilde{N}_1(\theta)[T_{\text{MV}} + T_{\text{sol}}(\theta)], \\ \tilde{T}_2(\theta) &= \tilde{T}_{\text{LU}}(\theta) + c \cdot \tilde{N}_2(\theta)[T_{\text{MV}} + \tilde{T}_{\text{sol}}(\theta)]. \end{aligned}$$

ただし、  $T(\theta) \approx \tilde{T}_1(\theta) \approx \tilde{T}_2(\theta)$  である。  $T(\theta)$  の第1近似  $\tilde{T}_1(\theta)$  は完全分解を行うため計算コストが大きい。一方、  $T(\theta)$  の第2近似  $\tilde{T}_2(\theta)$  は多くが近似値から成るため精度は低いと考えられるが、完全分解を行わずシンボリック分解までの適用としているため各 Cutoff パラメータでも適用が可能と考えられる。

### 4.3.3 自動選択アルゴリズム

本節では、4.3.1 節及び4.3.2 節で述べた Phase II, III を用いた提案法を述べる。 Fig. 4.1 に提案法のアルゴリズムを示す。本アルゴリズム Fig. 4.1 において、初めに  $s$  個の Cutoff パラメータ  $\theta$  を設定する。 Phase II では、残差減少率  $\tau(\theta)$  の第2近似  $\tilde{\tau}_2(\theta)$  を見積もる上限反復回



数  $m$  を設定後,  $\tau(\theta)$  の第1近似  $\bar{\tau}_1(\theta_l)$  が1を下回る  $\theta_l$  を推定する. 次に, Phase III では, Phase II で推定した  $\theta_l$  を基にシンボリック分解などを利用し, 合計時間  $T(\theta_l)$  が最小となる  $\theta_l$  を推定する. 最後に, Phase IV において推定した  $\theta_l$  を利用し連立一次方程式 (1.1) を解く.

Fig. 4.1 における計算の主要部は Phase II 及び Phase III のシンボリック分解と完全分解である. 演算量は  $\Theta$  と  $m$  の設定に依存するが, 大きい  $\theta$  から分解を行い, 多くの  $\theta$  においては完全分解よりも少ない演算量のシンボリック分解を行うため, 文献 [20] において小さすぎる  $\theta$  を選択するような場合と比べ, 少ない演算量になると考えられる.

Phase II, Phase III とともに, 比較的高コストではあるものの高精度な近似値  $\bar{\tau}_1(\theta)$ ,  $\bar{T}_1(\theta)$  と, 精度は必ずしも高いとは言えないが低コストで計算可能な近似値  $\bar{\tau}_2(\theta)$ ,  $\bar{T}_2(\theta)$  を併用することで, 最適な Cutoff パラメータ  $\theta_l$  の推定を低コストで実現できることが期待される.

## 4.4 数値実験

本節では, 疎行列係数連立一次方程式に対して数値実験を行う. 初めに, 4.2 節で述べた疎行列用直接解法を用いた前処理法の性能を検証する. その後, 前節で述べた提案法の有効性を検証する.

疎行列用直接解法には MUMPS [19] を, ND オーダリングとして SCOTCH [28] をそれぞれ用いた. Krylov 部分空間反復解法は非エルミート行列向き解法であるリスタート付き GCR 法を用い, リスタート周期を 50 とした. リスタート付き GCR 法は Arnoldi 系の解法であるためモデル式 (4.7) において  $c = 1$  となる. また,  $k$  反復目の残差ベクトル  $\mathbf{r}_k$  が  $\|\mathbf{r}_k\|_2 / \|\mathbf{b}\|_2 \leq 10^{-10}$  を満たした場合収束と判定した. 最大行列ベクトル積回数は 5000 回, 初期解ベクトル  $\mathbf{x}_0$  の成分はすべて 0, 右辺ベクトル  $\mathbf{b}$  の成分はランダムとした. 数値実験に用いた計算機は CPU: AMD Opteron Processor 6180 SE(2.50GHz) 12-Core×4, Memory: 256GB(8GB×32), OS: CentOS 5.4 であり, コンパイラは GNU Fortran ver. 4.5.2 を用い, 倍精度で計算した. 計算はすべて 1 コアで実行した.

### 4.4.1 数値例 1

数値例 1 では The University of Florida Sparse Matrix Collection [31] のマイクロ波帯域フィルタの有限要素離散化から現れる行列 “fem\_filter” を対象とした. 同行列  $A$  は式 (1.4) のように  $A = \omega I - F$  と表される. なお, 行列  $F$  は “fem\_filter” であり, 複素非エルミート行列である. 係数行列  $A$  の次元数は 74,062, 非零要素数は 1,731,206 (一行あたりの平均非零要素数は約 23) である. 行列  $A$  の非零構造図及び係数行列  $A$  の行列要素の絶対値の対数による分布をそれぞれ Fig. 4.2 (a) 及び (b) に示す,

### 疎行列用直接解法を用いた前処理法の性能

本小節では、Cutoff パラメータを変化させたときの同前処理法の性能を計算時間の観点から検証する。

Table 4.1 (a) 及び (b) にシフトパラメータ  $\omega$  を  $10^{-3}i$ ,  $-20 + 10^{-3}i$  とし、Cutoff パラメータ  $\theta$  を変化させたときの計算時間を示した。Table 4.1 における #MVs, Prec., Iter., Total はそれぞれ行列ベクトル積回数、前処理時間、反復時間、合計時間を表す。また、† は最大行列ベクトル積回数以内に収束条件を満たさなかったことを示す。

Table 4.1 (a), (b) 共に、 $\theta$  が大きくなるに従い行列ベクトル積回数が増大していく。それに伴い、反復時間も増加していき、 $\omega = 10^{-3}i$  における  $\theta = 1.0 \times 10^{-3}$  のときには最大反復回数以内に収束しなかった。また、 $\omega$  を変化した (a) 及び (b) を比較すると、(b) において、(a) のときでは収束しなかった  $\theta$  を用いても収束している。同実験から  $\theta$  が変化することで前処理時間及び反復時間も変化し、合計時間もそれに応じて変化していくことが分かる。

また、Fig. 4.3 に Cutoff パラメータ  $\theta$  を変化させたときの合計時間の推移をシフトパラメータ毎に示した。Fig. 4.3 (a) では  $10^{-5}$  付近から  $10^{-7}$  にかけて最小であり、(b) では  $10^{-3}$  付近が最小となることが分かる。

