Analisi di matrici sparse e data-sparse: metodi numerici ed applicazioni

Partecipanti: Elena Addis (UniPG), Giovanni Barbarino (SNS), Dario Bini (UniPI), Paola Boito (UniPI), Gianna M. Del Corso (UniPI), Massimiliano Fasi (Manchester), Dario Fasino (UniUD), Luca Gemignani (UniPI), Bruno Iannazzo (UniPG), Stefano Massei (EPF Losanna), Nicola Mastronardi (IAC-CNR), Beatrice Meini (UniPI), Federico Poloni (UniPI), Francesco Tudisco (GSSI).

Visite scientifiche

- Peter Benner (MPI- Magdeburd)
- Fernando De Téran (Universidad Carlos III de Madrid)

- Equazioni e funzioni di matrici
- Metodi per il calcolo degli autovalori di matrici o pencil strutturati
- Proprietà ed algoritmi per matrici data-sparse
- Metodi numerici per problemi agli autovalori non lineari
- Analisi e risoluzione di problemi computazionali per modelli markoviani.

Pubblicazioni I

- S. Ahn and B. Meini. Matrix equations in markov modulated brownian motion: theoretical properties and numerical solution. Stochastic Models, 2019.

- F. Arrigo and F. Tudisco. Multi-dimensional, multilayer, nonlinear and dynamic hits. pages 369–377, 2019.

- G. Barbarino. Spectral measures. Springer INdAM Series, 30:1–24, 2019.

- R. Bevilacqua, G.M. Del Corso, and L. Gemignani. Fast qr iterations for unitary plus low rank matrices. Numerische Mathematik, 144(1):23–53, 2020.

- D. Bini, K. Jbilou, M. Mitrouli, and L. Reichel. Numerical analysis and scientific computation (nasca18). Journal of Computational and Applied Mathematics, 2019.

- D.A. Bini, S. Massei, and L. Robol. Quasi-toeplitz matrix arithmetic: a matlab toolbox. Numerical Algorithms, 81(2):741–769, 2019.

Pubblicazioni II

- D.A. Bini and B. Meini. On the exponential of semi-infinite quasi-toeplitz matrices. Numerische Mathematik, 141(2):319–351, 2019.
- D. Camps, N. Mastronardi, R. Vandebril, and P. Van Dooren. Swapping 2×2 blocks in the schur and generalized schur form. Journal of Computational and Applied Mathematics, 2019.
- S. Cipolla, M. Redivo-Zaglia, and F. Tudisco. Extrapolation methods for fixed-point multilinear pagerank computations. Numerical Linear Algebra with Applications, 2020.
- F. De Terán, B. Iannazzo, F. Poloni, and L. Robol. Nonsingular systems of generalized sylvester equations: An algorithmic approach. Numerical Linear Algebra with Applications, 26(5), 2019.
- G.M. Del Corso, F. Poloni, L. Robol, and R. Vandebril. Factoring block Fiedler companion matrices. Springer INdAM Series, 30:129–155, 2019.

Pubblicazioni III

- G.M. Del Corso, F. Poloni, L. Robol, and R. Vandebril. When is a matrix unitary or hermitian plus low rank? Numerical Linear Algebra with Applications, 26(6), 2019.

- G.M. Del Corso and F. Romani. Adaptive nonnegative matrix factorization and measure comparisons for recommender systems. Applied Mathematics and Computation, 354:164–179, 2019.

- M. Fasi. Optimality of the paterson–stockmeyer method for evaluating matrix polynomials and rational matrix functions. Linear Algebra and Its Applications, 574:182–200, 2019.

- M. Fasi and N.J. Higham. An arbitrary precision scaling and squaring algorithm for the matrix exponential. SIAM Journal on Matrix Analysis and Applications, 40(4):1233–1256, 2019.

- M. Fasi and B. Iannazzo. Computing primary solutions of equations involving primary matrix functions. Linear Algebra and Its Applications, 560:17–42, 2019.

Pubblicazioni IV

- D. Fasino and E.E. Tyrtyshnikov. Error analysis of tt-format tensor algorithms. Springer INdAM Series, 30:91–106, 2019.
- A. Gautier and F. Tudisco. The contractivity of cone-preserving multilinear mappings. Nonlinearity, 32(12):4713–4728, 2019.
- A. Gautier, F. Tudisco, and M. Hein. The perron-frobenius theorem for multihomogeneous mappings. SIAM Journal on Matrix Analysis and Applications, 40(3):1179–1205, 2019.
- A. Gautier, F. Tudisco, and M. Hein. A unifying perron-frobenius theorem for nonnegative tensors via multihomogeneous maps. SIAM Journal on Matrix Analysis and Applications, 40(3):1206–1231, 2019.
- B. Iannazzo, B. Jeuris, and F. Pompili. The derivative of the matrix geometric mean with an application to the nonnegative decomposition of tensor grids. Springer INdAM Series, 30:107–128, 2019.

Pubblicazioni V

- E. Jarlebring and F. Poloni. Iterative methods for the delay lyapunov equation with t-sylvester preconditioning. Applied Numerical Mathematics, 135:173–185, 2019.
- D. Kressner, P. Kürschner, and S. Massei. Low-rank updates and divide-and-conquer methods for quadratic matrix equations. Numerical Algorithms, 2019.
- D. Kressner, S. Massei, and L. Robol. Low-rank updates and a divide-and-conquer method for linear matrix equations. SIAM Journal on Scientific Computing, 41(2):A848–A876, 2019.
- S. Massei, M. Mazza, and L. Robol. Fast solvers for two-dimensional fractional diffusion equations using rank structured matrices. SIAM Journal on Scientific Computing, 41(4):A2627–A2656, 2019.
- N. Mastronardi, H. Taeter, and P.V. Dooren. On computing eigenvectors of symmetric tridiagonal matrices. Springer INdAM Series, 30:181–195, 2019.

