A Verification Method of Generalized Eigenvalue Problems and Its Applications

Yoshitaka Watanabe[†],
Nobito Yamamoto[‡] and Mitsuhiro T.Nakao[‡]

[†]Computer Center, Kyushu University 33

[‡]Graduate School of Mathematics, Kyushu University 33
Fukuoka 812-81, JAPAN

Generalized Eigenvalue Problems

PROBLEM

Find the maximum absolute value of λ such that

$$A\boldsymbol{x} = \lambda B\boldsymbol{x}, \tag{1}$$

$\boldsymbol{x} \in \boldsymbol{R}^n$,
A: symmetric in $\boldsymbol{R}^{n\times n}$,
B: symmetric and positive definite in $\boldsymbol{R}^{n\times n}$.

○ Motivation ○

- As conserns the numerical verification methods for PDEs which we have developed, we need an *upper bound* of λ as sharp as possible.
- Aside from the numerical verification, this situation with the problem (1) often appears in many cases in mathematical science and engineering.
- Thus we believe that it is significant to investigate calculations for (1) with *guaranteed accuracy*.

Methods for Generalized Eigenvalue Probrems

We explain four kinds of methods:

METHODS

Method 1:	Approximate Diagonalization Method
Method 2:	Approximate Diagonalization Method Advanced
Method 3:	Rump's Method
Method 4:	Generalized Rump's Method

Other Methods

- G. ALEFELD: *Computing* **36** (1986) and *Z.angew.Math.Mech* **67** (1987), etc.
- H. BEHNKE: *Computing, Supplement* **6** (1988) and *Computing* **47** (1991)
- R. LOHNER: in *Accurate Numerical Algorithms: A Collection of Reserch Papers*(1989)
- G. MAYER: in *Computer Arithmetic and Enclosure Methods* (1992) and *Topics in Validated Computations* (1994)
- S. M. RUMP: *Computing* **42** (1989) and *Topics in Validated Computations*(1994)
- T. YAMAMOTO: *Numerische Mathematik* **34** (1980) and **40** (1982)

The Features of Methods

- Method 1 and Method 3 treat problems transformed to

$$\lambda = \sup_{\boldsymbol{y}\neq\mathbf{o}} \left| \frac{\boldsymbol{y}^T E \boldsymbol{y}}{\boldsymbol{y}^T \boldsymbol{y}} \right|, \tag{2}$$

$$E := C^{-1} A C^{-T},$$

  C: $n \times n$ lower triangular matrix obtained by the Cholesky decomposition of B, namely,

$$B = CC^T.$$

- Note that this transformation should be executed by a rigorous calculation using interval arithmetic considering the rounding error. Thus E is given as an interval matrix.
- The Method 2 and Method 4 do not need the transformation from (1) to (2) (i.e. $A\boldsymbol{x} = \lambda B\boldsymbol{x} \implies (C^{-1}AC^{-T})\boldsymbol{x} = \lambda\boldsymbol{x}$), thus we will find

$$\lambda = \sup_{\boldsymbol{x}\neq\mathbf{o}} \left| \frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T B \boldsymbol{x}} \right|.$$

Approximate Diagonalization Method

Algorithm

1. $B = CC^T$ — rigorously
2. $E := C^{-1}AC^{-T}$ — rigorously
3. Compute a $n \times n$ matrix $\tilde{T}$ which approximately diagonalizes $\mathrm{mid}(E)$ by some suitable method (eg. QR method). — approximately
4. $\tilde{D} := \tilde{T}^T E \tilde{T}$, — rigorously
   $\tilde{\lambda} := \max\limits_{1 \le i \le n} \sum\limits_{j=1}^{n} |\tilde{D}_{ij}|$ — rigorously
5. $\tilde{I} := (\tilde{T}\tilde{T}^T)^{-1}$, — rigorously
   $\lambda_T := \max\limits_{1 \le i \le n} \sum\limits_{j=1}^{n} |\tilde{I}_{ij}|$ — rigorously
6. $\lambda \le \tilde{\lambda}\lambda_T$ — rigorously

