On the Complex Falk-Langemeyer Method

Vjeran Hari

Faculty of Science, Department of Mathematics, University of Zagreb
hari@math.hr

89th GAMM Annual Meeting
March 19-23, 2018
Munich, Germany

OUTLINE

OUTLINE

- PGEP and DGEP

OUTLINE

- PGEP and DGEP
- Falk-Langemeyer algorithm, what is known

OUTLINE

- PGEP and DGEP
- Falk-Langemeyer algorithm, what is known
- Derivation of the complex FL algorithm (CFL)

OUTLINE

- PGEP and DGEP
- Falk-Langemeyer algorithm, what is known
- Derivation of the complex FL algorithm (CFL)
- Stability and relative accuracy, convergence

OUTLINE

- PGEP and DGEP
- Falk-Langemeyer algorithm, what is known
- Derivation of the complex FL algorithm (CFL)
- Stability and relative accuracy, convergence
- This work has been fully supported by Croatian Science Foundation under the project IP-09-2014-3670.

GEP and PGEP

Let $A = A^*, \quad B = B^*$. We consider the

Generalized Eigenvalue Problem (GEP): $\quad Ax = \lambda Bx, \quad x \neq 0.$

GEP and PGEP

Let $\quad A = A^*, \quad B = B^*.$ We consider the

Generalized Eigenvalue Problem (GEP): $\quad Ax = \lambda Bx, \quad x \neq 0.$

- If $B \succ O$, GEP is usually called positive definite GEP or shorter PGEP

GEP and PGEP

Let $A = A^*$, $B = B^*$. We consider the

$$\text{Generalized Eigenvalue Problem (GEP)}: \quad Ax = \lambda Bx, \quad x \neq 0.$$

- If $B \succ O$, GEP is usually called positive definite GEP or shorter PGEP
- If $sA + tB \succ O$, for some real s, t, we have definite GEP and also definite matrix pair (A, B)

GEP and PGEP

Let $\quad A = A^*, \quad B = B^*.$ We consider the

Generalized Eigenvalue Problem (GEP): $\quad Ax = \lambda Bx, \quad x \neq 0.$

- If $B \succ O$, GEP is usually called positive definite GEP or shorter PGEP
- If $sA + tB \succ O$, for some real s, t, we have definite GEP and also definite matrix pair (A, B)
- For a definite pair (A, B) there is a nonsingular matrix F such that

$$F^*AF = \Lambda_A \,, \qquad F^*BF = \Lambda_B ,$$

$\Lambda_A = \text{diag}(\alpha_1, \ldots, \alpha_n), \quad \Lambda_B = \text{diag}(\beta_1, \ldots, \beta_n)$ are real matrices

GEP and PGEP

Let $A = A^*$, $B = B^*$. We consider the

Generalized Eigenvalue Problem (GEP): $Ax = \lambda Bx, \quad x \neq 0.$

- If $B \succ O$, GEP is usually called positive definite GEP or shorter PGEP
- If $sA + tB \succ O$, for some real s, t, we have definite GEP and also definite matrix pair (A, B)
- For a definite pair (A, B) there is a nonsingular matrix F such that

$$F^* A F = \Lambda_A, \qquad F^* B F = \Lambda_B,$$

$\Lambda_A = \mathrm{diag}(\alpha_1, \ldots, \alpha_n)$, $\Lambda_B = \mathrm{diag}(\beta_1, \ldots, \beta_n)$ are real matrices

- The eigenpairs are: $(\alpha_i/\beta_i, Fe_i)$, $1 \leq i \leq n$; $I_n = [e_1, \ldots, e_n]$.

How to Solve Definite GEP?

- If $B \succ O$, use the transformation: $(A, B) \mapsto (L^{-1}AL^{-*}, I)$, $B = LL^*$.

How to Solve Definite GEP?

- If $B \succ O$, use the transformation: $(A, B) \mapsto (L^{-1}AL^{-*}, I)$, $B = LL^*$. This reduces PGEP to the EP for one Hermitian matrix. However, if L has small singular value(s), then the computed $L^{-1}AL^{-T}$ will have corrupt eigenvalues
- If $A \succ O$, apply the same procedure to (B, A)

How to Solve Definite GEP?

- If $B \succ O$, use the transformation: $(A, B) \mapsto (L^{-1}AL^{-*}, I)$, $B = LL^*$. This reduces PGEP to the EP for one Hermitian matrix. However, if L has small singular value(s), then the computed $L^{-1}AL^{-T}$ will have corrupt eigenvalues
- If $A \succ O$, apply the same procedure to (B, A)
- $A \succ O$ and $B \succ O$ apply one of the above procedures (take care which matrix has smaller condition number). Or employ the methods for the GSVD problem $L_A L_A^* x = \sigma^2 L_B L_B^* x$.

How to Solve Definite GEP?

- If $B \succ O$, use the transformation: $(A, B) \mapsto (L^{-1}AL^{-*}, I)$, $B = LL^*$. This reduces PGEP to the EP for one Hermitian matrix. However, if L has small singular value(s), then the computed $L^{-1}AL^{-T}$ will have corrupt eigenvalues
- If $A \succ O$, apply the same procedure to (B, A)
- $A \succ O$ and $B \succ O$ apply one of the above procedures (take care which matrix has smaller condition number). Or employ the methods for the GSVD problem $L_A L_A^* x = \sigma^2 L_B L_B^* x$.
- If neither A nor B is definite,one can try to maximize the minimum eigenvalue of B_φ by rotating the pair

$$(A, B) \mapsto (A_\varphi, B_\varphi) = (A\cos\varphi + B\sin\varphi, -A\sin\varphi + B\cos\varphi),$$

How to Solve Definite GEP?

If neither A nor B is definite, one can:

- use the indefinte Cholesky factorization to reduce the problem to the J-Hermitian EP

$$Hx = \lambda Jx, \qquad J \text{ is a matrix of signs}$$

How to Solve Definite GEP?

If neither A nor B is definite, one can:

- use the indefinte Cholesky factorization to reduce the problem to the J-Hermitian EP

$$Hx = \lambda Jx, \qquad J \text{ is a matrix of signs}$$

- employ the QZ method, which is complicated, slow and inaccurate

How to Solve Definite GEP?

If neither A nor B is definite, one can:

- use the indefinte Cholesky factorization to reduce the problem to the J-Hermitian EP

$$Hx = \lambda Jx, \qquad J \text{ is a matrix of signs}$$

- employ the QZ method, which is complicated, slow and inaccurate
- generalize the Falk-Langemeyer method to work with complex matrices

How to Solve Definite GEP?

If neither A nor B is definite, one can:

- use the indefinte Cholesky factorization to reduce the problem to the J-Hermitian EP

$$Hx = \lambda Jx, \qquad J \text{ is a matrix of signs}$$

- employ the QZ method, which is complicated, slow and inaccurate
- generalize the Falk-Langemeyer method to work with complex matrices

We follow the last choice!

Jacobi Methods for PGEP

We have at disposal several diagonalization methods for PGEP with real matrices:

- Falk-Langemeyer method (shorter: FL method)
  (Elektronische Datenverarbeitung, 1960)

Jacobi Methods for PGEP

We have at disposal several diagonalization methods for PGEP with real matrices:

- Falk-Langemeyer method (shorter: FL method)
  (Elektronische Datenverarbeitung, 1960)
- HZ (Hari-Zimmermann) method
  Numerical Algorithms, 2018 (to appear)

Jacobi Methods for PGEP

We have at disposal several diagonalization methods for PGEP with real matrices:

- Falk-Langemeyer method (shorter: FL method)
  (Elektronische Datenverarbeitung, 1960)
- HZ (Hari-Zimmermann) method
  Numerical Algorithms, 2018 (to appear)
- CJ (Cholesky-Jacobi) method
  Numerical Algorithms, 2018 (to appear)

Jacobi Methods for PGEP

We have at disposal several diagonalization methods for PGEP with real matrices:

- Falk-Langemeyer method (shorter: FL method)
  (Elektronische Datenverarbeitung, 1960)
- HZ (Hari-Zimmermann) method
  Numerical Algorithms, 2018 (to appear)
- CJ (Cholesky-Jacobi) method
  Numerical Algorithms, 2018 (to appear)

All three methods have excellent numerical properties, in particular they are indicated as high relative accurate on well-behaved positive definite matrices.

