SIAM J. MATRIX ANAL. APPL.
Vol. 32, No. 1, pp. 201–216

SOLVING RATIONAL EIGENVALUE PROBLEMS VIA LINEARIZATION*

YANGFENG SU† AND ZHAOJUN BAI‡

Abstract. The rational eigenvalue problem is an emerging class of nonlinear eigenvalue problems arising from a variety of physical applications. In this paper, we propose a linearization-based method to solve the rational eigenvalue problem. The proposed method converts the rational eigenvalue problem into a well-studied linear eigenvalue problem, and meanwhile, exploits and preserves the structure and properties of the original rational eigenvalue problem. For example, the low-rank property leads to a trimmed linearization. We show that solving a class of rational eigenvalue problems is just as convenient and efficient as solving linear eigenvalue problems.

Key words. rational eigenvalue problem, linearization, nonlinear eigenvalue problem

AMS subject classifications. 65F15, 65F50, 15A18

DOI. 10.1137/090777542

1. Introduction.

In recent years, there has been a great deal of interest in studying the rational eigenvalue problem (REP)

$$R(\lambda)x = 0, \tag{1.1}$$

where $R(\lambda)$ is an $n \times n$ matrix rational function of the form

$$R(\lambda) = P(\lambda) - \sum_{i=1}^{k} \frac{s_i(\lambda)}{q_i(\lambda)} E_i, \tag{1.2}$$

$P(\lambda)$ is an $n \times n$ matrix polynomial in λ of degree d, $s_i(\lambda)$ and $q_i(\lambda)$ are scalar polynomials of degrees n_i and d_i, respectively, and E_i are $n \times n$ constant matrices. The REP (1.2) arises from optimization of acoustic emissions of high speed trains [14], free vibration of plates with elastically attached masses [23], vibration of fluid-solid structure [25], free vibrations of a structure with a viscoelastic constitutive relation describing the behavior of a material [17], and electronic structure calculations of quantum dots [27, 11].

A brute-force approach to solve the REP (1.1) is to multiply (1.1) by the scalar polynomial $\prod_{i=1}^{k} q_i(\lambda)$ to turn it into a polynomial eigenvalue problem (PEP) of the degree $d_* = d + d_1 + \cdots + d_k$. Subsequently, the PEP is converted into a linear eigenvalue problem (LEP) by the process known as linearization. This approach is noted in [17] and employed in [11, 10] for electronic structure calculations of quantum

*Received by the editors November 18, 2009; accepted for publication (in revised form) January 20, 2011; published electronically March 9, 2011. Support for this work was provided in part by the National Science Foundation under grants DMS-0611548 and OCI-0749217 and by the DOE under grant DE-FC02-06ER25794.

http://www.siam.org/journals/simax/32-1/77754.html

†School of Mathematical Sciences, Fudan University, Shanghai 200433, China, affiliated with Scientific Computing Key Laboratory of Shanghai Normal University, Division of Computational Science, E-Institute of Shanghai Universities, Peoples Republic of China (yfsu@fudan.edu.cn). This author's work was supported in part by China NSF Project 10871049 and E-Institutes of Shanghai Municipal Education Commission, N. E03004.

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

dots. The recent study of the linearization techniques of the PEP can be found in [14, 15, 8, 9] and references therein. This is a practical approach only if the number k of the rational terms and the degree d_i of the polynomials $q_i(\lambda)$ are small, say, $k = d_1 = 1$. However, when k and/or d_i are large, it leads to a PEP of much higher degree, and this approach becomes impractical for large-scale problems. Furthermore, the possible low-rank property of the matrices E_i is lost in the linearized eigenvalue problem.

An alternative approach is to treat the REP (1.1) as a general nonlinear eigenvalue problem (NEP) and to solve it by a nonlinear eigensolver, such as Picard iteration (self-consistent iteration), Newton's method, nonlinear Rayleigh quotient method, nonlinear Jacobi–Davidson method, and nonlinear Arnoldi method [17, 19, 23, 26]. This approach limits the exploitation of the underlying rich structure and property of the REP (1.1), and it is challenging in convergence analysis and validation of computed eigenvalues.