### 提案法の有効性の検証

本小節では、前小節同様に  $\omega$  を変化させた数値例 1 に対して提案法を適用し有効性を検証する。提案した Cutoff パラメータ推定アルゴリズム Fig. 4.1 の Phase I における初期 Cutoff パラメータ  $\theta_0$  及び残差減少率  $\tau(\theta)$  の第 2 近似  $\bar{\tau}_2(\theta)$  を見積もる上限反復回数  $m$  を  $(\theta_0, m) = \{(10^{-2}, 5), (10^{-2}, 10), (10^{-2}, 15), (10^{-3}, 5), (10^{-3}, 10), (10^{-3}, 15)\}$  の 6 通りに設定したときの実験結果を Table 4.2 に示す。 $\theta_{\text{est}}$  は選択された Cutoff パラメータであり、Profiling は  $\theta_{\text{est}}$  が選択されるまでに要した時間である。また、Precond. は  $A(\theta_{\text{est}})$  に対する前処理に要した時間、Iteration は反復時間、Total は Profiling, Precond., Iteration の合計時間である。なお、時間の単位は [sec] である。 $\theta_p$  は初期値が  $1.0 \times 10^{-q}$  とするとき  $1.0 \times 10^{-q}$ ,  $9.0 \times 10^{-q-1}$ ,  $8.0 \times 10^{-q-1}$ , ... のように推移するよう設定した。ここで、 $q$  は正の整数である。また、 $\theta_p$  の個数は 55 としたため、 $\theta_0 = 10^{-2}$  のときは  $\theta_{54} = 10^{-8}$  であり、 $\theta_0 = 10^{-3}$  のときは  $\theta_{54} = 10^{-9}$  である。

Table 4.2 (a) に示すように、 $m$  が 5 か 10 の場合には  $\theta_0$  に関わらず  $8.0 \times 10^{-5}$  か  $9.0 \times 10^{-5}$  が選択されたことが分かる。また、(b) の  $\omega = -20 + 10^{-3}i$  の場合では、 $\theta_0 = 10^{-2}$  としたときは選択される  $\theta_{\text{est}}$  がすべて異なっており、 $\theta_0 = 10^{-3}$  としたときには  $\theta_{\text{est}}$  は  $3.0 \times 10^{-4}$  となっている。

#### 4.4.2 数値例 2

数値例 2 では EGFR2(Epidermal Growth Factor Receptor 2) と呼ばれる上皮成長因子受容体の分子軌道計算から現れる行列を対象とした。EGFR は皮膚移植や傷の回復促進など医療の分野で幅広く扱われる重要な分子の一つである。分子軌道計算では一般化固有値問題 (1.2) が現れ、1 節で述べた Sakurai-Sugiura 法 [27] などの固有値解法を適用することで連立一次方程式を解くことになる。同方程式の係数行列  $A$  は式 (1.3) のように  $A = \omega S - F$  と表される。なお、 $F$  は実対称行列であり、 $S$  は正定値実対称行列である。また、 $\omega$  は複素数とするため  $A$  は複素対称行列となる。そのため、完全分解は演算量を小さくするため  $LDL^T$  分解を用いた。係数行列  $A$  の次元数は 96,234、非零要素数は 456,807,964 (一行あたりの平均非零要素数は約 4,747) である。行列  $A$  の非零構造図及び係数行列  $A$  の行列要素の絶対値の対数による分布をそれぞれ Fig. 4.4 (a) 及び (b) に示す、

#### 疎行列用直接解法を用いた前処理法の性能

本小節では、数値例 1 と同様に Cutoff パラメータを変化させたときの同前処理法の性能を計算時間の観点から検証する。

Table 4.3 (a) 及び (b) にシフトパラメータ  $\omega$  を  $-0.6 + 10^{-3}i$ ,  $-30 + 10^{-3}i$  とし、Cutoff パラメータ  $\theta$  を変化させたときの計算時間を示した。Table 4.3 における #MVs, Prec., Iter., Total は数値例 1 同様、それぞれ行列ベクトル積回数、前処理時間、反復時間、合計時間を表す。また、† は最大行列ベクトル積回数以内に収束条件を満たさなかったことを示す。

Table 4.3 (a), (b) 共に、 $\theta$  が大きくなるに従い前処理時間は減少していく。しかしながら、反復時間は増加していき、Table 4.3 (a) における  $\theta = 1.0 \times 10^{-4}$  のときには最大反復回数以内に収束しなかった。また、 $\omega$  を変化した (a) 及び (b) を比較すると、(b) において、(a) のときでは収束しなかった  $\theta$  を用いても収束している。同実験から  $\theta$  が小さい場合には反復時間は短いが前処理時間は長くなり、一方  $\theta$  が大きいときには前処理時間は短いが反復時間が長くなるか解に収束しないことが分かる。

また、Fig. 4.5 に Cutoff パラメータ  $\theta$  を変化させたときの合計時間の推移をシフトパラメータ  $\omega$  毎に示した。Fig. 4.5 より、 $\omega$  によって合計時間が最小となる  $\theta$  が異なることが分かる。

#### 提案法の有効性の検証

本小節では、 $\omega$  を変化した数値例 2 に対して提案法を適用し有効性を検証する。本アルゴリズム Fig. 4.1 における初期 Cutoff パラメータ  $\theta_0$  及び残差減少率  $\tau(\theta)$  の第 2 近似  $\tilde{\tau}_2(\theta)$  を見積もる上限反復回数  $m$  を  $(\theta_0, m) = (10^{-2}, 5), (10^{-2}, 10), (10^{-2}, 15), (10^{-3}, 5), (10^{-3}, 10)$ ,

$(10^{-3}, 15)$  の 6 通りに設定したときの実験結果を Table 4.4 に示す. 数値例 1 と同様,  $\theta_{\text{est}}$  は選択された Cutoff パラメータであり, Profiling は  $\theta_{\text{est}}$  が選択されるまでに要した時間である. また, Precond. は  $A(\theta_{\text{est}})$  に対する前処理に要した時間, Iteration は反復時間, Total は Profiling, Precond., Iteration の合計時間である. なお, 時間の単位は [sec] である.  $\theta_p$  は初期値が  $1.0 \times 10^{-q}$  とするとき  $1.0 \times 10^{-q}$ ,  $9.0 \times 10^{-q-1}$ ,  $8.0 \times 10^{-q-1}$ , ... のように推移するよう設定した. ここで,  $q$  は正の整数であり,  $\theta_p$  の個数は 55 個とした.

Table 4.4 (a) より,  $\omega = -0.6 + 10^{-3}i$  の場合,  $\theta_0 = 10^{-2}$  のときは  $\theta_{\text{est}}$  は  $1.0 \times 10^{-5}$  か  $2.0 \times 10^{-5}$  が選択され,  $\theta_0 = 10^{-3}$  のときは  $m$  に依らず  $1.0 \times 10^{-5}$  が選択されたことが分かる. また, (b) の  $\omega = -30 + 10^{-3}i$  の場合では選択される  $\theta_{\text{est}}$  に差異が生じている.

$\theta_{\text{est}}$  が合計時間を最小とする  $\theta$  かを確認するため, Fig. 4.5 と比較する. Fig. 4.5 (a) の  $\omega = -0.6 + 10^{-3}i$  では,  $\theta = 1.0 \times 10^{-5}$  のときに合計時間が最小となっている. Fig. 4.5 (b) においては,  $\theta = 1.0 \times 10^{-4}$  付近の合計時間はほぼ同じになっており  $\theta$  は  $1.0 \times 10^{-5}$  から  $5.0 \times 10^{-4}$  付近で  $\theta_{\text{est}}$  が推定できれば良い. 以上より, 提案法を用いることで合計時間が最小となる  $\theta$  は見積もれたと考えられる.

