Numerical methods for the tridiagonal hyperbolic quadratic eigenvalue problem

Bor Plestenjak
Department of Mathematics
University of Ljubljana
Slovenia

Quadratic eigenvalue problem (QEP)

$Q(\lambda) = \lambda^2 M + \lambda C + K$, where M, C, K are $n \times n$ matrices.

If a scalar λ and a nonzero vector x satisfy $Q(\lambda)x = 0$ then λ is an eigenvalue and x is the (right) eigenvector.

If M is nonsingular then there are $2n$ finite eigenvalues that are the zeros of the characteristic polynomial $f(\lambda) = \det(Q(\lambda))$ and the eigenvalues of the $2n \times 2n$ matrix

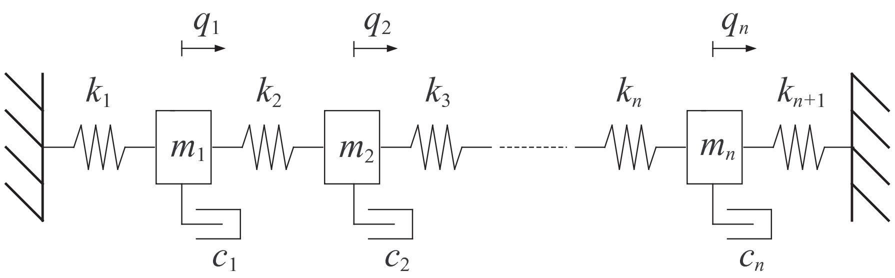
$$\begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix}.$$

We consider a special case when

- M, C, K are real symmetric tridiagonal matrices and
- QEP is hyperbolic

Our goal is to compute all the eigenvalues. The eigenvectors can be later obtained by inverse iteration.

Example: the overdamped mass-spring system



A damped system of n masses and $n+1$ springs leads to the QEP

$$\lambda^2 M + \lambda C + K,$$

where

$$M = \begin{bmatrix} m_1 & & \\ & \ddots & \\ & & m_n \end{bmatrix}, \quad C = \begin{bmatrix} c_1 & & \\ & \ddots & \\ & & c_n \end{bmatrix}, \quad K = \begin{bmatrix} k_1 + k_2 & -k_2 & & \\ -k_2 & \ddots & \ddots & \\ & \ddots & \ddots & -k_n \\ & & -k_n & k_n + k_{n+1} \end{bmatrix}.$$

M is the mass matrix, C is the damping matrix, and K is the stiffness matrix. The eigenvalues are the squares of the natural frequencies of the modes of vibration.

Overview

- Hyperbolic QEP
- Inertia of a hyperbolic QEP
- $\det(Q(\lambda))$ and its derivatives
- Rank two divide and conquer approach
- Laguerre's method and bisection
- Ehrlich–Aberth's method
- Durand–Kerner's method
- Numerical experiments
- Conclusion

Hyperbolic QEP

$Q(\lambda) = \lambda^2 M + \lambda C + K$ is hyperbolic if $M > 0$ and

$$(x^T Cx)^2 - 4(x^T Mx)(x^T Kx) > 0$$

for all $x \neq 0$.

Properties:

- all eigenvalues and eigenvectors are real
- eigenvalues are semisimple
- a gap between n largest (primary) and n smallest (secondary) eigenvalues
- n linearly independent eigenvectors associated with the primary and the secondary eigenvalues, respectively
- (Markus 1988) Q is hyperbolic iff there exists λ_0 such that $Q(\lambda_0) < 0$.

Minimax principle

For each $x \neq 0$ the equation $x^T Q(\mu) x = 0$

$$\mu^2 x^T M x + \mu x^T C x + x^T K x = 0$$

has two real solutions $\mu_1(x) < \mu_2(x)$.

Minimax principle (Duffin 1955): if $\lambda_{2n} \leq \cdots \leq \lambda_1$ are eigenvalues of a hyperbolic QEP Q then

$$\lambda_i = \max_{\substack{S \subset \mathbb{R}^n \\ \dim(S)=i}} \min_{0 \neq x \in S} \mu_2(x)$$

and

$$\lambda_{n+i} = \max_{\substack{S \subset \mathbb{R}^n \\ \dim(S)=i}} \min_{0 \neq x \in S} \mu_1(x)$$

for $i = 1, \dots, n$.