In this paper, we propose a linearization-based approach to solve the REP (1.1). Similar to the linearization of the PEP, the new approach converts the REP (1.1) into an LEP and exploits and preserves the structure and property of the REP (1.1) as much as possible. It has a number of advantages. For example, the low-rank property of matrices E_i as frequently encountered in applications leads to a trimmed linearization, namely, only a small increase of the size compared to the size of the original REP (1.1). The symmetry of the REP can also be preserved in the LEP. We show that under mild assumptions, the problem of solving a class of the REPs is just as convenient and efficient as the problem of solving an LEP of slightly larger size.

The rest of this paper is organized as follows. In section 2, we formalize the definition of the REP (1.1) and the assumptions we will use throughout. In section 3, we present a linearization scheme. In section 4, we show how to use the proposed linearization scheme on a number of REPs from different applications. Numerical examples are given in section 5.

2. Settings. We assume throughout that the matrix rational function $R(\lambda)$ is *regular*; that is, $\det(R(\lambda)) \not\equiv 0$. The roots of $q_i(\lambda)$ are the *poles* of $R(\lambda)$. $R(\lambda)$ is not defined on these poles. A scalar λ such that $\det(R(\lambda)) = 0$ is referred to as an *eigenvalue*, and a corresponding nonzero vector x satisfying (1.1) is called an *eigenvector*. The pair (λ, x) is referred to as an eigenpair.

Let us denote the matrix polynomial $P(\lambda)$ of degree d in λ as

$$P(\lambda) = \lambda^d A_d + \lambda^{d-1} A_{d-1} + \cdots + \lambda A_1 + A_0, \tag{2.1}$$

where A_i are $n \times n$ constant matrices. We assume throughout that the leading coefficient matrix A_d is *nonsingular*, which is equivalent to the assumption of a monic matrix polynomial ($A_d = I$) in the study of matrix polynomials [6]. The treatment in the presence of singular A_d is beyond the scope of this paper.

We assume that $s_i(\lambda)$ and $q_i(\lambda)$ are *coprime*, that is, having no common factors. Furthermore, the rational functions $\frac{s_i(\lambda)}{q_i(\lambda)}$ are *proper*, that is, $s_i(\lambda)$ having smaller degree than $q_i(\lambda)$. Otherwise, by the polynomial long division, an improper rational function can be written as the sum of a polynomial and a proper rational function:

$$\frac{s_i(\lambda)}{q_i(\lambda)} = p_i(\lambda) + \frac{\widehat{s}_i(\lambda)}{q_i(\lambda)}$$

with $\widehat{s}_i(\lambda)$ having smaller degree than $q_i(\lambda)$. Subsequently, the term $p_i(\lambda)E_i$ can be absorbed into the matrix polynomial term $P(\lambda)$:

$$P(\lambda) := P(\lambda) - p_i(\lambda)E_i.$$

In this case, the leading coefficient matrix of the updated matrix polynomial is assumed to be nonsingular.

Since $s_i(\lambda)$ and $q_i(\lambda)$ are coprime, the proper rational function $\frac{s_i(\lambda)}{q_i(\lambda)}$ can be represented as

$$\frac{s_i(\lambda)}{q_i(\lambda)} = a_i^T(C_i - \lambda D_i)^{-1}b_i \tag{2.2}$$

for some matrices $C_i, D_i \in \mathbb{R}^{d_i \times d_i}$, where D_i is nonsingular, and vectors $a_i, b_i \in \mathbb{R}^{d_i \times 1}$. The process of constructing the quadruple (C_i, D_i, a_i, b_i) satisfying (2.2) is called a *minimal realization* in the theory of control system; see, for example, [1, pp. 91–98] and [21].

Finally, we assume that the coefficient matrices E_i have the rank-revealing decompositions

$$E_i = L_i U_i^T, \tag{2.3}$$

where $L_i, U_i \in \mathbb{R}^{n \times r_i}$ are of full column rank r_i. In section 4, we will see that the decompositions (2.3) are often immediately available in the practical REPs. The rank of E_i is typically much smaller than the size n; that is, $r_i \ll n$. The algorithms for computing such sparse rank-revealing decompositions can be found in [18].

