Norm bounds on the complimentary error matrix function

Shinya Miyajima[1], Amir Sadeghi[2]
[1]Iwate University, [2]Islamic Azad University
e-mail : miyajima@iwate-u.ac.jp

1 Introduction

Let $A \in \mathbb{C}^{n\times n}$ be a square matrix such that $|\mathrm{Re}(\lambda)| > |\mathrm{Im}(\lambda)|$, $\forall\lambda \in \sigma(A)$, where $\sigma(A)$ is the spectrum of A. The error matrix function $\mathrm{erf}(A)$ and complimentary error matrix function $\mathrm{erfc}(A)$, which are introduced in [1], are defined as

$$\mathrm{erf}(A) := \frac{2A}{\sqrt{\pi}}\int_0^1 e^{-(Av)^2}dv \qquad \text{and} \qquad \mathrm{erfc}(A) := \frac{2A}{\sqrt{\pi}}\int_1^\infty e^{-(Av)^2}dv,$$

respectively. Hermitian invertible matrices satisfy the condition $|\mathrm{Re}(\lambda)| > |\mathrm{Im}(\lambda)|$, $\forall\lambda \in \sigma(A)$ as well as any matrix similar to an Hermitian invertible matrix. Similarly to the scalar case, we have $\mathrm{erf}(A) + \mathrm{erfc}(A) = I$, where I is the $n \times n$ identity matrix.

One of the most important application of the complimentary error matrix function is the solution to systems of partial differential equation.

Theorem 1 (Cortés, Company, Jódar and Ponsoda [1, Theorem 5.1])**.** *Let $u_0, u(x,t) \in \mathbb{C}^n$ and $A \in \mathbb{C}^{n\times n}$. If $\mathrm{Re}(\lambda) > |\mathrm{Im}(\lambda)|$, $\forall\lambda \in \sigma(A)$, then the solution $u(x,t)$ to the semi-finite coupled diffusion problem*

$$\begin{array}{llll} u_t = A^2u_{xx}, & x>0, & t>0, & u(x,0)=0, \quad x>0, \\ u(0,t)=u_0, & t>0, & & u(x,t)\to 0, \quad \text{as } x\to\infty, \quad t>0 \end{array}$$

can be represented by

$$u(x,t) = \begin{cases} u_0, & x=0, \quad t\geq 0, \\ \mathrm{erfc}\left(\dfrac{A^{-1}x}{2\sqrt{t}}\right)u_0, & x>0, \quad t>0, \\ 0, & x>0, \quad t=0. \end{cases}$$

According to the Taylor expansion of $e^{-(Av)^2}$ where $v \in [0,1]$, and integrating term by term, we obtain

$$\mathrm{erf}(A) = \frac{2A}{\sqrt{\pi}}\sum_{k=0}^{\infty}\frac{(-1)^k A^{2k}}{k!(2k+1)},$$

which is the Taylor expansion of $\mathrm{erf}(A)$ [1]. From this expansion and $\mathrm{erf}(A) + \mathrm{erfc}(A) = I$, we obtain

$$\mathrm{erfc}(A) = I - \frac{2A}{\sqrt{\pi}}\sum_{k=0}^{\infty}\frac{(-1)^k A^{2k}}{k!(2k+1)}.$$

Let $\|A\|$ be the 2-norm of A. The following norm bound have been given in [1]:

Theorem 2 (Cortés, Company, Jódar and Ponsoda [1, Theorem 2.1])**.** *Let $\zeta(A) := \min\{\mathrm{Re}(\lambda) : \lambda \in \sigma(A)\}$ for $A \in \mathbb{C}^{n\times n}$. If $\mathrm{Re}(\lambda) > |\mathrm{Im}(\lambda)|$, $\forall\lambda \in \sigma(A)$, then*

$$\|\mathrm{erfc}(A)\| \leq \frac{\|A\|}{\sqrt{\zeta(A^2)}}\mathrm{erfc}(\sqrt{\zeta(A^2)}).$$

The purpose of this talk is to present the following norm bounds:

Norm bound 1 a new upper bound on $\|\mathrm{erfc}(A)\|$ under a condition which is different from $\mathrm{Re}(\lambda) > |\mathrm{Im}(\lambda)|$, $\forall \lambda \in \sigma(A)$,

Norm bound 2 upper bounds on $\|\mathrm{erf}(A) - \mathrm{erf}(B)\|$ and $\|\mathrm{erfc}(A) - \mathrm{erfc}(B)\|$, where $B \in \mathbb{C}^{n\times n}$, under an assumption, and

Norm bound 3 upper bounds on

$$\left\|\mathrm{erf}(A) - \frac{2A}{\sqrt{\pi}}\sum_{k=0}^{m}\frac{(-1)^k A^{2k}}{k!(2k+1)}\right\| \qquad \text{and} \qquad \left\|\mathrm{erfc}(A) - \left(I - \frac{2A}{\sqrt{\pi}}\sum_{k=0}^{m}\frac{(-1)^k A^{2k}}{k!(2k+1)}\right)\right\|,$$

where m is a nonnegative integer, under an assumption.