Pubblicazioni VI

- P. Mercado, F. Tudisco, and M. Hein. Spectral clustering of signed graphs via matrix power means. volume 2019-June, pages 7940–7967, 2019.
- F. Poloni and G. Sbrana. Closed-form results for vector moving average models with a univariate estimation approach. Econometrics and Statistics, 10:27–52, 2019.
- H. Shojaei and D. Fasino. Isomorphism theorems in the primary categories of krasner hypermodules. Symmetry, 11(5), 2019.
- F. Tudisco and D.J. Higham. A fast and robust kernel optimization method for core–periphery detection in directed and weighted graphs. Applied Network Science, 4(1), 2019.

Backward stable methods for eigenvalue computation of perturbed unitary matrices

Gianna M. Del Corso
joint work with R. Bevilacqua and L. Gemignani

Dipartimento di Informatica, Università di Pisa

GNCS 2020

The problem

Efficient computation of the eigenvalues of perturbed unitary matrices

The most interesting case is related to the approximation of zeros of polynomials and eigenvalues of matrix polynomials.
In the scalar case:

- Let $p(z) = \sum_{i=0}^{n} a_i \phi_i(z)$
- $\Phi = \{\phi_0(z), \phi_1(z), \cdots, \phi_n(z)\}$ is a basis for Π_n.
- Two matrices A and B such that

$$p(z) = 0 \text{ related to } \det(A - z\,B) = 0$$

- computation of roots of polynomial equivalent to computation of eigenvalues of a matrix pencil

The problem

$p(z) = \sum_{i=0}^{n} a_i \phi_i(z)$

- If $\Phi = \{1, z, \dots, z^n\}$ we get the companion matrix.

$$A = \begin{bmatrix} -\frac{a_{n-1}}{a_n} & -\frac{a_{n-2}}{a_n} & \cdots & -\frac{a_0}{a_n} \\ 1 & 0 & \cdots & 0 \\ & \ddots & \ddots & \vdots \\ & & 1 & 0 \end{bmatrix}, \quad B = I$$

- Malab's command `roots` computes indeed the eigenvalues of the companion matrix
- Applying QR method we get a $O(n^3)$ complexity

Framework

- Fast and backwards stable method are mainly based on the representation fo companion matrices as unitary-plus rank-one

$$A = U + \mathbf{e}_1^T \mathbf{p}, \quad UU^T = I_n, \quad U \quad \text{Upper Hessenberg}$$

- This structure is preserved by QR iterations
- Using a data-sparse representation we can design $\mathcal{O}(n^2)$ algorithms
- We can obtain different structures using different basis

- $P(z) = A_0 + A_1 z + A_2 z^2 + \cdots + A_d z^d, \quad A_i \in \mathbb{C}^{k \times k}$
- In many applications we need to compute the eigenvalues and eigenvectors of $P(z)$, i.e. $\lambda \in \mathbb{C}$ and vector $\mathbf{x}$ such that

$$P(\lambda)\mathbf{x} = 0$$

- The problem is in general reduced to an equivalent generalized eigenproblem

$$L_0 \mathbf{v} = \lambda L_1 \mathbf{v}$$

  where $L_0, L_1 \in \mathbb{C}^{kd \times kd}$.
- Block-companion linearizations are unitary plus rank k matrices. Other structured linearizations possible

Related work

Most authors analyze the companion /block companion case... very little on the general unitary plus low rank case, mostly using QR iterations

- 2004 Bini, Daddi, Gemignani (explicit QR)
- 2007 Bini, Eidelman, Gemignani, Gohberg (explicit QR on unitary plus rank 1
- 2007 Chandrasekeran, Gu, Xia, Zhu (implicit QR on a QR factorization of the companion)
- 2012 Delvaux, Frederix, Van Barel (block companion, the matrix is stored using the Givens weight representation)
- 2015-2017 Aurentz, Mach, Robol, Vandebril, Watkins different papers where the representation uses only unitary matrices

The Problem

- Perfectly unitary structure corrupted by a low rank error
- Some arrowhead linearization
- Fellow matrices (whose eigenvalues are the zeros of a linear combination of Szegö polynomials)
- Diagonal plus low rank structures
- Finite truncation of matrices representing unitary operators on separable Hilbert spaces

Possible approaches

$A = G + UV^*$, $G \in \mathbb{C}^{n \times n}$, unitaria $U, V \in \mathbb{C}^{n \times k}$

- QR method
- Orthogonal iterations (if only a few eigenvalues are required)

For both approaches it is preferable to reduce the initial matrix to Hessenberg form.

We are looking for algorithms of cost is $O(n^2 k)$, where n is the dimension of the matrix and k is the rank of the correction.

R. Bevilacqua, G.D.C., L.Gemignani "Fast QR iterations for unitary plus low rank matrices", Numerische Mathematik, 2020.

R. Bevilacqua, G.D.C., L.Gemignani "Orthogonal iterations for the nonlinear eigenvalue problem", in preparation 2020.

Reduction to Hessenberg form

$A = G + UV^*$, $O(n^2k)$ algorithms only for particular classes of matrices, based on the structure of the unitary part

- G unitary block diagonal
- G unitary block Hessenberg
- G unitary block CMV and $U = [I_k, 0, \ldots, O]$

Not a problem if the eigenstructure of G is known.

CMV matrices are special unitary matrices with a block pentadiagonal structure (is not the definition) particularly important for the applications (related to orthogonal polynomials)

Effcient algorithms for dealing with block unitary diagonal plus low rank matrices have been developed by
L. Gemignani and L. Robol SIAM J. Matrix Anal. Appl. (2017).

