A Padé Approximate Linearization Algorithm for Solving the Quadratic Eigenvalue Problem with Low-Rank Damping

Ding Lu* Xin Huang† Zhaojun Bai‡ Yangfeng Su§

June 30, 2014

Abstract

The low-rank damping term appears commonly in quadratic eigenvalue problems arising from physical simulations. To exploit the low-rank damping property, we propose a Padé Approximate Linearization (PAL) algorithm. The advantage of the PAL algorithm is that the dimension of the resulting linear eigenvalue problem is only $n + \ell m$, which is generally substantially smaller than the dimension $2n$ of the linear eigenvalue problem produced by a direct linearization approach, where n is the dimension of the quadratic eigenvalue problem, ℓ and m are the rank of the damping matrix and the order of a Padé approximant, respectively. Numerical examples show that by exploiting the low-rank damping property, the PAL algorithm runs $33 - 47\%$ faster than the direct linearization approach for solving modest size quadratic eigenvalue problems.

1 Introduction

We consider the quadratic eigenvalue problem (QEP)

$$\mathcal{Q}(\lambda)x \equiv (\lambda^2 M + \lambda C + K)x = 0, \tag{1.1}$$

where M, C and K are $n \times n$ matrices, referred to as mass, damping and stiffness matrices, respectively, in structural dynamics analysis. The low-rank damping property refers to the case where the damping matrix C is of rank ℓ, $\ell \ll n$ and admits the rank-revealing decomposition

$$C = EF^{\mathrm{T}}, \tag{1.2}$$

where E and F are $n \times \ell$ full column rank matrices.

The QEP with the low-rank damping arises frequently from analysis of structural dynamics [10, 18] and structural-acoustic interaction [3, 6, 32]. In these applications, the damping

*School of Mathematical Sciences, Fudan University, Shanghai 200433, P. R. China. (dinglu@fudan.edu.cn). Part of this work was done while this author was visiting at University of California, Davis, supported by China Scholarship Council.

†School of Mathematical Sciences, Fudan University, Shanghai 200433, China. (xinhuang@fudan.edu.cn). Part of this work was done while this author was visiting at University of California, Davis, supported by China Scholarship Council.

‡Department of Computer Science and Department of Mathematics, University of California, Davis, CA 95616, USA. (bai@cs.ucdavis.edu).

§School of Mathematical Sciences, Fudan University, Shanghai 200433, China. (yfsu@fudan.edu.cn).

force is typically appled to the boundary and/or a small region. The finite element discretization of the governing equations leads to a QEP with an extremely sparse and low-rank damping matrix C.

To compute the eigenvalues of the QEP (1.1) close to a point σ of interest, a standard approach is to first apply the shift spectral transformation $\mu = \lambda - \sigma$ and then solve the QEP

$$(\mu^2 M + \mu C_\sigma + K_\sigma)x = 0, \tag{1.3}$$

by a linearization technique, where $C_\sigma = C + 2\sigma M$ and $K_\sigma = \sigma^2 M + \sigma C + K$. For example, in the first companion form, the QEP (1.3) is equivalent to the linear eigenvalue problem (LEP):

$$\begin{bmatrix} -C_\sigma & -K_\sigma \\ I_n & 0 \end{bmatrix} \begin{bmatrix} \mu x \\ x \end{bmatrix} = \mu \begin{bmatrix} M & 0 \\ 0 & I_n \end{bmatrix} \begin{bmatrix} \mu x \\ x \end{bmatrix}, \tag{1.4}$$

where I_n is the $n \times n$ identity matrix. For other forms of linearization, see [12, 28] and references therein. The task of finding eigenvalues λ of the QEP (1.1) close to the shift σ becomes one of extracting smallest (in modulus) few eigenvalues μ of the LEP (1.4).

After linearization, a variety of subspace-projection based methods and software for the resulting LEP can be applied. However, the dimension of the LEP (1.4) is twice the dimension of the QEP (1.1), and consequently, memory and computational costs are increased substantially. In particular, the Gram-Schmidt process for maintaining the orthogonality of the basis of the projection subspace in an LEP solver is observed as the dominant cost. The Jacobi-Davidson method [34], SOAR [1] and Q-Arnoldi [29] are memory-efficient QEP algorithms. However, none of these algorithms explicitly exploits the low-rank damping property for computational efficiency.

In this paper, we propose an algorithm to explicitly exploit the low-rank damping property for computational efficiency. The new algorithm is referred to as Padé Approximate Linearization, abbreviated as PAL. The dimension of the LEP produced by the PAL algorithm is $n_{\mathrm{L}} = n + \ell m$, where ℓ is the rank of C, and m is the order of Padé approximant. Since typically $\ell \ll n$ and m is a small positive integer, n_{L} is much smaller than the dimension $2n$ of the LEP derived by a direct linearization. Consequently, the PAL leads to a substantial reduction in memory and computational costs. Numerical examples show that with comparable accuracy, by exploiting the low-rank damping property, the new PAL algorithm runs 33 - 47% faster than the direct linearization approach for solving the QEPs of modest sizes.

The rest of this paper is organized as follows. In section 2, we introduce a spectral transformation that transforms the QEP (1.1) into an NEP. In section 3, we present the PAL algorithm. In section 4, we present a backward error analysis and a scaling scheme. In section 5, we give some implementation details of the PAL algorithm. In section 6, we present three numerical examples to demonstrate the accuracy and efficiency of the PAL algorithm. Concluding remarks are in section 7.

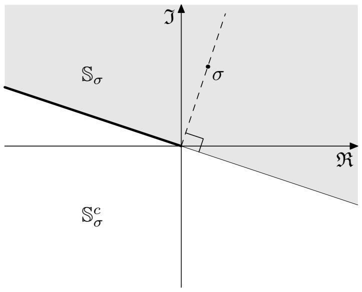

Figure 2.1: The domain $\mathbb{S}_\sigma$ (grey region including the solid line) and its complement $\mathbb{S}_\sigma^c$. If σ is a positive real number, then $\mathbb{S}_\sigma$ is the right half-plane including the non-negative imaginary axis, and if σ is a pure imaginary number with positive imaginary part, then $\mathbb{S}_\sigma$ is the upper half-plane including the non-positive real axis.

2 Spectral transformation

To compute eigenvalues of the QEP (1.1) close to a prescribed nonzero shift σ while preserving the low-rank damping property, let us consider the spectral transformation:

$$\begin{aligned} g_\sigma : \mathbb{S}_\sigma &\longrightarrow \mathbb{C} \\ \lambda &\longmapsto \mu = \frac{\lambda^2}{\sigma^2} - 1, \end{aligned} \tag{2.1}$$

where $\mathbb{S}_\sigma$, shown in Figure 2.1, defines a domain of the complex plane $\mathbb{C}$:

$$\mathbb{S}_\sigma \equiv \left\{ z \in \mathbb{C} \mid \arg\left(\frac{z}{\sigma}\right) \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right] \right\} \cup \{0\}. \tag{2.2}$$

In addition, let us define the mapping

$$\begin{aligned} f_\sigma : \mathbb{C} &\longrightarrow \mathbb{S}_\sigma \\ \mu &\longmapsto \lambda = \sigma\sqrt{\mu + 1}, \end{aligned} \tag{2.3}$$

where $\sqrt{\cdot}$ denotes the principal square root.[1]

The following lemma characterizes the relationship between mappings g_σ and f_σ.

Lemma 1. *(a) If $\lambda \in \mathbb{S}_\sigma$ and $\mu = g_\sigma(\lambda)$, then $f_\sigma(\mu) = \lambda$. (b) If $\mu \in \mathbb{C}$ and $\lambda = f_\sigma(\mu)$, then $\lambda \in \mathbb{S}_\sigma$ and $g_\sigma(\lambda) = \mu$.*

Proof. First let us show that if $\lambda \in \mathbb{S}_\sigma$, then $\sqrt{(\lambda/\sigma)^2} = \lambda/\sigma$. In fact, by the polar coordinate $\lambda/\sigma = te^{\theta i}$, where t is the modulus and $\theta \in (-\pi/2, \pi/2]$, we have $(\lambda/\sigma)^2 = t^2 e^{2\theta i}$. Consequently, by the definition of principal square root, we have the identity $\sqrt{(\lambda/\sigma)^2} = \lambda/\sigma$.

For (a) we can derive

$$f_\sigma(\mu) \equiv \sigma\sqrt{\mu + 1} = \sigma\sqrt{\frac{\lambda^2}{\sigma^2} - 1 + 1} = \sigma\sqrt{\frac{\lambda^2}{\sigma^2}} = \sigma\frac{\lambda}{\sigma} = \lambda. \tag{2.4}$$

[1] Using the polar coordinate system, a complex number z can be expressed as $z = te^{i\theta}$, where $t \geq 0$ is the modulus and the distance to the origin, and $\theta \in (-\pi, \pi]$ is the angle that the line from z to the origin makes with the positive real axis. The principal square root of z is then defined by $\sqrt{z} = \sqrt{t}e^{i\theta/2}$.

For (b), since $\lambda/\sigma = \sqrt{\mu+1}$, and by the definition of principal square root we have $\sqrt{\mu+1} = te^{\theta i}$ with t being the modular and $\theta \in (-\pi/2, \pi/2]$, which implies $\lambda \in \mathbb{S}_\sigma$. By (a) we have $\mu = g_\sigma(\lambda)$. □

Using the spectral transformation (2.1), the QEP (1.1) is transformed into the following nonlinear eigenvalue problem (NEP):

$$\mathcal{N}(\mu)x \equiv [K_\sigma - \mu M_\sigma + f_\sigma(\mu)C]\, x = 0, \tag{2.5}$$

where $K_\sigma = K + \sigma^2 M$ and $M_\sigma = -\sigma^2 M$.

The following theorem shows the relationship between the QEP (1.1) and the NEP (2.5) with respect to the domain $\mathbb{S}_\sigma$.