Inertia of a hyperbolic QEP

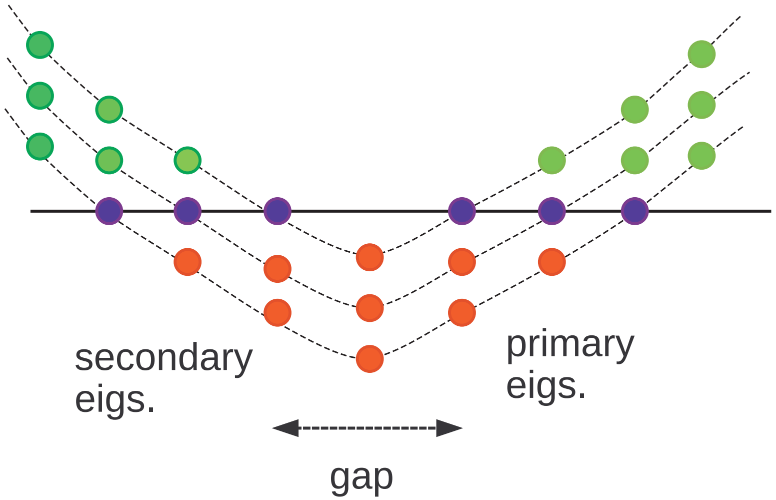
$Q(\lambda) = \lambda^2 M + \lambda C + K$

Inertia of a symmetric matrix A is a triple $(\nu(A), \zeta(A), \pi(A))$, where ν, ζ and π are the numbers of negative, zero and positive eigenvalues of A, respectively.

Theorem: Let γ be such that $Q(\gamma) < 0$.

a) If $\gamma \leq \mu$ then $\nu(Q(\mu))$ equals the number of eigenvalues of Q that are greater than μ.

b) If $\mu \leq \gamma$ then $\nu(Q(\mu))$ equals the number of eigenvalues of Q that are smaller than μ.



Bisection and other methods

Based on the inertia we can use bisection to obtain the k-th eigenvalue, but the convergence is slow. Therefore we apply methods that were successfully applied to tridiagonal generalized eigenproblems.

- Laguerre's iteration
  - K. Li, T.Y. Li (1993) - symmetric tridiagonal eigenproblem
  - K. Li, T.Y. Li, Z. Zeng (1994) - generalized symmetric tridiagonal eigenproblem
- Ehrlich–Abert iteration
  - D.A. Bini, L. Gemignani, F. Tisseur (2003) - nonsymmetric tridiagonal eigenproblem
- Durand–Kerner method
  - K. Li (1999) - generalized symmetric tridiagonal eigenproblem

The above methods require stable and efficient computation of $\nu(Q(\lambda))$, $f(\lambda)$, $f'(\lambda)/f(\lambda)$ and $f''(\lambda)/f(\lambda)$, where $f(\lambda) = \det(Q(\lambda))$.

Three term recurrences

Let

$$Q(\lambda) = \begin{bmatrix} a_1 & b_1 & & & 0 \\ b_1 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{n-2} & a_{n-1} & b_{n-1} \\ 0 & & & b_{n-1} & a_n \end{bmatrix},$$

where $a_i = a_i(\lambda)$ and $b_i = b_i(\lambda)$. Then

$f_0 = 1,\ f_1 = a_1,\quad f_{r+1} = a_{r+1} f_r - b_r^2 f_{r-1}.$

$f_0' = 0,\ f_1' = a_1',\quad f_{r+1}' = a_{r+1}' f_r + a_{r+1} f_r' - 2 b_r b_r' f_{r-1} - b_r^2 f_{r-1}'.$

$f_0'' = 0, f_1'' = a_1'',\quad f_{r+1}'' = a_{r+1}'' f_r + 2a_{r+1}' f_r' + a_{r+1} f_r'' - 2b_r'^2 f_{r-1} - 2b_r b_r'' f_{r-1} - 4b_r b_r' f_{r-1}' - b_r^2 f_{r-1}''.$