Jacobi Methods on Contemporary Computing Machines

- Element-wise Jacobi methods (two-sided or one-sided) are often used as kernel algorithms inside the corresponding block methods

Jacobi Methods on Contemporary Computing Machines

- Element-wise Jacobi methods (two-sided or one-sided) are often used as kernel algorithms inside the corresponding block methods
- One-sided block Jacobi methods are nicely adaptable to work with modern CPU and GPU parallel computing machines

Jacobi Methods on Contemporary Computing Machines

- Element-wise Jacobi methods (two-sided or one-sided) are often used as kernel algorithms inside the corresponding block methods
- One-sided block Jacobi methods are nicely adaptable to work with modern CPU and GPU parallel computing machines

A quote from V. Novaković, S. Singer, S. Singer (Parallel Comput., 2015):

Jacobi Methods on Contemporary Computing Machines

- Element-wise Jacobi methods (two-sided or one-sided) are often used as kernel algorithms inside the corresponding block methods
- One-sided block Jacobi methods are nicely adaptable to work with modern CPU and GPU parallel computing machines

A quote from V. Novaković, S. Singer, S. Singer (Parallel Comput., 2015):

> *Numerical tests on large matrices, on parallel machines, have confirmed the advantage of the HZ approach. When implemented as one-sided block algorithm for the GSVD, it is almost perfectly parallelizable, so parallel shared memory versions of the algorithm are highly scalable, and their speedup almost solely depends on the number of cores used.*

Jacobi Methods on Contemporary Computing Machines

- Element-wise Jacobi methods (two-sided or one-sided) are often used as kernel algorithms inside the corresponding block methods
- One-sided block Jacobi methods are nicely adaptable to work with modern CPU and GPU parallel computing machines

A quote from V. Novaković, S. Singer, S. Singer (Parallel Comput., 2015):

> *Numerical tests on large matrices, on parallel machines, have confirmed the advantage of the HZ approach. When implemented as one-sided block algorithm for the GSVD, it is almost perfectly parallelizable, so parallel shared memory versions of the algorithm are highly scalable, and their speedup almost solely depends on the number of cores used.*

The same can be said for the CJ and FL method.

Few Facts about Real FL method

- FL method is well defined for any definite matrix pair

(Slapničar, Hari: SIMAX, 1991)

Few Facts about Real FL method

- FL method is well defined for any definite matrix pair

(Slapničar, Hari: SIMAX, 1991)

- Quadratic convergence proved in the case of simple eigenvalues

(Slapničar, Hari: SIMAX, 1991)

Few Facts about Real FL method

- FL method is well defined for any definite matrix pair

(Slapničar, Hari: SIMAX, 1991)

- Quadratic convergence proved in the case of simple eigenvalues

(Slapničar, Hari: SIMAX, 1991)

- Relative accuracy investigated, general bounds obtained

(Matejaš, Numerical Algorithms, **2015**)

Few Facts about Real FL method

- FL method is well defined for any definite matrix pair

  (Slapničar, Hari: SIMAX, 1991)

- Quadratic convergence proved in the case of simple eigenvalues

  (Slapničar, Hari: SIMAX, 1991)

- Relative accuracy investigated, general bounds obtained

  (Matejaš, Numerical Algorithms, **2015**)

- Global convergence not yet proved

  (the proof will be similar to the one in Hari, Num. Algor., 2018)

Few Facts about Real FL method

- FL method is well defined for any definite matrix pair

  (Slapničar, Hari: SIMAX, 1991)

- Quadratic convergence proved in the case of simple eigenvalues

  (Slapničar, Hari: SIMAX, 1991)

- Relative accuracy investigated, general bounds obtained

  (Matejaš, Numerical Algorithms, **2015**)

- Global convergence not yet proved

  (the proof will be similar to the one in Hari, Num. Algor., 2018)

- High relative accuracy (HRA) of the FL method not yet proved

  (numerical tests indicate HRA of the method)

Derivation of the CFL Method

Starting with a definite pair (A, B) of complex Hermitian matrices, CFL generates a sequence of "congruent" matrix pairs

$$(A, B) = (A^{(0)}, B^{(0)}),\ (A^{(1)}, B^{(1)}),\ (A^{(2)}, B^{(2)}) \dots$$

by the rule

$$A^{(k+1)} = F_k^* A^{(k)} F_k \ , \qquad B^{(k+1)} = F_k^* B^{(k)} F_k \ , \quad k \geq 0.$$

Here F_k is an elementary plane matrix defined by the *pivot pair* $(i(k), j(k))$

$$F_k = \begin{bmatrix} I & & & & \\ & 1 & & \alpha_k & \\ & & I & & \\ & \beta_k & & 1 & \\ & & & & I \end{bmatrix} \begin{matrix} \\ i(k) \\ \\ j(k) \\ \\ \end{matrix} , \qquad \alpha_k, \beta_k \in \mathbf{C},$$

Derivation of the CFL Method

The goal is to compute complex numbers α_k, β_k such that the pivot elements $a_{ij}^{(k)}$, $b_{ij}^{(k)}$ of $A^{(k)}$, $B^{(k)}$ are annihilated.

Derivation of the CFL Method

The goal is to compute complex numbers α_k, β_k such that the pivot elements $a_{ij}^{(k)}$, $b_{ij}^{(k)}$ of $A^{(k)}$, $B^{(k)}$ are annihilated.

We simplify notation: $A \leftarrow A^{(k)}$, $A' \leftarrow A^{(k+1)}$, $F \leftarrow F_k$, $(i,j) \leftarrow (i(k),j(k))$.

Pivot submatrices $\hat{A}$, $\hat{B}$, $\hat{F}$ of A, B, F are 2×2 principal submatrices obtained on the intersection of pivot rows and columns i and j.

Derivation of the CFL Method

The goal is to compute complex numbers α_k, β_k such that the pivot elements $a_{ij}^{(k)}$, $b_{ij}^{(k)}$ of $A^{(k)}$, $B^{(k)}$ are annihilated.

We simplify notation: $A \leftarrow A^{(k)}$, $A' \leftarrow A^{(k+1)}$, $F \leftarrow F_k$, $(i,j) \leftarrow (i(k),j(k))$.

Pivot submatrices $\hat{A}$, $\hat{B}$, $\hat{F}$ of A, B, F are 2×2 principal submatrices obtained on the intersection of pivot rows and columns i and j.

We have

$$A' = F^* A F, \quad B' = F^* B F \qquad \left(\hat{A}' = \hat{F}^* \hat{A} \hat{F}, \quad \hat{B}' = \hat{F}^* \hat{B} \hat{F} \right)$$

and F is chosen to obtain $a'_{ij} = 0$, $b'_{ij} = 0$.

Derivation of the CFL Method ($n = 2$)

Further simplification: $(1,2) \leftarrow (i,j)$, $a_1 \leftarrow a_{ii}$, $a_2 \leftarrow a_{ij}$, $a_3 \leftarrow a_{jj}$, $a_1' \leftarrow a_{ii}'$, ...

Derivation of the CFL Method ($n = 2$)

Further simplification: $(1,2) \leftarrow (i,j)$, $a_1 \leftarrow a_{ii}$, $a_2 \leftarrow a_{ij}$, $a_3 \leftarrow a_{jj}$, $a'_1 \leftarrow a'_{ii}$, ...

The goal is to compute α and β which satisfy the matrix equations

$$\begin{bmatrix} 1 & \bar{\beta} \\ \bar{\alpha} & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 \\ \bar{a}_2 & a_3 \end{bmatrix} \begin{bmatrix} 1 & \alpha \\ \beta & 1 \end{bmatrix} = \begin{bmatrix} a'_1 & 0 \\ 0 & a'_3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & \bar{\beta} \\ \bar{\alpha} & 1 \end{bmatrix} \begin{bmatrix} b_1 & b_2 \\ \bar{b}_2 & b_3 \end{bmatrix} \begin{bmatrix} 1 & \alpha \\ \beta & 1 \end{bmatrix} = \begin{bmatrix} b'_1 & 0 \\ 0 & b'_3 \end{bmatrix}.$$

Derivation of the CFL Method ($n = 2$)

Further simplification: $(1,2) \leftarrow (i,j)$, $a_1 \leftarrow a_{ii}$, $a_2 \leftarrow a_{ij}$, $a_3 \leftarrow a_{jj}$, $a_1' \leftarrow a_{ii}'$, …

The goal is to compute α and β which satisfy the matrix equations

$$\begin{bmatrix} 1 & \bar{\beta} \\ \bar{\alpha} & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 \\ \bar{a}_2 & a_3 \end{bmatrix} \begin{bmatrix} 1 & \alpha \\ \beta & 1 \end{bmatrix} = \begin{bmatrix} a_1' & 0 \\ 0 & a_3' \end{bmatrix}$$

$$\begin{bmatrix} 1 & \bar{\beta} \\ \bar{\alpha} & 1 \end{bmatrix} \begin{bmatrix} b_1 & b_2 \\ \bar{b}_2 & b_3 \end{bmatrix} \begin{bmatrix} 1 & \alpha \\ \beta & 1 \end{bmatrix} = \begin{bmatrix} b_1' & 0 \\ 0 & b_3' \end{bmatrix}.$$