Theorem 1. *(a) If (λ, x) is an eigenpair of the QEP* (1.1) *and $\lambda \in \mathbb{S}_\sigma$, then $(\mu = g_\sigma(\lambda), x)$ is an eigenpair of the NEP* (2.5)*. (b) If (μ, x) is an eigenpair of the NEP* (2.5)*, then $\lambda = f_\sigma(\mu) \in \mathbb{S}_\sigma$ and (λ, x) is an eigenpair of the QEP* (1.1)*.*

Proof. (a) By Lemma 1(a), we have $\lambda = f_\sigma(\mu)$, where $\mu = g_\sigma(\lambda)$. Since (λ, x) is an eigenpair, it follows

$$\begin{aligned} 0 = \mathcal{Q}(\lambda)x &= (\lambda^2 M + \lambda C + K)x \\ &= [(\mu+1)\sigma^2 M + f_\sigma(\mu)C + K]x = \mathcal{N}(\mu)x. \end{aligned}$$

Therefore, (μ, x) is an eigenpair of the NEP (2.5).

(b) By Lemma 1(b), we have $\lambda \in \mathbb{S}_\sigma$, and $\mu = g_\sigma(\lambda)$. Since (μ, x) is an eigenpair of $\mathcal{N}(\mu)$, it follows

$$\begin{aligned} 0 = \mathcal{N}(\mu)x &= [K_\sigma - \mu M_\sigma + f_\sigma(\mu)C]\, x \\ &= [K_\sigma - g_\sigma(\lambda) M_\sigma + \lambda C]\, x = \mathcal{Q}(\lambda)x. \end{aligned}$$

Hence (λ, x) is an eigenpair of the QEP (1.1). □

For computing the eigenvalues of the QEP in the complement of $\mathbb{S}_\sigma$, i.e., $\mathbb{S}_\sigma^{\rm c} = \mathbb{S}_{-\sigma} \setminus \{0\}$, we consider the following NEP

$$\mathcal{N}^{\rm c}(\mu)x \equiv [K_\sigma - \mu M_\sigma - f_\sigma(\mu)C]\, x = 0, \tag{2.6}$$

The following theorem shows the equivalence between the QEP (1.1) and the NEP (2.6) with respect to the domain $\mathbb{S}_\sigma^{\rm c}$.

Theorem 2. *(a) If (λ, x) is an eigenpair of the QEP* (1.1) *and $\lambda \in \mathbb{S}_\sigma^{\rm c}$, then $(\mu = g_\sigma(\lambda), x)$ is an eigenpair of the NEP* (2.6)*. (b) If (μ, x) is an eigenpair of the NEP* (2.6) *and $\mu \neq 0$, then $\lambda = -f_\sigma(\mu) \in \mathbb{S}_\sigma^{\rm c}$ and (λ, x) is an eigenpair of the QEP* (1.1)*.*

Proof. Similar to the proof of Theorem 1. Note that $\mathbb{S}_\sigma^{\rm c} = \mathbb{S}_{-\sigma} \setminus \{0\}$ and $f_{-\sigma}(\mu) = -f_\sigma(\mu)$. □

By Theorems 1 and 2, the eigenvalues of the QEP (1.1) in $\mathbb{S}_\sigma$ are transformed to the eigenvalues of the NEP (2.5), while the eigenvalues of the QEP in $\mathbb{S}_\sigma^{\rm c}$ are transformed to the eigenvalues of the NEP (2.6). Since we are interested in extracting the eigenvalues of the QEP close to σ, we will focus on the NEP (2.5) in the rest of the paper.

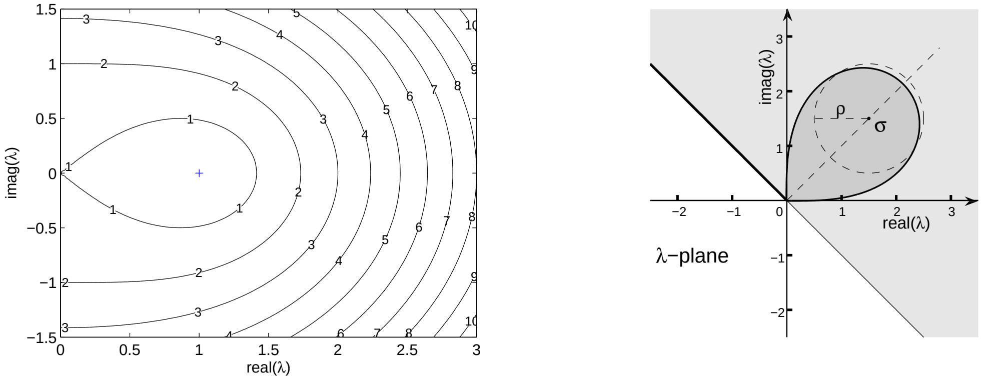

Figure 2.2: (a) The regions $\mathbb{B}_\rho$ with $\rho = 1, 2, \cdots, 10$ (left). (b) The region $\mathbb{A}_{\sigma,1} = \sigma\mathbb{B}_1$ with $\sigma = 1.5 + 1.5i$ (darker grey region) (right).

By the spectral transformation (2.1), λ close to σ corresponds to μ close to zero. Thus seeking eigenvalues λ of the QEP (1.1) close to the shift σ turns into seeking small (in modulus) eigenvalues μ of the NEP (2.5) in a disk $|\mu| \le \rho$. Specifically, the region in $\mathbb{S}_\sigma$ corresponding to the disk $|\mu| \le \rho$ is

$$\mathbb{A}_{\sigma,\rho} = \{\lambda \mid \lambda \in \mathbb{S}_\sigma \text{ and } |g_\sigma(\lambda)| \le \rho\} \equiv \sigma\mathbb{B}_\rho, \tag{2.7}$$

where $\mathbb{B}_\rho = \{\lambda \mid \lambda \in \mathbb{S}_1 \text{ and } |\lambda^2 - 1| \le \rho\}$. $\mathbb{A}_{\sigma,\rho}$ is the "unit" region $\mathbb{B}_\rho$ scaled by σ, as shown in Figure 2.2. As we can see $\mathbb{A}_{\sigma,\rho}$ leans toward the origin $(0, 0)$. $\mathbb{A}_{\sigma,\rho}$ can be regarded as the *domain of confidence* for the spectral transformation (2.1). This is similar to the notion for the shift spectral transformation [30].

We note that in practice, the shift $|\sigma|$ should not be chosen too small. Otherwise, the domain of confidence $\mathbb{A}_{\sigma,\rho}$ in the λ-plane corresponding to $|\mu| \le \rho$ is small. In this case the Padé approximant to be introduced in the next section will be able to approximate only a small number of eigenvalues of the QEP.

3 Padé approximate linearization

In this section, we start with an approximation of the NEP (2.5) by a rational eigenvalue problem (REP) via Padé approximation. Then we apply a trimmed linearization technique to convert the REP into an LEP.

3.1 Padé approximation

To find an accurate approximation of the NEP (2.5), let us consider an order-(m, m) diagonal Padé approximation [2] of the function $\sqrt{\mu + 1}$. In matrix-vector form, it can be written as

$$r_m(\mu) = -a^{\mathrm{T}}(I_m - \mu D_m)^{-1}a + d, \tag{3.1}$$

where a is a column vector $a = [(\gamma_1/\xi_1)^{\frac{1}{2}}, (\gamma_2/\xi_2)^{\frac{1}{2}}, \ldots, (\gamma_m/\xi_m)^{\frac{1}{2}}]^{\mathrm{T}}$, D_m is a diagonal matrix $D_m = -\operatorname{diag}(\xi_1, \xi_2, \ldots, \xi_m)$, $d = 2m + 1$ and

$$\gamma_j = \frac{2}{2m+1}\sin^2\frac{j\pi}{2m+1} \quad \text{and} \quad \xi_j = \cos^2\frac{j\pi}{2m+1}.$$

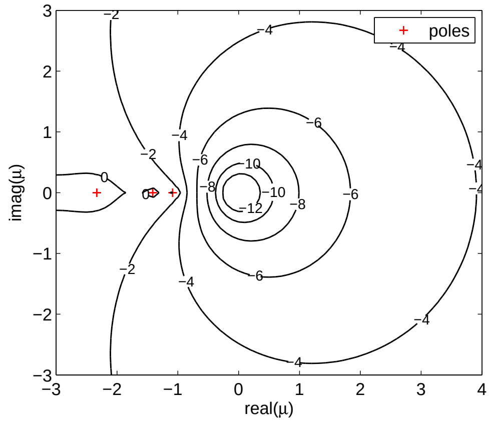

Figure 3.1: The contour plot of $\log_{10}|e(\mu)|$ with the order-5 diagonal Padé approximant. Three marked poles on the real axis are approximately -1.0862, -1.4130 and -2.3319. (The other two poles not presented are approximately -5.7948 and -49.3742.)

The poles of $r_m(\mu)$ are $-1/\xi_j$ for $j = 1, 2, \ldots, m$.

It can be shown [27] that an exact formula for the approximation error is given by

$$e(\mu) \equiv \sqrt{\mu+1} - r_m(\mu) = 2\sqrt{\mu+1}\,\frac{\theta^{2m+1}}{1+\theta^{2m+1}} \tag{3.2}$$

where $\theta = (\sqrt{\mu+1}-1)/(\sqrt{\mu+1}+1)$.[2] When $|\mu|$ is sufficiently small, $|e(\mu)| = O(\mu^{2m+1})$. The Padé approximation is more accurate than the polynomial-based Taylor approximation. For example, Figure 3.1 is a contour plot of the error $|e(\mu)|$ of $r_5(\mu)$. For $\mu = 2$, we have $|e(2)| \approx 1.77 \times 10^{-6}$. In contrast, the error of the 10th-order Taylor approximation is about 5.9938.

By the Padé approximant (3.1), the NEP (2.5) can be written as

$$\mathcal{N}(\mu)x = [K_\sigma - \mu M_\sigma + \sigma(r_m(\mu) + e(\mu))C]\,x = 0.$$

By truncating the error $e(\mu)$, it is then turned into the following REP:

$$\mathcal{R}(\mu)x \equiv [K_\sigma - \mu M_\sigma + \sigma\, r_m(\mu)C]\,x = 0. \tag{3.3}$$

Note that it is an abuse of notation that we use (μ, x) to denote the eigenpairs of the NEP (2.5) and the REP (3.3). However, we will connect the eigenpairs of the REP with the original QEP (1.1) directly. The eigenpairs of the NEP will no longer be referenced.