Slide 5

Approximate Diagonalization Method

Proof

$$
\begin{aligned}
\lambda &= \sup_{\boldsymbol{y} \neq \mathbf{o}} \left| \frac{\boldsymbol{y}^T E \boldsymbol{y}}{\boldsymbol{y}^T \boldsymbol{y}} \right| \\
&= \sup_{\boldsymbol{z} \neq \mathbf{o}} \left| \frac{\boldsymbol{z}^T \tilde{T}^T E \tilde{T} \boldsymbol{z}}{\boldsymbol{z}^T \tilde{T}^T \tilde{T} \boldsymbol{z}} \right| \qquad \longleftarrow \qquad \boldsymbol{z} = \tilde{T}^{-1}\boldsymbol{y} \\
&\le ||\tilde{T}^T E \tilde{T}||_2 \; ||(\tilde{T}^T \tilde{T})^{-1}||_2 \qquad \text{where} \quad ||A||_2 := \sup_{\boldsymbol{x} \neq \mathbf{o}} \frac{||A\boldsymbol{x}||}{||\boldsymbol{x}||} \\
&= ||\tilde{D}||_2 \; ||\tilde{I}||_2 \\
&\le \tilde{\lambda}\lambda_T \qquad \longleftarrow \qquad ||A||_2 \le \max_{1 \le i \le n} \sum_{j=1}^{n} |A_{ij}|
\end{aligned}
$$

(Corollary of Gerschgorin's thorem)

Slide 6

Approximate Diagonalization Method Advanced

Algorithm

1.	$B \approx CC^T$	approximately
2.	$\tilde{E} := \tilde{C}^{-1} A \tilde{C}^{-T}$	approximately
3.	Compute a $n \times n$ matrix $\tilde{T}$ which approximately diagonalizes $\tilde{E}$ by some suitable method (eg. QR method).	approximately
4.	$\tilde{P} := \tilde{T}^T \tilde{C}^{-1}$	approximately
5.	$X_1 := \tilde{P} A \tilde{P}^T$	rigorously
	$\lambda_1 := \max_{1 \le i \le n} \sum_{j=1}^{n} \lvert (X_1)_{ij} \rvert$	rigorously
6.	$X_2 := (\tilde{P} B \tilde{P}^T)^{-1}$	rigorously
	$\lambda_2 := \max_{1 \le i \le n} \sum_{j=1}^{n} \lvert (X_2)_{ij} \rvert$	rigorously
7.	$\lambda \le \lambda_1 \lambda_2$	rigorously

Slide 7

Approximate Diagonalization Method Advanced

Proof

$$
\begin{aligned}
\lambda &= \sup_{x \neq \mathbf{o}} \left| \frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T B \boldsymbol{x}} \right| \\
&= \sup_{\boldsymbol{y} \neq \mathbf{o}} \left| \frac{\boldsymbol{y}^T \tilde{C}^{-1} A \tilde{C}^{-T} \boldsymbol{y}}{\boldsymbol{y}^T \tilde{C}^{-1} B \tilde{C}^{-T} \boldsymbol{y}} \right| && \longleftarrow \quad \boldsymbol{y} = \tilde{C}^T \boldsymbol{x} \\
&= \sup_{\boldsymbol{z} \neq 0} \left| \frac{\boldsymbol{z}^T \tilde{T}^T \tilde{C}^{-1} A \tilde{C}^{-T} \tilde{T} \boldsymbol{z}}{\boldsymbol{z}^T \tilde{T}^T \tilde{C}^{-1} B \tilde{C}^{-T} \tilde{T} \boldsymbol{z}} \right| && \longleftarrow \quad \boldsymbol{z} = \tilde{T}^{-1} \boldsymbol{y} \\
&= \sup_{\boldsymbol{z} \neq \mathbf{o}} \left| \frac{\boldsymbol{z}^T \tilde{P} A \tilde{P}^T \boldsymbol{z}}{\boldsymbol{z}^T \tilde{P} B \tilde{P}^T \boldsymbol{z}} \right| && \longleftarrow \quad \boldsymbol{z} = \tilde{P}^{-T} \boldsymbol{x} \\
&\le ||\tilde{P} A \tilde{P}^T||_2 \; ||(\tilde{P} B \tilde{P}^T)^{-1}||_2 && \text{where} \quad ||A||_2 := \sup_{\boldsymbol{x} \neq \mathbf{o}} \frac{||A\boldsymbol{x}||}{||\boldsymbol{x}||} \\
&= ||X_1||_2 \; ||X_2||_2 \\
&\le \lambda_1 \lambda_2 && \longleftarrow \quad ||A||_2 \le \max_{1 \le i \le n} \sum_{j=1}^{n} |A_{ij}|
\end{aligned}
$$