This leads us to solving a system of two nonlinear equations:

$$e_1 = a_1\alpha + a_3\bar{\beta} + \bar{a}_2\alpha\bar{\beta} + a_2 = 0, \quad (1)$$

$$e_2 = b_1\alpha + b_3\bar{\beta} + \bar{b}_2\alpha\bar{\beta} + b_2 = 0. \quad (2)$$

Derivation of the CFL Method

To solve the obtained system of equation, we use the following quantities:

$$
\begin{aligned}
\Im_1 &= a_1 b_2 - a_2 b_1 = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} \\
\Im_3 &= a_3 b_2 - a_2 b_3 = \begin{vmatrix} a_3 & b_3 \\ a_2 & b_2 \end{vmatrix} \\
\Im_2 &= \Im_2' + i\Im_2'', \qquad\qquad \Im_2',\ \Im_2' \ \text{real} \\
\Im_2' &= a_1 b_3 - a_3 b_1 = \begin{vmatrix} a_1 & b_1 \\ a_3 & b_3 \end{vmatrix} \\
i\Im_2'' &= a_2 \bar{b}_2 - \bar{a}_2 b_2 = \begin{vmatrix} a_2 & b_2 \\ \bar{a}_2 & \bar{b}_2 \end{vmatrix} = i\left(-2\begin{vmatrix} \mathrm{Re}(a_2) & \mathrm{Re}(b_2) \\ \mathrm{Im}(a_2) & \mathrm{Im}(b_2) \end{vmatrix}\right) \\
\Im &= \Im_2^2 + 4\bar{\Im}_1\Im_3.
\end{aligned}
$$

The First Result

Recall, $\Im = \Im_2^2 + 4\bar{\Im}_1\Im_3$.

Lemma

Suppose the pair $(\hat{A}, \hat{B})$ *is definite. Then*

(i) $\Im \geq 0$

(ii) The following statements are equivalent

(a) $\Im = 0$

(b) $\Im_1 = \Im_2 = \Im_3 = 0$

(c) $\sigma\hat{A} + \omega\hat{B} = 0$ *for some real* σ, ω, $|\sigma| + |\omega| > 0$.

The Second Result

Lemma

Let $(\hat{A}, \hat{B})$ *be definite and* $\Im > 0$. *Then*

$$\begin{aligned} &(i) && \alpha = 0 \quad \textit{iff} \quad \Im_3 = 0 \\ &(ii) && \beta = 0 \quad \textit{iff} \quad \Im_1 = 0 \\ &(iii) && \alpha = \beta = 0 \quad \textit{iff} \quad \Im_1 = \Im_3 = 0. \end{aligned}$$

The Third Result

Lemma

Suppose $(\hat{A}, \hat{B})$ *is definite and* $\Im > 0$. *Then the solution* (α, β) *of the system* $e_1 - e_2$ *is given by*

$$\alpha = \frac{\Im_3}{\nu}, \quad \beta = -\frac{\bar{\Im}_1}{\nu}, \tag{3}$$

where ν *is any nonzero solution of the equation*

$$\nu^2 - \Im_2 \nu - \bar{\Im}_1 \Im_3 = 0. \tag{4}$$

The General Solution

Theorem

Let the pair $(\hat{A}, \hat{B})$ be definite.

(i) *If* $\Im > 0$ *then* $\alpha = \frac{\Im_3}{\nu}, \quad \beta = -\frac{\bar{\Im}_1}{\nu},$
where ν *is any nonzero solution of* $\nu^2 - \Im_2\nu - \bar{\Im}_1\Im_3 = 0$

The General Solution

Theorem

Let the pair $(\hat{A}, \hat{B})$ be definite.

(i) *If* $\Im > 0$ *then* $\alpha = \dfrac{\Im_3}{\nu}, \quad \beta = -\dfrac{\bar{\Im}_1}{\nu},$
where ν *is any nonzero solution of* $\nu^2 - \Im_2 \nu - \bar{\Im}_1 \Im_3 = 0$

(ii) *If* $\Im = 0$ *then the equations in the system* e_1*–*e_2 *are proportional and there is infinite number of solutions.*

The General Solution

Theorem

Let the pair $(\hat{A}, \hat{B})$ *be definite.*

(i) *If* $\Im > 0$ *then* $\alpha = \frac{\Im_3}{\nu}, \quad \beta = -\frac{\bar{\Im}_1}{\nu}$,
where ν *is any nonzero solution of* $\nu^2 - \Im_2 \nu - \bar{\Im}_1 \Im_3 = 0$

(ii) *If* $\Im = 0$ *then the equations in the system* e_1–e_2 *are proportional and there is infinite number of solutions.*

(a) *Let* $\hat{A} \neq 0$. *If* $|a_1| + |a_2| > 0$ *then*

$$\alpha = -\frac{\bar{\gamma} a_3 + a_2}{a_1 + \bar{\gamma} \bar{a}_2}, \quad \beta = \gamma, \quad \gamma \in \{z \in \mathbf{C}; a_1 + \bar{z} a_2 \neq 0\}.$$

If $|a_2| + |a_3| > 0$ *then*

$$\alpha = \gamma, \quad \beta = -\frac{\bar{\gamma} a_1 + \bar{a}_2}{\bar{\gamma} a_2 + a_3}, \quad \gamma \in \{c \in \mathbf{C}; a_3 + \bar{z} a_2 \neq 0\}.$$

The General Solution

Theorem

Let the pair $(\hat{A}, \hat{B})$ be definite.

(i) *If* $\Im > 0$ *then* $\alpha = \frac{\Im_3}{\nu}, \quad \beta = -\frac{\bar{\Im}_1}{\nu},$
where ν *is any nonzero solution of* $\nu^2 - \Im_2 \nu - \bar{\Im}_1 \Im_3 = 0$

(ii) *If* $\Im = 0$ *then the equations in the system* e_1–e_2 *are proportional and there is infinite number of solutions.*

(a) *Let* $\hat{A} \neq 0$. *If* $|a_1| + |a_2| > 0$ *then*

$$\alpha = -\frac{\bar{\gamma} a_3 + a_2}{a_1 + \bar{\gamma}\bar{a}_2}, \quad \beta = \gamma, \quad \gamma \in \{z \in \mathbf{C};\, a_1 + \bar{z} a_2 \neq 0\}.$$

If $|a_2| + |a_3| > 0$ *then*

$$\alpha = \gamma, \quad \beta = -\frac{\bar{\gamma} a_1 + \bar{a}_2}{\bar{\gamma} a_2 + a_3}, \quad \gamma \in \{c \in \mathbf{C};\, a_3 + \bar{z} a_2 \neq 0\}.$$

(b) *Let* $\hat{B} \neq 0$. *Then the solutions are as in the case (a) provided that* a_1, a_2, a_3 *are replaced by* b_1, b_2, b_3, *resp.*

Designing CFL Algorithm

Some natural criteria that should be observed, especially when $\Im \approx 0$:

Designing CFL Algorithm

Some natural criteria that should be observed, especially when $\Im \approx 0$:

1. $|\alpha| + |\beta| \to$ min

Designing CFL Algorithm

Some natural criteria that should be observed, especially when $\Im \approx 0$:

1. $|\alpha| + |\beta| \to \min$

The first criterion ensures the smallest norm of the transformation matrix $\hat{F}$. It is important for the faster asymptotic convergence.

Designing CFL Algorithm

Some natural criteria that should be observed, especially when $\Im \approx 0$:

1. $|\alpha| + |\beta| \to \min$
2. $\alpha \cdot \beta = 0 \qquad (\Im = 0)$

The first criterion ensures the smallest norm of the transformation matrix $\hat{F}$. It is important for the faster asymptotic convergence.

Designing CFL Algorithm

Some natural criteria that should be observed, especially when $\Im \approx 0$:

1. $|\alpha| + |\beta| \to \text{min}$
2. $\alpha \cdot \beta = 0 \qquad (\Im = 0)$

The first criterion ensures the smallest norm of the transformation matrix $\hat{F}$. It is important for the faster asymptotic convergence.

The second criterion ensures the smallest flop count per step of the method.

Designing CFL Algorithm

Some natural criteria that should be observed, especially when $\Im \approx 0$:

1. $|\alpha| + |\beta| \to \min$
2. $\alpha \cdot \beta = 0 \qquad (\Im = 0)$
3. (α, β) is determined from the pivot submatrix of larger norm $(\Im = 0)$

The first criterion ensures the smallest norm of the transformation matrix $\hat{F}$. It is important for the faster asymptotic convergence.

The second criterion ensures the smallest flop count per step of the method.

Designing CFL Algorithm

Some natural criteria that should be observed, especially when $\Im \approx 0$:

1. $|\alpha| + |\beta| \to$ min
2. $\alpha \cdot \beta = 0 \qquad (\Im = 0)$
3. (α, β) is determined from the pivot submatrix of larger norm $(\Im = 0)$

The first criterion ensures the smallest norm of the transformation matrix $\hat{F}$. It is important for the faster asymptotic convergence.

