

## 連立一次方程式に対する GMRES( $m$ ) 法の収束性改善

### 概要

大規模連立一次方程式に対する求解アルゴリズムとして、リスタート付き GMRES 法（以下、GMRES( $m$ ) 法と呼ぶ）が知られている。GMRES( $m$ ) 法は残差ノルムが単調に減少するという性質から多くの数値シミュレーションで用いられているものの、リスタートの影響から収束性が遅いという問題点を持つ。本報告では、GMRES( $m$ ) 法の収束性を改善する改良法として近年提案された Look-Back GMRES( $m$ ) 法のアルゴリズムの概略を紹介する。

### キーワード

大規模連立一次方程式, GMRES( $m$ ) 法, リスタート

### 1. 序

大規模連立一次方程式

$$Ax = b, \quad A \in \mathbb{C}^{n \times n}, \quad x, b \in \mathbb{C}^n \quad (1)$$

の求解は数値シミュレーションの計算時間の大部分を占めることもあり、その求解アルゴリズムは計算科学における最も基盤的なアルゴリズムの一つである。近年では、問題サイズの大規模化に伴って反復法の一つである Krylov 部分空間法が有効な解法として広く用いられている。

本報告では、特にリスタート付き GMRES 法 [5]（以下、GMRES( $m$ ) 法と呼ぶ）に注目する。GMRES( $m$ ) 法は残差ノルムが単調に減少するという性質から多くの数値シミュレーションで用いられているものの、リスタートの影響から収束性が遅いという問題点を持つ。本報告では、GMRES( $m$ ) 法の収束性を改善する改良法として近年提案された Look-Back GMRES( $m$ ) 法 [3] のアルゴリズムの概略を紹介する。

以下、第 2 節では Krylov 部分空間法の基本的なアイデアを示し、GMRES( $m$ ) 法のアルゴリズムの概略を記す。第 3 節では、GMRES( $m$ ) 法の収束性を改善する Look-Back GMRES( $m$ ) 法のアルゴリズムの概略を記し、第 4 節において数値実験を通しその有効性を検証する。最後に第 5 節で本報告のまとめを行う。

---

**Algorithm 1** The GMRES method

---

- 1: Choose the initial guess  $x_0$  and compute  $r_0 = b - Ax_0$
  - 2: Set  $\beta = \|r_0\|_2, v_1 = r_0/\beta$
  - 3: **For**  $j = 1, 2, \dots, k$ , **Do**:
  - 4:   Compute  $w_j = Av_j$
  - 5:   **For**  $i = 1, 2, \dots, j$ , **Do**:
  - 6:      $h_{i,j} = (w_j, v_i)$
  - 7:      $w_j = w_j - h_{i,j}v_i$
  - 8:   **End For**
  - 9:    $h_{j+1,j} = \|w_j\|_2$ . **If**  $h_{j+1,j} = 0$  **then** Set  $k := j$  and go to 12
  - 10:    $v_{j+1} = w_j/h_{j+1,j}$
  - 11: **End For**
  - 12: Define the  $(k+1) \times k$  Hessenberg matrix  $\tilde{H}_k = \{h_{i,j}\}_{1 \leq i \leq k+1, 1 \leq j \leq k}$
  - 13:  $x_k = x_0 + V_k s_k$ , where  $s_k = \arg \min_{s \in \mathbb{C}^k} \|\beta e_1 - \tilde{H}_k s\|_2$
- 

## 2 . GMRES( $m$ ) 法

連立一次方程式 (1) に対する Krylov 部分空間法は, 係数行列  $A$  および初期近似解  $x_0 \in \mathbb{C}^n$  に対応する初期残差  $r_0 := b - Ax_0$  で張られる Krylov 部分空間に基づく射影法の一つとして定義される. これまでに, 係数行列の性質, Krylov 部分空間の基底ベクトルの生成アルゴリズムおよび近似解を設定するための条件等によって様々は解法が提案されている. 詳細は [4, 6] 等を参照されたい.

非エルミート連立一次方程式に対する GMRES 法は Arnoldi 原理に基づき Krylov 部分空間の正規直交基底を生成し, 最小残差条件に基づき近似解を構築する (Algorithm 1 参照). GMRES 法は最小残差条件に基づく優れた収束性を持つ反面, Arnoldi 原理に基づく長い漸化式を持つため, 反復回数の増加に伴って 1 反復あたりの演算量および記憶容量が増大するという問題点を持つ. このため, 実用上の観点から, 所定の  $m$  反復毎に反復を停止し, 得られた近似解を初期近似解として再び GMRES 法を適用する「リスタート」が用いられる.