3. Linearization. Under the assumptions presented in the previous section, let us consider a linearization method for solving the REP (1.1). By the realizations (2.2) of the rational functions $s_i(\lambda)/q_i(\lambda)$ and the factorizations (2.3) of the coefficient matrices E_i, the rational terms of the matrix rational function $R(\lambda)$ can be rewritten as the following:

$$\begin{aligned}
\sum_{i=1}^{k} \frac{s_i(\lambda)}{q_i(\lambda)} E_i &= \sum_{i=1}^{k} a_i^T(C_i - \lambda D_i)^{-1} b_i L_i U_i^T \\
&= \sum_{i=1}^{k} L_i \left[a_i^T(C_i - \lambda D_i)^{-1} b_i \cdot I_{r_i} \right] U_i^T \\
&= \sum_{i=1}^{k} L_i (I_{r_i} \otimes a_i)^T (I_{r_i} \otimes C_i - \lambda\, I_{r_i} \otimes D_i)^{-1} (I_{r_i} \otimes b_i) U_i^T,
\end{aligned}$$

where $\otimes$ is the Kronecker product. Define

$$\begin{aligned}
C &= \operatorname{diag}(I_{r_1} \otimes C_1,\, I_{r_2} \otimes C_2,\, \ldots,\, I_{r_k} \otimes C_k), \\
D &= \operatorname{diag}(I_{r_1} \otimes D_1,\, I_{r_2} \otimes D_2,\, \ldots,\, I_{r_k} \otimes D_k), \\
L &= \begin{bmatrix} L_1(I_{r_1} \otimes a_1)^T & L_2(I_{r_2} \otimes a_2)^T & \cdots & L_k(I_{r_k} \otimes a_k)^T \end{bmatrix}, \\
U &= \begin{bmatrix} U_1(I_{r_1} \otimes b_1)^T & U_2(I_{r_2} \otimes b_2)^T & \cdots & U_k(I_{r_k} \otimes b_k)^T \end{bmatrix},
\end{aligned}$$

where the size of C and D is $m \times m$, the size of L and U is $n \times m$, and $m = r_1 d_1 + r_2 d_2 + \cdots + r_k d_k$. Then the rational terms of $R(\lambda)$ can be compactly represented

in a realization form

$$\sum_{i=1}^{k} \frac{s_i(\lambda)}{q_i(\lambda)} E_i = L(C - \lambda D)^{-1} U^T. \tag{3.1}$$

We note that the matrix D is nonsingular since the matrices D_i in (2.2) are nonsingular. The eigenvalues of the matrix pencil $C - \lambda D$ are the poles of $R(\lambda)$.

Using the representation (3.1), the REP (1.1) can be equivalently written in the following compact form

$$\left[P(\lambda) - L(C - \lambda D)^{-1} U^T\right] x = 0, \tag{3.2}$$

and the matrix rational function $R(\lambda)$ is written as

$$R(\lambda) = P(\lambda) - L(C - \lambda D)^{-1} U^T. \tag{3.3}$$

If $P(\lambda)$ is linear and is denoted as $P(\lambda) = A - \lambda B$, then the REP (3.2) is of the form

$$\left[A - \lambda B - L(C - \lambda D)^{-1} U^T\right] x = 0. \tag{3.4}$$

By introducing the auxiliary vector

$$y = -(C - \lambda D)^{-1} U^T x,$$

(3.2) can be written as the following LEP:

$$(\mathcal{A} - \lambda \mathcal{B}) z = 0, \tag{3.5}$$

where

$$\mathcal{A} = \begin{bmatrix} A & L \\ U^T & C \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} B & \\ & D \end{bmatrix}, \quad z = \begin{bmatrix} x \\ y \end{bmatrix}.$$

In general, if the matrix polynomial $P(\lambda)$ is of the form (2.1), we can first write the REP (3.2) as a "PEP" of the form

$$\left(\lambda^d A_d + \lambda^{d-1} A_{d-1} + \cdots + \lambda A_1 + \widetilde{A}_0(\lambda)\right) x = 0, \tag{3.6}$$

where $\widetilde{A}_0(\lambda) \triangleq A_0 - L(C - \lambda D)^{-1} U^T$. Then by symbolically applying the well-known (first) companion form linearization to (3.6), we have

$$\left(\begin{bmatrix} A_{d-1} & A_{d-2} & \cdots & \widetilde{A}_0(\lambda) \\ -I & 0 & \cdots & 0 \\ & \ddots & \ddots & \vdots \\ & & -I & 0 \end{bmatrix} - \lambda \begin{bmatrix} A_d & & & \\ & I & & \\ & & \ddots & \\ & & & I \end{bmatrix} \right) \begin{bmatrix} \lambda^{d-1} x \\ \lambda^{d-2} x \\ \vdots \\ x \end{bmatrix} = 0, \tag{3.7}$$