最後に, Profiling の時間の内訳を Fig. 4.6 に示す. Fig. 4.6 (a) は  $\omega = -0.6 + 10^{-3}i$  の場合, (b) は  $\omega = -30 + 10^{-3}i$  の場合である. Fig. 4.6 における “Calc. of  $\eta(\theta)$ ” はアルゴリズム Fig. 4.1 の Phase II 及び Phase III の Frobenius ノルムを含む  $\eta(\theta)$  の計算時間であり, “Cutoff” は  $A(\theta)$  の作成に要する時間, “Symb.” 及び “Fact.” は Phase II 及び Phase III のシンボリック分解時間, 完全分解時間を表す.  $\theta_0$  と  $m$  の選択によって計算時間に差が生じており, 全体的に完全分解が Profiling 時間の多くを占めていることが分かる.

## 4.5 結言

非零要素数が多く, 絶対値の小さい要素を多く含む行列を係数行列にもつ連立一次方程式に対して有効な前処理法として近似係数行列に対する疎行列用直接解法を用いた前処理法 [20] が提案されている. 本章では, 同前処理法の性能に大きく影響する Cutoff パラメータの自動推定法を提案した. また, 絶対値の小さい要素が多い行列を 3 つ例に挙げ, 提案法の有効性を検証した.

提案法は, 初期 Cutoff パラメータ  $\theta_0$ , 近似残差減少率を見積もる上限反復回数  $m$  の値に依らず概ね適切な Cutoff パラメータを選択することに成功した. しかしながら, 適切な Cutoff パラメータが選択されるまでに要する時間の割合が多かった. そのため現段階では, 計算時間の観点からシフトパラメータ毎に本提案法を適用することは厳しいが, 本提案法で選択された Cutoff パラメータを 1 つの基準値として他のシフトパラメータにおける Cutoff パラメータの選択に利用できると考えられる.

**1: Phase I**

Set  $\Theta = \{\theta_p \mid 0 \leq \theta_p \leq 1, p = 0, 1, \dots, s-1\}$  ( $\theta_0 > \theta_1 > \dots > \theta_{s-1}$ )  
 Compute  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$

**2: Phase II**

Set  $l = 0, m (< s)$

Measure  $T_{\text{MV}}$

For  $k = 1, 2, \dots$

    Compute  $A(\theta_l) = LU$  by a sparse direct solver

    Measure  $T_{\text{LU}}(\theta_l), T_{\text{sol}}(\theta_l)$

$\tilde{\tau}_1(\theta_l) = \|\mathbf{r}_0 - AA(\theta_l)^{-1}\mathbf{r}_0\|_2 / \|\mathbf{r}_0\|_2$

$\eta(\theta_l) = \|A(\theta_l) - A\|_{\text{F}} / \|A(\theta_l)\|_{\text{F}}$

    If  $\tilde{\tau}_1(\theta_l) < 1$  exit

    If  $l = s - 1$  go to Phase IV

    For  $i = 1, 2, \dots, \min(m, s - l - 1)$

$\eta(\theta_{l+i}) = \|A(\theta_{l+i}) - A\|_{\text{F}} / \|A(\theta_{l+i})\|_{\text{F}}$

$\tilde{\tau}_2(\theta_{l+i}) = \tilde{\tau}_1(\theta_l) [\eta(\theta_{l+i}) / \eta(\theta_l)]$

        If  $\tilde{\tau}_2(\theta_{l+i}) < 1$  exit

    End For

$l = l + i$

End For

**3: Phase III**

For  $k = 1, 2, \dots$

$\tilde{N}_1(\theta_l) = \log \varepsilon / \log \tilde{\tau}_1(\theta_l)$

$\tilde{T}_1(\theta_l) = T_{\text{LU}}(\theta_l) + c \cdot \tilde{N}_1(\theta_l) [T_{\text{MV}} + T_{\text{sol}}(\theta_l)]$

    For  $j = l + 1, l + 2, \dots, s - 1$

        Estimate  $f_j, nz_j$  by Symbolic Factorization of  $A(\theta_j)$

$\tilde{T}_{\text{LU}}(\theta_j) = T_{\text{LU}}(\theta_l) \cdot (f_j / f_l)$

$\tilde{T}_{\text{sol}}(\theta_j) = T_{\text{sol}}(\theta_l) \cdot (nz_j / nz_l)$

$\eta(\theta_j) = \|A(\theta_j) - A\|_{\text{F}} / \|A(\theta_j)\|_{\text{F}}$

$\tilde{\tau}_2(\theta_j) = \tilde{\tau}_1(\theta_l) [\eta(\theta_j) / \eta(\theta_l)]$

$\tilde{N}_2(\theta_j) = \log \varepsilon / \log \tilde{\tau}_2(\theta_j)$

$\tilde{T}_2(\theta_j) = \tilde{T}_{\text{LU}}(\theta_j) + c \cdot \tilde{N}_2(\theta_j) [T_{\text{MV}} + \tilde{T}_{\text{sol}}(\theta_j)]$

        If  $\tilde{T}_2(\theta_j) > \tilde{T}_2(\theta_{j-1})$  exit

    End For

    If  $\tilde{T}_2(\theta_{j-1}) > \tilde{T}_1(\theta_l) - T_{\text{LU}}(\theta_l)$  or  $l = s - 1$  exit

$l = j - 1$

    Compute  $A(\theta_l) = LU$  by a sparse direct solver

    Measure  $T_{\text{LU}}(\theta_l), T_{\text{sol}}(\theta_l)$

$\tilde{\tau}_1(\theta_l) = \|\mathbf{r}_0 - AA(\theta_l)^{-1}\mathbf{r}_0\|_2 / \|\mathbf{r}_0\|_2$

$\eta(\theta_l) = \|A(\theta_l) - A\|_{\text{F}} / \|A(\theta_l)\|_{\text{F}}$

End For

**4: Phase IV**

Solve  $A\mathbf{x} = \mathbf{b}$  using the preconditioner  $K = A(\theta_l) = LU$

Fig. 4.1: Algorithm of our proposed method.