2 Norm bounds

Let $\nu(A) := \min\{\lambda : \lambda \in \sigma((A + A^*)/2)\}$, where A^* is the conjugate transpose of A. We present Theorems 3, 4 and 5 which correspond to Norm bounds 1, 2 and 3, respectively.

Theorem 3. *If $\nu(A^2) > 0$, then we have the following estimation:*

$$\|\mathrm{erfc}(A)\| \le \frac{\|A\|}{\sqrt{\nu(A^2)}}\mathrm{erfc}(\sqrt{\nu(A^2)}).$$

Theorem 4. *Let $\omega := \min(\nu(A^2), \nu(B^2))$ for $A, B \in \mathbb{C}^{n\times n}$. If $\omega > 0$, it then follows that*

$$\|\mathrm{erf}(A) - \mathrm{erf}(B)\| \le \|A - B\|\left(\frac{\mathrm{erf}(\sqrt{\nu(A^2)})}{\sqrt{\nu(A^2)}} + \frac{\|B\|(\|A\| + \|B\|)}{\sqrt{\pi}\omega}\left(\frac{\sqrt{\pi}\mathrm{erf}(\sqrt{\omega})}{2\sqrt{\omega}} - e^{-\omega}\right)\right),$$

$$\|\mathrm{erfc}(A) - \mathrm{erfc}(B)\| \le \|A - B\|\left(\frac{1 - \mathrm{erfc}(\sqrt{\nu(A^2)})}{\sqrt{\nu(A^2)}} + \frac{\|B\|(\|A\| + \|B\|)}{\sqrt{\pi}\omega}\left(\frac{\sqrt{\pi}(1 - \mathrm{erfc}(\sqrt{\omega}))}{2\sqrt{\omega}} - e^{-\omega}\right)\right).$$

Theorem 5. *If $\nu(A^2) \ge 0$, then we have*

$$\left\|\mathrm{erf}(A) - \frac{2A}{\sqrt{\pi}}\sum_{k=0}^{m}\frac{(-1)^k A^{2k}}{k!(2k+1)}\right\| \le \frac{2\|A\|^{2m+3}}{\sqrt{\pi}(m+1)!(2m+3)},$$

$$\left\|\mathrm{erfc}(A) - \left(I - \frac{2A}{\sqrt{\pi}}\sum_{k=0}^{m}\frac{(-1)^k A^{2k}}{k!(2k+1)}\right)\right\| \le \frac{2\|A\|^{2m+3}}{\sqrt{\pi}(m+1)!(2m+3)}.$$

We will report results of numerical experiments at the talk in order to observe how much larger the presented bounds are compared to the corresponding norms.

Acknowledgments This work was partially supported by JSPS KAKENHI Grant Number JP21K03363.

References

[1] J. Cortés, R. Company, L. Jódar and E. Ponsoda, The complementary error matrix function and its role solving coupled diffusion mathematical models, Math. Comput. Modell, 42(9–10) (2005), 1023–1034.

前処理付き CGS 法に対する Deflation の適用

Application of Deflation to the Preconditioned CGS Method

高谷 周平 (Shuhei Takaya)[1]
[1] 個人 (Individual)
e-mail :　shuhei.takaya@gmail.com

1　概要

積型解法 [1] に前処理と deflation を同時に適用していくための端緒として，CGS 法 [2] を取り上げる．議論の単純さと一貫性を保つために，伊藤ら [3] による改善版前処理付き CGS 法のアプローチに倣い，前処理付き deflated BiCG 法 [4] に CGS 法の導出過程を適用する．標準的な前処理付き CGS 法 [5] に deflation を適用して得られるアルゴリズムとの比較も行う．