(Corollary of Gerschgorin's thorem)

Slide 8

Rump's Method

Algorithm

1.	$B = CC^T$	rigorously
2.	$E := C^{-1}AC^{-T}$	rigorously
3.	Compute an approximate value $\gamma \approx \lambda$ by some favorite method.	approximately
4.	$\tilde{\gamma} := (1+\delta)\gamma \quad$ for some $0 < \delta \ll 1$	rigorously
5.	$X_1 := \tilde{\gamma}I - E, \quad X_2 := \tilde{\gamma}I + E$	rigorously
6.	$\mathrm{mid}(X_1) \approx \tilde{C}_1\tilde{C}_1{}^T, \quad \mathrm{mid}(X_2) \approx \tilde{C}_2\tilde{C}_2{}^T$	approximately
7.	$Y_1 := \tilde{C}_1\tilde{C}_1{}^T - X_1, \quad \lambda_1 := \max_{1\le i\le n} \sum_{j=1}^{n} \lvert(Y_1)_{ij}\rvert$	rigorously
8.	$Y_2 := \tilde{C}_2\tilde{C}_2{}^T - X_2, \quad \lambda_2 := \max_{1\le i\le n} \sum_{j=1}^{n} \lvert(Y_2)_{ij}\rvert$	rigorously
9.	$\lambda \le \tilde{\gamma} + \max\{\lambda_1, \lambda_2\}$	rigorously

Rump's Method

Proof

For any vector $\boldsymbol{x} \in \boldsymbol{R}^n$ under $\boldsymbol{x}^T\boldsymbol{x} = 1$,

$$|\boldsymbol{x}^T(\tilde{C}_1\tilde{C}_1{}^T - \tilde{\gamma}I + E)\boldsymbol{x}| = |\boldsymbol{x}^T Y_1 \boldsymbol{x}| \le \lambda_1,$$

$$|\boldsymbol{x}^T(\tilde{C}_2\tilde{C}_2{}^T - \tilde{\gamma}I - E)\boldsymbol{x}| = |\boldsymbol{x}^T Y_2 \boldsymbol{x}| \le \lambda_2.$$

Since $\tilde{C}_1\tilde{C}_1{}^T$ and $\tilde{C}_2\tilde{C}_2{}^T$ are positive definite, we obtain

$$-\tilde{\gamma} - \lambda_2 \le \boldsymbol{x}^T E \boldsymbol{x} \le \tilde{\gamma} + \lambda_1.$$

Then, $\lambda \le \tilde{\gamma} + \max\{\lambda_1, \lambda_2\}$ holds.

∘ Note ∘

If A is known as positive definite, the process for X_2 can be omitted.

Generalized Rump's Method

Algorithm

1. Compute an approximate value $\gamma \approx \lambda$ by some favorite method. — approximately
2. $\tilde{\gamma} := (1+\delta_1)\gamma \quad$ for some $0 < \delta_1 \ll 1$ — rigorously
3. $X_1 := \tilde{\gamma}B - A, \quad X_2 := \tilde{\gamma}B + A$ — rigorously

   We will verify each X_k $(k = 1, 2)$ is **positive definite**
4. Compute an approximate value ρ_k to the minimum value of the eigenvalues of X_k. — approximately
5. $\tilde{\rho_k} := (1-\delta_2)\rho_k \quad$ for some $0 < \delta_2 \ll 1$ — rigorously
6. $Y_k := X_k - \tilde{\rho_k} I$ — rigorously
7. $Y_k \approx \tilde{C_k}\tilde{C_k}^T$ — approximately
8. $Z_k := \tilde{C_k}\tilde{C_k}^T - Y_k, \quad \lambda_k := \max\limits_{1\le i\le n} \sum\limits_{j=1}^{n} |(Z_k)_{ij}|$ — rigorously
9. If $\tilde{\rho_k} - \lambda_k > 0$ then X_k is positive definite and $\lambda \le \tilde{\gamma}$.