An $\mathcal{O}(nk)$ representation

A matrix $A = G + UV^*$ is representable in LFR format if

- $A = LFR$
- $L = L_1 \cdots L_k$, each L_i unitary lower Hessenberg
- $R = R_1 \cdots R_k$, each R_i unitary upper Hessenberg
- $F = Q + [I_k, 0]Z^*$,

  $$Q = \begin{bmatrix} I_k & 0 \\ 0 & \hat{Q} \end{bmatrix}$$

  is unitary.

Any unitary Hessenberg matrix can be decomposed as a sequence of $n - 1$ Givens rotations acting on two consecutive rows

To represent L and R we need only $\mathcal{O}(nk)$ parameters. In the cases of interest also F is representable with $\mathcal{O}(nk)$ parameters.

Reduction to Hessenberg form

- Working on the LFR-representation
- We do not give the details in this talk (see "Efficient Reduction of compressed unitary plus low-rank matrices to Hessenberg form" by Bevilacqua, Del Corso, Gemignani, arxiv 2019).
- The reduction to Hessenberg form is done "embedding" the original matrix adding k null rows and k columns
- Represent the matrix in terms of $\mathcal{O}(nk)$ parameters, convenient for the iterations for the computation of the eigenvalues
- The algorithm costs $\mathcal{O}(n^2k)$ and is backward stable

Properties of the representation

After the embedding

$$\widehat{A} = \begin{bmatrix} A & * \\ 0_{k,n} & 0_{kk} \end{bmatrix} = \widehat{L}\widehat{F}\widehat{R}$$

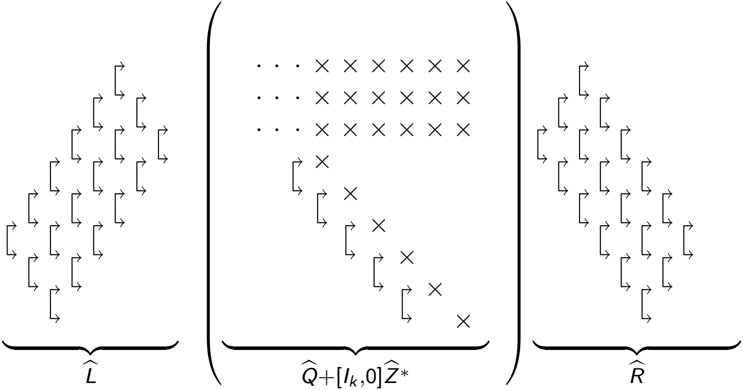
still unitary-plus-low-rank.

- $\widehat{F}$ inherits the bandwidth profile of A
- This is a key ingredient for eigenvalues iterations
- Each lower (upper) Hessenberg matrix can be decomposed as an ascending (descending) sequence of $n-1$ Givens rotations acting on two consecutive rows

$$\Gamma = G = \begin{bmatrix} c & s \\ -s & \overline{c} \end{bmatrix}, \quad |c|^2 + s^2 = 1, c \in \mathbb{C}, s \in \mathbb{R}.$$

Iterations on the structure

From a computational point of view it is preferable to work with A in Hessenberg form



The implicit QR algorithm

On upper Hessenberg matrices

- Inizialization phase:
  - Pick the shift μ,
  - Retrieve the vector $x = (A - \mu I)e_1$.
  - x has only two nonzeros.
  - Compute G_1 such that $G_1 x = \alpha e_1$.
  - Perform the similarity transformation $\tilde{A} = G_1 A G_1^H$.

$\tilde{A}$ is no more in Hessenberg form: a bulge is formed

- Chasing the bulge to restore the Hessenberg structure

No QR decomposition needed!! Only 2×2 Givens rotations

The implicit QR algorithm

On upper Hessenberg matrices

- Inizialization phase:
  - Pick the shift μ,
  - Retrieve the vector $x = (A - \mu I)e_1$.
  - x has only two nonzeros.
  - Compute G_1 such that $G_1 x = \alpha e_1$.
  - Perform the similarity transformation $\tilde{A} = G_1 A G_1^H$.

$\tilde{A}$ is no more in Hessenberg form: a bulge is formed

- Chasing the bulge to restore the Hessenberg structure

No QR decomposition needed!! Only 2×2 Givens rotations

Implicit Q Theorem $\rightarrow$ the same of an explicit QR step

Works fabulously with multiple shifts

The implicit QR algorithm

On upper Hessenberg matrices

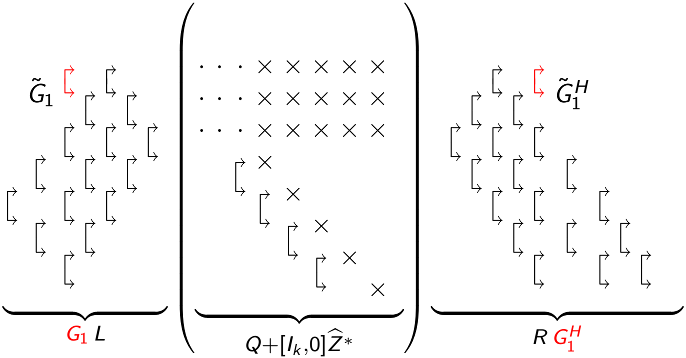
- Inizialization phase:
  - Pick the shift μ,
  - Retrieve the vector $x = (A - \mu I)e_1$.
  - x has only two nonzeros.
  - Compute G_1 such that $G_1 x = \alpha e_1$.
  - Perform the similarity transformation $\tilde{A} = G_1 A G_1^H$.