2　前処理付き Deflated CGS 法の導出

2.1　漸化式の係数

係数行列が $A \in \mathbb{R}^{n\times n}$ の n 次連立 1 次方程式を，$LR \approx A$ による両側前処理及び Y と $Z(\in \mathbb{R}^{n\times m}, m \ll n)$ による deflation の双方を適用した BiCG 法 [4] で解くと，k 回目の反復における漸化式の係数 ρ_k と μ_k は，前処理系の係数行列を $\mathfrak{A}$，残差多項式を $\phi_k(\mathfrak{A})$，方向多項式を $\pi_k(\mathfrak{A})$，初期残差を $\mathfrak{r}_0 = L^{-1}P\boldsymbol{r}_0$，初期シャドウ残差を $\mathfrak{r}^*_{\mathrm{BiCG},0} = R^{-T}P'\boldsymbol{r}^*_0$ として，

$$\rho_k = \left(\phi_k(\mathfrak{A}^T)\mathfrak{r}^*_{\mathrm{BiCG},0}, \phi_k(\mathfrak{A})\mathfrak{r}_0\right),\ \mu_k = \left(\pi_k(\mathfrak{A}^T)\mathfrak{r}^*_{\mathrm{BiCG},0}, \mathfrak{A}\pi_k(\mathfrak{A})\mathfrak{r}_0\right) \tag{1}$$

となる．ただし，deflation を適用する際に斜交射影 $P = I - AZ\left(Y^TAZ\right)^{-1}Y^T$ を A にかける場合 (DEF1) は $\mathfrak{A} = L^{-1}PAR^{-1}$，方向ベクトルに斜交射影 $P'^T = I - Z\left(Y^TAZ\right)^{-1}Y^TA$ をかける場合 (DEF2) は $\mathfrak{A} = L^{-1}AP'^TR^{-1}$ となる．

CGS 法では A^T を避けて $\phi_k^2(\mathfrak{A})\mathfrak{r}_0$ と $\pi_k^2(\mathfrak{A})\mathfrak{r}_0$ の漸化式を解き，式 (1) を以下のように変形する．

$$\rho_k = \left(\mathfrak{r}^*_{\mathrm{CGS},0}, \phi_k^2(\mathfrak{A})\mathfrak{r}_0\right),\ \mu_k = \left(\mathfrak{r}^*_{\mathrm{CGS},0}, \mathfrak{A}\pi_k^2(\mathfrak{A})\mathfrak{r}_0\right) \tag{2}$$

シャドウ残差 $\mathfrak{r}^*_{\mathrm{CGS},0}$ は，従来型前処理 [5] では $L^TP'\boldsymbol{r}^*_0$，伊藤らによる前処理 [3] では $R^{-T}P'\boldsymbol{r}^*_0$ となるが，$P'\boldsymbol{r}^*_0$ の計算には A^T が必要となってしまうので，伊藤らの前処理と DEF2 を組み合わせる場合を除いて，斜交射影 P' を正射影 $P_{\mathrm{orth}Z} = I - Z\left(Z^TZ\right)^{-1}Z^T$ に置き換える．

2.2　アルゴリズム

前処理と deflation の組み合わせにより $2\times 2 = 4$ 通りの前処理付き deflated CGS 法が得られる．従来型前処理 [5] による手法をアルゴリズム 1 に，伊藤ら [3] の前処理による手法をアルゴリズム 2 に示す．DEF1 と DEF2 を一括して記述するために表 1 の記号を使用する．

表 1. アルゴリズムの記述に使用する記号

	$\overline{\boldsymbol{x}}_0$	$\hat{P}'$	$\hat{A}$	$\hat{K}$	$\boldsymbol{x}_{\mathrm{approx}}$
DEF1	$\boldsymbol{x}_0$	$P_{\mathrm{orth}Z}$	PA	M^{-1}	$P'^T\overline{\boldsymbol{x}}_{\mathrm{conv}} + Q\boldsymbol{b}$
DEF2	$P'^T\boldsymbol{x}_0 + Q\boldsymbol{b}$	I	A	P'^TM^{-1}	$\overline{\boldsymbol{x}}_{\mathrm{conv}}$

Algorithm 1 従来型前処理 [5] による Deflated CGS 法 (DEF1,DEF2)