ここで, GMRES( $m$ ) 法のリスタートは以下の 3 つの部分に分けて記述することができる.

- Part 1. リスタート周期  $m$  および第 1 リスタートサイクルの初期近似解  $x_0^{(1)}$  を設定する.
- Part 2. 第  $l$  リスタートサイクルにおいて, 方程式  $Ax = b$  を初期近似解を  $x_0^{(l)}$  とした  $m$  反復の GMRES 法で (近似的に) 解き, 近似解  $x_m^{(l)}$  を得る.
- Part 3. 得られた近似解  $x_m^{(l)}$  を, 第  $(l+1)$  リスタートサイクルの初期近似解  $x_0^{(l+1)}$  として更新する,  $x_0^{(l+1)} := x_m^{(l)}$ .

上記 Part 1–3 に基づき簡略化した GMRES( $m$ ) 法のアルゴリズムを Algorithm 2 に示す.

---

**Algorithm 2** The GMRES( $m$ ) method

---

- 1: Choose the restart frequency  $m$  and the initial guess  $\mathbf{x}_0^{(1)}$
  - 2: **For**  $l = 1, 2, \dots$ , until convergence **Do**:
  - 3: Solve (approximately)  $A\mathbf{x} = \mathbf{b}$  by  $m$  iterations of GMRES with the initial guess  $\mathbf{x}_0^{(l)}$ , and get the approximate solution  $\mathbf{x}_m^{(l)}$
  - 4: Update the initial guess  $\mathbf{x}_0^{(l+1)} := \mathbf{x}_m^{(l)}$
  - 5: **End For**
- 

### 3 . Look-Back GMRES( $m$ ) 法

リスタートは, 1 反復あたりの演算量および記憶容量が増大するという GMRES 法の問題点を抑制するものの, 一般にその収束性を悪化させる. このため, これまでに上記の Part 1 および Part 2 に着目した様々な改良法が提案されてきた. 例えば, 近似固有ベクトルを利用するデフレーションは Part 2 を, またリスタート周期の動的な決定に関する技法は Part 1 を対象とした改良法である.

一方近年, これまで改良の対象と認識されてこなかった Part 3 に着目した GMRES( $m$ ) 法の拡張法が提案された [2]. この拡張法の基本的アイデアは, GMRES( $m$ ) 法 (および従来の改良法) が初期近似解  $\mathbf{x}_0^{(l+1)}$  を

$$\mathbf{x}_0^{(l+1)} := \mathbf{x}_m^{(l)}$$

のように近似解  $\mathbf{x}_m^{(l)}$  を単純に引き継ぐのに対し, ある任意のベクトル  $\mathbf{y}^{(l+1)} \in \mathbb{C}^n$  を用い,

$$\mathbf{x}_0^{(l+1)} := \mathbf{x}_m^{(l)} + \mathbf{y}^{(l+1)}$$

のように設定するというものである. この拡張法は誤差方程式および反復改良法の観点から自然な拡張であることが示されている [2].

また, 拡張した GMRES( $m$ ) 法の残差多項式に対する解析を通し, GMRES( $m$ ) 法の過去の残差多項式を逐次再構築する Look-Back 戦略に基づき, Look-Back GMRES( $m$ ) 法が提案された [3]. アルゴリズムの導出等の詳細は割愛するが, Look-Back GMRES( $m$ ) 法においてベクトル  $\mathbf{y}^{(l+1)}$  は

$$\mathbf{y}^{(l+1)} := \mu^{(l)} \Delta \mathbf{x}^{(l)}, \quad \Delta \mathbf{x}^{(l)} := \begin{cases} \mathbf{x}_m^{(l)} - \mathbf{x}_m^{(l-\frac{k}{2})} & (k : \text{偶数}) \\ \mathbf{x}_m^{(l)} - \mathbf{x}_0^{(l-\frac{k-1}{2})} & (k : \text{奇数}) \end{cases}$$