$\tilde{A}$ is no more in Hessenberg form: a bulge is formed

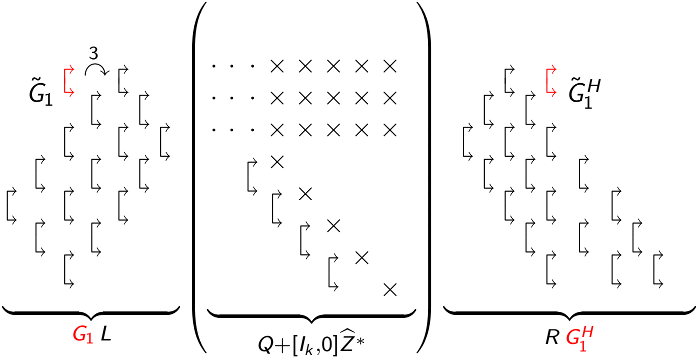
- Chasing the bulge to restore the Hessenberg structure

No QR decomposition needed!! Only 2×2 Givens rotations
Implicit Q Theorem $\rightarrow$ the same of an explicit QR step
Works fabulously with multiple shifts

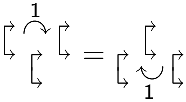
Inizialization Step



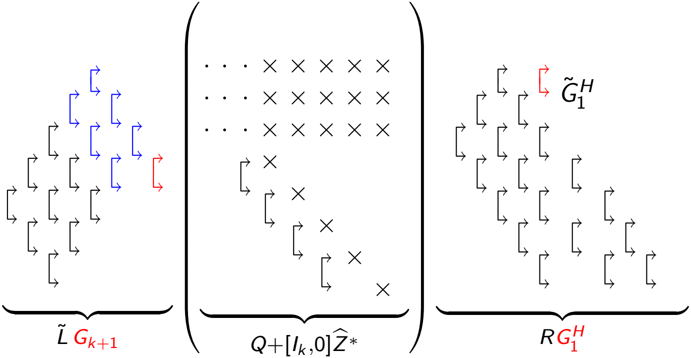
Inizialization Step



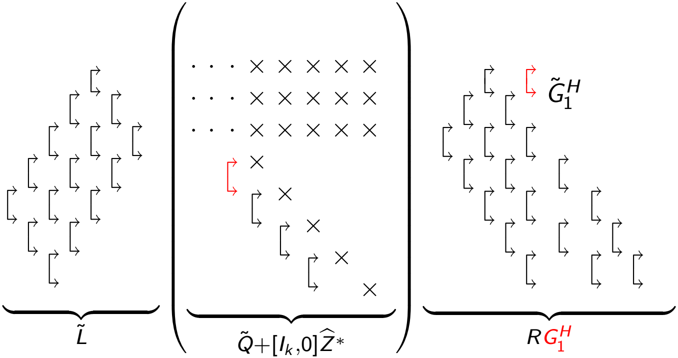
k turnovers



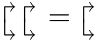
Inizialization Step



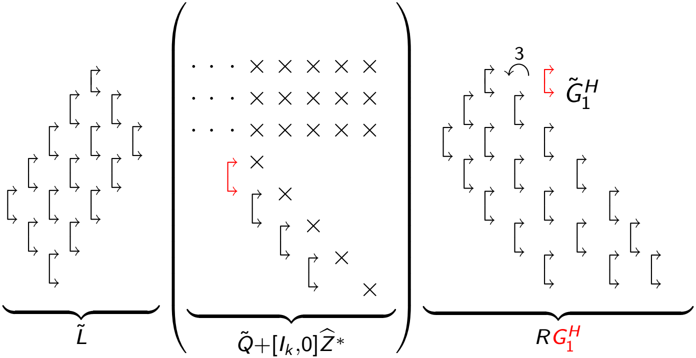
Inizialization Step



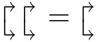
fusion



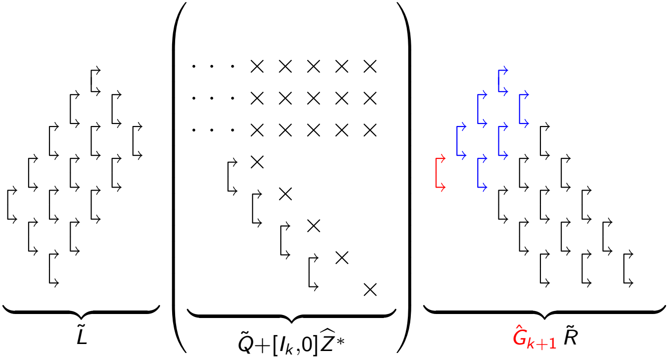
Inizialization Step



fusion

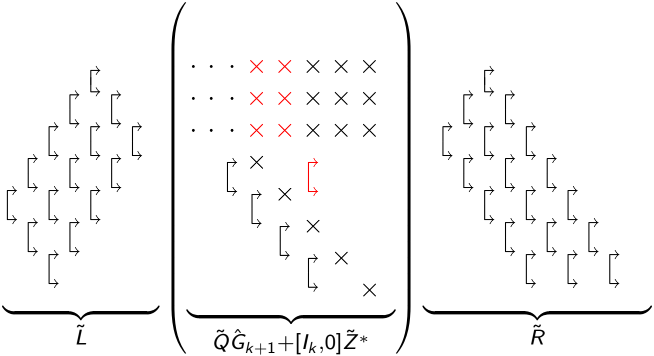