The second criterion ensures the smallest flop count per step of the method.

The third criterion ensures that (α, β) is determined by a more reliable set of input data.

The Case $\Im > 0$: The Standard Solution

The theorem gives the solution:

$$\alpha = \frac{\Im_3}{\nu}, \qquad \beta = -\frac{\bar{\Im}_1}{\nu}$$

where ν is any nonzero solution of the equation

$$\nu^2 - \Im_2 \nu - \bar{\Im}_1 \Im_3 = 0.$$

The Case $\Im > 0$: The Standard Solution

The theorem gives the solution:

$$\alpha = \frac{\Im_3}{\nu}, \qquad \beta = -\frac{\bar{\Im}_1}{\nu}$$

where ν is any nonzero solution of the equation

$$\nu^2 - \Im_2 \nu - \bar{\Im}_1 \Im_3 = 0.$$

Respecting the first criterion we choose larger (by absolute value) ν:

$$\nu = \frac{\Im_2' + \imath \Im_2'' + \mathrm{sgn}(\Im_2')\sqrt{\Im}}{2}.$$

The Case $\Im > 0$: The Standard Solution

The theorem gives the solution:

$$\alpha = \frac{\Im_3}{\nu}, \qquad \beta = -\frac{\bar{\Im}_1}{\nu}$$

where ν is any nonzero solution of the equation

$$\nu^2 - \Im_2 \nu - \bar{\Im}_1 \Im_3 = 0.$$

Respecting the first criterion we choose larger (by absolute value) ν:

$$\nu = \frac{\Im_2' + \imath \Im_2'' + \mathrm{sgn}(\Im_2')\sqrt{\Im}}{2}.$$

This is referred to as the standard solution.

The Case $\Im = 0$

We have $\Im_1 = \Im_2 = \Im_3 = 0$ and $s\hat{A} + t\hat{B} = 0$, real s, t, $|s| + |t| > 0$.

The Case $\Im = 0$

We have $\Im_1 = \Im_2 = \Im_3 = 0$ and $s\hat{A} + t\hat{B} = 0$, real s, t, $|s| + |t| > 0$.

The standard solution does not exists.

The Case $\Im = 0$

We have $\Im_1 = \Im_2 = \Im_3 = 0$ and $s\hat{A} + t\hat{B} = 0$, real s, t, $|s| + |t| > 0$.

The standard solution does not exists.

The theorem and the three criteria imply the following solution:

if $|a_1| + |b_1| \geq |a_3| + |b_3|$ **then** $\beta = 0, \quad \alpha = -\dfrac{a_2}{a_1} \quad \left(= -\dfrac{b_2}{b_1}\right),$

else $\alpha = 0, \quad \beta = -\dfrac{\bar{a}_2}{a_3} \quad \left(= -\dfrac{\bar{b}_2}{b_3}\right)$

end

The Case $\Im = 0$

We have $\Im_1 = \Im_2 = \Im_3 = 0$ and $s\hat{A} + t\hat{B} = 0$, real s, t, $|s| + |t| > 0$.

The standard solution does not exists.

The theorem and the three criteria imply the following solution:

$$\textbf{if} \quad |a_1| + |b_1| \geq |a_3| + |b_3| \quad \textbf{then} \quad \beta = 0, \qquad \alpha = -\frac{a_2}{a_1} \quad \left(= -\frac{b_2}{b_1} \right),$$

$$\textbf{else} \quad \alpha = 0, \qquad \beta = -\frac{\bar{a}_2}{a_3} \quad \left(= -\frac{\bar{b}_2}{b_3} \right)$$

end

The probability for $\Im = 0$ is zero. We have to consider the case $\Im \approx 0$.

The Case $\Im \approx 0$

Let $\Im_1 = \Im_1' + \imath \Im_1''$, $\Im_3 = \Im_3' + \imath \Im_3''$, $a_2 = a_2' + \imath a_2''$, $b_2 = b_2' + \imath b_2''$.

$$\begin{aligned}
|\Im| &= |(\Im_2' - \Im_2'')(\Im_2' + \Im_2'') + 4\mathrm{Re}(\bar{\Im}_1 \Im_3)| \\
&\leq \max\{(\Im_2')^2\,,\,(\Im_2'')^2\} + 4|\Im_1'\Im_3' + \Im_1''\Im_3''| \\
&\leq \max\{(|a_1 b_3| + |b_1 a_3|)^2, 4(|a_2' b_2''| + |a_2'' b_2'|)^2\} + \\
&\quad 4[|a_1 a_3||b_2|^2 + |b_1 b_3||a_2|^2 + (|a_1 b_3| + |b_1 a_3|)(|a_2' b_2'| + |a_2'' b_2''|)|] \\
&\equiv \varrho.
\end{aligned}$$

The Case $\Im \approx 0$

Let $\Im_1 = \Im_1' + \imath\Im_1''$, $\Im_3 = \Im_3' + \imath\Im_3''$, $a_2 = a_2' + \imath a_2''$, $b_2 = b_2' + \imath b_2''$.

$$\begin{aligned}
|\Im| &= |(\Im_2' - \Im_2'')(\Im_2' + \Im_2'') + 4\mathrm{Re}(\bar{\Im}_1 \Im_3)| \\
&\leq \max\{(\Im_2')^2\,,\,(\Im_2'')^2\} + 4|\Im_1'\Im_3' + \Im_1''\Im_3''| \\
&\leq \max\{(|a_1 b_3| + |b_1 a_3|)^2, 4(|a_2' b_2''| + |a_2'' b_2'|)^2\} + \\
&\quad 4[|a_1 a_3||b_2|^2 + |b_1 b_3||a_2|^2 + (|a_1 b_3| + |b_1 a_3|)(|a_2' b_2'| + |a_2'' b_2''|)|] \\
&\equiv \varrho.
\end{aligned}$$

- ϱ is a reasonable upper bound for $|\mathrm{fl}(\Im)|$

The Case $\Im \approx 0$

Let $\Im_1 = \Im_1' + \imath \Im_1''$, $\Im_3 = \Im_3' + \imath \Im_3''$, $a_2 = a_2' + \imath a_2''$, $b_2 = b_2' + \imath b_2''$.

$$\begin{aligned} |\Im| &= |(\Im_2' - \Im_2'')(\Im_2' + \Im_2'') + 4\mathrm{Re}(\bar{\Im}_1 \Im_3)| \\ &\leq \max\{(\Im_2')^2, (\Im_2'')^2\} + 4|\Im_1'\Im_3' + \Im_1''\Im_3''| \\ &\leq \max\{(|a_1 b_3| + |b_1 a_3|)^2, 4(|a_2' b_2''| + |a_2'' b_2'|)^2\} + \\ &\quad 4[|a_1 a_3||b_2|^2 + |b_1 b_3||a_2|^2 + (|a_1 b_3| + |b_1 a_3|)(|a_2' b_2'| + |a_2'' b_2''|)|] \\ &\equiv \varrho. \end{aligned}$$

- ϱ is a reasonable upper bound for $|\mathrm{fl}(\Im)|$
- Let ϵ be a modest multiple of $\mathbf{u}$ (say of $\mathbf{u} \leq \epsilon \leq 10\mathbf{u}$).

The Case $\Im \approx 0$

Let $\Im_1 = \Im_1' + \imath \Im_1''$, $\Im_3 = \Im_3' + \imath \Im_3''$, $a_2 = a_2' + \imath a_2''$, $b_2 = b_2' + \imath b_2''$.

$$
\begin{aligned}
|\Im| &= |(\Im_2' - \Im_2'')(\Im_2' + \Im_2'') + 4\mathrm{Re}(\bar{\Im}_1 \Im_3)| \\
&\leq \max\{(\Im_2')^2\,,\,(\Im_2'')^2\} + 4|\Im_1'\Im_3' + \Im_1''\Im_3''| \\
&\leq \max\{(|a_1 b_3| + |b_1 a_3|)^2, 4(|a_2' b_2''| + |a_2'' b_2'|)^2\} + \\
&\quad 4[|a_1 a_3||b_2|^2 + |b_1 b_3||a_2|^2 + (|a_1 b_3| + |b_1 a_3|)(|a_2' b_2'| + |a_2'' b_2''|)|] \\
&\equiv \varrho.
\end{aligned}
$$

- ϱ is a reasonable upper bound for $|\mathrm{fl}(\Im)|$
- Let ϵ be a modest multiple of $\mathbf{u}$ (say of $\mathbf{u} \leq \epsilon \leq 10\mathbf{u}$).
- If $\mathrm{fl}(\Im) < -\varrho\epsilon$ we consider (A, B) not definite and abort comput.