As the above recurrences may suffer from overflow–underflow problems, we define $d_i = \frac{f_i}{f_{i-1}}$, $g_i = \frac{f_i'}{f_i}$, $h_i = \frac{f_i''}{f_i}$:

$d_1 = a_1,\qquad d_{r+1} = a_{r+1} - \frac{b_r^2}{d_r}.$

$g_0 = 0,\ g_1 = \frac{a_1'}{a_1},\quad g_{r+1} = \frac{1}{d_{r+1}}(a_{r+1}' + a_{r+1} g_r - \frac{1}{d_r}(2b_r b_r' + b_r^2 g_{r-1})).$

$h_0 = 0, h_1 = \frac{a_1''}{a_1},\quad h_{r+1} = \frac{1}{d_{r+1}}(a_{r+1}'' + 2a_{r+1}' g_r + a_{r+1} h_r - \frac{1}{d_r}(2b_r'^2 + 2b_r b_r'' + 4b_r b_r' g_{r-1} + b_r^2 h_{r-1})).$

Using QR decomposition

It is known that

$$f'(\lambda)/f(\lambda) = \mathrm{Tr}(Q(\lambda)^{-1}Q'(\lambda)).$$

(Bini, Gemignani, Tisseur 2003) A stable $\mathcal{O}(n)$ computation for $\mathrm{Tr}(A^{-1})$, where A is unreduced nonsymmetric tridiagonal matrix.

(Kressner 2004) Generalization to $\mathcal{O}(n)$ algorithm for the computation of $\mathrm{Tr}(A^{-1}B)$ where A and B are tridiagonal.

Using LU decomposition

(Bohte 1979) Suppose that $\det(Q(\lambda)) \neq 0$ and that $PQ(\lambda) = LU$ is LU decomposition for $Q(\lambda)$. Then

$$f(\lambda) = \det(Q(\lambda)) = \det(P) \cdot u_{11}u_{22} \cdots u_{nn}.$$

By differentiation we have $PQ'(\lambda) = L'U + LU' = MU + LV$, where $M = L'$ is lower triangular with zero diagonal and $V = U'$ is upper triangular. M and V of the proper form and such that $PQ'(\lambda) = MU + LV$ can be computed from $Q'(\lambda)$, P, L, and U. It follows that

$$f'(\lambda) = \det(P) \sum_{i=1}^{n} v_{ii} \prod_{\substack{j=1 \\ j \neq i}}^{n} u_{jj}$$

and

$$f'(\lambda)/f(\lambda) = \sum_{i=1}^{n} \frac{v_{ii}}{u_{ii}}.$$

For the second derivative we have $PQ''(\lambda) = L''U + 2L'U' + LU'' = NU + 2MV + LW$, where $N = L''$ is lower triangular with zero diagonal and $V = U''$ is upper triangular. It follows that

$$f''(\lambda)/f(\lambda) = \sum_{i=1}^{n} \frac{w_{ii}}{u_{ii}} + \left(\sum_{i=1}^{n} \frac{v_{ii}}{u_{ii}} \right)^2 - \sum_{i=1}^{n} \frac{v_{ii}^2}{u_{ii}^2}.$$

Divide and conquer

Let

$$Q(\lambda) = \begin{bmatrix} a_1 & b_1 & & & 0 \\ b_1 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{n-2} & a_{n-1} & b_{n-1} \\ 0 & & & b_{n-1} & a_n \end{bmatrix},$$

where $a_i = a_i(\lambda)$ and $b_i = b_i(\lambda)$. We choose $m \approx n/2$ and set $b_m = 0$. We obtain

$$Q_0(\lambda) = \begin{bmatrix} Q_1(\lambda) & 0 \\ 0 & Q_2(\lambda) \end{bmatrix}.$$

Q_0 is hyperbolic. We take the eigenvalues $\widetilde{\lambda}_{2n} \leq \cdots \leq \widetilde{\lambda}_1$ of Q_0 as approximations to the eigenvalues $\lambda_{2n} \leq \cdots \leq \lambda_1$ of Q.