Inizialization Step

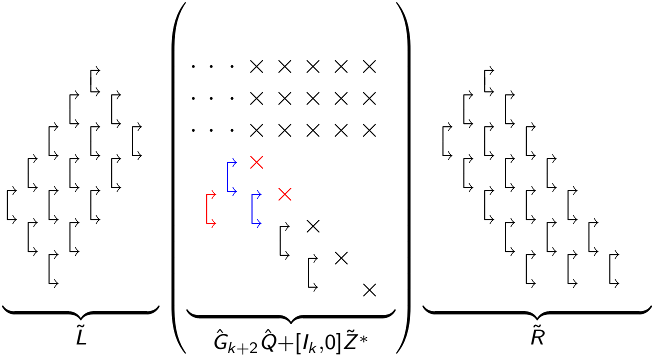


k turnovers

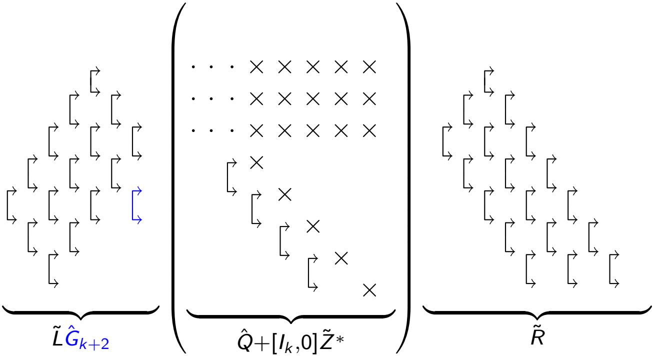
Inizialization Step



Inizialization Step

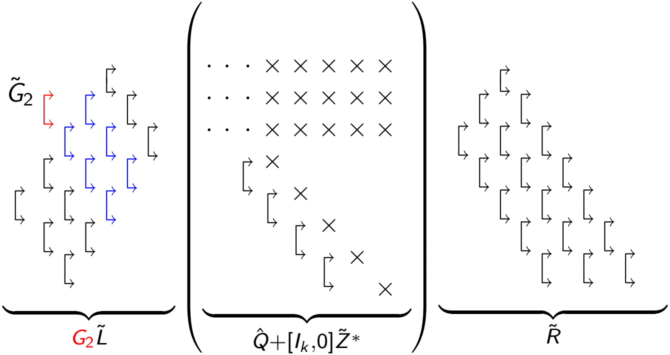


Inizialization Step



Move out - I_k is not affected since the Givens acts on rows $k+2$ and $k+3$

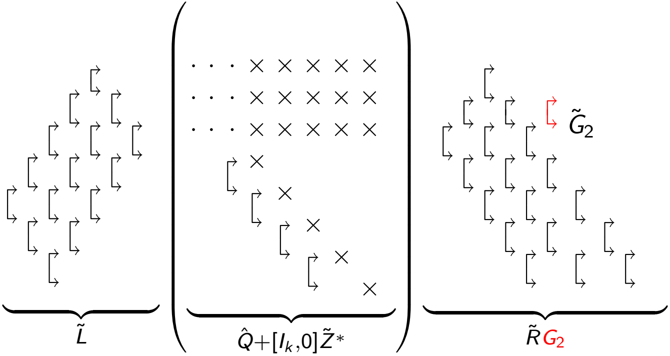
Inizialization Step



G_2 is the transformation which produces the bulge

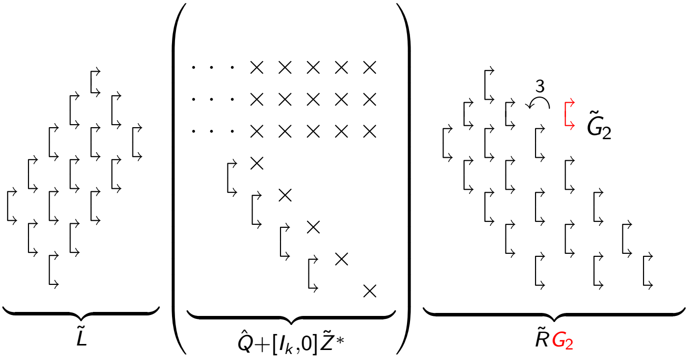
Chasing Steps

We have to chase the bulge form the top o the bottom until it gets absorbed with a fusion!

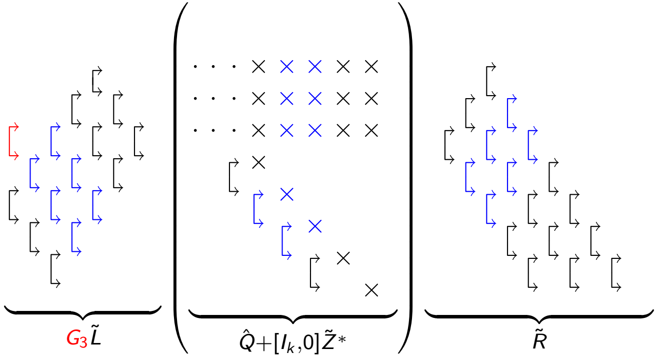


Chasing Steps

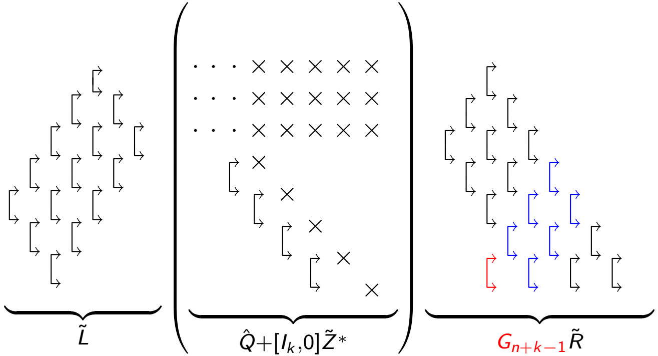
We have to chase the bulge form the top o the bottom until it gets absorbed with a fusion!