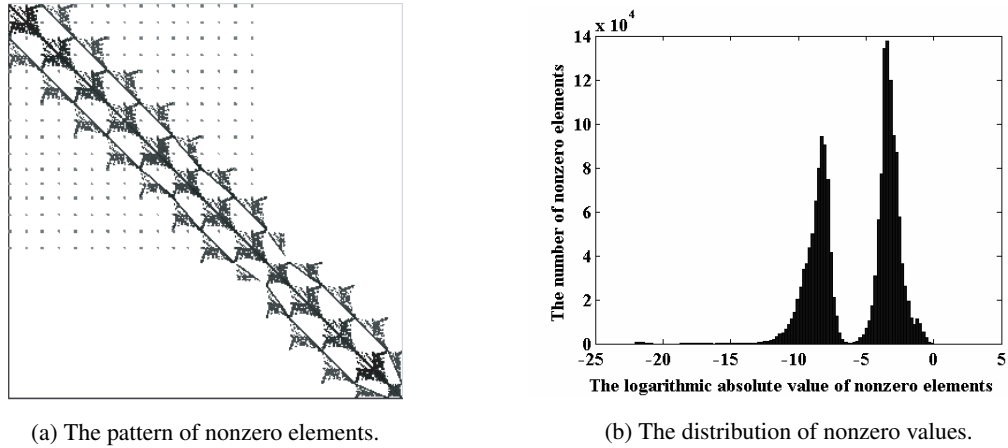


Fig. 4.2: The features of  $A$  for `fem_filter`.

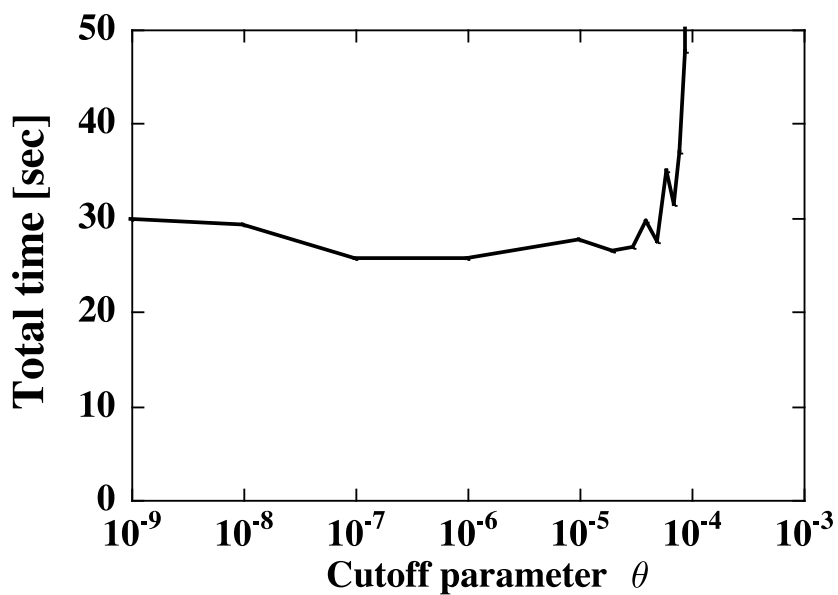
Table 4.1: Computation time corresponding to  $\theta$  for `fem_filter`.

(a)  $\omega = 10^{-3}i$

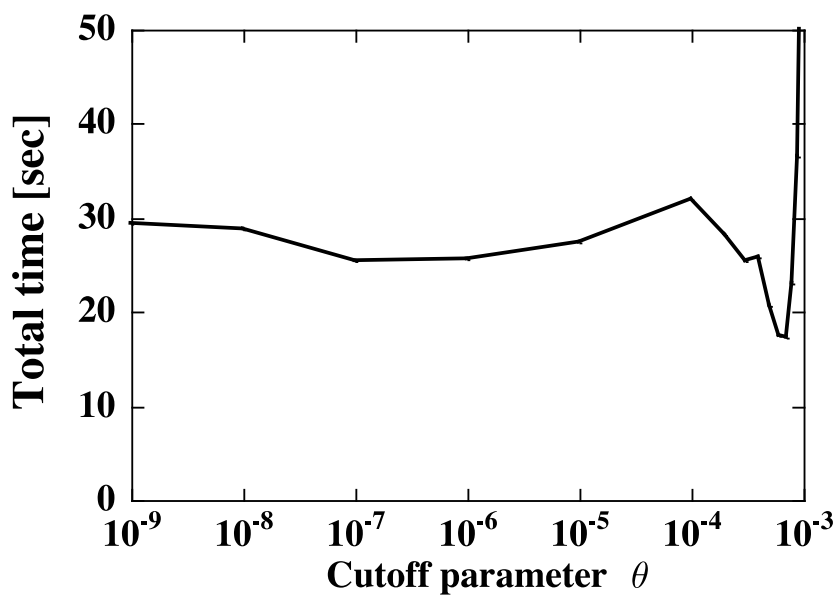
$\theta$	$1.0 \times 10^{-9}$	$1.0 \times 10^{-7}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-3}$	
#MVs	3	4	8	190	†	
Time [sec]	Prec.	28.95	24.73	25.98	28.77	2.78
	Iter.	0.72	0.85	1.51	35.31	—
	Total	29.67	25.58	27.49	64.08	—

(b)  $\omega = -20 + 10^{-3}i$

$\theta$	$1.0 \times 10^{-9}$	$1.0 \times 10^{-7}$	$1.0 \times 10^{-5}$	$5.0 \times 10^{-4}$	$1.0 \times 10^{-3}$	
#MVs	2	3	4	42	779	
Time [sec]	Prec.	28.81	24.74	26.48	14.21	2.81
	Iter.	0.55	0.69	0.90	6.41	73.59
	Total	29.36	25.43	27.38	20.62	76.40



(a)  $\omega = 10^{-3}i$



(b)  $\omega = -20 + 10^{-3}i$

Fig. 4.3: Total time corresponding to  $\theta$  for fem\_filter.

Table 4.2: Results of our proposed method for fem\_filter.

(a)  $\omega = 10^{-3}i$

$(\theta_0, m)$	$(10^{-2}, 5)$	$(10^{-2}, 10)$	$(10^{-2}, 15)$	$(10^{-3}, 5)$	$(10^{-3}, 10)$	$(10^{-3}, 15)$
$\theta_{\text{est}}$	$8.0 \times 10^{-5}$	$8.0 \times 10^{-5}$	$5.0 \times 10^{-5}$	$9.0 \times 10^{-5}$	$9.0 \times 10^{-5}$	$4.0 \times 10^{-5}$
Profiling	29.28	8.85	26.17	22.98	4.91	4.94
Precond.	24.72	24.72	26.04	25.40	25.40	23.17
Iteration	13.78	13.78	4.68	22.93	22.93	3.52
Total	67.78	47.35	56.89	71.31	53.24	31.63

(b)  $\omega = -20 + 10^{-3}i$

$(\theta_0, m)$	$(10^{-2}, 5)$	$(10^{-2}, 10)$	$(10^{-2}, 15)$	$(10^{-3}, 5)$	$(10^{-3}, 10)$	$(10^{-3}, 15)$
$\theta_{\text{est}}$	$4.0 \times 10^{-4}$	$2.0 \times 10^{-4}$	$5.0 \times 10^{-4}$	$3.0 \times 10^{-4}$	$3.0 \times 10^{-4}$	$3.0 \times 10^{-4}$
Profiling	11.24	6.83	7.67	14.33	14.33	14.33
Precond.	18.96	22.96	14.21	21.81	21.81	21.81
Iteration	5.49	2.45	6.41	3.50	3.50	3.50
Total	35.69	32.24	28.29	39.64	39.64	39.64