Calculate $\bar{\boldsymbol{x}}_0$
$\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0,\ \bar{\boldsymbol{r}}_0 = P\boldsymbol{r}_0$
$\boldsymbol{r}_0^* = \boldsymbol{r}_0,\ \bar{\boldsymbol{r}}_0^* = P_{\mathrm{orth}Z}\boldsymbol{r}_0^*$
$\boldsymbol{p}_0 = \boldsymbol{0},\ \boldsymbol{q}_0 = \boldsymbol{0},\ \rho_0 = 1.0$
for $k = 1, \ldots, n$, until convergence, **do**
  $\rho_k = (\bar{\boldsymbol{r}}_0^*, \bar{\boldsymbol{r}}_{k-1}),\ \beta_{k-1} = \rho_k/\rho_{k-1}$
  $\boldsymbol{u}_k = \bar{\boldsymbol{r}}_{k-1} + \beta_{k-1}\boldsymbol{q}_{k-1}$
  $\boldsymbol{p}_k = \boldsymbol{u}_k + \beta_{k-1}\left(\boldsymbol{q}_{k-1} + \beta_{k-1}\boldsymbol{p}_{k-1}\right)$
  $\mu_k = (\bar{\boldsymbol{r}}_0^*, \hat{A}\hat{K}\boldsymbol{p}_k),\ \alpha_k = \rho_k/\mu_k$
  $\boldsymbol{q}_k = \boldsymbol{u}_k - \alpha_k\hat{A}\hat{K}\boldsymbol{p}_k$
  $\bar{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_{k-1} + \alpha_k\hat{K}\boldsymbol{u}_k + \alpha_k\hat{K}\boldsymbol{q}_k$
  $\bar{\boldsymbol{r}}_k = \bar{\boldsymbol{r}}_{k-1} - \alpha_k\hat{A}\hat{K}\boldsymbol{u}_k - \alpha_k\hat{A}\hat{K}\boldsymbol{q}_k$
end for
Calculate $\boldsymbol{x}_{\mathrm{approx}}$

Algorithm 2 伊藤らの前処理 [3] による Deflated CGS 法 (DEF1,DEF2)

Calculate $\bar{\boldsymbol{x}}_0$
$\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0,\ \bar{\boldsymbol{r}}_0 = P\boldsymbol{r}_0$
$\boldsymbol{r}_0^* = \boldsymbol{r}_0,\ \bar{\boldsymbol{r}}_0^* = \hat{P}'\boldsymbol{r}_0^*$
$\boldsymbol{w}_0 = \boldsymbol{0},\ \boldsymbol{q}_0 = \boldsymbol{0},\ \rho_0 = 1.0$
for $k = 1, \ldots, n$, until convergence, **do**
  $\rho_k = (\bar{\boldsymbol{r}}_0^*, \hat{K}\bar{\boldsymbol{r}}_{k-1}),\ \beta_{k-1} = \rho_k/\rho_{k-1}$
  $\boldsymbol{u}_k = \hat{K}\bar{\boldsymbol{r}}_{k-1} + \beta_{k-1}\boldsymbol{q}_{k-1}$
  $\boldsymbol{w}_k = \hat{A}\boldsymbol{u}_k + \beta_{k-1}(\hat{A}\boldsymbol{q}_{k-1} + \beta_{k-1}\boldsymbol{w}_{k-1})$
  $\mu_k = (\bar{\boldsymbol{r}}_0^*, \hat{K}\boldsymbol{w}_k),\ \alpha_k = \rho_k/\mu_k$
  $\boldsymbol{q}_k = \boldsymbol{u}_k - \alpha_k\hat{K}\boldsymbol{w}_k$
  $\bar{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_{k-1} + \alpha_k\boldsymbol{u}_k + \alpha_k\boldsymbol{q}_k$
  $\bar{\boldsymbol{r}}_k = \bar{\boldsymbol{r}}_{k-1} - \alpha_k\hat{A}\boldsymbol{u}_k - \alpha_k\hat{A}\boldsymbol{q}_k$
end for
Calculate $\boldsymbol{x}_{\mathrm{approx}}$

3 数値実験

SuiteSparse Matrix Collection[6] で公開されている行列で基本的な収束性を，並列 FEM コード FrontISTR[7] から得られた行列で実行時間を検討した．これらの実験の詳細は講演で報告する．

参考文献

[1] 相原 研輔, 連立一次方程式に対する積型 Bi-CG 法の発展, 応用数理, 30 巻 3 号 (2020), pp. 16–23.

[2] P. Sonneveld, CGS, a fast Lanczos-type solver for nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing, Vol. 10 (1989), pp. 36–52.

[3] 伊藤 祥司, 杉原 正顯, 導出過程に着目した CGS 法の前処理付きアルゴリズム, 日本応用数理学会論文誌, 23 巻 2 号 (2013), pp. 253–286.

[4] 高谷 周平, 前処理つき Deflated BiCG 法及び BiCR 法の 5 つのバリアントの性能検証, 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 38 回研究会 (2024)

[5] H. A. van der Vorst, Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems, SIAM Journal on Scientific and Statistical Computing, Vol. 13(1992), pp. 631–644.