We repeat the procedure recursively for the eigenvalues of Q_1 and Q_2.

Interlacing property

$$Q_0(\lambda) = \begin{bmatrix} Q_1(\lambda) & 0 \\ 0 & Q_2(\lambda) \end{bmatrix}.$$

Let Q_t be a QEP with

$$Q_t(\lambda) = tQ(\lambda) + (1-t)Q_0(\lambda).$$

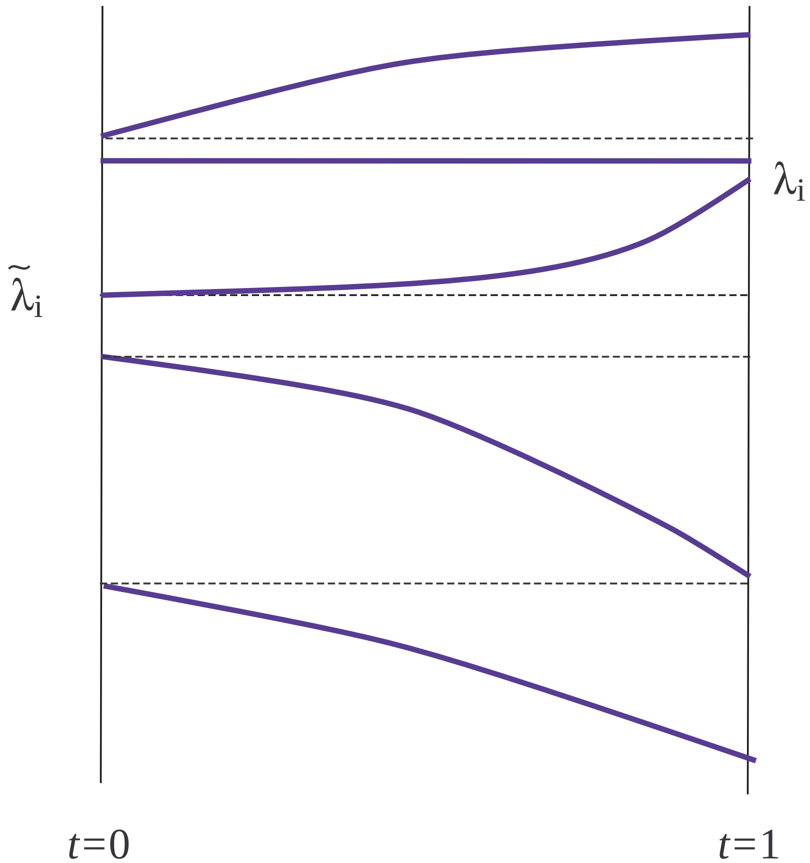
Can show that Q_t is hyperbolic for $t \in [0, 1]$.

Theorem: Let $\widetilde{\lambda}_{2n} \leq \cdots \leq \widetilde{\lambda}_1$ be the eigenvalues of Q_0 and $\lambda_{2n} \leq \cdots \leq \lambda_1$ the eigenvalues of Q. Then:

a) $\widetilde{\lambda}_1 \leq \lambda_1$ and $\lambda_{2n} \leq \widetilde{\lambda}_{2n}$,

b) $\widetilde{\lambda}_{i+1} \leq \lambda_i \leq \widetilde{\lambda}_{i-1}$, for $i = 2, \ldots, n-1$ and $i = n+2, \ldots, 2n-1$,

c) $\widetilde{\lambda}_{n+1} \leq \lambda_{n+1} < \lambda_n \leq \widetilde{\lambda}_n$.


λ_i
$\tilde{\lambda}_i$
t=0
t=1

Laguerre's method

Let $f(\lambda) = \det(Q(\lambda))$. Laguerre's iteration is

$$L_{\pm}(x) = x + \frac{2n}{\left(\frac{-f'(x)}{f(x)} \pm \sqrt{(2n-1)\left((2n-1)\left(\frac{-f'(x)}{f(x)}\right)^2 - 2n\frac{f''(x)}{f(x)}\right)}\right)}.$$

The method has cubic convergence in a neighbourhood of a simple eigenvalue.