Generalized Rump's Method

Proof

For any vector $\boldsymbol{x} \in \boldsymbol{R}^n$ under $\boldsymbol{x}^T\boldsymbol{x} = 1$,

$$|\boldsymbol{x}^T(\tilde{C_k}\tilde{C_k}^T - X_k + \tilde{\rho_k} I)\boldsymbol{x}| = |\boldsymbol{x}^T Z_k \boldsymbol{x}| \le \lambda_k,$$

Since $\tilde{C_k}\tilde{C_k}^T$ positive definite, we obtain

$$\boldsymbol{x}^T X_k \boldsymbol{x} \ge \tilde{\rho_k} - \lambda_k > 0$$

Then, X_k is positive definite.

Hence,

$$\boldsymbol{x}^T(\tilde{\gamma}B - A)\boldsymbol{x} \ge 0, \quad \boldsymbol{x}^T(\tilde{\gamma}B + A)\boldsymbol{x} \ge 0,$$

then

$$-\tilde{\gamma}\boldsymbol{x}^T B\boldsymbol{x} \le \boldsymbol{x}^T A\boldsymbol{x} \le \tilde{\gamma}\boldsymbol{x}^T B\boldsymbol{x}.$$

Since B is positive definite, we conclude that

$$\lambda = \sup_{\boldsymbol{x}^T\boldsymbol{x}=1} \left|\frac{\boldsymbol{x}^T A\boldsymbol{x}}{\boldsymbol{x}^T B\boldsymbol{x}}\right| \le \tilde{\gamma}.$$

Numerical Experiments

○ Note ○

- The results are obtained by rigorous calculation taking account of the effects of the rounding error with interval arithmetic.
  In practice, the interval library `INTLIB_90` is used on Sun Ultra Enterprise 450 (300MHz) and FUJITSU VPP700/56(2.2GFLOPS).
- In order to check "diagonally dominancy", we define

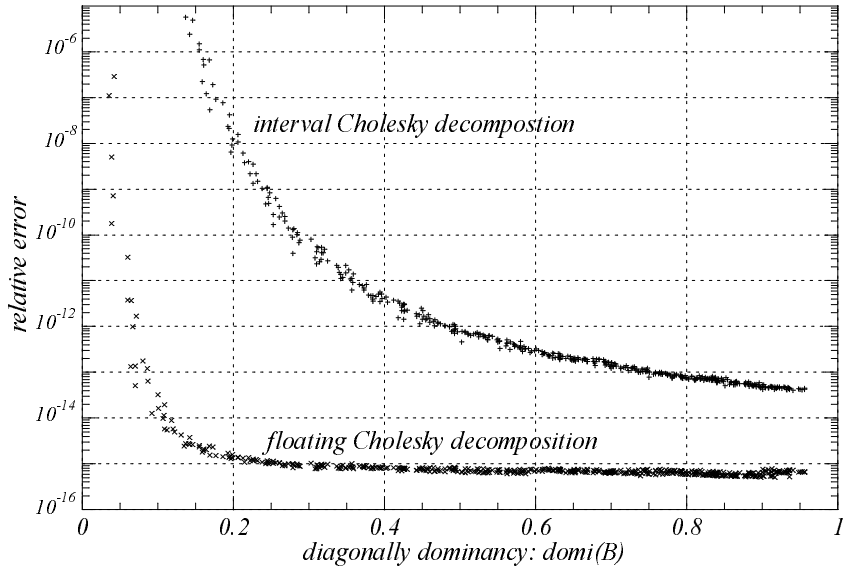
$$\mathrm{domi}(B) := \min_{1 \le i \le n} \frac{|B_{ii}|}{\sum_{j \neq i} |B_{ij}|}.$$

  If $\mathrm{domi}(B) > 1$ then B is strictry diagonally dominant matrix.
- "An interval matrix is symmetric or positive definite." means "We will consider a set of symmetric or positive definite (non interval) matrices within the interval matrix."