と設定される. ここで,  $k$  は整数パラメータ ( $k \geq 2, k \in \mathbb{N}$ ) であり,  $\mu^{(l)} \in \mathbb{C}$  は

$$\mu^{(l)} = \arg \min_{\mu \in \mathbb{C}} \|\mathbf{r}_m^{(l)} - \mu A \Delta \mathbf{x}^{(l)}\|_2 = \frac{(\mathbf{r}_m^{(l)}, A \Delta \mathbf{x}^{(l)})}{(A \Delta \mathbf{x}^{(l)}, A \Delta \mathbf{x}^{(l)})}$$

である. ここで, Look-Back GMRES( $m$ ) 法は GMRES( $m$ ) 法と同様に残差ノルムの単調減少性が保証される. Look-Back GMRES( $m$ ) 法のアルゴリズムを Algorithm 3 に示す.

また, GMRES( $m$ ) 法および Look-Back GMRES( $m$ ) 法の 1 リスタートサイクルあたりの演算量および記憶容量の主要部を Table 1 に示す. ここで, Mat-Vec, AXPY および Inner-Product は, それ

---

**Algorithm 3** A Look-Back GMRES( $m$ ) method
 

---

- 1: Choose the restart frequency  $m$ , the parameter  $k \geq 2$  and the initial guess  $\mathbf{x}_0^{(1)}$
  - 2: **For**  $l = 1, 2, \dots$ , until convergence **Do**:
  - 3: Solve (approximately)  $A\mathbf{x} = \mathbf{b}$  by  $m$  iterations of GMRES with the initial guess  $\mathbf{x}_0^{(l)}$ , and get the approximate solution  $\mathbf{x}_m^{(l)}$
  - 4: Compute the vector  $\mathbf{y}^{(l+1)}$  as follows:
    - If**  $l = 1$  **then**  $\mathbf{y}^{(l+1)} = \mathbf{0}$
    - If**  $l \geq 2$  **then**
      - If**  $(l = k = 2)$  **or**  $(k : \text{even}, l \leq \frac{k}{2})$  **or**  $(k : \text{odd}, l \leq \frac{k-1}{2})$  **then**

$$\Delta\mathbf{x}^{(l)} := \mathbf{x}_m^{(l)} - \mathbf{x}_0^{(1)}$$
      - Else**

$$\Delta\mathbf{x}^{(l)} := \begin{cases} \mathbf{x}_m^{(l)} - \mathbf{x}_m^{(l-\frac{k}{2})} & (k : \text{even}) \\ \mathbf{x}_m^{(l)} - \mathbf{x}_0^{(l-\frac{k-1}{2})} & (k : \text{odd}) \end{cases}$$
      - End If**
      - $\mathbf{y}^{(l+1)} = \mu^{(l)}\Delta\mathbf{x}^{(l)}, \mu^{(l)} = \arg \min_{\mu \in \mathbb{C}} \|\mathbf{r}_m^{(l)} - \mu A\Delta\mathbf{x}^{(l)}\|_2$
      - End If**
  - 5: Update the initial guess  $\mathbf{x}_0^{(l+1)} := \mathbf{x}_m^{(l)} + \mathbf{y}^{(l+1)}$
  - 6: **End For**
- 

Table 1 The number of operations per restart cycle and storage requirements of GMRES( $m$ ) and Look-Back GMRES( $m$ ).

Method	GMRES( $m$ )	Look-Back GMRES( $m$ )	
		$k : \text{even}$	$k : \text{odd}$
Mat-Vec	$m + 1$	$(m + 1) + \mathbf{1}$	
AXPY	$(m^2 + 5m + 4)/2$	$(m^2 + 5m + 4)/2 + \mathbf{3}$	
Inner-Product	$(m^2 + 3m + 2)/2$	$(m^2 + 3m + 2)/2 + \mathbf{2}$	
Storage	$(m + 2)n$	$(m + 2)n + \frac{k}{2}\mathbf{n}$	$(m + 2)n + \frac{k+1}{2}\mathbf{n}$

ぞれ 1 リスタートサイクル当たりの行列ベクトル積, スカラー倍のベクトルの和算および内積計算の回数を意味する. また, Strage は係数行列を除く記憶容量の主要部を示す. ただし, これらの計算量および記憶容量は著者らの実装法に由来する. GMRES( $m$ ) 法のリスタートサイクル当たりの行列ベクトル積は  $m + 1$  から  $m$  に削減可能であり [5], また同様の方法で Look-Back GMRES( $m$ ) 法は  $m + 2$  から  $m$  に削減可能である. しかしながら, これらの実装法は丸め誤差の影響を増大させる可能性がある点に注意する必要がある.