The Padé approximation induces m poles $\{-1/\xi_1, \ldots, -1/\xi_m\}$ in the REP (3.3). All these poles are real and less than -1. Large Padé error only occurs in a small region around the poles, as illustrated in Figure 3.1. Since the eigenvalues μ of interest of the REP (3.3) are close to zero, the presence of the poles is generally not a concern in practice.

[2] In [27], this result is shown for the case μ is real and $\mu > -1$. However, it can be extended directly to complex μ.

The REP (3.3) can be interpreted as a perturbation of the original QEP (1.1). Specifically, if (μ_*, x_*) is an eigenpair of the REP (3.3), then it is easy to verify that $(\lambda_* = f_\sigma(\mu_*), x_*)$ is an eigenpair of the QEP

$$\widetilde{\mathcal{Q}}(\lambda)x = \left[\lambda^2 M + \lambda(C + \Delta C) + K\right] x = 0, \tag{3.4}$$

where

$$\Delta C = -\frac{e(\mu_*)}{\sqrt{\mu_* + 1}} C \frac{x_* x_*^{\mathrm{H}}}{\|x_*\|^2}.$$

The QEP (3.4) is a perturbation of the original QEP (1.1). The perturbation only occurs in the damping matrix C and the perturbed damping term is still of low-rank. Furthermore, the relative perturbation error

$$\frac{\|\Delta C\|}{\|C\|} = \frac{|e(\mu_*)|}{|\sqrt{\mu_* + 1}|} \frac{\|Cx_*\|}{\|C\| \|x_*\|}$$

is expected to be small due to small Padé approximation error $|e(\mu_*)|$. In addition, the quantity $\|Cx_*\|/(\|C\|\|x_*\|)$ is expected to be small too due to the low-rank damping property. It provides extra accuracy to the approximation. In the extreme case where $Cx_* = 0$, the REP (3.3) and the QEP (1.1) share the same eigenpair. See Example 1 in section 6.

The idea of approximating a nonlinear eigenvalue problem by a simple eigenvalue problem has been proposed repeatedly. In [33, 31], a successive linear approximation method is used to solve a nonlinear eigenvalue problem by successively solving a sequence of linear eigenvalue problems. Instead of linear approximation, high-order polynomial and rational function approximations are studied [8, 38, 22, 21, 20, 15]. Although the notion of using one type of eigenvalue problem to approximate another type is widely adopted, a comprehensive error analysis of such an approach is not trivial. This is particularly true for nonlinear eigenvalue problems. Recently, the eigenvalue approximation error is characterized using the first order perturbation theory [8] and the nonlinear perturbation of linear eigenvalue problems [5]. Here, for the REP approximation (3.3) of the NEP (2.5), the approximation error can be interpreted as the backward error to the original QEP (1.1). We can apply the well-studied perturbation theory of the QEP, see for example [37]. This is another advantage of our proposed approach.

3.2 Trimmed linearization

To solve the the REP (3.3), we apply the trimmed linearization technique [35]. It converts the REP (3.3) to an LEP. Specifically, by the Padé approximant (3.1) and the factorization (1.2) of the damping matrix C, the rational term of the REP (3.3) can be rewritten as

$$\begin{aligned}
\sigma r_m(\mu) C &= -\sigma a^{\mathrm{T}} \big(I_m - \mu D_m\big)^{-1} a \cdot EF^{\mathrm{T}} + \sigma d C \\
&= -\sigma E \Big(I_\ell \cdot a^{\mathrm{T}} \big(I_m - \mu D_m\big)^{-1} a\Big) F^{\mathrm{T}} + \sigma d C \\
&= -\sigma E (I_\ell \otimes a^{\mathrm{T}}) \big(I_\ell \otimes I_m - \mu I_\ell \otimes D_m\big)^{-1} (I_\ell \otimes a) F^{\mathrm{T}} + \sigma d C \\
&= -E_{\sigma_1} (I_{\ell m} - \mu I_\ell \otimes D_m)^{-1} F_{\sigma_2}^{\mathrm{T}} + \sigma d C,
\end{aligned} \tag{3.5}$$

where $E_{\sigma_1} = \sigma_1 E(I_\ell \otimes a^{\mathrm{T}})$, $F_{\sigma_2} = \sigma_2 F(I_\ell \otimes a^{\mathrm{T}})$, $\otimes$ is the Kronecker product and $\sigma = \sigma_1 \sigma_2$ with σ_1 and σ_2 being two scalars.[3] By (3.5), the REP (3.3) can be written as

$$\mathcal{R}(\mu)x = \left[K_\sigma + \sigma d C - \mu M_\sigma - E_{\sigma_1} (I_{\ell m} - \mu I_\ell \otimes D_m)^{-1} F_{\sigma_2}^{\mathrm{T}}\right] x = 0. \tag{3.6}$$

[3] The decomposition $\sigma = \sigma_1 \sigma_2$ is not unique. A desirable choice of $\sigma = \sigma_1 \sigma_2$ will be discussed in section 5.

Applying the trimmed linearization proposed in [35], the REP (3.6) can be recast as the LEP of dimension $n_{\mathrm{L}} = n + \ell m$:

$$\mathcal{L}(\mu)x_{\mathrm{L}} \equiv (A - \mu B)x_{\mathrm{L}} = 0, \tag{3.7}$$

where

$$A = \begin{bmatrix} K_\sigma + \sigma d C & E_{\sigma_1} \\ F_{\sigma_2}^{\mathrm{T}} & I_{\ell m} \end{bmatrix}, \quad B = \begin{bmatrix} M_\sigma & 0 \\ 0 & I_\ell \otimes D_m \end{bmatrix},$$

and

$$x_{\mathrm{L}} = Hx \quad \text{with} \quad H = \begin{bmatrix} I_n \\ -(I_{\ell m} - \mu I_\ell \otimes D_m)^{-1} F_{\sigma_2}^{\mathrm{T}} \end{bmatrix}.$$

The connection between the REP and the LEP is shown in the following theorem, where $y(i\!:\!j)$ denotes the entries i to j of a vector y.

Theorem 3. *[35, Theorem 3.1] (a) If μ is an eigenvalue of the REP* (3.3)*, then it is an eigenvalue of the LEP* (3.7)*. (b) Let (μ, x_{L}) be an eigenpair of the LEP* (3.7) *and μ be not a pole of the REP* (3.3) *and $x_{\mathrm{L}}(1\!:\!n) \neq 0$. Then $(\mu, x_{\mathrm{L}}(1\!:\!n))$ is an eigenpair of the REP* (3.3)*. Moreover, the algebraic and geometric multiplicities of μ for the REP* (3.3) *and the LEP* (3.7) *are the same.*

Note that it is imposed that μ is not a pole of the REP (3.3). This condition can be easily verified since all poles $\{-1/\xi_1, \ldots, -1/\xi_m\}$ of the REP (3.3) are known from the choice of the Padé approximant $r_m(\mu)$.

3.3 Summary

The following is a summary of the proposed algorithm for computing a few eigenpairs of the QEP (1.1) around the shift σ.

1. Use the spectral transformation (2.1) to convert the QEP (1.1) to the NEP (2.5).
2. Select a Pade approximant $r_m(\mu)$ by (3.1).
3. Approximate the NEP (2.5) by the REP (3.3).
4. Use the trimmed linearization to the REP (3.3), and obtain the LEP (3.7).
5. Compute a few small (in modulus) eigenpairs (μ, x_{L}) of the LEP (3.7),
6. Return $(\lambda, x) = (f_\sigma(\mu), x_{\mathrm{L}}(1\!:\!n))$ as approximate eigenpairs of the QEP (1.1).

We call this approach the *Padé Approximate Linearization* (PAL). A discussion on some implementation aspects will be presented in section 5.

4 Error bound and scaling

In this section we provide a backward error analysis for the proposed PAL algorithm, and discuss a scaling scheme to reduce the backward error.

4.1 Error bound

Let $(\widehat{\mu}, \widehat{x}_{\mathrm{L}})$ be a computed eigenpair of the LEP (3.7) with the backward error $\eta_{\mathrm{L}}(\widehat{\mu}, \widehat{x}_{\mathrm{L}})$. Then by the PAL algorithm, $(\widehat{\lambda}, \widehat{x}) = (f_\sigma(\widehat{\mu}), \widehat{x}_{\mathrm{L}}(1:n))$ is an approximate eigenpair of the original QEP (1.1) with the backward error $\eta_{\mathrm{Q}}(\widehat{\lambda}, \widehat{x})$, where

$$\eta_{\mathrm{L}}(\widehat{\mu}, \widehat{x}_{\mathrm{L}}) = \frac{\|\mathcal{L}(\widehat{\mu})\widehat{x}_{\mathrm{L}}\|}{\varphi(\widehat{\mu})\|\widehat{x}_{\mathrm{L}}\|}, \quad \eta_{\mathrm{Q}}(\widehat{\lambda}, \widehat{x}) = \frac{\|\mathcal{Q}(\widehat{\lambda})\widehat{x}\|}{\rho(\widehat{\lambda})\|\widehat{x}\|}, \tag{4.1}$$

and $\varphi(\widehat{\mu}) = \|A\| + |\widehat{\mu}|\|B\|$ and $\rho(\widehat{\lambda}) = |\widehat{\lambda}|^2\|M\| + |\widehat{\lambda}|\|C\| + \|K\|$.