Global convergence: if we add $\lambda_{2n+1} = -\infty$ and $\lambda_0 = \infty$ then for $x \in (\lambda_{i+1}, \lambda_i)$ we have

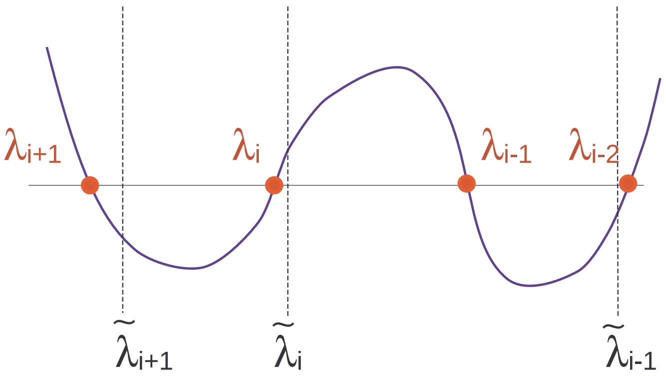
$$\lambda_{i+1} < L_-(x) < x < L_+(x) < \lambda_i.$$

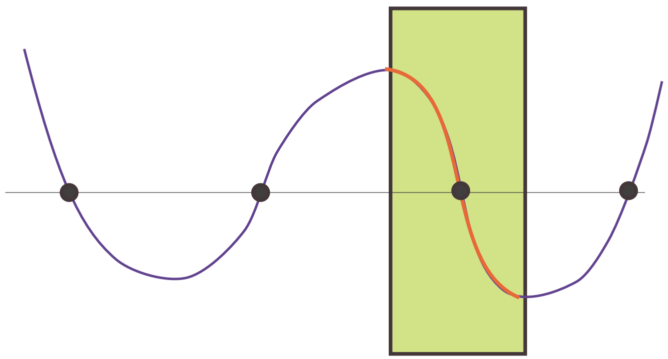
Divide and conquer: eigenvalues $\widetilde{\lambda}_{2n} \leq \cdots \leq \widetilde{\lambda}_1$ of $Q_0(\lambda)$ are initial approximations for $\lambda_{2n} \leq \cdots \leq \lambda_1$.

As $\widetilde{\lambda}_{i+1} \leq \lambda_i \leq \widetilde{\lambda}_{i-1}$ we can always use $\widetilde{\lambda}_i$ as an initial approximation for λ_i. From $\nu(Q(\widetilde{\lambda}_i))$ we see if $\lambda_i > \widetilde{\lambda}_i$ or $\widetilde{\lambda}_i < \lambda_i$ and then use L_+ or L_- sequence.

Bisection and Laguerre's method

If $\widetilde{\lambda}_i$ is close to λ_{i-1} or λ_{i+1}, then the convergence can be very slow.





The necessary condition for the cubic convergence near a single eigenvalue λ is that the sign of $-f'(x)/f(x)$ agrees with the sign of $\lambda - x$. To improve the convergence we first use bisection on interval $[\widetilde{\lambda}_i, \widetilde{\lambda}_{i+1}]$ (or $[\widetilde{\lambda}_i, \widetilde{\lambda}_{i-1}]$) until the condition is achieved.

Ehrlich–Aberth's method

The method simultaneously approximates all the zeros of a polynomial $f(\lambda) = \det(Q(\lambda))$. From an initial approximation $x^{(0)} \in \mathbb{C}^{2n}$ the method generates a sequence $x^{(j)} \in \mathbb{C}^{2n}$ which locally converges to the eigenvalues of Q. The equation is

$$x_j^{(k+1)} = x_j^{(k)} - \frac{\frac{f(x_j^{(k)})}{f'(x_j^{(k)})}}{1 - \frac{f(x_j^{(k)})}{f'(x_j^{(k)})} \sum_{\substack{l=1 \\ l \neq j}}^{2n} \frac{1}{x_j^{(k)} - x_l^{(k)}}}$$

for $j = 1, \ldots, 2n$.