The Case $\Im \approx 0$

Let $\Im_1 = \Im_1' + \imath \Im_1''$, $\Im_3 = \Im_3' + \imath \Im_3''$, $a_2 = a_2' + \imath a_2''$, $b_2 = b_2' + \imath b_2''$.

$$\begin{aligned} |\Im| &= |(\Im_2' - \Im_2'')(\Im_2' + \Im_2'') + 4\mathrm{Re}(\bar{\Im}_1 \Im_3)| \\ &\leq \max\{(\Im_2')^2, (\Im_2'')^2\} + 4|\Im_1'\Im_3' + \Im_1''\Im_3''| \\ &\leq \max\{(|a_1 b_3| + |b_1 a_3|)^2, 4(|a_2' b_2''| + |a_2'' b_2'|)^2\} + \\ &\quad 4[|a_1 a_3||b_2|^2 + |b_1 b_3||a_2|^2 + (|a_1 b_3| + |b_1 a_3|)(|a_2' b_2'| + |a_2'' b_2''|)|] \\ &\equiv \varrho. \end{aligned}$$

- ϱ is a reasonable upper bound for $|\mathrm{fl}(\Im)|$
- Let ϵ be a modest multiple of $\mathbf{u}$ (say of $\mathbf{u} \leq \epsilon \leq 10\mathbf{u}$).
- If $\mathrm{fl}(\Im) < -\varrho\epsilon$ we consider (A, B) not definite and abort comput.
- If $\varrho\epsilon^2 \leq \mathrm{fl}(\Im)$, we employ the standard solution for α, β.

The Case $\Im \approx 0, \quad \mathrm{fl}(\Im) \in (-\varrho\epsilon^2, \varrho\epsilon^2)$

If $\mathrm{fl}(\Im) \in (0, \varrho\epsilon^2)$, then severe cancelations take place and the computed ν, α and β will have large relative errors.

The Case $\Im \approx 0, \quad \text{fl}(\Im) \in (-\varrho\epsilon^2, \varrho\epsilon^2)$

If $\text{fl}(\Im) \in (0, \varrho\epsilon^2)$, then severe cancelations take place and the computed ν, α and β will have large relative errors.

If $\text{fl}(\Im) \in (-\varrho\epsilon^2, 0)$ we can still speculate that the rounding errors have caused $\text{fl}(\Im)$ to be negative. How to compute the solution (α, β)?

The Case $\Im \approx 0, \quad \mathrm{fl}(\Im) \in (-\varrho\epsilon^2, \varrho\epsilon^2)$

If $\mathrm{fl}(\Im) \in (0, \varrho\epsilon^2)$, then severe cancelations take place and the computed ν, α and β will have large relative errors.

If $\mathrm{fl}(\Im) \in (-\varrho\epsilon^2, 0)$ we can still speculate that the rounding errors have caused $\mathrm{fl}(\Im)$ to be negative. How to compute the solution (α, β)?

We can assume $\alpha\beta = 0$. Let $\beta = 0$. Then the equations

$$\begin{aligned} e_1 &= a_1\alpha + a_3\bar{\beta} + \bar{a}_2\alpha\bar{\beta} + a_2 = 0 \\ e_2 &= b_1\alpha + b_3\bar{\beta} + \bar{b}_2\alpha\bar{\beta} + b_2 = 0 \end{aligned}$$

become

The Case $\Im \approx 0, \quad \mathrm{fl}(\Im) \in (-\varrho\epsilon^2, \varrho\epsilon^2)$

If $\mathrm{fl}(\Im) \in (0, \varrho\epsilon^2)$, then severe cancelations take place and the computed ν, α and β will have large relative errors.

If $\mathrm{fl}(\Im) \in (-\varrho\epsilon^2, 0)$ we can still speculate that the rounding errors have caused $\mathrm{fl}(\Im)$ to be negative. How to compute the solution (α, β)?

We can assume $\alpha\beta = 0$. Let $\beta = 0$. Then the equations

$$\begin{aligned} e_1 &= a_1\alpha + a_3\bar{\beta} + \bar{a}_2\alpha\bar{\beta} + a_2 = 0 \\ e_2 &= b_1\alpha + b_3\bar{\beta} + \bar{b}_2\alpha\bar{\beta} + b_2 = 0 \end{aligned}$$

become

$$\begin{aligned} e_1 &= a_1\alpha + a_2 = 0 \\ e_2 &= b_1\alpha + b_2 = 0 \end{aligned}$$

and we can look for the least square (LS) solution.

The Case $\Im \approx 0, \quad \beta = 0$

Let $\tilde{a}_1 = \sqrt{a_1^2 + b_1^2}$, $c_1 = a_1/\tilde{a}_1$, $s_1 = b_1/\tilde{a}_1$. We obtain

$$\begin{aligned}
\| \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \alpha + \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \|_2^2 &= \| \begin{bmatrix} \tilde{a}_1 \\ 0 \end{bmatrix} \alpha + \begin{bmatrix} c_1 & s_1 \\ -s_1 & c_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \|_2^2 \\
&= \left| \tilde{a}_1 \alpha + \frac{a_1 a_2 + b_1 b_2}{\tilde{a}_1} \right|^2 + \frac{|\Im_1|^2}{a_1^2 + b_1^2},
\end{aligned}$$

The Case $\Im \approx 0, \quad \beta = 0$

Let $\tilde{a}_1 = \sqrt{a_1^2 + b_1^2}$, $c_1 = a_1/\tilde{a}_1$, $s_1 = b_1/\tilde{a}_1$. We obtain

$$\left\| \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \alpha + \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \right\|_2^2 = \left\| \begin{bmatrix} \tilde{a}_1 \\ 0 \end{bmatrix} \alpha + \begin{bmatrix} c_1 & s_1 \\ -s_1 & c_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \right\|_2^2 = \left| \tilde{a}_1 \alpha + \frac{a_1 a_2 + b_1 b_2}{\tilde{a}_1} \right|^2 + \frac{|\Im_1|^2}{a_1^2 + b_1^2},$$

where $\| \cdot \|_2$ stands for the Euclidean vector norm. The solution is

$$\alpha = -\frac{a_1 a_2 + b_1 b_2}{a_1^2 + b_1^2} \qquad \text{with the residual error} \qquad \frac{|\Im_1|}{\sqrt{a_1^2 + b_1^2}}.$$

The Case $\Im \approx 0$, the LS solution

The case $\alpha = 0$ is treated in the similar way. We obtain

$$\beta = -\frac{a_3 \bar{a}_2 + b_3 \bar{b}_2}{a_3^2 + b_3^2} \qquad \text{with the residual error} \qquad \frac{|\Im_3|}{\sqrt{a_3^2 + b_3^2}},$$

The Case $\Im \approx 0$, the LS solution

The case $\alpha = 0$ is treated in the similar way. We obtain

$$\beta = -\frac{a_3 \bar{a}_2 + b_3 \bar{b}_2}{a_3^2 + b_3^2} \qquad \text{with the residual error} \qquad \frac{|\Im_3|}{\sqrt{a_3^2 + b_3^2}},$$

This leads us to the following algorithm:

$$\textbf{if} \quad \frac{|\Im_1|}{\sqrt{a_1^2 + b_1^2}} \leq \frac{|\Im_3|}{\sqrt{a_3^2 + b_3^2}} \quad \textbf{then} \quad \alpha = -\frac{a_1 a_2 + b_1 b_2}{a_1^2 + b_1^2}, \quad \beta = 0$$

$$\textbf{else} \quad \alpha = 0, \quad \beta = -\frac{a_3 \bar{a}_2 + b_3 \bar{b}_2}{a_3^2 + b_3^2}$$

endif

The Case $\Im \approx 0$, the LS solution

The case $\alpha = 0$ is treated in the similar way. We obtain

$$\beta = -\frac{a_3\bar{a}_2 + b_3\bar{b}_2}{a_3^2 + b_3^2} \qquad \text{with the residual error} \qquad \frac{|\Im_3|}{\sqrt{a_3^2 + b_3^2}},$$

This leads us to the following algorithm:

$$\textbf{if} \quad \frac{|\Im_1|}{\sqrt{a_1^2 + b_1^2}} \le \frac{|\Im_3|}{\sqrt{a_3^2 + b_3^2}} \quad \textbf{then} \quad \alpha = -\frac{a_1 a_2 + b_1 b_2}{a_1^2 + b_1^2}, \quad \beta = 0$$

$$\textbf{else} \quad \alpha = 0, \quad \beta = -\frac{a_3\bar{a}_2 + b_3\bar{b}_2}{a_3^2 + b_3^2}$$

endif

Since $(\hat{A}, \hat{B})$ is definite, we should have $a_1^2 + b_1^2 > 0$ and $a_3^2 + b_3^2 > 0$.

Toward the Complex Falk-Langemeyer Algorithm

Moving from 2×2 to $n \times n$ GEP. We are dealing with an iterative process.

Toward the Complex Falk-Langemeyer Algorithm

Moving from 2×2 to $n \times n$ GEP. We are dealing with an iterative process.