Now let us derive an upper bound of $\eta_{\rm Q}(\widehat{\lambda}, \widehat{x})$ in terms of $\eta_{\rm L}(\widehat{\mu}, \widehat{x}_{\rm L})$. First, we observe that the residual of the eigenpair $(\widehat{\lambda}, \widehat{x})$ of the QEP satisfies

$$\begin{aligned}\mathcal{Q}(\widehat{\lambda})\widehat{x} &= \mathcal{N}(\widehat{\mu})\widehat{x} = \mathcal{R}(\widehat{\mu})\widehat{x} + \sigma e(\widehat{\mu})C\widehat{x} \\ &= \mathcal{R}(\widehat{\mu})[\, I_n \ 0\,]\widehat{x}_{\rm L} + \sigma e(\widehat{\mu})C\widehat{x} = G\mathcal{L}(\widehat{\mu})\widehat{x}_{\rm L} + \sigma e(\widehat{\mu})C\widehat{x},\end{aligned} \tag{4.2}$$

where for the last equality, we used the identity

$$\mathcal{R}(\widehat{\mu})[\, I_n \ 0\,] = G\mathcal{L}(\widehat{\mu}) \quad \text{with} \quad G = [\, I_n \ \ -E_{\sigma_1}(I - \widehat{\mu}I_\ell \otimes D_m)^{-1}\,].$$

Then by (4.2), we have the bound

$$\begin{aligned}\frac{\|\mathcal{Q}(\widehat{\lambda})\widehat{x}\|}{\|\widehat{x}\|} &\le \|G\|\,\frac{\|\mathcal{L}(\widehat{\mu})\widehat{x}_{\rm L}\|}{\|\widehat{x}\|} + |\sigma e(\widehat{\mu})|\,\frac{\|C\widehat{x}\|}{\|\widehat{x}\|} \\ &\le \|G\|\frac{\|\widehat{x}_{\rm L}\|}{\|\widehat{x}\|}\,\frac{\|\mathcal{L}(\widehat{\mu})\widehat{x}_{\rm L}\|}{\|\widehat{x}_{\rm L}\|} + |\sigma e(\widehat{\mu})|\frac{\|C\widehat{x}\|}{\|\widehat{x}\|}.\end{aligned} \tag{4.3}$$

In terms of the backward errors $\eta_{\rm Q}(\widehat{\lambda}, \widehat{x})$ and $\eta_{\rm L}(\widehat{\mu}, \widehat{x}_{\rm L})$, the inequality (4.3) can be written as

$$\eta_{\rm Q}(\widehat{\lambda}, \widehat{x}) \le \alpha\, \eta_{\rm L}(\widehat{\mu}, \widehat{x}_{\rm L}) + \beta, \tag{4.4}$$

where α and β are given by

$$\alpha = \|G\|\frac{\|\widehat{x}_{\rm L}\|}{\|\widehat{x}\|}\frac{\varphi(\widehat{\mu})}{\rho(\widehat{\lambda})} \quad \text{and} \quad \beta = \frac{|\sigma e(\widehat{\mu})|}{\rho(\widehat{\lambda})}\frac{\|C\widehat{x}\|}{\|\widehat{x}\|}. \tag{4.5}$$

The quantity α is an error growth factor from the solution of the LEP (3.7) to the solution of the QEP (1.1). Later we will show how to reduce α via a proper scaling scheme. The quantity β is dominated by the Padé approximation error $e(\widehat{\mu})$, which is small in practice as we have discussed in Section 3.1. Another contributing factor to make the term β even smaller is the quantity $\|C\widehat{x}\|$. If $C\widehat{x} = 0$, then the approximate eigenpair $(\widehat{\lambda}, \widehat{x})$ of QEP (1.1) is also an approximate eigenpair of the undamped eigenvalue problem $(\lambda^2 M + K)x = 0$. In this case the bound (4.4) implies that there is no Padé approximation error contributing to the overall error. Since the dimension of the null space of C is expected to be large due to its low-rank property, the value $\|C\widehat{x}\|$ is generally very small. This so-called *extra accuracy* phenomenon has been observed in all of our numerical experiments as shown in Section 6.

In summary, the upper bound (4.4) indicates that in order to have an accurate approximation of the QEP (1.1) by the LEP (3.7), the Padé approximation error β should be within the desired threshold and the error growth factor α be bounded and small.

4.2 Scaling

It is a common practice to use a proper scaling scheme to the QEP for obtaining an LEP with a better condition number and smaller backward error [9, 18, 11, 16, 39]. A popular scaling scheme is to scale the QEP by a pair of parameters ω and ζ such that the coefficient matrices of the following scaled QEP have nearly unit 2-norms:

$$\mathcal{Q}_{\rm s}(\lambda_{\rm s})x_{\rm s} \equiv (\lambda_{\rm s}^2 M_{\rm s} + \lambda_{\rm s} C_{\rm s} + K_{\rm s})x_{\rm s} = 0, \tag{4.6}$$

where $\lambda_{\rm s} = \omega^{-1}\lambda$, $M_{\rm s} = \omega^2\zeta M$, $C_{\rm s} = \omega\zeta C$ and $K_{\rm s} = \zeta K$. If the shift σ for $\mathcal{Q}_{\rm s}$ is applied, then we should use the scaled shift $\sigma_{\rm s} = \omega^{-1}\sigma$. It is shown [9, 17, 18] that with the choice of scaling parameters

$$\omega = (\|K\|/\|M\|)^{1/2} \quad \text{and} \quad \zeta = 2(\|K\| + \omega\|C\|)^{-1}, \tag{4.7}$$

the companion form linearization of the scaled QEP (4.6) generally yields a better conditioned LEP.

Applying the PAL algorithm to the scaled QEP (4.6), we obtain the following scaled LEP

$$\mathcal{L}_{\rm s}(\mu_{\rm s})x_{\rm L_s} \equiv (A_{\rm s} - \mu_{\rm s}B_{\rm s})x_{\rm L_s} = 0, \tag{4.8}$$

where

$$A_{\rm s} = \begin{bmatrix} \zeta(K_\sigma + \sigma dC) & \sqrt{\zeta}E_{\sigma_1} \\ \sqrt{\zeta}F_{\sigma_2}^{\rm T} & I_{\ell m} \end{bmatrix} \quad \text{and} \quad B_{\rm s} = \begin{bmatrix} \zeta M_\sigma & \\ & I_\ell \otimes D_m \end{bmatrix}.$$

If $(\widehat{\mu}_{\rm s}, \widehat{x}_{\rm L_s})$ is an approximate eigenpair of the LEP (4.8), then $(\widehat{\lambda}_{\rm s}, \widehat{x}_{\rm s}) = (\sigma_{\rm s}\sqrt{\widehat{\mu}_{\rm s}+1}, \widehat{x}_{\rm L_s}(1\!:\!n))$ is an approximate eigenpair of the scaled QEP (4.6). Subsequently,

$$\widehat{\lambda} = \omega\widehat{\lambda}_{\rm s} = \omega\sigma_{\rm s}\sqrt{\widehat{\mu}_{\rm s}+1} = \sigma\sqrt{\widehat{\mu}_{\rm s}+1} = f_\sigma(\widehat{\mu}_{\rm s}) \quad \text{and} \quad \widehat{x} = \widehat{x}_{\rm s}. \tag{4.9}$$

is an approximate eigenpair of the original QEP (1.1).

We observe that the scaled LEP (4.8) does not depend on the scaling parameter ω, and neither does the approximate eigenpair $(\widehat{\lambda}, \widehat{x})$ of the QEP. Furthermore, if the eigenpair $(\widehat{\mu}_{\rm s}, \widehat{x}_{\rm L_s})$ of the scaled LEP (4.8) is computed with the backward error `rtol`, then by (4.4), we have

$$\eta_{\rm Q}(\widehat{\lambda}, \widehat{x}) \le \alpha_{\rm s} \cdot \texttt{rtol} + \beta, \tag{4.10}$$

where

$$\alpha_{\rm s} = \|G_{\rm s}\| \frac{\varphi_{\rm s}(\widehat{\mu}_{\rm s})}{\zeta\rho(\widehat{\lambda})} \frac{\|\widehat{x}_{\rm Ls}\|}{\|\widehat{x}\|}, \tag{4.11}$$

and $\varphi_{\rm s}(\widehat{\mu}_{\rm s}) = \|A_{\rm s}\| + |\widehat{\mu}_{\rm s}|\|B_{\rm s}\|$, and $G_{\rm s} = [I_n, \ -\sqrt{\zeta}E_{\sigma_1}(I_{\ell m} - \widehat{\mu}_{\rm s}I_\ell \otimes D_m)^{-1}]$.

By (4.10), we see that to reduce backward error $\eta_{\rm Q}$, the scaling parameter ζ should be chosen to yield a small growth factor $\alpha_{\rm s}$. Towards this goal, we have the following theorem to give an upper bound of $\alpha_{\rm s}$.

Theorem 4. *Let the rank-revealing decomposition $C = EF^{\rm T}$ and the shift splitting $\sigma = \sigma_1\sigma_2$ be chosen such that*

$$|\sigma_1|\|E\| = |\sigma_2|\|F\| = \sqrt{|\sigma|\,\|C\|}. \tag{4.12}$$

Then by the scaling parameter

$$\zeta = \frac{1}{\max\{\|\sigma^2 M\|, 2m\|\sigma C\|, \|K\|\}}, \tag{4.13}$$

we have

$$\alpha_{\rm s} \le \left(\frac{4m\tau}{\tau+2} + 2 + |\widehat{\mu}_{\rm s}|\right)\left(\frac{1+\delta^2}{1-\delta\nu}\right)\frac{\rho(\sigma)}{\rho(\widehat{\lambda})}, \tag{4.14}$$

where $\tau = \|C\|/\sqrt{\|M\|\|K\|}$, $\delta = \|(I_{\ell m} - \widehat{\mu}_{\rm s}I_\ell \otimes D_m)^{-1}\|$ and $\nu = \|\mathcal{L}_{\rm s}(\widehat{\mu}_{\rm s})\widehat{x}_{\rm Ls}\|/\|\widehat{x}_{\rm Ls}\|$.