Test Matrices

1. Give a list of eigenvalues arbitrarily, and a diagonal matrix D which has these eigenvalues as its diagonal elements.
2. Take a vector $\boldsymbol{v} \in \boldsymbol{R}^n$ with random elements,
   and set $\boldsymbol{u} := \boldsymbol{v}/\|\boldsymbol{v}\|$, where $\|\boldsymbol{v}\| = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}$.
3. Calculate $M := I - 2\boldsymbol{u}\boldsymbol{u}^T$.
   Then M is an orthogonal matrix : $M^T = M^{-1}$.
4. Determine a lower triangular matrix Q by random numbers.
5. Take $A := QM^T DMQ^T$ and $B := QQ^T$.

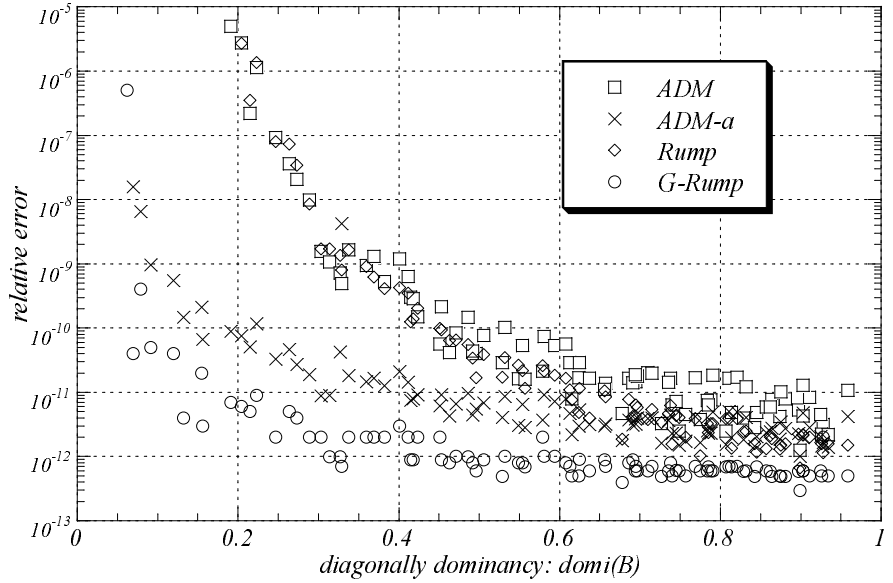
Interval Cholesky Decomposition



$n = 100$. If "diagonally dominancy" is small, the Cholesky decomposition of $B := QQ^T$ fails or the inflation of intervals caused by the rounding error. Method 2 and Method 4 are available.

Slide 15

Performance(accuracy)



$n = 100$. When the condition of the matrix B is not so good, Method 2 and Method 4 give desiable results. In Rump and G-Rump, the best parameters δ are chosen from several values.

Slide 16

Performance(efficiency)

Table 1. Elapsed time for each algorithms(sec.)

computer(sizes)	ADM	ADM-a	Rump	G-Rump
VPP700/56(100×100)	39	40	5.3	2.8
VPP700/56(200×200)	337	313	41	19
Ultra Enterprise 450(100×100)	22	23	3.6	3.2
Ultra Enterprise 450(200×200)	169	183	24	23

Generalized Rump's Method is faster than other methods because interval Cholesky decomposition and interval matrix inversion (based on Brouwer's fixed point theorem) are not required.

Numerical Experiments(conclusion)

- In view of the cost of computation, Method 1 and Method 2 are expensive because of the rigorous calculation of inverse matrices.
- Method 3 is an effective method for standard eigenvalue problem, however, the inflation of intervals might be occured if "diagonally dominancy" is small.

$\Downarrow$

Generalized Rump's Method(Method 4) is the best of the four if it works with a sufficiently small parameter δ_1.

Application to the Stokes Problems

• **Stokes Equations**

$$\begin{cases} -\nu\Delta u + \nabla p = f & \text{in} \quad \Omega, \\ \operatorname{div} u = 0 & \text{in} \quad \Omega, \\ u = 0 & \text{on} \quad \partial\Omega, \end{cases}$$

where Ω is a convex polygon in $\mathbb{R}^2$.