which can be equivalently written as

$$\left(\begin{bmatrix} A_{d-1} & A_{d-2} & \cdots & A_0 \\ -I & 0 & \cdots & 0 \\ & \ddots & \ddots & \vdots \\ & & -I & 0 \end{bmatrix} - \lambda \begin{bmatrix} A_d & & & \\ & I & & \\ & & \ddots & \\ & & & I \end{bmatrix} - \begin{bmatrix} L \\ 0 \\ \vdots \\ 0 \end{bmatrix} (C - \lambda D)^{-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ U \end{bmatrix}^T \right) \begin{bmatrix} \lambda^{d-1} x \\ \lambda^{d-2} x \\ \vdots \\ x \end{bmatrix} = 0.$$

The above equation is of the same form as (3.4). Therefore, by introducing the variable

$$y = -(C-\lambda D)^{-1}\begin{bmatrix} 0 & 0 & \cdots & U^T \end{bmatrix}\begin{bmatrix} \lambda^{d-1}x \\ \lambda^{d-2}x \\ \vdots \\ x \end{bmatrix} = -(C-\lambda D)^{-1}U^T x,$$

we derive the following linearization of the REP (1.1):

$$(\mathcal{A}-\lambda\mathcal{B})z = 0, \tag{3.8}$$

where

$$\mathcal{A} = \left[\begin{array}{cccc|c} A_{d-1} & A_{d-2} & \cdots & A_0 & L \\ -I & 0 & \cdots & 0 & \\ & \ddots & \ddots & \vdots & \\ & & -I & 0 & \\ \hline & & & U^T & C \end{array}\right], \quad \mathcal{B} = -\left[\begin{array}{cccc|c} A_d & & & & \\ & I & & & \\ & & \ddots & & \\ & & & I & \\ \hline & & & & -D \end{array}\right],$$

$$z = \begin{bmatrix} \lambda^{d-1}x \\ \lambda^{d-2}x \\ \vdots \\ x \\ \hline y \end{bmatrix}.$$

The size of matrices $\mathcal{A}$ and $\mathcal{B}$ is $nd+m$, where $m = r_1d_1 + r_2d_2 + \cdots + r_kd_k$. In the case that all the coefficient matrices E_i are of full rank, i.e., $r_i = n$, the LEP (3.8) is of the size nd_*, where $d_* = d + d_1 + \cdots + d_k$. This is the same size as the one derived by the brute-force approach. However, it is typical that $r_i \ll n$ in practice; then $nd+m \ll nd_*$. The LEP (3.8) is a trimmed linearization of the REP (1.1). This will be illustrated by four REPs from applications in section 4.

Note that under the assumption of nonsingularity of the matrix A_d, the matrix $\mathcal{B}$ is nonsingular. Therefore all eigenvalues of the LEP (3.8) are finite. There is no infinite eigenvalue.

The following theorem shows the connection between eigenvalues of the REP (1.1) and the LEP (3.8).

THEOREM 3.1.

(a) *If λ is an eigenvalue of the REP* (1.1)*, then it is an eigenvalue of the LEP* (3.8)*.*

(b) *Let λ be an eigenvalue of the LEP* (3.8) *and be not a pole of $R(\lambda)$, $z = [z_1^T, z_2^T, \cdots, z_d^T, y^T]^T$ be the corresponding eigenvector, where z_i are vectors of length n for $i = 1, 2, \ldots n$. Then $z_d \neq 0$ and $R(\lambda)z_d = 0$, namely, λ is an eigenvalue of the REP* (1.1) *and z_d is the corresponding eigenvector. Moreover, the algebraic and geometric multiplicities of λ for REP* (1.1) *and LEP* (3.8) *are the same.*

Proof. Define dn-by-dn matrices

$$V_L(\lambda) = \begin{bmatrix} I & -\lambda A_d - A_{d-1} & -A_{d-2} & \cdots & -A_1 \\ & I & -\lambda I & & \\ & & \ddots & \ddots & \\ & & & \ddots & -\lambda I \\ & & & & I \end{bmatrix},$$

$$V_R(\lambda) = \begin{bmatrix} \lambda^{d-1} I & -I & & \\ \vdots & & \ddots & \\ \lambda I & & & -I \\ I & & & \end{bmatrix}.$$