[6] The University of Florida Sparse Matrix Collection, `https://sparse.tamu.edu/`.

[7] 一般社団法人 FrontISTR Commons, `https://www.frontistr.org/`.

微分作用素の固有値問題に対する複素モーメントを用いた精度保証付き数値計算と Mathieu 方程式および Schrödinger 程式への応用

Verified numerical computation using complex moments for eigenvalue problems of differential operators and its applications to Mathieu and Schrödinger equations

今倉 暁 (Akira Imakura)[1], 保國 惠一 (Keiichi Morikuni)[1], 高安 亮紀 (Akitoshi Takayasu)[1]
[1] 筑波大学 (University of Tsukuba)
e-mail : morikuni.keiichi.fw@u.tsukuba.ac.jp

1 Introduction

An operator analogue of our matrix eigenvalue verification technique [3] is established in parallel with a numerical verification analogue of our operator eigensolver [4] to enclose eigenvalues of a linear, self-adjoint, and ordinary differential operator $\mathcal{A} : \mathrm{dom}(A) \to \mathcal{H}$ densely defined on a domain $\mathrm{dom}(A)$ in a Hilbert space $\mathcal{H}$. The eigenvalue problem considered is

$$\mathcal{A}u = \lambda u \text{ in } \Omega \subset \mathbb{R}, \quad u = 0 \text{ on } \partial\Omega \tag{1}$$

for eigenvalue λ's in an interval $I = [a, b] \subset \mathbb{R}$, where $u \in \mathrm{dom}(A) \setminus \{0\}$ is the eigenfunction. The spectrum of $\mathcal{A}$ is assumed to be discrete and bounded below. The problem (1) includes Mathieu's and Schrödinger's equations.

Our operator eigensolver [4] on (1) delays discretization until the last step. This avoids the effect of discretization on accuracy. The numerical verification analogue inherits this property.

2 Reduction to a matrix eigenvalue problem

Our operator eigensolver uses the complex moment $\mathsf{M}_k = V^{\mathsf{T}}\mathcal{M}_k V \in \mathbb{C}^{L\times L}$ of order k with

$$\mathcal{M}_k = \frac{1}{2\pi \mathrm{i}} \oint_{\Gamma} (z-\gamma)^k (z-\mathcal{A})^{-1} \mathrm{d}z, \quad k = 0, 1, \ldots, 2M-1,$$

where $V : \mathbb{R}^L \to \mathcal{H}$ is a quasimatrix, whose columns are functions in $L^\infty(\Omega)$, and Γ is the circle with center γ and radius ρ of which the interval I is interior. Cauchy's integral formula shows $\mathcal{M}_k = \sum_{i:\lambda_i\in I} (\lambda_i - \gamma)^k \mathcal{P}_i$ with the spectral projector $\mathcal{P}_i = \frac{1}{2\pi \mathrm{i}} \oint_{\Gamma_i} (z-\mathcal{A})^{-1} \mathrm{d}z$ onto the invariant subspace of $\mathcal{A}$ for an eigenvalue λ_i, where Γ_i is a Jordan curve in which λ_i lies.

The problem (1) and the block Hankel matrix pencil $zH_M - H_M^<$ with $H_M^< = (\mathsf{M}_{i+j-1})$, $H_M = (\mathsf{M}_{i+j-2}) \in \mathbb{C}^{LM\times LM}$ have the same eigenvalues in I, if $\mathrm{rank}(H_M^<) = \mathrm{rank}(H_M) = n_\Omega$, the number of eigenvalues of (1) in I [4]. Rather than enclosing $H_M, H_M^<$, we consider an alternative way in section 3.

In the approximation of $\mathsf{M}_k \simeq \mathsf{M}_k^{(N)} = \sum_{i=1}^{\infty} (\lambda_i - \gamma)^k d_i^{(N)} V^* \mathcal{P}_i V$ by using the N-point trapezoidal rule, the truncation error lies in $\mathsf{M}_{k,\mathrm{out}}^{(N)}$ of the splitting $\mathsf{M}_k^{(N)} = \mathsf{M}_{k,\mathrm{in}}^{(N)} + \mathsf{M}_{k,\mathrm{out}}^{(N)}$, where $\mathsf{M}_{k,\mathrm{in}}^{(N)}$ depends on the eigenvalues in I, $\mathsf{M}_{k,\mathrm{out}}^{(N)}$ depends on those outside I, $k = 0, 1, \ldots, 2M-1$, $d_i^{(N)} = \frac{1}{1-\left(\frac{\lambda_i-\gamma}{\rho}\right)^N}$ for $\lambda_i \in I$, and $d_i^{(N)} = \frac{-\left(\frac{\rho}{\lambda_i-\gamma}\right)^N}{1-\left(\frac{\rho}{\lambda_i-\gamma}\right)^N}$ for $\lambda_i \notin I$.