If we implement the method in the Gauss–Seidel style then the convergence for simple roots is cubical and linear for multiple roots. We iterate only those eigenvalues that have not converged yet.

For initial approximations we again use divide and conquer with rank two modifications. To eliminate multiple values in the initial approximation we slightly perturb the eigenvalues of Q_0.

Durand–Kerner's method

Another method that simultaneously approximates all the zeros of a polynomial. As the method requires that the leading coefficient of the polynomial is one we apply it on

$$p(\lambda) = \frac{1}{\det(M)} \det(Q(\lambda)).$$

Similar to Ehrlich–Aberth the method generates a sequence $x^{(j)} \in \mathbb{R}^{2n}$ which locally converges to the eigenvalues of Q. The equation for $j = 1, \ldots, n$ is

$$x_j^{(k+1)} = x_j^{(k)} - \frac{p(x_j^{(k)})}{\prod_{\substack{l=1 \\ l \neq j}}^{2n} (x_j^{(k)} - x_l^{(k)})}.$$

If we implement the method in the Gauss–Seidel style then the convergence for simple roots is superquadratical and linear for multiple roots. As before, we iterate only the eigenvalues that have not converged yet.

Comparison and numerical results

In Matlab 6.5 we tested all three methods on a limited set of tridiagonal hyperbolic QEPs. We compared the average number of iterations. In all three methods one step (for one eigenvalue approximation) has linear time complexity.

- One step of Durand–Kerner's method is the cheapest as it requires only values of f.
- One step of Ehrlich–Aberth's method requires f and f' and is roughly equivalent to 2 Durand–Kerner steps.
- One step of Laguerre's method is the most expensive. It requires f, f' and f'', and is roughly equivalent to 3.7 Durand–Kerner steps.

Numerical example 1

- M and K: diagonals are random values from $[0.5, 1]$, codiagonals are random values from $[0, 0.1]$,
- C: diagonals are random values from $[4, 5]$, codiagonals are random values from $[0, 0.5]$,

n	Ehrlich-Aberth $\mathbb{R}$	Durand-Kerner $\mathbb{R}$	Ehrlich-Aberth $\mathbb{C}$	Durand-Kerner $\mathbb{C}$	Laguerre-bisection
avg. iteration in last D&C					
50	2.20	3.40	2.51	3.77	2.19
100	1.58	2.49	2.24	3.32	1.71
200	1.35	2.28	2.13	3.82	1.91
400	1.12	1.25	2.01	2.17	1.16
800	1.07	1.13	1.99	2.17	1.22
time					
50	0.52	0.77	0.67	1.16	0.86
100	1.25	2.36	1.66	2.92	2.09
200	3.33	5.89	5.11	8.30	6.63
400	9.63	13.28	17.17	21.09	20.45
800	29.19	33.27	70.56	64.09	76.47
error (compared to `polyeig`)					
50	$7e-15$	$7e-15$	$7e-15$	$7e-15$	$7e-15$
100	$1e-14$	$1e-14$	$1e-14$	$1e-14$	$1e-14$
200	$1e-14$	$1e-12$	$1e-14$	$1e-12$	$1e-14$
400	$2e-14$	$2e-14$	$2e-14$	$2e-14$	$2e-14$

- all eigenvalues are simple,
- as the dimension of the matrices increases, better the eigenvalues of $Q_0(\lambda)$ approximate eigenvalues of $Q(\lambda)$ and fewer steps are required in the final divide and conquer phase.

Numerical example 2

- $M = \mathrm{tridiag}(0.1, 1, 0.1)$, $C = \mathrm{tridiag}(0.5, 5, 0.5)$, $K = \mathrm{tridiag}(0.2, 1, 0.2)$
- all eigenvalues are simple, D&C approximations are double