We have $\det(V_L(\lambda)) = \det(V_R(\lambda)) = 1$ and

$$\begin{bmatrix} \lambda A_d + A_{d-1} & A_{d-2} & \cdots & A_0 \\ -I & \lambda I & & \\ & \ddots & \ddots & \\ & & -I & \lambda I \end{bmatrix} V_R(\lambda) = V_L(\lambda) \begin{bmatrix} P(\lambda) & & & \\ & I & & \\ & & \ddots & \\ & & & I \end{bmatrix}.$$

Therefore,

$$\begin{aligned} \det(\mathcal{A} - \lambda\mathcal{B}) &= \det\left((\mathcal{A} - \lambda\mathcal{B}) \left[\begin{array}{c|c} V_R(\lambda) & \\ \hline & I_m \end{array}\right] \right) \\ &= \det\left(\left[\begin{array}{cccc|c} \lambda A_d + A_{d-1} & A_{d-2} & \cdots & A_0 & L \\ -I & \lambda I & & & \\ & \ddots & \ddots & & \\ & & -I & \lambda I & \\ \hline & & & U^T & C - \lambda D \end{array}\right] \left[\begin{array}{c|c} V_R(\lambda) & \\ \hline & I_m \end{array}\right] \right) \\ &= \det\left(\left[\begin{array}{c|c} V_L(\lambda) & \\ \hline & I_m \end{array}\right] \left[\begin{array}{ccc|c} P(\lambda) & & & L \\ & I & & \\ & & \ddots & \\ & & I & \\ \hline U^T & & & C - \lambda D \end{array}\right] \right) \\ &= \det\left(\left[\begin{array}{ccc|c} P(\lambda) & & & L \\ & I & & \\ & & \ddots & \\ & & I & \\ \hline U^T & & & C - \lambda D \end{array}\right] \right), \end{aligned} \tag{3.9}$$

where, for the third equality, we use the identities

$$\begin{bmatrix} 0 & \cdots & 0 & U^T \end{bmatrix} V_R(\lambda) = \begin{bmatrix} U^T & 0 & \cdots & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} L \\ 0 \\ \vdots \\ 0 \end{bmatrix} = V_L(\lambda) \begin{bmatrix} L \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

By exploiting the block structure of the matrix in the determinant of the right-hand side of (3.9), we derive

$$\det(\mathcal{A}-\lambda\mathcal{B}) = \det\left(\begin{bmatrix} P(\lambda) & L \\ U^T & C-\lambda D \end{bmatrix}\right). \tag{3.10}$$

If λ is an eigenvalue of the REP (1.1), then it is not a pole of the REP. It implies that λ is not an eigenvalue of $C-\lambda D$ and $C-\lambda D$ is nonsingular. Therefore, we have the block factorization

$$\begin{bmatrix} P(\lambda) & L \\ U^T & C-\lambda D \end{bmatrix} = \begin{bmatrix} I & L(C-\lambda D)^{-1} \\ & I \end{bmatrix} \begin{bmatrix} R(\lambda) & \\ U^T & C-\lambda D \end{bmatrix},$$

where $R(\lambda)$ is defined by (3.3). The proof of (a) immediately follows from the identity

$$\det(\mathcal{A}-\lambda\mathcal{B}) = \det(R(\lambda)) \cdot \det(C-\lambda D). \tag{3.11}$$

For (b), if (λ, z) is an eigenpair of the pencil $\mathcal{A}-\lambda\mathcal{B}$ and λ is not a pole of $R(\lambda)$, then the matrix $C-\lambda D$ is nonsingular. From the block forms of $\mathcal{A}$ and $\mathcal{B}$, we know that $z_i = \lambda^{d-i} z_d$ for $i = 1, 2, \ldots, d-1$, and

$$\sum_{i=0}^{d} \lambda^i A_i z_d + Ly = 0,$$
$$U^T z_d + (C-\lambda D)y = 0.$$

Since $C-\lambda D$ is nonsingular, we have

$$y = -(C-\lambda D)^{-1} U^T z_d,$$

and

$$\left(P(\lambda) - L(C-\lambda D)^{-1}U^T\right) z_d = R(\lambda) z_d = 0.$$

If $z_d = 0$, then $z_i = \lambda^{d-i} z_d = 0$ for $i = 1, 2, \ldots, d-1$ and $y = -(C-\lambda D)^{-1}U^T z_d = 0$. Consequently, $z = 0$. This contradicts the fact that z is the eigenvector of $\mathcal{A}-\lambda\mathcal{B}$. Hence λ is an eigenvalue of $R(\lambda)$, and $z_d \neq 0$ is the corresponding eigenvector.