3 Enclosure of the complex moment

The block Hankel matrix pencils $zH_M - H_M^<$ and $zH_{M,\mathrm{in}}^{(N)} - H_{M,\mathrm{in}}^{<,(N)}$ with $H_{M,\mathrm{in}}^{<,(N)} = (\mathsf{M}_{i+k-1,\mathrm{in}}^{(N)})$, $H_{M,\mathrm{in}}^{(N)} = (\mathsf{M}_{i+j-2,\mathrm{in}}^{(N)}) \in \mathbb{C}^{LM\times LM}$ turn out to have the same eigenvalues if $\mathrm{rank}(H_M) = n_\Omega$. This motivates us to enclose $\mathsf{M}_{k,\mathrm{in}}^{(N)}$ as

$$\mathsf{M}_{k,\mathrm{in}}^{(N)} \in \langle \mathsf{M}_k^{(N)}, |\mathsf{M}_{k,\mathrm{out}}^{(N)}| \rangle \subset \langle \widetilde{\mathsf{M}}_k^{(N)}, |\mathsf{M}_{k,\mathrm{out}}^{(N)}| + |\widetilde{\mathsf{M}}_k^{(N)} - \mathsf{M}_k^{(N)}| \rangle, \quad k = 0, 1, \ldots, 2M-1,$$

where $\langle C, R\rangle$ is the interval matrix with radius $R \in \mathbb{R}_+^{L\times L}$ and center at $C \in \mathbb{R}^{L\times L}$, $|\cdot|$ is the nonnegative matrix of entrywise absolute values, and $\widetilde{\mathsf{M}}_k^{(N)}$ is a matrix obtained by numerically computing $\mathsf{M}_k^{(N)}$. Note that $|\widetilde{\mathsf{M}}_k^{(N)} - \mathsf{M}_k^{(N)}|$ is regarded as the rounding error.

To evaluate $|\mathsf{M}_{k,\mathrm{out}}^{(N)}|$, we use the fact that the eigenvalues of the Laplacian $-\Delta$ are known and regard the operator in (1) as a perturbed Laplacian $\mathcal{A} = -\Delta + \delta$ with perturbation $\delta \in L^\infty(\Omega)$. The effect of δ on the eigenvalues of $-\Delta$ is evaluated via the Krylov–Weinstein bound. This leads to evaluating $|\mathsf{M}_{k,\mathrm{out}}^{(N)}|$ by using the Gauss hypergeometric function (cf. [2]).

4 Practical realization

Rather than discretizing the equation (1), our approach adaptively approximates analytic functions to within machine precision using Chebyshev expansions [1], where the coefficients are computed via a discrete Chebyshev transform and stored in interval form [6]. The truncation error for the Chebyshev polynomial approximation of a function is computed by evaluating the function in the Bernstein ellipse. The rounding error caused in operations on Chebyshev polynomials is rigorously evaluated. The inner product between functions is computed by using the Clenshaw–Curtis quadrature rule, which has polynomial exactness. Our implementation of this method is realized by means of INTLAB [6], Chebfun [1], and kv [2]. Numerical experiments on Mathieu's and Schrödinger's equations illustrate the performance of the method.

Acknowledgement Supported by JSPS grant numbers JP21H03451, JP24K00535.

参考文献

[1] T. A. Driscoll, N. Hale, L. N. Trefethen, ed., Chebfun Guide, Pafnuty Publications, 2014.

[2] M. Kashiwagi, kv – a C++ Library for Verified Numerical Computation.

[3] A. Imakura, K. Morikuni, A. Takayasu, Verified partial eigenvalue computations using contour integrals for Hermitian generalized eigenproblems, J. Comput. Appl. Math., Vol. 369 (2020), 112543.

[4] A. Imakura, K. Morikuni, A. Takayasu, Complex moment-based methods for differential eigenvalue problems, Numer. Algorithms, Vol. 92 (2022), 693–721.

[5] A. Horning, A. Townsend, FEAST for differential eigenvalue problems, SIAM J. Sci. Comput., Vol. 58 (2020), 1239–1262.

[6] S. M. Rump, INTLAB - INTerval LABoratory, in Developments in Reliable Computing, Tibor Csendes, ed., Kluwer Academic Publishers, pp. 77–104, 1999.