Notation: k numbers iterations $(k = 0, 1, 2, \ldots)$

$(1, 2) \longrightarrow (i, j) = (i(k), j(k))$ pivot pair in step k

$\left(\hat{A}, \hat{B}\right) \longrightarrow \left(\hat{A}_{ij}^{(k)}, \hat{B}_{ij}^{(k)}\right)$

$a_1,\ a_2,\ a_3 \longrightarrow a_{ii}^{(k)},\ a_{ij}^{(k)},\ a_{jj}^{(k)}, \qquad b_1,\ b_2,\ b_3 \longrightarrow b_{ii}^{(k)},\ b_{ij}^{(k)},\ b_{jj}^{(k)}$

$\Im_1,\ \Im_3 \longrightarrow \Im_i^{(k)},\ \Im_j^{(k)},$

$\Im_2 = \Im_2' + \imath \Im_2'' \longrightarrow \Im_{ij}^{(k)} = \mathrm{Re}(\Im_{ij}^{(k)}) + \imath \mathrm{Im}(\Im_{ij}^{(k)})$

Pivot strategy:

Toward the Complex Falk-Langemeyer Algorithm

Moving from 2×2 to $n \times n$ GEP. We are dealing with an iterative process.

Notation: k numbers iterations $(k = 0, 1, 2, \ldots)$

$(1, 2) \longrightarrow (i, j) = (i(k), j(k))$ pivot pair in step k

$$\left(\hat{A}, \hat{B}\right) \longrightarrow \left(\hat{A}_{ij}^{(k)}, \hat{B}_{ij}^{(k)}\right)$$

$$a_1,\ a_2,\ a_3 \longrightarrow a_{ii}^{(k)},\ a_{ij}^{(k)},\ a_{jj}^{(k)}, \qquad b_1,\ b_2,\ b_3 \longrightarrow b_{ii}^{(k)},\ b_{ij}^{(k)},\ b_{jj}^{(k)}$$

$$\Im_1,\ \Im_3 \longrightarrow \Im_i^{(k)},\ \Im_j^{(k)},$$

$$\Im_2 = \Im_2' + \imath \Im_2'' \longrightarrow \Im_{ij}^{(k)} = \mathrm{Re}(\Im_{ij}^{(k)}) + \imath \mathrm{Im}(\Im_{ij}^{(k)})$$

Pivot strategy: assume the serial one, say, the row-cyclic one

The Complex Falk-Langemeyer Method

Input data: $A = A^*$, $B = B^*$ of order n and the logical variable *eivec*

The Complex Falk-Langemeyer Method

Input data: $A = A^*$, $B = B^*$ of order n and the logical variable *eivec*

Output data: the diagonal matrices A and B obtained by the method and, if $eivec = \texttt{true}$, the matrix F of the eigenvectors of (A, B).

The Complex Falk-Langemeyer Method

Input data: $A = A^*$, $B = B^*$ of order n and the logical variable *eivec*

Output data: the diagonal matrices A and B obtained by the method and, if $eivec = \texttt{true}$, the matrix F of the eigenvectors of (A, B).

1. Set $k = 0$, $A^{(k)} = A$, $B^{(k)} = B$. If *eivec* then set $F^{(k)} = I_n$

The Complex Falk-Langemeyer Method

Input data: $A = A^*$, $B = B^*$ of order n and the logical variable *eivec*

Output data: the diagonal matrices A and B obtained by the method and, if *eivec* $=$ `true`, the matrix F of the eigenvectors of (A, B).

1. Set $k = 0$, $A^{(k)} = A$, $B^{(k)} = B$. If *eivec* then set $F^{(k)} = I_n$
2. Repeat
   - (a) Choose the pivot pair $(i, j) = (i(k), j(k))$
   - (b) Compute the parameters (α_k, β_k) of F_k
   - (c) Compute $A^{(k+1)} = F_k^* A^{(k)} F_k$, $B^{(k+1)} = F_k^* B^{(k)} F_k$

     if *eivec* then compute $F^{(k+1)} = F^{(k)} F_k$.

   Until convergence

One Step of the CFL Method: **2(b)**-part

The superscipt (k) is omitted, $\quad \mathbf{u}$ is the unit round-off

One Step of the CFL Method: **2(b)**-part

The superscipt (k) is omitted, $\mathbf{u}$ is the unit round-off
$job = -1$ indicates that the computation should be terminated

One Step of the CFL Method: **2(b)**-part

The superscipt (k) is omitted, $\quad \mathbf{u}$ is the unit round-off
$job = -1$ indicates that the computation should be terminated
Notation: $a'_{ij} = \mathrm{Re}(a_{ij})$, $a''_{ij} = \mathrm{Im}(a_{ij})$, $b'_{ij} = \mathrm{Re}(b_{ij})$, $b''_{ij} = \mathrm{Im}(bij)$

One Step of the CFL Method: **2(b)**-part

The superscipt (k) is omitted, $\mathbf{u}$ is the unit round-off
$job = -1$ indicates that the computation should be terminated
Notation: $a'_{ij} = \mathrm{Re}(a_{ij})$, $a''_{ij} = \mathrm{Im}(a_{ij})$, $b'_{ij} = \mathrm{Re}(b_{ij})$, $b''_{ij} = \mathrm{Im}(bij)$

if $|a_{ij}| + |b_{ij}| = 0$ **then** $\alpha = \beta = 0$ **else**

One Step of the CFL Method: **2(b)**-part

The superscipt (k) is omitted, **u** is the unit round-off
$job = -1$ indicates that the computation should be terminated
Notation: $a'_{ij} = \mathrm{Re}(a_{ij})$, $a''_{ij} = \mathrm{Im}(a_{ij})$, $b'_{ij} = \mathrm{Re}(b_{ij})$, $b''_{ij} = \mathrm{Im}(bij)$

if $|a_{ij}| + |b_{ij}| = 0$ **then** $\alpha = \beta = 0$ **else**

(i) Renormalize $\hat{A}$, $\hat{B}$ and compute:

$$\Im'_{ij} = a_{ii} b_{jj} - a_{jj}\, b_{ii}; \quad \Im''_{ij} = -2\,(a'_{ij}\, b''_{ij} - b'_{ij}\, a''_{ij}); \quad \Im_{ij} = \Im'_{ij} + \imath\, \Im''_{ij};$$

$$\Im_i = a_{ii}\, b_{ij} - a_{ij}\, b_{ii}; \quad \Im_j = a_{jj}\, b_{ij} - a_{ij}\, b_{jj};$$

$$\Im = (\Im'_{ij} - \Im''_{ij})\,(\Im'_{ij} + \Im''_{ij}) + 4\,\mathrm{Re}(\bar{\Im}_1\, \Im_3);$$

One Step of the CFL Method: **2(b)**-part

The superscipt (k) is omitted, $\mathbf{u}$ is the unit round-off
$job = -1$ indicates that the computation should be terminated
Notation: $a'_{ij} = \mathrm{Re}(a_{ij})$, $a''_{ij} = \mathrm{Im}(a_{ij})$, $b'_{ij} = \mathrm{Re}(b_{ij})$, $b''_{ij} = \mathrm{Im}(bij)$

if $|a_{ij}| + |b_{ij}| = 0$ **then** $\alpha = \beta = 0$ **else**

(i) Renormalize $\hat{A}$, $\hat{B}$ and compute:

$$\Im'_{ij} = a_{ii} b_{jj} - a_{jj}\, b_{ii}; \quad \Im''_{ij} = -2\,(a'_{ij}\, b''_{ij} - b'_{ij}\, a''_{ij}); \quad \Im_{ij} = \Im'_{ij} + \imath\, \Im''_{ij};$$

$$\Im_i = a_{ii}\, b_{ij} - a_{ij}\, b_{ii}; \quad \Im_j = a_{jj}\, b_{ij} - a_{ij}\, b_{jj};$$

$$\Im = (\Im'_{ij} - \Im''_{ij})\,(\Im'_{ij} + \Im''_{ij}) + 4\,\mathrm{Re}(\bar{\Im}_1\, \Im_3);$$

$$\varrho = \max\{(|a_{ii} b_{jj}| + |b_{ii} a_{jj}|)^2, 4(|a'_{ij} b''_{ij}| + |a''_{ij} b'_{ij}|)^2\} +$$

$$4\left[\, |a_{ii} a_{jj}| |b_{ij}|^2 + |b_{ii} b_{jj}| |a_{ij}|^2 + (|a_{ii} b_{jj}| + |b_{ii} a_{jj}|)(|a'_{ij} b'_{ij}| + |a''_{ij} b''_{ij}|)\,\right];$$

One Step of the CFL Method: **(b)**-part

(ii) Set $job = 0$;