#### 4 . 数値実験・結果

本節では, The University of Florida Sparse Matrix Collection [1] に掲載されている 6 個の問題に対し, GMRES( $m$ ) 法および Look-Back GMRES( $m$ ) 法 (Algorithm 3) を適用し, その性能を比較する. 以下の数値実験は AMD Phenom II X4 940 (3.0GHz) において FORTRAN 77 コンパイラの倍精度演算で行われた.

Table 2 Characteristics of the coefficient matrices of the test problems for GMRES( $m$ ) and Look-Back GMRES( $m$ ).

Matrix (Type)	$n$	$Nnz$	$Ave.Nnz$	Application area
CAVITY10 (R)	2597	76367	29.41	Computational fluid dynamics
KIM1 (C)	38415	933195	24.29	2D/3D problem
LIGHT_IN_TISSUE (C)	29282	406084	13.87	Electromagnetics problem
RAJAT03 (R)	7602	32653	4.30	Circuit simulation
WAVEGUIDE3D (C)	21036	303468	14.43	Electromagnetics problem
XENON2 (R)	157464	3866688	24.56	Materials problem

テスト問題の特徴を Table 2 に示す. ここで, (R) および (C) はそれぞれ, 実非対称行列, 複素非エルミート行列を示す. また,  $n, Nnz, Ave.Nnz$  はそれぞれ, 次元数, 非零要素数および 1 列 (1 行) 当たりの平均非零要素数を示す.

リスタート周期を  $m = 30$ , Look-Back GMRES( $m$ ) 法のパラメータを  $k = 3$  とする. また, 右辺ベクトルを  $\mathbf{b} = [1, 1, \dots, 1]^T$  および初期近似解を  $\mathbf{x}_0^{(1)} = [0, 0, \dots, 0]^T$  とし, 収束判定条件を  $\|\mathbf{r}_k\|_2 / \|\mathbf{b}\|_2 \leq 10^{-10}$  と設定する.

#### [実験結果]

実験結果を Table 3 に, 収束履歴を Fig. 1 にそれぞれ示す. ここで, Table の † は解法が最大反復回数 (100000 反復) で収束しなかったことを意味する. 以下では, 収束性, 1 リスタートサイクル ( $m$  反復) 当たりの計算時間および合計計算時間の 3 つの観点から実験結果の解析を行う.

まず, 両解法の収束性について議論する. 反復回数 (Iter) の観点では, Look-Back GMRES( $m$ ) 法は, GMRES( $m$ ) 法に比べ, 多くの問題において同程度もしくは優れた収束性を示すことが分かる. 特に, CAVITY10 および RAJAT03 に対しては, GMRES( $m$ ) 法は最大反復回数では収束しなかったのに対し, Look-Back GMRES( $m$ ) 法は要求精度を満たす近似解を効率的に計算することができた (Table 3 の TRR 参照). この結果から, Look-Back GMRES( $m$ ) 法は GMRES( $m$ ) 法に比べ優れた収束性を持つことが示された.

また, Fig. 1 から Look-Back GMRES( $m$ ) 法の残差ノルムは GMRES( $m$ ) 法と同様に単調に減少していることが分かる. KIM1 および WAVEGUIDE3D に対しては Look-Back GMRES( $m$ ) 法は GMRES( $m$ ) 法と同程度の収束性を示すものの, CAVITY10, LIGHT\_IN\_TISSUE, RAJAT03 および

Table 3 Convergence results (Iter : number of iterations,  $t_{\text{Total}}$  : total computation time,  $t_{\text{Restart}}$  : computation time per restart cycle, TRR :  $\log_{10}$  of explicitly computed relative residual 2-norm) of GMRES( $m$ ) and Look-Back GMRES( $m$ ) for  $m = 30$ .