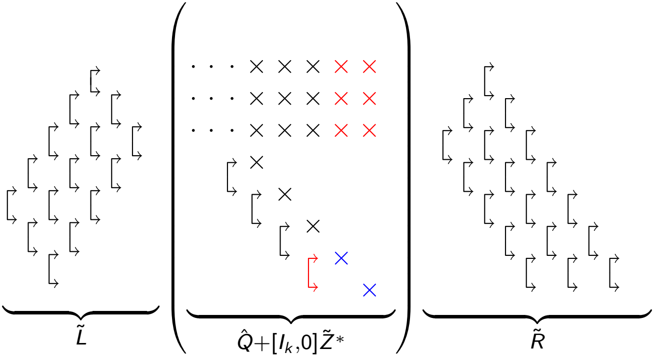
Chasing Steps



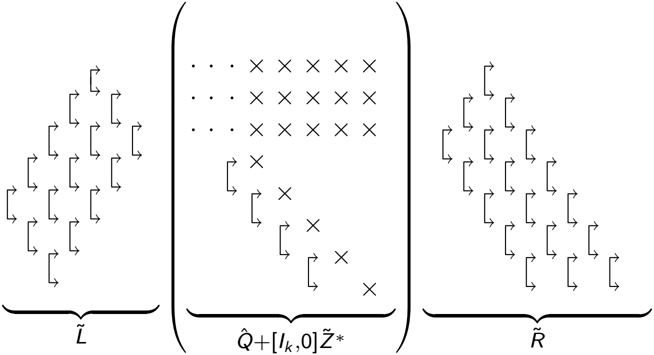
Final Step



Final Step



Final Step



The initial representation has been restored.

Deflation

- Deflations can be detected directly on the representation
- Because the outermost diagonal entries of L and R are always nonzero,
- $a_{i+1,i} = 0, i = 1, \ldots n-1$ iff Q_{i+k} is essentially the 2×2 identity matrix.
- $a_{i+1,i}$ is numerically zero if one of the Givens matrices of $\hat{Q}$ is "close" to a phase matrix.

Computational cost

For each step:

- Inizialization: 3k+1 turnovers + 1 fusion
- Chasing: (n-2) chasing steps, each one requiring 2k+1 turnovers
- Total of $\mathcal{O}(nk)$ turnovers per step, each one requiting a constant number of flops
- Total cost of the procedure $\mathcal{O}(n^2 k)$.

Applying it to the matrix polynomial eigenvalue problem, with $P(\lambda) = A_0 + A_1\lambda + \cdots + I_k\lambda^d$ and $A_i \in \mathbb{C}^{k\times k}$, we have $n = kd$, and we get a cost of $O(k^3 d^2)$ which is claimed to be the best achievable by implicit QR.

Backward Stability

- Backward stability of the representation:

$$\tilde{A} = A + E, \quad \|E\|_2 \approx \varepsilon \|A\|_2$$

- Backward stability of the QR steps:
  Let $A^{(1)} = P^* A P$ the matrix after one step of QR method.
  Let $\tilde{A}^{(1)}$ the matrix obtained in floating point arithmetic

$$\tilde{A}^{(1)} = P^*(A + \Delta_A)P, \quad \|\Delta_A\|_2 \leq O(\varepsilon)\|A\|_2.$$

Numerical Results

- Backward stability
- Cost of the eigenvalues computation, to show the dependence on n and on k.

Test suite

- Companion and block companion matrices associated to scalar and matrix polynomials
- Random unitary-plus-rank-k
- Random unitary-diagonal-plus-rank-k
- Random fellow matrices

Backward error

$$\mathrm{bw}_{\mathrm{err}}(A) = \frac{\|P_m A P_m^* - \tilde{A}^{(m)}\|_\infty}{\|A\|_\infty}$$

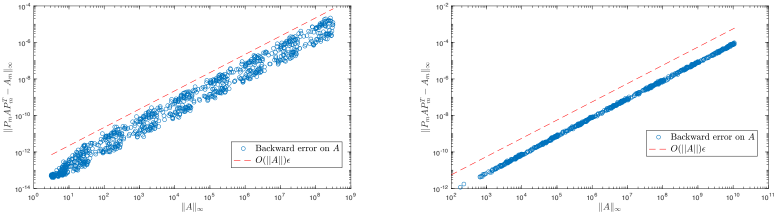


Figure: Absolute backward error in the computation of the eigenvalues respect to the norm of the matrix. On the left the results obtained from 1000 random unitary-plus-rank-5 matrices of size 50. On the right the absolute backward error is plotted against the norm of the matrix for one 1000 unitary diagonal-plus-rank-5 matrices of size 100.

Name	n	k	$\|\mathrm{ceig}(A)\|_\infty$	forw_{err}	back_{err}
acousticwave1d	20	10	5.96e+01	1.42e-15	1.02e-14
bicycle	4	2	5.70e+02	2.60e-15	8.29e-16
cdplayer	120	60	4.50e+03	5.17e-16	5.85e-15
closedloop	4	2	9.00e+00	8.99e-16	1.42e-15
dirac	160	80	2.11e+03	5.24e-14	1.39e-13
hospital	48	24	4.49e+01	7.84e-13	2.57e-14
metalstrip	18	9	1.71e+02	7.78e-16	2.42e-15
omnicam2	30	15	4.66e+17	1.16e-02	4.90e-15
powerplant	24	8	1.72e+05	7.13e-08	2.68e-15
sign1	162	81	3.29e+09	4.10e-09	5.84e-14
sign2	162	81	9.61e+02	4.27e-13	3.54e-14
spring	10	5	2.33e+00	3.00e-16	1.93e-15
wiresaw1	20	10	1.57e+01	6.00e-14	4.20e-15
butterfly	240	64	2.97e+01	5.15e-14	1.29e-13
orrsommerfeld	40	10	1.88e+06	1.83e-14	6.35e-15
plasmadrift	384	128	6.64e+04	1.02e-13	4.86e-14

Table: Results on the NLEVP collection.