If $\Im > \varrho \mathbf{u}^2$ then $\nu = \frac{1}{2}(\Im_{ij} + \mathrm{sgn}(\Im'_{ij})\sqrt{\Im})$, $\alpha = \frac{\Im_j}{\nu}$, $\beta = -\frac{\bar{\Im}_i}{\nu}$

elseif $\Im < -\varrho \mathbf{u}$ then $job = -1$

else if $|\Im_i| \sqrt{a_{jj}^2 + b_{jj}^2} \leq |\Im_j| \sqrt{a_{ii}^2 + b_{ii}^2}$

then $\alpha = -\frac{a_{ii}\, a_{ij} + b_{ii}\, b_{ij}}{a_{ii}^2 + b_{ii}^2}$, $\beta = 0$

else $\alpha = 0$, $\beta = -\frac{a_{jj}\, \bar{a}_{ij} + b_{jj}\, \bar{b}_{ij}}{a_{jj}^2 + b_{jj}^2}$

endif

endif

Properties of the CFL Method

Theorem

Let (A, B) be a definite pair of Hermitian matrices and let $(A^{(k)}, B^{(k)}), k \geq 0$ be the sequence of pairs generated by applying the CFL algorithm to (A, B). Then for each k the following assertions hold:

(i) F_k *is nonsingular*

(ii) $|\alpha_k \beta_k| \leq 1$

(iii) $|\alpha_k \beta_k| = 1$ *iff* $Re(\Im_{ij}^{(k)}) = 0$ *and* $|a_{ij}^{(k)}| + |b_{ij}^{(k)}| > 0.$

We also have $\alpha_k \beta_k = -1$ *iff* $\Im_{ij}^{(k)} = 0.$

Properties of the CFL Method

Theorem

Let (A, B) be a definite pair of Hermitian matrices and let $(A^{(k)}, B^{(k)}), k \geq 0$ be the sequence of pairs generated by applying the CFL algorithm to (A, B). Then for each k the following assertions hold:

(i) F_k *is nonsingular*

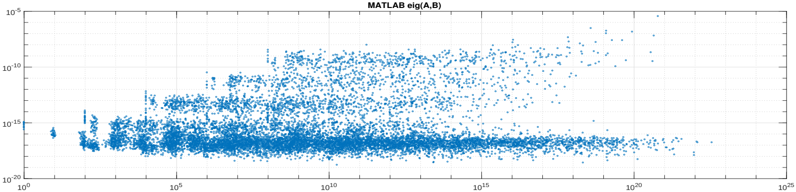
(ii) $|\alpha_k \beta_k| \leq 1$

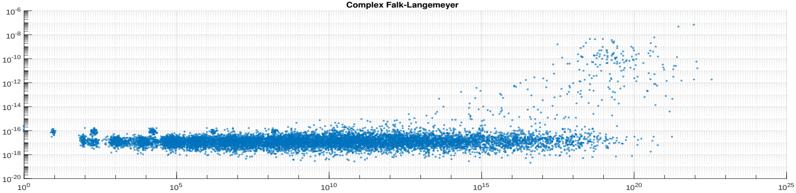
(iii) $|\alpha_k \beta_k| = 1 \quad \textit{iff} \quad Re(\Im_{ij}^{(k)}) = 0$ *and* $|a_{ij}^{(k)}| + |b_{ij}^{(k)}| > 0.$

We also have $\alpha_k \beta_k = -1 \quad \textit{iff} \quad \Im_{ij}^{(k)} = 0.$

Next we consider high relative accuracy (HRA) of the method!

Relative errors: *CFL* vs. MATLAB eig(A,B)





Theorem

Let $A = A^T \succ O$, $B = B^T \succ O$ *and* $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, $\lambda_i \in \sigma(A, B)$.
Let $A_S = D_A^{-1/2} A D_A^{-1/2}$, $B_S = D_B^{-1/2} B D_B^{-1/2}$, $D_A = diag(A)$, $D_B = diag(B)$

Let δA, δB *be symmetric perturbations and* $\tilde{\lambda}_1 \geq \tilde{\lambda}_2 \geq \cdots \geq \tilde{\lambda}_n$ *the eigenvalues of* $(A + \delta A, B + \delta B)$.

Let

$$\varepsilon_{A_S} = \|(\delta A)_S\|_2 / \|A_S\|_2, \quad \varepsilon_{B_S} = \|(\delta B)_S\|_2 / \|B_S\|_2$$

where

$$(\delta A)_S = D_A^{-1/2} \delta A D_A^{-1/2}, \quad (\delta B)_S = D_B^{-1/2} \delta B D_B^{-1/2} \ .$$

If

$$\varepsilon_{A_S} \kappa_2(A_S) < 1 \quad \textit{and} \quad \varepsilon_{B_S} \kappa_2(B_S) < 1,$$

then

$$\max_{1 \leq i \leq n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i} \leq \frac{\varepsilon_{A_S} \kappa_2(A_S) + \varepsilon_{B_S} \kappa_2(B_S)}{1 - \varepsilon_{B_S} \kappa_2(B_S)}.$$

Theoretical Background

- From the theorem we see that one class of "well-behaved matrix pairs" is made ofpairs of Hermitian positive definite matrices that can be well-scaled, i.e. for which $\kappa_2(A_S)$ and $\kappa_2(B_S)$ are small.
- For a well-behaved pair, the perturbations also have to be special, i.e. the numbers ε_{A_S} and ε_{B_S} have to be small. Then we shall have tiny relative errors.
- For those well-behaved pairs we have to find out what methods generate at every step only tiny relative errors $\varepsilon_{A_S^{(k)}}$, $\varepsilon_{B_S^{(k)}}$ and in the same time matrices with small or modest $\kappa_2(A_S^{(k)})$ and $\kappa_2(B^{(k)})$.

Nonetheless, this is a demanding task, so we shall go for a shortcut.

How to detect high relative accuracy of a method?

Recall the assertion of the theorem

$$\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i} \le \frac{\varepsilon_{A_S}\kappa_2(A_S) + \varepsilon_{B_S}\kappa_2(B_S)}{1 - \varepsilon_{B_S}\kappa_2(B_S)}, \qquad \text{it implies}$$

$$\varrho_{(A,B)} \equiv \frac{\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i}}{\sqrt{\kappa_2^2(A_S) + \kappa_2^2(B_S)}} \le \frac{\sqrt{\varepsilon_{A_S}^2 + \varepsilon_{B_S}^2}}{1 - \varepsilon_{B_S}\kappa_2(B_S)} \approx \max\{|\varepsilon_{A_S}|, |\varepsilon_{B_S}|\},$$

How to detect high relative accuracy of a method?

Recall the assertion of the theorem

$$\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i} \le \frac{\varepsilon_{A_S}\kappa_2(A_S) + \varepsilon_{B_S}\kappa_2(B_S)}{1 - \varepsilon_{B_S}\kappa_2(B_S)}, \qquad \text{it implies}$$

$$\varrho_{(A,B)} \equiv \frac{\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i}}{\sqrt{\kappa_2^2(A_S) + \kappa_2^2(B_S)}} \le \frac{\sqrt{\varepsilon_{A_S}^2 + \varepsilon_{B_S}^2}}{1 - \varepsilon_{B_S}\kappa_2(B_S)} \approx \max\{|\varepsilon_{A_S}|, |\varepsilon_{B_S}|\},$$

We can check numerically whether the inequality

$$\varrho_{(A,B)} \le f(n)\mathbf{u}, \tag{5}$$

holds for a larger sample Υ of well-behaved pairs (A, B)!

How to detect high relative accuracy of a method?

Recall the assertion of the theorem

$$\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i} \le \frac{\varepsilon_{A_S}\kappa_2(A_S) + \varepsilon_{B_S}\kappa_2(B_S)}{1 - \varepsilon_{B_S}\kappa_2(B_S)}, \qquad \text{it implies}$$

$$\varrho_{(A,B)} \equiv \frac{\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i}}{\sqrt{\kappa_2^2(A_S) + \kappa_2^2(B_S)}} \le \frac{\sqrt{\varepsilon_{A_S}^2 + \varepsilon_{B_S}^2}}{1 - \varepsilon_{B_S}\kappa_2(B_S)} \approx \max\{|\varepsilon_{A_S}|, |\varepsilon_{B_S}|\},$$

We can check numerically whether the inequality

$$\varrho_{(A,B)} \le f(n)\mathbf{u}, \tag{5}$$

holds for a larger sample Υ of well-behaved pairs (A, B)! Here

- $\tilde{\lambda}_i$ are the computed eigenvalues of (A, B)

How to detect high relative accuracy of a method?