Proof. To show the upper bound (4.14), we start with the definition (4.11) of $\alpha_{\rm s}$. For the term $\|G_{\rm s}\|$ of $\alpha_{\rm s}$, we have

$$\|G_{\rm s}\|^2 \leq 1 + \|\sqrt{\zeta}E_{\sigma_1}\|^2 \|(I_{\ell m} - \widehat{\mu}_{\rm s} I_\ell \otimes D_m)^{-1}\|^2. \tag{4.15}$$

Due to the assumption (4.12) and the choice of the scaling ζ as in (4.13), we have

$$\|\sqrt{\zeta}E_{\sigma_1}\| = \|\sqrt{\zeta}\sigma_1 E(I_\ell \otimes a^{\rm T})\| \leq \sqrt{\zeta}|\sigma_1|\|E\|\|a\| = \sqrt{\zeta 2m\|\sigma C\|} < 1, \tag{4.16}$$

where we used the identity $\|I_\ell \otimes a^{\rm T}\| = \|a\| = (2m)^{1/2}$.[4] Therefore by (4.15) and the definition of δ, we have

$$\|G_{\rm s}\| \leq \sqrt{1+\delta^2}. \tag{4.17}$$

For the second quantity $\varphi_{\rm s}(\widehat{\mu}_{\rm s})/\zeta\rho(\widehat{\lambda})$ of $\alpha_{\rm s}$, let us first bound $\|A_{\rm s}\|$ and $\|B_{\rm s}\|$.

$$\begin{aligned}\|A_{\rm s}\| &\leq 2\max\left\{1,\ \sqrt{\zeta}\|E_{\sigma_1}\|,\ \sqrt{\zeta}\|F_{\sigma_2}\|,\ \zeta\|K_\sigma + \sigma d C\|\right\} \\ &\leq 2\max\{1, \zeta\|K_\sigma + \sigma d C\|\} \\ &\leq 2\max\{1, \zeta\left(|\sigma|^2\|M\| + (2m+1)|\sigma|\|C\| + \|K\|\right)\} \\ &= 2\max\{1, \zeta\left(\rho(\sigma) + 2m|\sigma|\|C\|\right)\} \\ &= 2\zeta\rho(\sigma) + 4m\zeta|\sigma|\|C\|, \end{aligned} \tag{4.18}$$

where for the first inequality, we repeatedly apply the inequality $\|[A_1, A_2]\| \leq \sqrt{2}\max\{\|A_1\|, \|A_2\|\}$. For the second inequality we use the inequalities (4.16) and $\|\sqrt{\zeta}F_{\sigma_2}\| \leq 1$, which is derived by using an analogous derivation of (4.16). The last equality uses the choice of scaling parameter ζ.

Meanwhile, the choice of scaling ζ yields

$$\|B_{\rm s}\| \leq \max\left\{1,\ (|\sigma|^2\|M\|)\zeta\right\} = 1. \tag{4.19}$$

Combining (4.18) and (4.19), we have

$$\begin{aligned}\varphi_{\rm s}(\widehat{\mu}_{\rm s}) &= \|A_{\rm s}\| + |\widehat{\mu}_{\rm s}|\|B_{\rm s}\| \\ &\leq 2\zeta\rho(\sigma) + 4m\zeta|\sigma|\|C\| + |\widehat{\mu}_{\rm s}| \\ &\leq 2\zeta\rho(\sigma)\left(1 + \frac{2m\tau}{\tau+2}\right) + |\widehat{\mu}_{\rm s}|, \end{aligned} \tag{4.20}$$

where for the last inequality we use the inequality

$$\frac{|\sigma|\|C\|}{\rho(\sigma)} = \frac{|\sigma|}{|\sigma^2|\|M\|/\|C\| + \|K\|/\|C\| + |\sigma|} \leq \frac{|\sigma|}{2|\sigma|/\tau + |\sigma|} = \frac{\tau}{\tau+2}.$$

Dividing the inequality (4.20) by $\zeta\rho(\widehat{\mu}_{\rm s})$ on both sides gives rise to

$$\frac{\varphi_{\rm s}(\widehat{\mu}_{\rm s})}{\zeta\rho(\widehat{\lambda})} \leq \left(2\left(1 + \frac{2m\tau}{\tau+2}\right) + \frac{|\widehat{\mu}_{\rm s}|}{\zeta\rho(\sigma)}\right)\frac{\rho(\sigma)}{\rho(\widehat{\lambda})} \leq \left(2 + \frac{4m\tau}{\tau+2} + |\widehat{\mu}_{\rm s}|\right)\frac{\rho(\sigma)}{\rho(\widehat{\lambda})} \tag{4.21}$$

where the second inequality uses the inequality $\zeta\rho(\sigma) \geq 1$.

[4] Note that the identity $\sum_{j=1}^m \tan^2 \frac{j\pi}{2m+1} \equiv 2m^2 + m$. See [19].

Finally, we bound the quantity $\|\widehat{x}_{\mathrm{Ls}}\|/\|\widehat{x}\|$ of α_{s}. By the definition $\widehat{x} = \widehat{x}_{\mathrm{Ls}}(1\!:\!n)$, it holds that

$$\widehat{x}_{\mathrm{Ls}} = H_{\mathrm{s}}\widehat{x} + \begin{bmatrix} 0 \\ (I_{\ell m} - \widehat{\mu}_{\mathrm{s}} I_\ell \otimes D_m)^{-1}[0, I_{\ell m}]\mathcal{L}_{\mathrm{s}}(\widehat{\mu}_{\mathrm{s}})\widehat{x}_{\mathrm{Ls}}. \end{bmatrix},$$

where

$$H_{\mathrm{s}} = \begin{bmatrix} I_n \\ -\sqrt{\zeta}(I_{\ell m} - \widehat{\mu}_{\mathrm{s}} I_\ell \otimes D_m)^{-1} F_{\sigma_2}^{\mathrm{T}} \end{bmatrix}.$$

Therefore we have

$$\frac{\|\widehat{x}_{\mathrm{Ls}}\|}{\|\widehat{x}\|} \le \|H_{\mathrm{s}}\| + \|(I_{\ell m} - \widehat{\mu}_{\mathrm{s}} I_\ell \otimes D_m)^{-1}\| \frac{\|\mathcal{L}_{\mathrm{s}}(\widehat{\mu}_{\mathrm{s}})\widehat{x}_{\mathrm{Ls}}\|}{\|\widehat{x}_{\mathrm{Ls}}\|} \frac{\|\widehat{x}_{\mathrm{Ls}}\|}{\|\widehat{x}\|}, \tag{4.22}$$

which yields

$$\frac{\|\widehat{x}_{\mathrm{Ls}}\|}{\|\widehat{x}\|} \le \frac{\|H_{\mathrm{s}}\|}{1 - \delta\nu} \le \frac{\sqrt{1+\delta^2}}{1 - \delta\nu}, \tag{4.23}$$

where the bound $\|H_{\mathrm{s}}\| \le \sqrt{1+\delta^2}$ can be derived similarly to the derivation for the upper bound (4.17) of $\|G_{\mathrm{s}}\|$.

Combining (4.17), (4.21) and (4.23), we have the bound (4.14). □

We note that since the eigenvalues $\widehat{\mu}_{\mathrm{s}}$ of interest of the scaled LEP (4.8) are small, i.e., $|\widehat{\mu}_{\mathrm{s}}| \approx 0$, then $\delta \approx 1$ and $\rho(\widehat{\lambda}) \approx \rho(\sigma)$. Consequently, if the scaled LEP (4.8) has been solved with the residual norm $\nu \ll 1$, then the bound (4.14) is simplified to

$$\alpha_{\mathrm{s}} \lesssim 4\left(\frac{2m\tau}{\tau+2} + 1\right).$$

Moreover, if $\tau \ll 1$, known as a heavily underdamped system, then $\alpha_{\mathrm{s}} \lesssim 4$.

The assumption (4.12) is mild in practice and will be discussed in detail in Section 5.

5 PAL algorithm

Algorithm 1 is a complete description of the PAL algorithm for computing a few eigenpairs of the QEP (1.1) around the prescribed shift σ.

Algorithm 1 PAL

```
Initialize
    (a) the shift $\sigma \neq 0$
    (b) $k$ for the desired number of eigenpairs, and rtol for the backward error tolerance
    (c) the order $m$ of Padé approximant $r_m(\mu)$
Compute the scaling factor $\zeta$ by (4.13)
Compute the shift splitting $\sigma = \sigma_1\sigma_2$ to satisfy the condition (4.12)
Compute the LU factorization of $\mathcal{Q}(\sigma)$
Compute the $k$ smallest (in modulus) eigenpairs $(\widehat{\mu}_{\mathrm{s}}, \widehat{x}_{\mathrm{L_s}})$ of the scaled LEP (4.8) with the
    backward errors $\eta_{\mathrm{L_s}}(\widehat{\mu}_{\mathrm{s}}, \widehat{x}_{\mathrm{L_s}}) \le$ rtol
Discard those $\widehat{\mu}_{\mathrm{s}}$ which coincide with the poles of $r_m(\mu)$
Compute the approximate eigenpairs $(\widehat{\lambda}, \widehat{x}) = (\sigma\sqrt{\widehat{\mu}_{\mathrm{s}} + 1}, \widehat{x}_{\mathrm{L_s}}(1:n))$ of the QEP (1.1) and
    the corresponding backward errors $\eta_{\mathrm{Q}}(\widehat{\lambda}, \widehat{x})$
```

To apply the proposed scaling parameter ζ in (4.13), we assume that the rank-revealing decomposition (1.2) and the shift splitting $\sigma = \sigma_1\sigma_2$ are chosen to satisfy the assumption (4.12). If C is symmetric positive semi-definite, then $E = F$ in the rank-revealing factorization (1.2) of C. We can then select $\sigma_1 = \sigma_2 = \sqrt{\sigma}$. In general, given the rank-revealing decomposition (1.2), one can compute the QR factorization $E = QR$, where Q is $n \times \ell$ orthogonal and R is $\ell \times \ell$, then with an updated rank-revealing factorization of C with $E = Q$ and $F := FR^{\mathrm{T}}$, we can let $\sigma_1 = \sqrt{\sigma\|F\|}$ and $\sigma_2 = \sqrt{\sigma/\|F\|}$ to satisfy the assumption (4.12).