Dependence on k and n

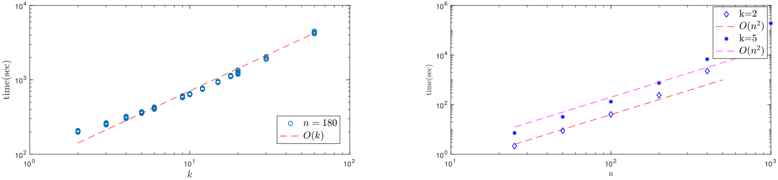


Figure: On the right the double logarithmic plot for random matrices of size 180 that are unitary-plus-rank-k with k ranging from 1 to 60. The reference line shows the linear dependence on k. On the right, for $k = 2$ and $k = 5$ and matrices of size ranging from 25 to 1000. The dashed lines represent the $O(n^2)$ slope.

The NEP Problem (non PEP)

$$T : \Omega \to \mathbb{C}^{k\times k}$$

(NEP) find the pair $(\lambda, v), v \neq 0$ such that $T(\lambda)v = 0$.

Consider $\Delta \in \Omega$, and find eigenvalues in Δ.

Possible approaches (see Güttel-Tisseur)

- Newton's method
- Contour integrals
- Approximation of the nonlinear function with a matrix polynomial
- Linearize to obtain a pencil (A, B)

$T : \Omega \to \mathbb{C}^{k\times k}$ holomorphic matrix-valued function, $\Omega \subseteq \mathbb{C}$ open and connected.
Focus on computing eigenvalues in a selected subset $\Delta \in \Omega$
A possible approach

$$\text{(NEP)} \Longrightarrow \text{(PEP)} \Longrightarrow \text{(LIN)}$$

$$T(z) \Longrightarrow P(z) \Longrightarrow (A, B)$$

If $T(z)$ is not a polynomial, we seek an approximation $P(z)$ such that

$$\|T(z) - P(z)\| \leq \varepsilon, \qquad z \in \Delta \subseteq \Omega$$

Approximating polynomial

- Taylor polynomial $P_d(z) = T(\sigma) + T'(\sigma)(z-\sigma) + \cdots + T^{(d)}(\sigma)\frac{(z-\sigma)^d}{d!}$. This approximation is appropriate if $\lambda \in \bar{\mathbb{D}}_{\sigma,\rho} \subset \Omega$
- Interpolation on suitable nodes
  - Chebyshev nodes Suitable if the wanted eigenvalues of T lie in or near the interval $[-1;1]$
  - $d+1$ Roots of unity Suitable if the wanted eigenvalues of T lie in open unit disk
  - d Roots of unity plus $\sigma = 0$ Suitable if the wanted eigenvalues of T lie in open unit disk

Approximating polynomial

For these approximations ...

- We can prove uniform convergence of the interpolant $P_d(z)$ to the $T(z)$ inside "circular" regions in which T is holomorphic.
- If $P_d(z)$ such that $\|T(z) - P_d(z)\| \leq \varepsilon, z \in \Delta$, and (λ, v) is an eigenpair for $P_d(z)$, $\lambda \in \Delta \implies \|T(\lambda)v\| < \varepsilon$.
- if $\det(T(\mu)) \neq 0, \mu \in \Delta \implies \det(P_d(\mu)) \neq 0$.

Not always easy to choose the degree of the approximating polynomial

Linearizations

Interested in Unitary-plus-low rank pencils...
We tested and compared different linearizations

- Generalized companion
- Unitary diagonal plus low rank
- Arrowed linearization $\rightarrow$ obtained considering the matrix polynomial written in the Lagrange basis

In all these cases we get a pencil (A, B) with A and B unitary plus rank k.

Linearizations

Interested in Unitary-plus-low rank pencils...
We tested and compared different linearizations

- Generalized companion
- Unitary diagonal plus low rank
- Arrowed linearization $\rightarrow$ obtained considering the matrix polynomial written in the Lagrange basis

In all these cases we get a pencil (A, B) with A and B unitary plus rank k.

The linearizations considered only need the evaluation of $T(z)$ on the interpolation nodes.

The generalized eigenvalue problem

To find the eigenvalue of (A, B)

- QR on $B^{-1}A$
- QZ on the pencil
- Arnoldi or other Krylov methods
- Since most of the eigenvalues have to be thrown away: inverse orthogonal iterations
- This is important when we have clustered eigenvalues

Inverse orthogonal iterations

- Generalization of inverse Power iterations
- Approximate the invariant subspace associated with the s eigenvalues with smallest modulus
- Approximate the s eigenvalues with smallest modulus
- The matrices of the pencil (A, B) are unitary plus low rank. and can be represented in the LFR format

Inverse orthogonal iterations: pencil

- On the pencil (A, B)

$$\begin{cases} AZ_i = BQ_{i-1} \\ Q_i R_i = Z_i \quad \text{economy size QR fact} \end{cases}$$

- Compute $V_{i-1}S_{i-1} = BQ_{i-1}$, full QR factorization.
- $AZ_i = V_{i-1}S_{i-1}$, that is

$$V_{i-1}^H A Z_i = S_{i-1}, \qquad S_{i-1} \text{ is rectangular upper triang}$$

  Consider the RQ factorization of $V_{i-1}^H A$.
- $V_{i-1}^H A = \tilde{R}_i \tilde{Q}_i$ we get

$$V_{i-1}^H A Z_i = S_{i-1}.$$

- Then $Z_i = \tilde{Q}_i^H \tilde{R}_i^{-1} S_{i-1}$ and $Q_i = \tilde{Q}_i^H(:, 1 : s)$.