縦長行列に対する列選択付きハウスホルダ型 QR 分解とその分散並列化

Distributed parallelization of Householder-QR factorization with column pivotings for tall and skinny matrix

村上 弘 (Hiroshi Murakami)[1]
[1] 東京都立大学 (Tokyo Metropolitan University)
e-mail : mrkmhrsh@tmu.ac.jp

1 概要

ハウスホルダ型 QR 分解法では行列 A に列の順に作成した鏡映を毎回適用して上三角行列 R を作り，鏡映を逆順に用いて列正規直交行列 Q を得て A=QR とする．それに対して毎回の鏡映を列交換を行ってノルム最大の列から決めると列置換を P として AP=QR となり，上三角行列 R の性質が良くなる [1]．A が極めて縦長なとき，列交換なしの QR 分解は TSQR 法で容易に分散並列化できるが [2]，列交換付きの場合についても同様にできることを示す [3]．さらに簡単な実験結果を示す．

■はじめに 鏡映変換を用いる行列 A の QR 分解は分解精度と基底の正規直交性が非常に良い．列選択なしの分解法では，列の順に鏡映変換を決めて後続の列に適用を繰り返して上三角行列 R を作り，次に鏡映変換を逆順に繰り返して列正規直交行列 Q を作り，分解 A=QR を得る．それに対してノルム最大の列を選んで鏡映変換を決める方法では，列交換を蓄積した列置換を P として分解 AP=QR を得る [1]．列選択と交換を行うので手間が増えるが，それと引き換えに上三角行列 R は対角要素の絶対値が単調減少し，対角要素が列内で絶対値が最大であるという良い性質を持つので，数値的に悪条件でランク落ちに近い A に対しても得られた分解の結果を後で利用する上で有利である．

行列 A が極めて縦長である場合には列選択なしの QR 分解の計算を階層的に行う TSQR 法と呼ばれる手法がある [2]．たとえば A を縦ブロック分割し，各小行列の QR 分解から得た上三角行列を縦に並べた行列に対して再び QR 分解を行うことで，計算の主要部分の処理を容易に分散並列化できる．そこで行列 A が極めて縦長である場合には，同様の階層的な手法が列選択付きの QR 分解についても可能で，容易に分散並列化ができることを示す（ただし今回は 2 階層の場合だけを扱う）[3]．

■T-S 行列用の列交換付きの QR 分解法 以下では 2 階層方式による T-S 行列 A に対する列選択付きの QR 分解法で，両方の階層で列選択を行う方法の今回の実験で用いた実装法について述べる．

M×N 行列 A を縦の寸法がほぼ均等な NBLK 個の小行列 $A^{(\ell)}$，ℓ=1, 2, . . . , NBLK に分割して考える．実験では $A^{(\ell)}$ の縦の寸法 BLKSZ(ℓ) は $\ell = 1$ のときは $\lfloor M/\mathtt{NBLK} \rfloor + \mathrm{mod}(M, \mathtt{NBLK})$ とし，$\ell \neq 1$ のときは $\lfloor M/\mathtt{NBLK} \rfloor$ と設定した．さらに N 次行列 $B^{(\ell)}$，ℓ=1, 2, . . . , NBLK を縦に順番に小行列として並べた (NBLK×N)×N 行列 B を用意する． **Step-1)** 第 1 階層では各 ℓ に対して，$A^{(\ell)}$ の列を列交換はせずに毎回選択してそれから決まる鏡映変換を繰り返して $H^{(\ell)}A^{(\ell)} = U^{(\ell)}$ とする．直交行列 $H^{(\ell)}$ は毎回の鏡映の情報の形式で陰的に保持する．M×N の行列 $U^{(\ell)}$ は先頭の N 個以外の行は零で，一般には上三角にはならない．そうして $U^{(\ell)}$ の先頭の N 個の行を N 次行列 $B^{(\ell)}$ の場所に格納する． **Step-2)** 第 2 階層では $B^{(\ell)}$ を縦に順番に NBLK 個並べた行列 B に対して列交換付きの QR 分解を行い $B\,P = Q'\,R$ とする．列置換 P の情報は要素数 N の整数配列に格納し，N 次上三角行列