n	Ehrlich-Aberth $\mathbb{R}$	Durand-Kerner $\mathbb{R}$	Ehrlich-Aberth $\mathbb{C}$	Durand-Kerner $\mathbb{C}$	Laguerre-bisection
avg. iterations in last D&C					
50	16.80	47.55	17.29	18.94	4.32
100	16.44	67.87	16.73	19.72	4.34
200	15.44	200.12	16.37	25.57	4.32
400	15.15	200.53	15.73	26.99	4.30
800	14.56	200.09	15.24	29.57	4.29
time					
50	1.81	3.16	2.22	2.28	0.97
100	5.97	15.72	7.69	7.75	3.03
200	20.31	133.20	28.50	32.22	10.44
400	76.89	575.33	114.64	126.80	39.50
800	303.86	2477.45	486.31	554.66	163.61
error (compared to `polyeig`)					
50	$4e-15$	$4e-15$	$4e-15$	$4e-15$	$4e-15$
100	$3e-15$	$2e-14$	$3e-15$	$3e-15$	$4e-15$
200	$5e-15$	$5e-02$	$5e-15$	$5e-15$	$5e-15$
400	$4e-15$	$1e+00$	$4e-15$	$4e-15$	$5e-15$

Numerical example 3

- $Q(\lambda) = \begin{bmatrix} Q_1(\lambda) & & \\ & Q_1(\lambda) & \\ & & Q_1(\lambda) \end{bmatrix}$, where Q_1 is from the first numerical example,
- all eigenvalues are triple.

n	Ehrlich-Aberth $\mathbb{R}$	Durand-Kerner $\mathbb{R}$	Ehrlich-Aberth $\mathbb{C}$	Durand-Kerner $\mathbb{C}$	Laguerre-bisection
avg. iteration in last D&C					
51	6.59	17.29	7.07	18.20	10.70
102	6.11	14.73	6.41	15.63	7.95
201	5.60	17.62	6.40	20.71	6.09
402	5.31	16.07	6.53	17.26	3.57
time					
51	0.72	1.64	0.89	1.98	1.36
102	1.94	3.89	2.64	5.02	3.58
201	5.89	13.16	8.75	20.75	11.38
402	19.11	36.72	35.16	57.28	31.97
error (compared to `polyeig`)					
51	$1e-14$	$9e-12$	$3e-15$	$6e-13$	$7e-14$
102	$1e-14$	$1e-12$	$5e-15$	$1e-12$	$7e-14$
201	$1e-14$	$8e-11$	$8e-15$	$7e-11$	$8e-14$
402	$1e-14$	$8e-11$	$1e-14$	$8e-11$	$6e-14$

Ehrlich-Aberth and general tridiagonal QEP

- M: diagonals are random values from $[0, 1]$, codiagonals are random values from $[0, 0.1]$,
- C: diagonals are random values from $[0, 1]$, codiagonals are random values from $[0, 0.5]$,
- K: diagonals are random values from $[0, 1]$, codiagonals are random values from $[0, 0.2]$,
- all eigenvalues are simple.

n	polyeig time	EA $\mathbb{C}$ time	avg. iter	error
50	0.09	0.84	2.30	9e-15
100	0.91	2.81	2.44	5e-14
200	6.17	7.19	2.15	8e-14
400	68.70	260.8	50.00	2e-13

Example from Tisseur and Meerbergen (2000)

- $M = \mathrm{tridiag}(0.1, 1, 0.1)$, $C = \mathrm{tridiag}(-3, 9, -3)$, $K = \mathrm{tridiag}(-5, 15, -5)$
- all eigenvalues are simple, $\mathrm{Re}(\lambda) < 0$, D&C approximations are double.

n	polyeig time	EA $\mathbb{C}$ time	avg. iter	error
50	0.08	2.05	15.12	9e-15
100	0.75	6.73	13.96	9e-15
200	4.95	24.56	13.48	1e-14
400	49.44	264.69	50.00	2e-14

Conclusions

Three eigensolvers for tridiagonal hyperbolic QEPs.

Generalizations:

- All methods can be easily parallelized.
- Similar approach (EA and DK) might be applied to:
  - nonsymmetric and non hyperbolic tridiagonal quadratic polynomial problems
  - tridiagonal polynomial problems
  - banded polynomial eigenvalue problems

  Algorithm based on LU decomposition might be used for an efficient computing of the derivative of the determinant.

Future work:

- handling of multiple eigenvalues
- more numerical tests
- stable computation of f, f' and f''.
- could continuation and path following be as efficient as other three methods