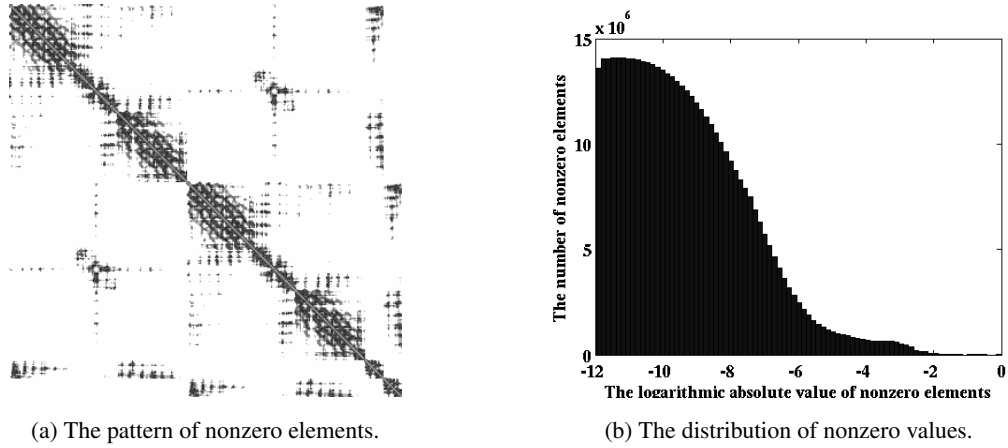


Fig. 4.4: The features of  $A$  for EGFR2.

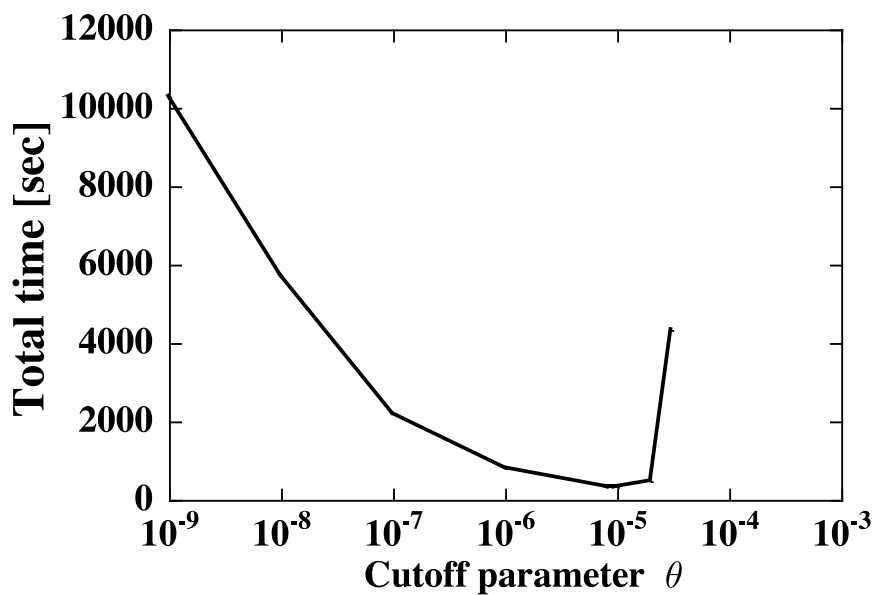
Table 4.3: Computation time corresponding to  $\theta$  for EGFR2.

(a)  $\omega = -0.6 + 10^{-3}i$

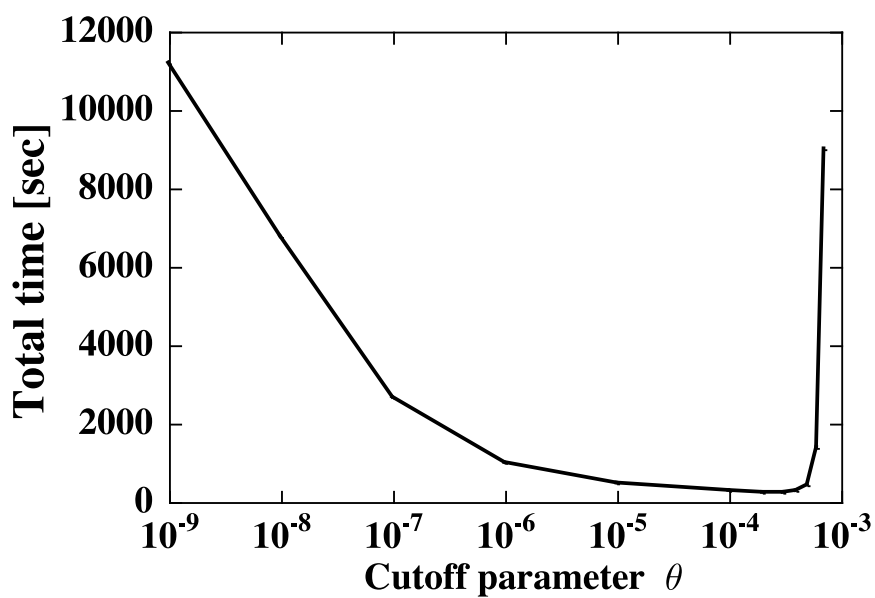
$\theta$	$1.0 \times 10^{-8}$	$1.0 \times 10^{-7}$	$1.0 \times 10^{-6}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-4}$	
#MVs	4	5	8	23	†	
Time [sec]	Prec.	5658.51	2142.56	768.19	236.82	47.76
	Iter.	39.31	36.13	42.21	90.69	—
	Total	5697.82	2178.69	810.40	327.51	—

(b)  $\omega = -30 + 10^{-3}i$

$\theta$	$1.0 \times 10^{-7}$	$1.0 \times 10^{-6}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-4}$	$5.0 \times 10^{-4}$	
#MVs	3	4	6	12	83	
Time [sec]	Prec.	2619.72	977.61	442.62	219.26	133.31
	Iter.	26.25	26.13	30.52	48.93	288.29
	Total	2645.97	1003.74	473.14	268.19	421.60



(a)  $\omega = -0.6 + 10^{-3}i$



(b)  $\omega = -30 + 10^{-3}i$

Fig. 4.5: Total time corresponding to  $\theta$  for EGFR2.