Inverse orthogonal iterations: pencil

- On the pencil (A, B)

$$\begin{cases} AZ_i = BQ_{i-1} \\ Q_i R_i = Z_i \quad \text{economy size QR fact} \end{cases}$$

- Compute $V_{i-1}S_{i-1} = BQ_{i-1}$, full QR factorization.
- $AZ_i = V_{i-1}S_{i-1}$, that is

$$V_{i-1}^H A Z_i = S_{i-1}, \quad S_{i-1} \text{ is rectangular upper triang}$$

Consider the RQ factorization of $V_{i-1}^H A$.

- $V_{i-1}^H A = \tilde{R}_i \tilde{Q}_i$ we get

$$\tilde{R}_i \tilde{Q}_i Z_i = S_{i-1}.$$

- Then $Z_i = \tilde{Q}_i^H \tilde{R}_i^{-1} S_{i-1}$ and $Q_i = \tilde{Q}_i^H(:, 1 : s)$.

The algorithm on the structure

- We can describe inverse orthogonal iterations using LFR format
- At convergence construct the $s \times s$ pencil $(Q_i^H A Q_i, Q_i^H B Q_i)$
- The smallest s eigenvalues of (A, B) are approximated by the eigenvalues of $(Q_i^H A Q_i, Q_i^H B Q_i)$
- In our experiments we used Matlab `eig`

Cost and Forward error

This method costs $O(nks)$ flops per iterations if we start from a pencil (A, B) in the LFR format and with A and B as in the linearizations considered.

We consider the following relative forward error estimate

$$err(\lambda) = \frac{\|T(\lambda)v\|_2}{\|T(\lambda)\|_2\|v\|_2} = \frac{\sigma_k}{\sigma_1}, \text{for } k > 1, \quad err(\lambda) = \sigma_1, \text{for } k = 1,$$

- λ is the approximated eigenvalue inside the unit circle,
- $\sigma_i = \sigma_i(T(\lambda)), 1 = 1, \ldots, k$ are the singular values of $T(\lambda)$
- v is its $k-$th right singular vector of $T(\lambda)$.

Backward Error

A possible measure of the backward error is

$$bk_{err} = \frac{\sigma_{s+1}([AQ_i, BQ_i])}{\sigma_1([AQ_i, BQ_i])}$$

at convergence $AQ_i \approx BQ_i$

Numerical Experiments

Time Delay (k=2, s=3))

nodes	Structure	*deg*	$err(\lambda_1)$	$err(\lambda_2)$	$err(\lambda_3)$	it	bk_{err}
ω^j_{deg+1}	comp	32	8.47e-13	1.20e-12	1.21e-12	50	7.90e-13
$\omega^j_{deg}, 0$	comp	32	8.47e-13	8.48e-13	2.56e-12	53	7.90e-13
$\omega^j_{deg}, 0$	u.diag	40	9.43e-15	4.34e-14	3.43e-13	51	1.06e-12
Cheb	comp	32	5.07e-13	6.25e-10	6.25e-10	53	4.73e-13

Cancer growth (k=2, s=4)

nodes	Structure	*deg*	$err(\lambda_1)$	$err(\lambda_2)$	$err(\lambda_3)$	it	bk_{err}
ω^j_{deg+1}	comp	29	2.65e-14	2.87e-14	8.83e-14	34	9.57e-14
$\omega^j_{deg}, 0$	comp	29	1.67e-15	1.83e-14	2.52e-14	33	9.57e-14
$\omega^j_{deg}, 0$	unit diag	32	8.14e-16	3.47e-15	6.16e-15	36	7.68e-15
Cheb	comp	29	1.77e-15	1.27e-14	4.40e-12	36	7.53e-14

Complex eigenvalues: Interpolation on Chebyshev point is not adequate

Numerical Experiments

Neutral functional differential equation (k=1, s=2)

nodes	Structure	*deg*	$err(\lambda_1)$	$err(\lambda_2)$	it	bk_{err}
ω^j_{deg+1}	comp	60	3.19e-13	3.70e-13	37	4.36e-15
$\omega^j_{deg}, 0$	comp	60	1.71e-13	2.11e-13	37	3.85e-15
$\omega^j_{deg}, 0$	unit diag	64	9.35e-12	9.64e-12	40	3.19e-13
Cheb	comp	62	8.90e-07	8.90e-07	55	5.66e-15

Complex eigenvalues: Interpolation on Chebyshev point is not adequate

Spectral abscissa optimization (k=3, s=6)

nodes	Structure	*deg*	$err(\lambda_1)$	$err(\lambda_2)$	$err(\lambda_3)$	it	bk_{err}
ω^j_{deg+1}	comp	26	7.09e-12	7.09e-12	5.37e-11	89	4.34e-14
$\omega^j_{deg}, 0$	comp	26	1.02e-11	1.02e-11	2.42e-11	90	4.33e-14
$\omega^j_{deg}, 0$	unit diag	48	1.49e-14	1.58e-14	1.73e-14	116	8.20e-13
Cheb	comp	30	6.87e-15	7.08e-15	2.73e-09	116	1.21e-13

Numerical Experiments

Hadeler function (k=8, s=2)

nodes	Structure	deg	$err(\lambda_1)$	$err(\lambda_2)$	it	bk_{err}
ω^j_{deg+1}	comp	14	7.51e-14	3.38e-12	75	1.13e-14
$\omega^j_{deg}, 0$	comp	14	7.84e-14	6.98e-13	77	8.39e-15
$\omega^j_{deg}, 0$	unit diag	16	1.76e-15	8.78e-15	81	4.82e-15
Cheb	comp	16	1.35e-15	3.29e-13	80	4.53e-14

Conclusions

- The LFR factorization is a very useful representation for unitary-plus-low-rank matrices
- From F we can "read" the structure of A without multiplying the factors.
- Currently considering other linearizations
- Other experiments for orthogonal iterations