To solve the scaled LEP (4.8) by an iterative solver, such as the Arnoldi method [13], we need to provide the product of the matrix $A_{\mathrm{s}}^{-1}B_{\mathrm{s}}$ with an arbitrary vector u, that is

$$v = A_{\mathrm{s}}^{-1}B_{\mathrm{s}}u. \tag{5.1}$$

By exploiting the structure of A_{s}, we can implement the matrix-vector product efficiently, Specifically, we first note that the matrix A_{s} can be factorized as

$$A_{\mathrm{s}} = \begin{bmatrix} I_n & \sqrt{\zeta}E_{\sigma_1} \\ & I_{\ell m} \end{bmatrix} \begin{bmatrix} \zeta(K_\sigma + \sigma dC - E_{\sigma_1}F_{\sigma_2}^{\mathrm{T}}) & \\ & I_{\ell m} \end{bmatrix} \begin{bmatrix} I_n & \\ \sqrt{\zeta}F_{\sigma_2}^{\mathrm{T}} & I_{\ell m} \end{bmatrix}. \tag{5.2}$$

By the identity (3.5) and $r_m(0) = 1$, we have

$$\begin{aligned} K_\sigma + \sigma dC - E_{\sigma_1}F_{\sigma_2}^{\mathrm{T}} &= K_\sigma + \sigma dC - E_{\sigma_1}(I_{\ell m} - 0 \cdot I_\ell \otimes D_m)^{-1}F_{\sigma_2}^{\mathrm{T}} \\ &= K_\sigma + \sigma r_m(0)C = \sigma^2 M + \sigma C + K = \mathcal{Q}(\sigma). \end{aligned}$$

Therefore, the inverse of A_{s} is given by

$$A_{\mathrm{s}}^{-1} = \begin{bmatrix} I_n & \\ -\sqrt{\zeta}F_{\sigma_2}^{\mathrm{T}} & I_{\ell m} \end{bmatrix} \begin{bmatrix} \mathcal{Q}(\sigma)^{-1}/\zeta & \\ & I_{\ell m} \end{bmatrix} \begin{bmatrix} I_n & -\sqrt{\zeta}E_{\sigma_1} \\ & I_{\ell m} \end{bmatrix}.$$

If vectors $v = [v_1^{\mathrm{T}} \;\; v_2^{\mathrm{T}}]^{\mathrm{T}}$ and $u = [u_1^{\mathrm{T}} \;\; u_2^{\mathrm{T}}]^{\mathrm{T}}$ are partitioned to be conformal with the blocks of matrices A_{s} and B_{s}, then the matrix-vector product (5.1) can be computed by the following formulae:

$$\begin{aligned} v_1 &= (\mathcal{Q}(\sigma)^{-1}/\zeta)\left(\zeta M_\sigma u_1 - \sqrt{\zeta}E_{\sigma_1}(I_\ell \otimes D_m)u_2\right) \\ &= -\mathcal{Q}(\sigma)^{-1}\left(\sigma^2 M u_1 + (\sigma_1/\sqrt{\zeta})E(I_\ell \otimes a^{\mathrm{T}}D_m)u_2\right) && \text{(5.3a)} \\ v_2 &= (I_\ell \otimes D_m)u_2 - \sqrt{\zeta}F_{\sigma_2}^{\mathrm{T}}v_1 \\ &= (I_\ell \otimes D_m)u_2 - \sqrt{\zeta}\sigma_2(I_\ell \otimes a)F^{\mathrm{T}}v_1, && \text{(5.3b)} \end{aligned}$$

where the identity $(A \otimes B)(C \otimes D) = AC \otimes BD$ is used in (5.3a) for matrices A, B, C and D of sizes that the matrix products AC and BD are defined.

By (5.3), we can compute the LU factorization of $\mathcal{Q}(\sigma)$ once and then apply it for the matrix-vector multiplication with $\mathcal{Q}(\sigma)^{-1}$. Hence the PAL algorithm takes about the same amount of work as the direct linearization in terms of the matrix-vector products in an iterative linear eigenvalue problem solver.

6 Numerical examples

In this section, we present three numerical examples to demonstrate the accuracy and efficiency of the PAL algorithm. The accuracy of a computed eigenpair $(\widehat{\lambda}, \widehat{x})$ is measured by the QEP normwise backward error $\eta_{\mathrm{Q}}(\widehat{\lambda}, \widehat{x})$ defined in (4.1), where 1-norm $\|\cdot\|_1$ is used for computing the norms of matrices M, C and K.

In Matlab implementation of the PAL algorithm, we use the functions `eig` or `eigs` to solve the the LEP (4.8). The function `eigs` is an implementation of the implicitly restarted Arnoldi method [23]. We use the function `lu` for computing the LU factorization of $\mathcal{Q}(\sigma)$. For sparse matrices, the function `lu` is from UMFPACK [7]. The testing data is collected on a Dell computer with an Intel(R) Dual Core(TM) 2.20GHz i7-3632QM CPU and 6 GB RAM.

We have also implemented the PAL algorithm in C++. For comparison, we have also implemented a Direct Linearization (DLIN) algorithm in C++. The DLIN algorithm is based on the linearizaiton (1.4) and the two-parameter scaling (4.7). The LEPs (1.4) and (4.8) are solved by using `ARPACK++` [14], which is based on the implicitly restarted Arnoldi method (IRAM) [23]. We use the default parameters provided in `ARPACK++`. Specifically, the number of Lanczos vectors is $p = 2k + 1$ with k being the number of eigenvalues required. The residual error tolerance `rtol` is the machine precision. The sparse LU factorization of $\mathcal{Q}(\sigma)$ is computed using `SuperLU` [24] with a threshold pivoting parameter $u = 0.1$ to control numerical stability. The testing data is collected on a cluster with two Intel Xeon X5670 2.93GHz CPUs and 94 GB RAM. No parallelization is attempted.

Example 1. In this example, we demonstrate numerical accuracy of the PAL algorithm, and effectiveness of the backward error bound (4.10) with the scaling scheme (4.13). We use a QEP arising from the vibration analysis of a slender beam supported at both ends and damped at the midpoint [18, 4]. The $n \times n$ mass and stiffness matrices M and K are positive definite. The damping matrix C has only one nonzero positive entry at the center position $(n/2, n/2)$. Therefore, the rank of C is one, $\ell = 1$, and has the decomposition $C = EE^{\mathrm{T}}$ where $E = \delta^{\frac{1}{2}} e_{n/2}$, $\delta > 0$, and $e_{n/2}$ is the unit column vector with only one entry at the position $n/2$ and zeros at the others. It is known [18] that half of the eigenvalues in this example are pure imaginary and are eigenvalues of the undamped problem $(\lambda^2 M + K)x = 0$, so the corresponding eigenvectors satisfy $Cx = 0$. PAL will introduce no truncation errors for these eigenpairs.

To demonstrate the accuracy of the PAL algorithm, we take the dimension $n = 200$ and $\delta = 5$. It is an underdamped QEP with $\tau = \|C\|/\sqrt{\|M\|\|K\|} \approx 0.0153598$. By the analysis in Section 4.2, we expect the error growth factor $\alpha_{\mathrm{s}} \leq 4$. The left plot in Figure 6.1 shows all eigenvalues computed by the Matlab function `polyeig` with the scaling strategy (4.7).

Let us compute a few eigenvalues of the QEP around the shift $\sigma = 10^6 i$. With the diagonal Padé order $m = 1$, the PAL leads to the LEP (3.7) of dimension $n_{\mathrm{L}} = n + \ell m = 201$, which is then solved by Matlab function `eig`. The right plot of Figure 6.1 shows some of the computed eigenvalues by the PAL. The following table is a profile of six selected eigenvalues and the corresponding backward errors of the LEP and the QEP:

#	$\mathrm{Re}(\widehat{\lambda})$	$\mathrm{Im}(\widehat{\lambda}/10^6)$	$\eta_{\mathrm{Ls}}(\widehat{\mu}_{\mathrm{s}}, \widehat{x}_{\mathrm{Ls}})$	$\eta_{\mathrm{Q}}(\widehat{\lambda}, \widehat{x})$
1	$+4.787700 \times 10^{-7}$	0.993105	7.87×10^{-16}	6.44×10^{-16}
2	$+2.828370 \times 10^{-7}$	1.573793	5.68×10^{-16}	5.21×10^{-16}
3	-9.193417×10^{-6}	2.097337	5.53×10^{-16}	5.27×10^{-16}
4	-6.423440	1.013141	1.26×10^{-15}	8.55×10^{-14}
5	-6.745303	1.545041	6.22×10^{-16}	1.71×10^{-9}
6	-5.595220	2.060988	5.80×10^{-16}	4.06×10^{-9}

Furthermore, the following table shows the corresponding error bound (4.10):

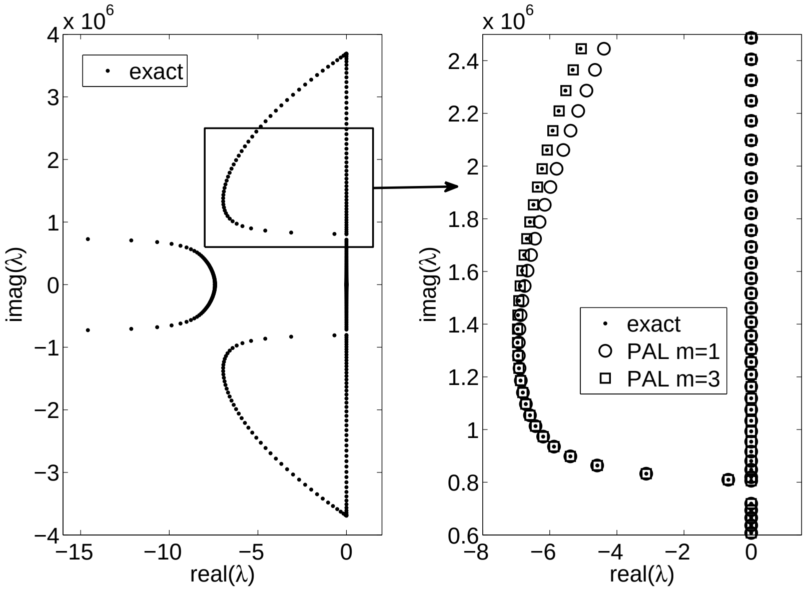

Figure 6.1: Left: “Exact” eigenvalues. Right: a fraction of exact eigenvalues and approximated eigenvalues by PAL.