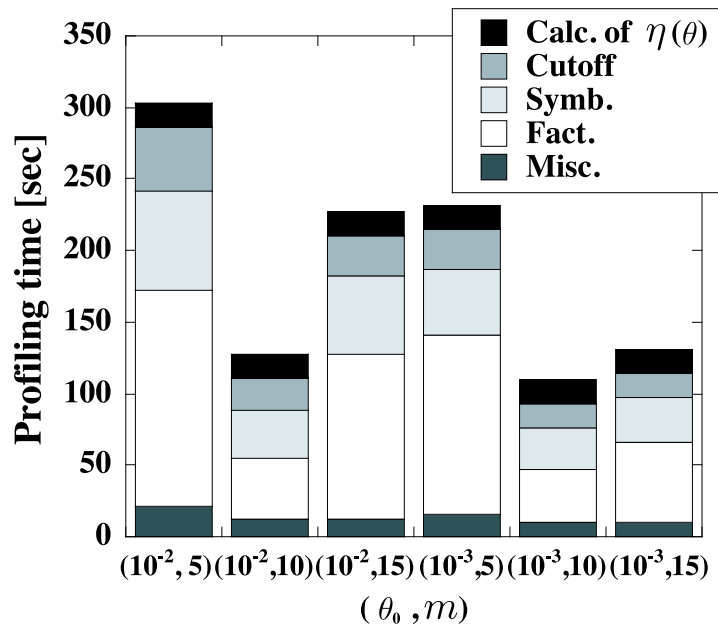
Table 4.4: Results of our proposed method for Example 2.

(a)  $\omega = -0.6 + 10^{-3}i$

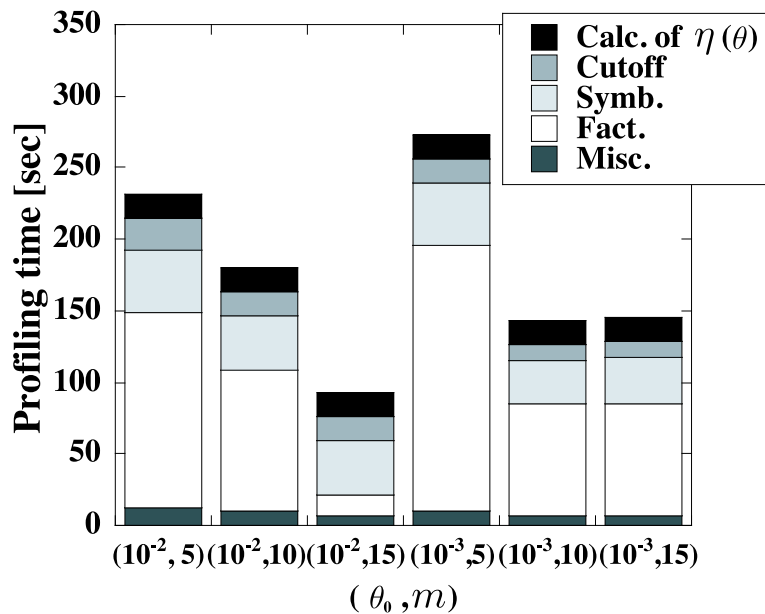
$(\theta_0, m)$	$(10^{-2}, 5)$	$(10^{-2}, 10)$	$(10^{-2}, 15)$	$(10^{-3}, 5)$	$(10^{-3}, 10)$	$(10^{-3}, 15)$
$\theta_{\text{est}}$	$2.0 \times 10^{-5}$	$1.0 \times 10^{-5}$	$2.0 \times 10^{-5}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-5}$
Profiling	303.47	127.31	227.38	231.76	109.92	130.57
Precond.	161.65	236.82	161.65	236.82	236.82	236.82
Iteration	308.18	90.69	308.18	90.69	90.69	90.69
Total	773.30	454.82	697.21	559.27	437.43	458.08

(b)  $\omega = -30 + 10^{-3}i$

$(\theta_0, m)$	$(10^{-2}, 5)$	$(10^{-2}, 10)$	$(10^{-2}, 15)$	$(10^{-3}, 5)$	$(10^{-3}, 10)$	$(10^{-3}, 15)$
$\theta_{\text{est}}$	$4.0 \times 10^{-4}$	$8.0 \times 10^{-5}$	$4.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$9.0 \times 10^{-5}$	$4.0 \times 10^{-5}$
Profiling	231.17	180.14	93.07	273.43	142.80	145.38
Precond.	137.57	235.04	137.57	219.26	230.43	286.15
Iteration	138.51	45.53	138.51	48.93	45.53	35.93
Total	507.25	460.71	369.15	541.62	418.76	467.46



(a)  $\omega = -0.6 + 10^{-3}i$



(b)  $\omega = -30 + 10^{-3}i$

Fig. 4.6: Breakdown of Profiling time for EGFR2.

## 第5章 結論

本論文では、1行あたりの非零要素数が多く、行列要素の値が広く分布し絶対値の小さい要素が多い行列（準疎行列）を係数行列にもつ連立一次方程式（準疎行列係数連立一次方程式）に対する前処理法の研究を行った。まず、準疎行列係数連立一次方程式に対して有効な高い計算並列性をもつ前処理法を提案した。また、準疎行列係数連立一次方程式に対する前処理法の性能に大きく影響を与える Cutoff パラメータの自動推定法を確立した。

第2章では、Krylov 部分空間反復解法と同解法的一种である GMRES 法、リスタート付き GMRES 法、GMRES-IR 法、GCR 法及びリスタート付き GCR 法について述べ、それらのアルゴリズムを示した。また、前処理法の意義と代表的な前処理法である不完全分解前処理法及び近似逆行列前処理法（SAI）に関して述べた。

第3章では、まず、SAI の Block 版である Block SAI と疎行列化手法である Cutoff について述べた。次に、準疎行列係数連立一次方程式に対する並列性が高い前処理法として SAI を改良した提案法である Cutoff 付きブロック近似逆行列前処理法（BSAIC）について述べた。また、BSAIC における Cutoff と Block 化の影響についても言及した。加えて、収束しなくなる原因を固有値の観点から解析し改善策を提示した。得られた結果をまとめると以下の通りとなる。

1. 提案法である BSAIC を用いることによって、従来法である SAI や Block SAI に比べ前処理時間が減少した。
2. BSAIC と Cutoff を適用した SAI (SAIC) を比較した場合、BSAIC は Block 化の影響で前処理時間及び行列ベクトル積回数が減少し、合計時間も減少する傾向が見られた。
3. 0 付近の微小な固有値と Krylov 部分空間反復解法の収束性に関連性があることを、係数行列  $A$  と前処理行列  $M$  の固有値分布から観測できた。
4. 微小な固有値を取り除くデフレーション系の解法を用いることで BSAIC を付加した Krylov 部分空間反復解法の収束性を向上させることが可能になった。

以上より、BSAIC は様々な分野から現れ準疎行列係数連立一次方程式に対して有効な前処理法と言え、デフレーション系の解法を用いることでよりロバストな前処理法になると言える。しかしながら、BSAIC における多くのパラメータ、特に Cutoff パラメータの選択によっ

て収束性や計算時間が大きく変化することが問題点として浮かび上がった。そのような状況を回避する手段として、次の章で提案した Cutoff パラメータの自動推定法である。