Matrix	Method	Iter	Time[sec.]		TRR
			$t_{\text{Total}}$	$t_{\text{Restart}}$	
CAVITY10	GMRES( $m$ )	†	†	$8.48 \times 10^{-3}$	-7.03
	LB-GMRES( $m$ )	24811	$7.28 \times 10^0$	$8.80 \times 10^{-3}$	-10.00
KIM1	GMRES( $m$ )	2824	$3.72 \times 10^1$	$3.96 \times 10^{-1}$	-10.00
	LB-GMRES( $m$ )	3971	$5.35 \times 10^1$	$4.05 \times 10^{-1}$	-10.00
LIGHT_IN_TISSUE	GMRES( $m$ )	2964	$2.60 \times 10^1$	$2.62 \times 10^{-1}$	-10.00
	LB-GMRES( $m$ )	938	$8.29 \times 10^0$	$2.66 \times 10^{-1}$	-10.00
RAJAT03	GMRES( $m$ )	†	†	$1.43 \times 10^{-2}$	-0.55
	LB-GMRES( $m$ )	14881	$8.24 \times 10^0$	$1.65 \times 10^{-2}$	-10.01
WAVEGUIDE3D	GMRES( $m$ )	35268	$2.36 \times 10^2$	$2.03 \times 10^{-1}$	-10.00
	LB-GMRES( $m$ )	29745	$2.02 \times 10^2$	$2.03 \times 10^{-1}$	-10.00
XENON2	GMRES( $m$ )	15677	$3.92 \times 10^2$	$7.50 \times 10^{-1}$	-10.00
	LB-GMRES( $m$ )	2371	$6.05 \times 10^1$	$7.65 \times 10^{-1}$	-10.00

XENON2 に対しては, Look-Back GMRES( $m$ ) 法は反復全体を通して極めて高い収束性を示しており, Look-Back 戦略が有効に働くことを示している.

次に, 1 リスタートサイクル ( $m$  反復) 当たりの計算時間 ( $t_{\text{Restart}}$ ) について議論する. Look-Back GMRES( $m$ ) 法は GMRES( $m$ ) 法に比べ, 1 リスタートサイクル当たりの計算時間が最大で 10% 程度増加している. これは, 著者らの実装が Look-Back 戦略に基づき 1 リスタートサイクル当たりに 1 回の行列ベクトル積および数回の AXPY と内積計算が増加するという事実に基づく (Table 1 参照).

最後に合計計算時間 ( $t_{\text{Total}}$ ) について考察する. Look-Back GMRES( $m$ ) 法は, GMRES( $m$ ) 法に比べ, 優れた収束性 (Iter) を持つこと, また, 1 リスタートサイクル当たりの計算時間 ( $t_{\text{Restart}}$ ) がほぼ同程度であるという事実から, 結果として, KIM1 を除く問題に対し少ない計算時間で解を計算することができた.

## 5 . まとめ

本報告では, GMRES( $m$ ) 法の改良法である Look-Back GMRES( $m$ ) 法のアルゴリズムの概略を紹介し, 数値実験によりその有効性を検証した. 数値実験から, Look-Back GMRES( $m$ ) 法は GMRES( $m$ ) 法と 1 反復あたりの演算量および記憶容量が同程度でありながら, 同等以上の収束性