#	α_{s}	$\lvert e(\widehat{\mu}_{\mathrm{s}})\rvert$	$\lVert C\widehat{x}\rVert / \lVert\widehat{x}\rVert$	β	$\alpha_{\mathrm{s}} \cdot \eta_{\mathrm{Ls}} + \beta_{\mathrm{s}}$
1	0.844	8.22×10^{-8}	1.19×10^{-13}	1.16×10^{-24}	6.65×10^{-16}
2	0.919	3.45×10^{-2}	1.49×10^{-13}	2.78×10^{-19}	5.23×10^{-16}
3	0.952	1.79×10^{-1}	2.97×10^{-13}	1.69×10^{-18}	5.29×10^{-16}
4	0.828	5.64×10^{-7}	1.32×10^{-3}	8.55×10^{-14}	8.67×10^{-14}
5	0.916	3.01×10^{-2}	1.02×10^{-3}	1.71×10^{-9}	1.71×10^{-9}
6	0.951	1.65×10^{-1}	7.49×10^{-4}	4.06×10^{-9}	4.06×10^{-9}

where the α_{s} values are computed by the definition (4.11).

We observe that the first three approximate the pure imaginary eigenvalues of the original QEP. Here the PAL algorithm introduces nearly no error as shown by the β and η_Q values. In particular, note that the 2nd and 3rd eigenvalues, although the Padé errors $\lvert e(\widehat{\mu})\rvert$ are not small, $\lVert C\widehat{x}\rVert / \lVert\widehat{x}\rVert$ are small. This is the so-called extra precision phenomenon as discussed in Section 4.1.

Furthermore, we observe that for all six eigenvalues, $\eta_{\mathrm{Q}} \approx \alpha_{\mathrm{s}} \cdot \eta_{\mathrm{Ls}} + \beta$, which suggests the error bound in (4.10) is tight. In particular, for the last three eigenvalues, we actually have $\eta_Q \approx \beta$. The errors of these computed eigenvalues are dominated by the Padé approximation errors. To improve the accuracy, we use a higher Padé order $m = 9$. It leads to an LEP of dimension $n_{\mathrm{L}} = 209$. Consequently, η_Q for the last three approximate eigenvalues are all reduced to the machine precision, namely about 10^{-16}, although $\lVert C\widehat{x}\rVert / \lVert\widehat{x}\rVert$ remains unchanged.

Example 2. This example shows the computational efficiency of the PAL algorithm. We consider an acoustic wave problem to model acoustic pressure in a two-dimensional bounded domain with boundary conditions that are partly pressure release and partly impedance [6]. By the finite element discretization of the wave equation on the unit square $[0, 1] \times [0, 1]$ with

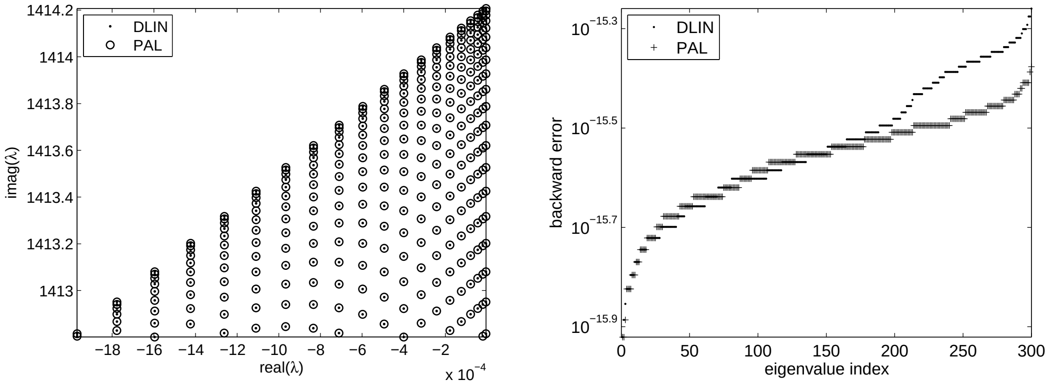

Figure 6.2: Left: computed eigenvalues. Right: backward errors $\eta_{\mathcal{Q}}(\widehat{\lambda}, \widehat{x})$.

mesh size h, it leads to the QEP (1.1) of dimension $n = q(q-1)$, where $q = 1/h$. The mass and stiffness matrices M and K are both symmetric positive definite. The rank of the damping matrix $C = EE^{\mathrm{T}}$ is $q-1$, where $E = (h/\xi)^{\frac{1}{2}} I_{q-1} \otimes e_q$, and ξ is an impedance parameter. This example is available in the NLEVP collection [4] labeled as `acoustic_wave_2d`.

To show the computational efficiency of PAL, we consider $h = 1/500$ and impedance $\xi = 1$. Consequently, the QEP has the dimension $n = q(q-1) = 249500$ and the damping matrix C has the rank $\ell = 499$. We compute $k = 300$ eigenvalues close to the shift $\sigma = 2\sqrt{2}q\mathtt{i}$. The diagonal Padé order is chosen to be $m = 3$. Consequently, the scaled LEP (4.8) has the dimension $n_{\mathrm{L}} = n + \ell m = 250997$, which is slightly larger than n but much smaller than the dimension $2n = 499000$ of the LEP (1.4) produced by the direct linearization.

The IRAM of `ARPACK++` takes 3 update iterations (or 2 restarts) to converge for the LEPs (1.4) and (4.8). The computed eigenvalues and their corresponding backward errors are shown in Figure 6.2. As we can see there are high agreements between the two linearizations in terms of computed eigenvalues and backward errors.

The computational costs of `ARPACK++` are dominated by four parts, namely, (a) sparse matrix-vector multiplications (SpMVs); (b) Gram-Schmidt (GS) process to maintain the orthogonality of basis vectors of the projection subspaces; (c) the eigenvector computation (EvComp); (d) costs of updating, such as restarting processes and solving small Hessenberg eigenvalue subproblems. The following table profiles the CPU time for each of these four parts:

	SpMVs	GS	EvComp	Updating	subtotal
DLIN	168.95	1130.44	314.65	304.00	1918.04
PAL	162.26	562.37	137.66	156.10	1018.39

From the above table, we see that the SpMV costs for the two linearizations are almost the same, which confirms that the discussion in Section 5. The bulk of computational time lies in the Gram-Schmidt orthogonalization process, where PAL reduces the cost by almost half.

By adding 6.41 seconds for the LU factorization of $\mathcal{Q}(\sigma)$ and other setting up costs, the total CPU elapsed time of the DLIN method is 1931.89 seconds. On the other hand, the PAL algorithm is 1028.54 seconds. PAL runs 47% faster than DLIN.

Example 3. This is a modest industrial size QEP arising from the automobile industry to analyze the modal frequency responses of a car body [26]. The QEP size is $n = 655812$.

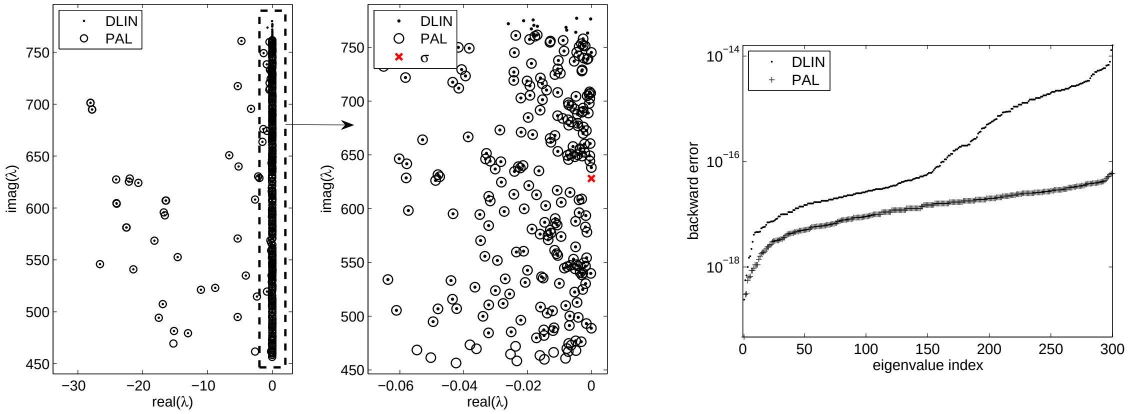

Figure 6.3: Left: computed eigenvalues. Right: backward errors $\eta_{Q}(\widehat{\lambda}, \widehat{x})$ of the eigenpairs.

Matrices M, C and K are real symmetric with the numbers of nonzero elements being 394508, 294 and 31775679, respectively. The damping matrix C is extremely sparse. Indeed, the nonzero elements of C form a 144-by-144 principal symmetric positive semi-definite submatrix. We first compute the eigenvalue decomposition of this submatrix and then truncate these eigenvalues whose magnitude less than $10^{-16} \cdot \lambda_{\max}$ to obtain the rank-revealing factorization $C = EE^{\mathrm{T}}$, where the rank of E is $\ell = 126$, and $\lambda_{\max}$ is the largest eigenvalue of the submatrix.

We compute 300 eigenvalues near the shift $\sigma = 200\pi\mathtt{i}$. The diagonal Padé order is chosen to be $m = 3$. The PAL algorithm leads to the LEP (4.8) of size $n_{\mathrm{L}} = n + \ell m = 656190$. In contrast, the size of the LEP (1.4) by the direct linearization is $2n = 1311624$.