第4章では、先行研究である近似係数行列に対する疎行列用直接解法を用いた前処理法 [20] が第3章で述べた提案法である BSAIC に比べ設定するパラメータが Cutoff パラメータのみであり、前処理法の性能が Cutoff パラメータのみに影響を受けることに着目し、同前処理法に対する Cutoff パラメータの自動推定法を提案した。自動推定法を提案するにあたり、残差が収束する Cutoff パラメータの推定及び合計時間が最小となるような Cutoff パラメータの推定を考えた。得られた結果をまとめると以下の通りとなる。

1. Cutoff パラメータの選び方によっては BSAIC 同様、前処理に要する時間が多くなる場合や収束しない、または収束に要する時間が多くなる場合があった。
2. 初期 Cutoff パラメータの設定等に依らず概ね適切な Cutoff パラメータの選択ができた。
3. 係数行列  $A$  の作成に用いるシフトパラメータを変化させた場合にも適切な Cutoff パラメータが選択された。

以上より、Cutoff パラメータの自動推定法は準疎行列係数連立一次方程式に対して有効であると言える。しかしながら、適切な Cutoff パラメータが選択されるまでに要する時間が多いことなどから、さらに改善すべき必要がある。また、より高い精度で適切な Cutoff パラメータを見積もることができる指標の導出も今後の課題の1つである。

## 参考文献

- [1] 阿部 邦美, 張 紹良, 残差最小性に基づく Krylov 部分空間解法に対する可変的前処理, 数理解析研究所講究録, **1265**(2002), 118–128.
- [2] Abe, K. and Zhang, S.-L., A variable preconditioning using the SOR method for GCR-like methods, *Int. J. Numer. Anal. Model.*, **2**(2005), 147–161.
- [3] Arnoldi, W. E., The principle of minimized iterations in the solution of the matrix eigenvalue problem, *Quart. Appl. Math.*, **9**(1951), 17–29.
- [4] Bai, J. Demmel, J. Dongarra, A. Ruhe and H. A. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems*, SIAM, Philadelphia (2000).
- [5] Barnard, S. T. and Grote, M. J., A block version of the SPAI preconditioner, in *Proc. of the 9th SIAM conf. on Parallel Process. for Sci. Comput*, San Antonio, TX (1999).
- [6] Barret, R., Berry, M., Chan, T., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romime, C. and H. A. van der Vorst, H.A., *Templates for the solution of linear systems: Building blocks for iterative methods*, SIAM (1994).
- [7] Benzi, M. and Tuma, M., A sparse approximate inverse preconditioner for nonsymmetric linear systems, *SIAM J. Sci. Comput.*, **19**(1998), 968–994.
- [8] Chow, E. and Saad, Y., Approximate inverse preconditioners via sparse-sparse iterations, *SIAM J. Sci. Comput.*, **19**(1998), 995–1023.
- [9] 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.
- [10] Francis, J. G. F., The QR transformation: A unitary analogue to the LR transformation, Parts I and II, *Comp. J.* **4**(1961), 265–272, 332–345.
- [11] 藤野 清次, 張 紹良, 反復法の数理, 朝倉書店, 東京 (1996).

- [12] George, A., Nested dissection of a regular finite element mesh, *SIAM J. Numer. Anal.*, **10**(1973), 345–363.
- [13] Gould, N. I. M. and Scott, J. A., Sparse Approximate-inverse preconditioners using norm-minimization techniques, *SIAM J. Sci. Comput.*, **19**(1998), 605–625.
- [14] Goossens, S. and Roose, D., Ritz and harmonic Ritz values and the convergence of FOM and GMRES, *Numer. Linear Algebra Appl.*, **6**(1999), 281–293.
- [15] Grote, M. and Huckel, T., Parallel preconditioning with sparse approximate inverses, *SIAM J. Sci. Comput.*, **18**(1997), 838–853.
- [16] 卷幡憲俊, 宇都宮智昭, 渡邊英一, 波浪回折問題のための境界要素法への GMRES-IR の適用, *応用力学論文集*, **6**(2003), 275–281.
- [17] 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, *Math. Comput.* **31**(1997), 148–162.
- [18] Morgan, R. B., Implicitly restarted GMRES and Arnoldi methods for nonsymmetric systems of equations, *SIAM J. matrix Anal. Appl.*, **21**(2000), 1112–1135.
- [19] MUMPS, <http://mumps.enseeiht.fr/>.
- [20] 岡田 真幸, 櫻井 鉄也, 寺西 慶太, 近似係数行列に対する疎行列用直接解法を用いた前処理, *応用数学会論文誌*, **17**(2007), 319–329.
- [21] Paige, C. C., Parlett, B. N. and van der Vorst, H. A., Approximate solution and eigenvalue bounds from Krylov subspaces, *Numer. Linear Algebra Appl.*, **2**(1995), 115–133.
- [22] ParaSails, <https://computation.llnl.gov/casc/parasails/parasails.html>.
- [23] PARDISO, <http://www.pardiso-project.org/>.
- [24] Saad, Y. and Schultz, M. H., GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.*, **7**(1986), 856–869.
- [25] Saad, Y., ILUT : A dual threshold incomplete ILU factorization, *Num. Lin. Alg. Appl.*, **1**(1994), 387–402.
- [26] Saad, Y., *Iterative methods for sparse linear systems*, SIAM, Philadelphia (2003).



- [27] Sakurai, T. and Sugiura, H., A projection method for generalized eigenvalue problems using numerical integration, *J. Comput. Appl. Math.*, **159**(2003), 119–128.
- [28] SCOTCH, <http://www.labri.fr/perso/pelegrin/scotch/>.
- [29] Sorensen, D. C., Implicit application of polynomial filters in a  $k$ -step Arnoldi method, *SIAM J. Matrix Anal. Appl.* **13**(1992), 357–385.
- [30] SPAI, <http://www.computational.unibas.ch/software/spai/>.
- [31] The University of Florida Sparse Matrix Collection,  
<http://www.cise.ufl.edu/research/sparse/matrices/>.
- [32] 山崎 育朗, 今倉 暁, 多田野 寛人, 櫻井 鉄也, 残差最小性に基づく Krylov 部分空間反復解法に対する疎行列用直接解法を用いた前処理のパラメータ推定, *日本応用数学会論文誌*, **23**(2013), (採録予定).
- [33] Yamazaki, I., Okada, M., Tadano, H., Sakurai, T. and Teranishi, K., A block sparse approximate inverse with cutoff preconditioner for semi-sparse linear systems derived from Molecular Orbital calculations, *JSIAM Letters*, **2**(2010), 41–44.
- [34] 山崎 育朗, 岡田 真幸, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff を 2 重に用いた前処理の性能評価, *日本応用数学会論文誌*, **18**(2008), 631–651.
- [35] Yamazaki, I., Tadano, H., Sakurai, T. and Teranishi, K., A convergence improvement of the BSAIC preconditioner by deflation, *JSIAM Letters*, **3**(2011), 5–8.