R を配列に格納し，列正規直交行列 Q' は B 全体に上書きで格納する．そうして $B^{(\ell)}$ の場所に格納された Q' の ℓ 番目の N 次の小行列を $Q'^{(\ell)}$ とする． **Step-3)** 各 ℓ に対して，第 1 階層で既に作って保持しておいた毎回の鏡映変換の情報を逆順に用いて BLKSZ(ℓ)×N の一般化単位行列 $I'^{(\ell)}$ から $\widehat{Q}^{(\ell)} := (H^{(\ell)})^{-1} I'^{(\ell)}$ を作り，それを $A^{(\ell)}$ の場所に上書きする（注記：この $\widehat{Q}^{(\ell)}$ を作り $A^{(\ell)}$ の場所に上書きする処理は，第 2 階層での処理と並行して行うことも可能）．次に行列積 $Q^{(\ell)} := \widehat{Q}^{(\ell)} Q'^{(\ell)}$ を作って $A^{(\ell)}$ の場所に上書きする．あるいは $A^{(\ell)} := A^{(\ell)} B^{(\ell)}$ とも書ける．

以上の結果，元の M×N 行列 A に対する列選択付きの QR 分解の結果である $AP = QR$ が得られる，列置換 P の情報は第 2 階層での N 要素の整数配列に，行列 R の内容は第 2 階層での配列 R に，M×N の列正規直交行列 Q は元の行列 A の場所を上書きすることで，それぞれ格納されている．

■実験に用いた計算機システムの仕様 CPU は intel Core i7-5960X（8 コア，3.0GHz（T-B 時最大 3.5GHz））が 1 つであり，主記憶は DDR4-17000 の 16GB が 8 個で合計容量 128GB である．コンパイラには intel Fortran (ifort) v15.00 for Linux を使用して，コンパイルオプションは`-Ofast -qopenmp -D'MY_RKIND=8' -lmkl` とし，数学ライブラリには intel MKL v11.2 を用いた．OS は CentOS 7 for x86_64 である．OpenMP の最大スレッド数は CPU のコア数と同じ 8 に設定した．

■実験例 テスト用の M×N の行列 A の要素の式は $a_{i,j} = 1/(M+N+1-i-j)$, $1 \le i \le M$, $1 \le j \le N$ とした．「分解の残差」は残差行列 $AP-QR$ の要素の大きさの最大値で，「正規直交性誤差」は $c_{i,j}$ を Q の i 列と j 列の内積，$\delta_{i,j}$ をクロネッカー記号として $|c_{i,j}-\delta_{i,j}|$ の最大値である．数値と演算には IEEE754 の倍精度を用いた．2 階層法による T-S 行列用の列交換付き QR 分解法をテスト用の N=3,000,000，N=50 の行列に適用した結果の両対数グラフを示す（図 1，2，3）．ブロック分割数 NBLK が CPU のコア数 8 を越えても経過時間がさらに減少しているのは，小行列への処理が頻繁に参照する記憶の容量が CPU 内の 3 段階の各キャッシュの容量を占める割合の変化を反映している．

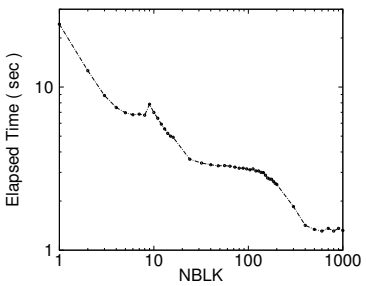

図 1. 経過時間 vs. NBLK

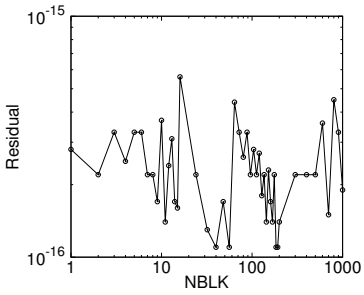

図 2. 分解の残差 vs. NBLK

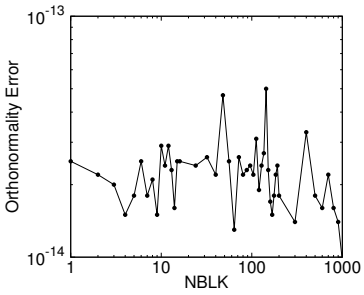

図 3. 正規直交性誤差 vs. NBLK

参考文献

[1] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., *The John Hopkins University Press*, Baltimore and London, 1996.

[2] E. Agullo, C. Coti, J. Dongarra, T. Herâult and J. Langem, "QR factorization of tall and skinny matrices in a grid computing environment", in: Proc. of *2010 IEEE International Symposium on Parallel Distributed Processing (IPDPS)*, pp.1–11, 2010.

[3] 村上弘, T-S 行列に対する列選択付きハウスホルダ型 QR 分解法の並列処理に向けた実装法について, 情報処理学会研究報告 *(HPC)*, Vol.2023-HPC-192, No.38 (2023), pp.1–21.