The IRAM takes 2 iterations to converge for the LEPs (1.4) and (4.8). The computed eigenvalues are shown in Figure 6.3. There are high agreements of computed eigenvalues. Most of the eigenvalues are close to the imaginary axis. Zooming into this part, we can see that PAL computes few more eigenvalues of small modulus, while DLIN computes more of large modulus. The backward errors of the eigenpairs are shown in the left plot of Figure 6.3. PAL is more accurate than DLIN.

The following table profiles the CPU timing of four parts of `ARPACK++` for solving the LEPs.

	SpMV	GS	EvComp	Update	Subtotal
DLIN	1305.94	2277.01	791.25	393.81	4768.01
PAL	1297.92	1146.89	394.37	202.43	3459.99

The SpMV cost is high in this example, but is almost the same for PAL and DLIN. The bulk of computational time still lies in the Gram-Schmidt orthogonalization process and PAL reduces it by almost half.

By adding 408.59 seconds for computing the LU factorization of $\mathcal{Q}(\sigma)$, and other setting up costs, the total CPU elapsed time of DLIN is 5196.12 seconds. On the other hand, PAL is 3459.99 seconds. PAL runs 33.4% faster than DLIN.

7 Concluding remarks

We presented the PAL algorithm to solve the QEP with low-rank damping. The PAL algorithm combines Padé approximation and the trimmed linearization, and produces an LEP with slightly larger dimension than the original QEP. Numerical experiments have demonstrated the accuracy and saving in memory and computational time comparing with the direct linearization. One interesting future research problem is to determine the Padé approximation order m adaptively based on the desired accuracy.

It is still an open problem how to efficiently exploit the low-rank property in eigenvalue computation. Recently, in [36], an algorithm was proposed to compute all eigenpairs of the QEP with low-rank damping. However, due to the computational cost, it is not designed for large scale problems.

The PAL algorithm proposed in this paper can be naturally extended to computing NEPs of the form

$$\big[K - \lambda M + \sum_{\ell=1}^{\mathrm{L}} f_\ell(\lambda) C_\ell\big] x = 0,$$

where $f_\ell(\lambda)$ are nonlinear functions in λ, C_ℓ are low rank matrices. Such NEPs are found, for example, in the cavity design of a linear accelerator [25]. To solve this problem, one can first generate an approximate REP by replacing $f_\ell(\lambda)$ with properly chosen Padé approximants, then apply the trimmed linearization. This would fall in the same idea as recently proposed algorithm in [15].

Acknowledgement. We would like to thank Dr. Benshan Liao of SIMENS for the QEP data of Example 3.

References

[1] Z. Bai and Y. Su. SOAR: a second-order Arnoldi method for the solution of the quadratic eigenvalue problem. *SIAM Journal on Matrix Analysis and Applications*, 26(3):640–659, 2005.

[2] G.A. Baker and P.R. Graves-Morris. *Padé Approximants*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2nd edition, 1996.

[3] A. Bermúdez, R.G. Durán, R. Rodríguez, and J. Solomin. Finite element analysis of a quadratic eigenvalue problem arising in dissipative acoustics. *SIAM Journal on Numerical Analysis*, 38(1):267–291, 2000.

[4] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder, and F. Tisseur. NLEVP: A collection of nonlinear eigenvalue problems. *ACM Transactions on Mathematical Software (TOMS)*, 39(2):7:1–7:28, 2013.

[5] D. Bindel and A. Hood. Localization theorems for nonlinear eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 34(4):1728–1749, 2013.

[6] F. Chaitin-Chatelin and M.B. Van Gijzen. Analysis of parameterized quadratic eigenvalue problems in computational acoustics with homotopic deviation theory. *Numerical Linear Algebra with Applications*, 13(6):487–512, 2006.

[7] T. A. Davis. Algorithm 832: UMFPACK v4. 3—an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software (TOMS)*, 30(2):196–199, 2004.

[8] C. Effenberger and D. Kressner. Chebyshev interpolation for nonlinear eigenvalue problems. *BIT Numerical Mathematics*, 52(4):933–951, 2012.

[9] H. Y. Fan, W. W. Lin, and P. Van Dooren. Normwise scaling of second order polynomial matrices. *SIAM Journal on Matrix Analysis and Applications*, 26(1):252–256, 2004.

[10] A. Feriani, F. Perotti, and V. Simoncini. Iterative system solvers for the frequency analysis of linear mechanical systems. *Computer Methods in Applied Mechanics and Engineering*, 190(13):1719–1739, 2000.

[11] S. Gaubert and M. Sharify. Tropical scaling of polynomial matrices. In *Positive Systems*, pages 291–303. Springer Berlin Heidelberg, 2009.

[12] I. Gohberg, P. Lancaster, and L. Rodman. *Matrix Polynomials*, volume 58. SIAM, 2009.

[13] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University, Press, Baltimore, MD, USA, 1996.

[14] F. M. Gomes and D. C. Sorensen. *ARPACK++: An Object-oriented Version of ARPACK Eigenvalue Package, Version 1.2*. Available at `http://www.ime.unicamp.br/~chico/arpack++/`, 2000. Accessed on February 25, 2014.

[15] S. Güttel, R. Van Beeumen, K. Meerbergen, and W. Michiels. NLEIGS: A class of robust fully rational Krylov methods for nonlinear eigenvalue problems. Technical report, MIMS Preprint: 2013.49, available at `http://eprints.ma.man.ac.uk/2019/01/covered/MIMS_ep2013_49.pdf`. Accessed on May 20, 2014.

[16] S. Hammarling, C. J. Munro, and F. Tisseur. An algorithm for the complete solution of quadratic eigenvalue problems. *ACM Transactions on Mathematical Software (TOMS)*, 39(3):18:1–18:19, 2013.

[17] N. J. Higham, R. C. Li, and F. Tisseur. Backward error of polynomial eigenproblems solved by linearization. *SIAM Journal on Matrix Analysis and Applications*, 29(4):1218–1241, 2007.

[18] N. J. Higham, D. S. Mackey, F. Tisseur, and S. D. Garvey. Scaling, sensitivity and stability in the numerical solution of quadratic eigenvalue problems. *International Journal for Numerical Methods in Engineering*, 73(3):344–360, 2008.

[19] Wolfram Research Inc. *Tangent.* `http://functions.wolfram.com/01.08.23.0007.01/`, accessed on February 25, 2014.

[20] E. Jarlebring and S. Güttel. A spatially adaptive iterative method for a class of nonlinear operator eigenproblems. *Electronic Transactions on Numerical Analysis*, 41:21–41, 2014.

[21] E. Jarlebring, K. Meerbergen, and W. Michiels. Computing a partial Schur factorization of nonlinear eigenvalue problems using the infinite Arnoldi method. *SIAM Journal on Matrix Analysis and Applications*, 35(2):411–436, 2014.

[22] E. Jarlebring, W. Michiels, and K. Meerbergen. A linear eigenvalue algorithm for the nonlinear eigenvalue problem. *Numerische Mathematik*, 122(1):169–195, 2012.

[23] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK Users' Guide: Solution of Large-scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, volume 6. SIAM, Philadelphia, 1998.

[24] X. S. Li. An overview of SuperLU: algorithms, implementation, and user interface. *ACM Transactions on Mathematical Software (TOMS)*, 31(3):302–325, 2005.

[25] B. S. Liao, Z. Bai, L. Q. Lee, and K. Ko. Nonlinear Rayleigh-Ritz iterative method for solving large scale nonlinear eigenvalue problems. *Taiwanese Journal of Mathematics*, 14(3A):pp–869, 2010.

[26] K. Louis. *What Every Engineer Should Know about Computational Techniques of Finite Element Analysis*. CRC Press, 2005.

[27] Y. Y. Lu. A Padé approximation method for square roots of symmetric positive definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 19(3):833–845, 1998.

[28] D. S. Mackey, N. Mackey, C. Mehl, and V. Mehrmann. Vector spaces of linearizations for matrix polynomials. *SIAM Journal on Matrix Analysis and Applications*, 28(4):971–1004, 2006.

[29] K. Meerbergen. The quadratic Arnoldi method for the solution of the quadratic eigenvalue problem. *SIAM Journal on Matrix Analysis and Applications*, 30(4):1463–1482, 2008.

[30] K. Meerbergen, A. Spence, and D. Roose. Shift-invert and Cayley transforms for detection of rightmost eigenvalues of nonsymmetric matrices. *BIT Numerical Mathematics*, 34(3):409–423, 1994.

[31] V. Mehrmann and H. Voss. Nonlinear eigenvalue problems: A challenge for modern eigenvalue methods. *Mitt. der Ges. f. Ang. Mathematik und Mechanik*, 27:121–151, 2005.

[32] R. S. Puri. *Krylov Subspace Based Direct Projection Techniques for Low Frequency, Fully Coupled, Structural Acoustic Analysis and Optimization*. PhD thesis, Oxford Brookes Universiy, 2008.

[33] A. Ruhe. Algorithms for the nonlinear eigenvalue problem. *SIAM Journal on Numerical Analysis*, 10(4):674–689, 1973.

[34] G. L. G. Sleijpen, A. G. L. Booten, D.R. Fokkema, and H. A. Van der Vorst. Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems. *BIT Numerical Mathematics*, 36(3):595–633, 1996.

[35] Y. Su and Z. Bai. Solving rational eigenvalue problems via linearization. *SIAM Journal on Matrix Analysis and Applications*, 32(1):201–216, 2011.

[36] L. Taslaman. An algorithm for quadratic eigenproblems with low rank damping. Technical report, MIMS Preprint: 2014.21, available at `http://eprints.ma.man.ac.uk/2132/01/covered/MIMS_ep2014_21.pdf`. Accessed on May 20, 2014.

[37] F. Tisseur. Backward error and condition of polynomial eigenvalue problems. *Linear Algebra and its Applications*, 309(1):339–361, 2000.

[38] R. Van Beeumen, K. Meerbergen, and W. Michiels. A rational Krylov method based on Hermite interpolation for nonlinear eigenvalue problems. *SIAM Journal on Scientific Computing*, 35(1):A327–A350, 2013.

[39] L. Zeng and Y. Su. A backward stable algorithm for quadratic eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 35(2):499–516, 2014.