◦ **Weak Formulation**

$$\nu(\nabla u, \nabla v) - (p, \operatorname{div} v) - (q, \operatorname{div} u) = (f, v) \qquad \forall [v, q] \in H_0^1(\Omega)^2 \times L_0^2(\Omega).$$

- $X_h \subset H_0^1(\Omega) \cap C(\bar{\Omega})$: approximation of the velocity u satisfies

$$\inf_{\xi \in X_h} |v - \xi|_1 \le C_0 h |v|_2, \quad \forall v \in H_0^1(\Omega) \cap H^2(\Omega),$$

where C_0 is a positive constant numerically determined.

- $Y_h \subset L_0^2(\Omega) \cap C(\bar{\Omega})$: approximation of the pressure p.

◦ **Finite Element Solution**

$$\nu(\nabla u_h, \nabla v_h) - (p_h, \operatorname{div}\, v_h) - (q_h, \operatorname{div}\, u_h) = (f, v_h) \quad \forall [v_h, q_h] \in X_h^2 \times Y_h.$$

A Priori Error Estimates for the Stokes Equations

THEOREM

$$|u - u_h|_1 \le C(h) \left(\frac{1}{\nu^2} + \frac{1}{\beta^2} \right)^{\frac{1}{2}} |f|_0,$$

$$|p - p_h|_0 \le C(h) \left(\frac{1}{\beta} + \frac{\nu}{\beta^2} \right) |f|_0.$$

where

$$C(h) \equiv \sqrt{(K_1 + C_0 h K_2)^2 + (C_0 h)^2},$$

$$|v|_1 \le \frac{1}{\beta} |\operatorname{div} v|_0 \quad v \in \{u \in H_0^1(\Omega)^2; \operatorname{div} u = 0\}^{\perp},$$

$$K_1 := \left(\sup_{\boldsymbol{x} \in \boldsymbol{R}^n} \frac{\boldsymbol{x}^T A_1 \boldsymbol{x}}{\boldsymbol{x}^T L \boldsymbol{x}} \right)^{1/2},$$

$$K_2 := \left(\sup_{\boldsymbol{x} \in \boldsymbol{R}^n} \frac{\boldsymbol{x}^T A_2 \boldsymbol{x}}{\boldsymbol{x}^T L \boldsymbol{x}} \right)^{1/2}.$$

Numerical Example

- $\Omega = (0,1) \times (0,1)$.
- uniform partition. L denotes the number of partitions for the interval $(0,1)$, i.e. $h = 1/L$.
- X_h : continuous piecewise quadratic polynomials, Y_h : piecewise linear.
- $\nu = 1$, $C_0 = 1/(2\pi)$ and $1/\beta^2 = 4 + 2\sqrt{2}$.

Error Estimates for Stokes equations

Table 2. A priori error constants for Stokes equations

L	algorithms	K_1	K_2	$C(h)$
5	G-Rump	0.05100876310491353	1.268294925398748	0.09676511668110258
	Rump	0.05100876311238357	1.268294926799123	0.09676511673025147
	ADM	0.05100876357504820	1.268294978633361	0.09676511872527857
	ADM-a	0.05100876404340230	1.268294979559628	0.09676511919541045
	floating	0.05100876307940880	1.268294924764600	0.09676511663795448
10	G-Rump	0.02794084133393883	1.238111303510752	0.05023389068930387
	Rump	0.02794084209468706	1.238111523884439	0.05023389473752864
	ADM	0.02794094026027832	1.238127882787599	0.05023423479314133
	ADM-a	0.02794085315113312	1.238113249056619	0.05023393126685157
	floating	0.02794083993689644	1.238111241605190	0.05023388842973234
15	G-Rump	0.01896605324389127	1.238019686150493	0.03380987832209943
	Rump	0.01896606508645449	1.238023659225624	0.03380992959238906
	ADM	0.01897072488949515	1.238612263477653	0.03382028394542623
	ADM-a	0.01898434068678022	1.242378307644777	0.03387115806908502
	floating	0.01896595841433506	1.238013496098524	0.03380972592273357

"*floating*"s are computed on FUJITSU VP2600/10 vector processor by IBM floating format with double precision and Householder-Bisection method.