## 謝辞

本研究を行うにあたり、常日頃から終始御指導いただきました筑波大学大学院システム情報工学研究科 櫻井鉄也 教授に心から感謝致します。櫻井先生の熱心な御指導、そして大変有益な御助言の恩恵で様々な経験を積むことができました。櫻井先生の下で研究を行えたことは掛け替えのない素晴らしいものとなりました。

研究を行う上で、最高の研究環境を整えて下さいました筑波大学大学院システム情報工学研究科 北川高嗣 教授に深く感謝致します。また、大変お忙しい中、本論文の副査を引き受けて頂き、誠にありがとうございました。

筑波大学大学院システム情報工学研究科 徳永隆治 教授、高橋大介 教授、亀山啓輔 准教授には大変お忙しい中本論文の審査を引き受けて頂きました。心より感謝致します。

終始お忙しい中、同分野の先達として多くの御助言を頂きました Cray, Inc. 寺西慶太 氏に厚く御礼申し上げます。

情報数理研究室の OB として多くの相談にのって頂いた筑波大学大学院システム情報工学研究科 HPCS 研究室 多田野寛人 助教に心より感謝致します。多田野先生には、色々ご迷惑をおかけしてしまいましたが、最後まで面倒を見て頂き誠にありがとうございました。

本研究を行う上で、多くのご助言とご協力を頂いた筑波大学大学院システム情報工学研究科 今倉暁 助教に深く御礼申し上げます。今倉先生にも、最後まで御助言や御指摘を頂きました。心より御礼申し上げます。

現日立製作所の 岡田真幸 氏には研究室に配属された時から、研究だけでなく私事も含め密度の濃い時間を過ごさせて頂きました。同氏とは、研究を共に行い、趣味を共有でき、本当に楽しい日々でした。

常日頃から過ごしやすい雰囲気を作って下さっている筑波大学情報数理研究室の皆様にも心より感謝致します。岡田龍太郎 氏とは研究の専門分野が異なるにも関わらず、多くの場面を通して、有意義な御助言を頂きました。心より感謝致します。また、同じ櫻井先生グループの後輩である 二村保徳 氏には、ゼミなどを介して研究に対するモチベーションを高めさせて頂きました。深く感謝致します。

最後に、9年間という長い期間、経済的・精神的に支えてくださった両親に感謝致します。本当にありがとうございました。現在は無事就職し、一社会人として頑張っております。独

り立ちした私にどうぞ御期待ください。

# 研究業績一覧

## 査読付き論文

1. 山崎 育朗, 今倉 暁, 多田野 寛人, 櫻井 鉄也, 残差最小性に基づく Krylov 部分空間反復解法に対する疎行列用直接解法を用いた前処理のパラメータ推定, 日本応用数学会論文誌, **23**(2013), (採録予定).
2. Yamazaki, I., Okada, M., Tadano, H., Sakurai, T. and Teranishi, K., A block sparse approximate inverse with cutoff preconditioner for semi-sparse linear systems derived from Molecular Orbital calculations, *JSIAM Letters*, **2**(2010), 41–44.
3. 山崎 育朗, 岡田 真幸, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff を 2 重に用いた前処理の性能評価, 日本応用数学会論文誌, **18**(2008), 631–651.
4. Yamazaki, I., Tadano, H., Sakurai, T. and Teranishi, K., A convergence improvement of the BSAIC preconditioner by deflation, *JSIAM Letters*, **3**(2011), 5–8.

## 査読無し国際会議

1. Yamazaki, I., Okada, M., Tadano, H., Sakurai, T. and Teranishi, K., A Block Approximate Inverse With Cutoff Preconditioner For Semi-Sparse Linear Systems, International Conference On Preconditioning Techniques For Scientific And Industrial Applications (Preconditioning 2009), Hong Kong, Aug. 2009.
2. Yamazaki, I., Tadano, H., Sakurai, T. and Teranishi, K., A performance evaluation of a block sparse approximate inverse with cutoff (BSAIC) preconditioner for semi-sparse linear systems, In 6th International Workshop on Parallel Matrix Algorithms and Applications (PMAA '10), Switzerland, Jul. 2010. (Poster session)

## 全国大会, 研究会における発表

1. 山崎 育朗, 岡田 真幸, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff を 2 重に用いた前処

理の性能評価, 平成 20 年日本応用数理学会研究部会連合発表会, 首都大学東京 南大沢キャンパス, 2008 年 3 月.

2. 山崎 育朗, 岡田 真幸, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Semi-Sparse 行列に対する Cutoff を 2 重に用いた前処理, 日本応用数理学会 2008 年度年会, 東京大学 柏キャンパス, 2008 年 9 月.
3. 山崎 育朗, 岡田 真幸, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff 付きブロック近似逆行列前処理の提案, 平成 21 年日本応用数理学会研究部会連合発表会, 京都大学 理学部, 2009 年 3 月.
4. 山崎 育朗, 岡田 真幸, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff 付きブロック近似逆行列前処理の提案, 第 6 回計算数学研究会, ウェルハートピア熱海, 2009 年 3 月.
5. 山崎 育朗, 岡田 真幸, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff 付きブロック近似逆行列前処理の性能評価, 日本応用数理学会 2009 年度年会, 大阪大学 豊中キャンパス, 2009 年 9 月.
6. 山崎 育朗, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff 付きブロック近似逆行列前処理の性能評価, 日本応用数理学会「行列・固有値問題の解法とその応用」研究部会 第 8 回研究会, 国立情報学研究所, 2009 年 11 月.
7. 山崎 育朗, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff 付きブロック近似逆行列前処理の性能評価, 特異値・固有値合同ワークショップ, つくば国際会議場, 2009 年 11 月.
8. 山崎 育朗, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff 付きブロック近似逆行列前処理における収束性改善, 2010 年度数値解析研究集会, 国立信州高遠青少年自然の家, 2010 年 8 月. (ポスターセッション)
9. 山崎 育朗, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, Cutoff 付きブロック近似逆行列前処理における収束性改善, 日本応用数理学会 2010 年度年会, 明治大学 駿河台キャンパス, 2010 年 9 月.
10. 山崎 育朗, 多田野 寛人, 櫻井 鉄也, 寺西 慶太, 準疎行列に対する近似逆行列前処理の性能評価, 特異値・固有値合同ワークショップ, 筑波大学 総合研究棟 B, 2010 年 11 月.
11. 山崎 育朗, 多田野 寛人, 櫻井 鉄也, Cutoff を組み合わせた前処理について, 日本応用数理学会 2011 年度年会, 同志社大学 今出川キャンパス, 2011 年 9 月.
12. 山崎 育朗, 多田野 寛人, 櫻井 鉄也, Cutoff を用いた前処理のパラメータ選択, 並列固有値解析研究会, 伊香保温泉旅館 よろこびの宿 しん喜, 2011 年 11 月.

## 受賞

1. 筑波大学 2010 年コンピュータサイエンス専攻長表彰