Recall the assertion of the theorem

$$\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i} \le \frac{\varepsilon_{A_S}\kappa_2(A_S) + \varepsilon_{B_S}\kappa_2(B_S)}{1 - \varepsilon_{B_S}\kappa_2(B_S)}, \qquad \text{it implies}$$

$$\varrho_{(A,B)} \equiv \frac{\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i}}{\sqrt{\kappa_2^2(A_S) + \kappa_2^2(B_S)}} \le \frac{\sqrt{\varepsilon_{A_S}^2 + \varepsilon_{B_S}^2}}{1 - \varepsilon_{B_S}\kappa_2(B_S)} \approx \max\{|\varepsilon_{A_S}|, |\varepsilon_{B_S}|\},$$

We can check numerically whether the inequality

$$\varrho_{(A,B)} \le f(n)\mathbf{u}, \tag{5}$$

holds for a larger sample Υ of well-behaved pairs (A, B)! Here

- $\tilde{\lambda}_i$ are the computed eigenvalues of (A, B)
- $f(n)$ is a slowly growing function of n and $\mathbf{u}$ is the round off unit

How to detect high relative accuracy of a method?

Recall the assertion of the theorem

$$\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i} \le \frac{\varepsilon_{A_S}\kappa_2(A_S) + \varepsilon_{B_S}\kappa_2(B_S)}{1 - \varepsilon_{B_S}\kappa_2(B_S)}, \qquad \text{it implies}$$

$$\varrho_{(A,B)} \equiv \frac{\max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i}}{\sqrt{\kappa_2^2(A_S) + \kappa_2^2(B_S)}} \le \frac{\sqrt{\varepsilon_{A_S}^2 + \varepsilon_{B_S}^2}}{1 - \varepsilon_{B_S}\kappa_2(B_S)} \approx \max\{|\varepsilon_{A_S}|, |\varepsilon_{B_S}|\},$$

We can check numerically whether the inequality

$$\varrho_{(A,B)} \le f(n)\mathbf{u}, \tag{5}$$

holds for a larger sample Υ of well-behaved pairs (A, B)! Here

- $\tilde{\lambda}_i$ are the computed eigenvalues of (A, B)
- $f(n)$ is a slowly growing function of n and $\mathbf{u}$ is the round off unit
- Rel. (5) should not depend on $\kappa_2(A^{(0)})$ and $\kappa_2(B^{(0)})$.

Therefore, we are interested in how $\varrho_{(A,B)}$ behaves with respect to $\chi_{(A,B)}$,

$$\chi_{(A,B)} \equiv \kappa_2(A^{(0)}, B^{(0)}) = \sqrt{\kappa_2^2(A^{(0)}) + \kappa_2^2(B^{(0)})}.$$

- For the given sample of well behaved pairs Υ, and for each method, we shall make its graph of relative errors: $\mathcal{E}$,

$$\mathcal{E} = \{(\chi_{(A,B)}\ ,\ \varrho_{(A,B)})\ :\ (A,B) \in \Upsilon\}.$$

- Then we shall depict that graph $\mathcal{E}$ using the M-function
  `scatter(x,y,3)`

- The method will be indicated high relative accurate if the ordinates of the points on the graph are of order $\mathcal{O}(\mathbf{u})$ where $\mathbf{u} \approx 2.2 \cdot 10^{-16}$.

How to generate matrix pairs?

The starting pair $(A^{(0)}, B^{(0)})$ is generated by

- 4 the diagonal matrices : Δ_A, Δ_B, Σ, Δ and

How to generate matrix pairs?

The starting pair $(A^{(0)}, B^{(0)})$ is generated by

- 4 the diagonal matrices : Δ_A, Δ_B, Σ, Δ and
- 2 orthogonal matrices U, V of order n.

It is done in two steps:

1: $F = U\Sigma V^T, \quad A = F^T\Delta_A F, \quad B = F^T\Delta_B F,$

How to generate matrix pairs?

The starting pair $(A^{(0)}, B^{(0)})$ is generated by

- 4 the diagonal matrices : Δ_A, Δ_B, Σ, Δ and
- 2 orthogonal matrices U, V of order n.

It is done in two steps:

1: $F = U\Sigma V^T, \quad A = F^T \Delta_A F, \quad B = F^T \Delta_B F,$

2: $B^{(0)} = B_S = D_B^{-1/2} B D_B^{-1/2}, \quad A^{(0)} = \Delta A_S \Delta,\ A_S = D_A^{-1/2} A D_A^{-1/2},$

where D_A and D_B are the diagonal parts of A and B.

How to generate matrix pairs?

The starting pair $(A^{(0)}, B^{(0)})$ is generated by

- 4 the diagonal matrices : Δ_A, Δ_B, Σ, Δ and
- 2 orthogonal matrices U, V of order n.

It is done in two steps:

1: $F = U\Sigma V^T, \quad A = F^T \Delta_A F, \quad B = F^T \Delta_B F,$

2: $B^{(0)} = B_S = D_B^{-1/2} B D_B^{-1/2}, \quad A^{(0)} = \Delta A_S \Delta, \; A_S = D_A^{-1/2} A D_A^{-1/2},$

where D_A and D_B are the diagonal parts of A and B. Then $\kappa_2(A_S^{(0)})$ and $\kappa_2(B^{(0)})$ can be controlled by the diagonal elements of Δ_A, Δ_B, Σ,

How to generate matrix pairs?

The starting pair $(A^{(0)}, B^{(0)})$ is generated by

- 4 the diagonal matrices : Δ_A, Δ_B, Σ, Δ and
- 2 orthogonal matrices U, V of order n.

It is done in two steps:

1: $F = U\Sigma V^T, \quad A = F^T \Delta_A F, \quad B = F^T \Delta_B F,$

2: $B^{(0)} = B_S = D_B^{-1/2} B D_B^{-1/2}, \quad A^{(0)} = \Delta A_S \Delta,\ A_S = D_A^{-1/2} A D_A^{-1/2},$

where D_A and D_B are the diagonal parts of A and B. Then $\kappa_2(A_S^{(0)})$ and $\kappa_2(B^{(0)})$ can be controlled by the diagonal elements of Δ_A, Δ_B, Σ, since

$$\kappa_2(A_S^{(0)}) \leq n\kappa_2^2(\Sigma)\kappa_2(\Delta_A) \quad \text{and} \quad \kappa_2(B^{(0)}) \leq n\kappa_2^2(\Sigma)\kappa_2(\Delta_B),$$

although most often $\kappa_2(A_S^{(0)})$ and $\kappa_2(B^{(0)})$ are much smaller than these bounds.

How to generate matrix pairs?

To simplify the construction we set $\Delta_B = I_n$.

If the method is high relative accurate, then $\varrho_{(A,B)}$ from the relation (5) should not depend on $\kappa_2(\Delta)$.

Note that

$$\kappa_2(A^{(0)}) \leq \kappa_2(A_S^{(0)})\kappa_2^2(\Delta).$$

If we set $\Delta = I_n$ i $(A^{(0)}, B^{(0)}) = (D_B^{-1/2} A D_B^{-1/2}, B_S)$, then we know in advance the eigenvalues of $(A^{(0)}, B^{(0)})$ These are the quotients

$$(\Delta_A)_{jj}/(\Delta_B)_{jj}, \qquad 1 \leq j \leq n.$$

This way can be used when considering behavior of the methods on pairs with multiple eigenvalues.

More Details

- Diagonal matrices are constructed by help of the M-function `diag(d)`
- `d` is a vector, and vectors are constructed by the M-function `logspace(x1,x2,n)`. We use it for the diagonal matrices Σ and Δ_A.
- For the construction of Δ we use our m-function

  `scalvec(k1,k2,k3,n,k)`

  which generates vector of length n, $d = [10^{\mathrm{k1}}, \ldots, 10^{\mathrm{k2}}, \ldots, 10^{\mathrm{k3}}]$ where `k` determines the position of 10^{k2} within the components of d.
- To compute Δ, the function `scalvec` is used within triple loop controlled by the indices `k1`, `k2` and `k3`
- Orthogonal matrices U and V are computed by the command

  `[Q,~]=qr(rand(n))`
- We have generated the sample Υ of 18900 pairs of matrices of order 10. As "exact eigenvalues" we have used the eigenvalues computed by the M-function `eig(A,B)` in variable precision arithmetic (VPA) using 80 decimal digits.

Relative Accuracy

$$\varrho_{(A,B)} = \max_{1\le i\le n} \frac{|\tilde{\lambda}_i - \lambda_i|}{\lambda_i} / \sqrt{\kappa_2^2(A_S) + \kappa_2^2(B_S)} \le \frac{\sqrt{\varepsilon_{A_S}^2 + \varepsilon_{B_S}^2}}{1 - \varepsilon_{B_S}\kappa_2(B_S)}.$$

$$\chi_{(A,B)} = \sqrt{\kappa_2^2(A^{(0)}) + \kappa_2^2(B^{(0)})}\ .$$

$$\mathcal{E} = \{(\chi_{(A,B)}\ ,\ \varrho_{(A,B)}) : \ (A,B) \in \Upsilon\}.$$

Relative errors: *CFL* vs. MATLAB eig(A,B)