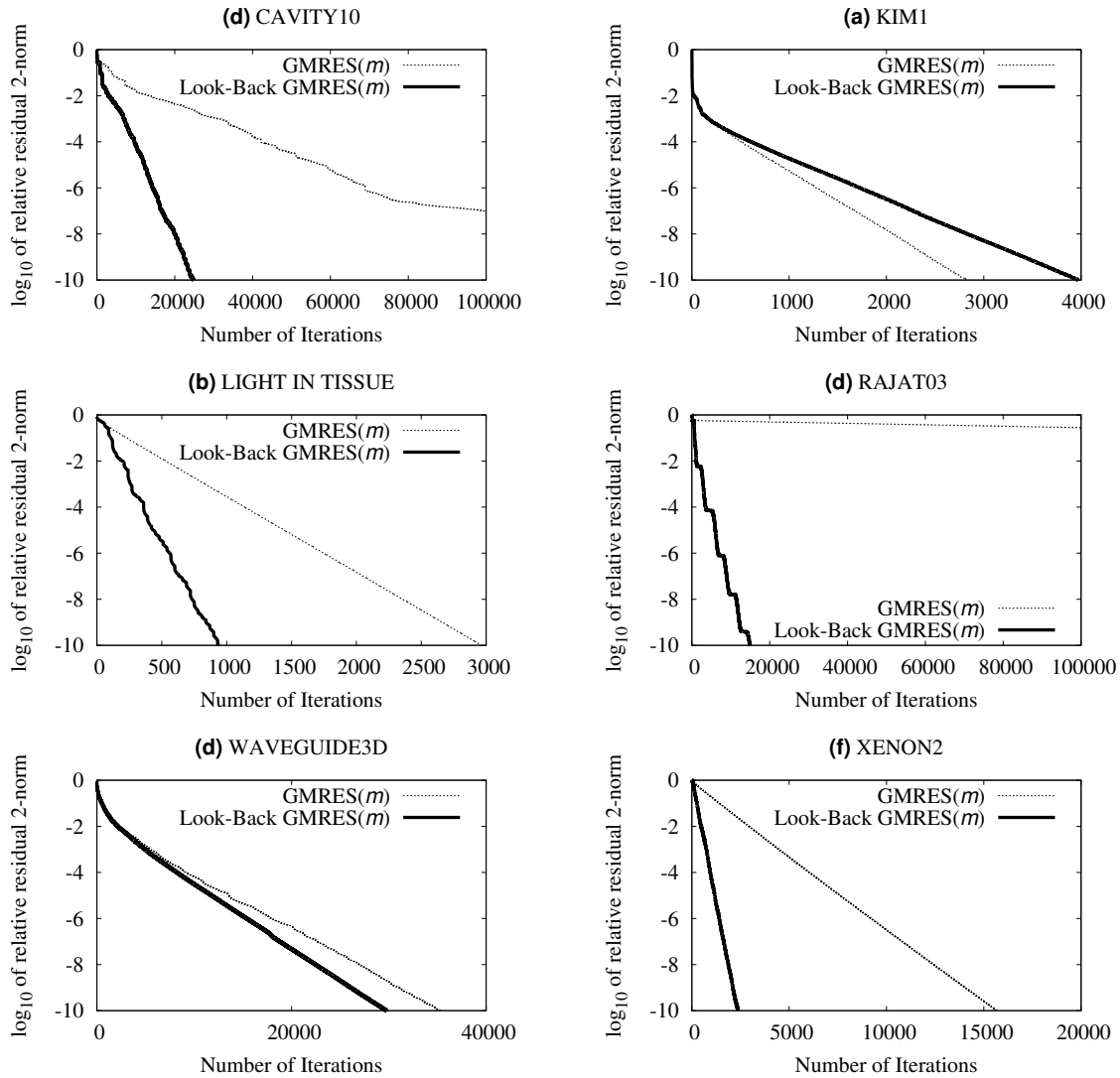


Fig.1 The relative residual 2-norm history of GMRES( $m$ ) and Look-Back GMRES( $m$ ) of  $m = 30$  without preconditioners for KIM1, LIGHT\_IN\_TISSUE, NS3DA, RAJAT03, RDB5000 and XENON2.

を示すことが明らかとなった。また、Look-Back GMRES( $m$ ) 法の特徴として、

- GMRES( $m$ ) 法に対する従来の改良法とは着眼点異なるため、他の改良法と競合せず併用することが可能である。
- プログラムの変更は GMRES( $m$ ) 法のリスタートサイクル毎の初期近似解の設定部分のみであり、実装が比較的容易である。
- 計算の主要部は GMRES( $m$ ) 法と同様に行列ベクトル積、AXPY および内積計算であり、GMRES( $m$ ) 法と比較して並列化効率が大きく減少しないことが予想される。

等が挙げられる。

## 参考文献

- [1] Davis, T. A., The University of Florida Sparse Matrix Collection,  
<http://www.cise.ufl.edu/research/sparse/matrices/>.
- [2] 今倉暁, 曾我部知広, 張紹良, GMRES( $m$ ) 法のリスタートについて, 日本応用数学会論文誌, **19**(2009), 551–564.
- [3] 今倉暁, 曾我部知広, 張紹良, Look-Back GMRES( $m$ ) 法の収束の振る舞いについて, 日本応用数学会 2010 年度年会 講演予稿集, (2010), 25–26.
- [4] Saad, Y., Iterative methods for sparse linear systems. 2nd edition, SIAM, Philadelphia, PA, 2003.
- [5] Saad, Y. and Schultz, M. H., GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems, SIAM J. Sci. Stat. Comput., **7**(1986), 856–869.
- [6] 杉原正顯, 室田一雄, 線形計算の数理, 岩波書店, 東京, 2009.