Furthermore, the result that the multiplicities of λ for REP (1.1) and LEP (3.8) are the same is a direct consequence of identity (3.11). ∎

We note that the condition that λ is not a pole of the $R(\lambda)$ in Theorem 3.1(b) is necessary. Consider the following example:

$$\left(\lambda I_2 - \frac{1}{\lambda} e_2 e_2^T\right) x = 0, \tag{3.12}$$

where I_2 is the 2-by-2 identity matrix, and e_2 is the second column of I_2. Since $\det(R(\lambda)) = \lambda(\lambda - 1/\lambda)$, the REP (3.12) has two eigenvalues, 1 and -1. Moreover, $\lambda = 0$ is a pole. Let $y = \lambda^{-1} e_2^T x$; then the corresponding LEP is given by

$$(\mathcal{A}-\lambda\mathcal{B})z = \left(\left[\begin{array}{c|c} 0 & e_2 \\ \hline e_2^T & 0 \end{array}\right] - \lambda \left[\begin{array}{c|c} I_2 & \\ \hline & 1 \end{array}\right]\right) \begin{bmatrix} x \\ y \end{bmatrix} = 0.$$

It has three eigenvalues, -1, 0, and 1. But $\lambda = 0$ is not an eigenvalue of the REP (3.12).

Two additional remarks are in order. First, the realization of a rational function can be represented in different forms. For example, the realization of $1/(\sigma-\lambda)^2$ could be given by

$$\frac{1}{(\sigma-\lambda)^2} = \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\begin{bmatrix} \sigma & -1 \\ 0 & \sigma \end{bmatrix} - \lambda I \right)^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

or in a symmetric form

$$\frac{1}{(\sigma-\lambda)^2} = \begin{bmatrix} 1 & 0 \end{bmatrix} \left(\begin{bmatrix} & \sigma \\ \sigma & -1 \end{bmatrix} - \lambda \begin{bmatrix} & 1 \\ 1 & \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

The study of realization can be found in [1, pp. 91–98] and references therein.

Second, there are many different ways of linearization for the matrix polynomials, including recent work [8, 9, 14, 15]. Many of these linearizations can be easily integrated into the proposed linearization of the REP. For example, if the original REP (1.1) is symmetric, namely, A_i are symmetric, matrices C and D in the realization are symmetric, and $L = U$, then we can use a symmetric linearization proposed in [8]. Specifically, let us consider a symmetric matrix polynomial of degree $d = 3$; then a symmetric linearization is given by

$$\left(\begin{bmatrix} & & -A_3 & \\ & -A_3 & -A_2 & \\ -A_3 & -A_2 & & \\ & & A_0 & U \\ & & U^T & C \end{bmatrix} + \lambda \begin{bmatrix} & & & A_3 & \\ & & A_3 & A_2 & \\ & A_3 & A_2 & A_1 & \\ A_3 & A_2 & A_1 & & \\ & & & & -D \end{bmatrix} \right) \begin{bmatrix} \lambda^2 x \\ \lambda x \\ x \\ y \end{bmatrix} = 0, \tag{3.13}$$

where $y = -(C - \lambda D)^{-1} U^T x$.

4. Applications. In this section, we apply the proposed linearization in section 3 to four REPs from applications.

4.1. Loaded elastic string. We consider the following REP arising from the finite element discretization of a boundary value problem describing the eigenvibration of a string with a load of mass attached by an elastic spring:

$$\left(A - \lambda B + \frac{\lambda}{\lambda - \sigma} E \right) x = 0, \tag{4.1}$$

where A and B are tridiagonal and symmetric positive definite, and $E = e_n e_n^T$, e_n is the last column of the identity matrix. $\sigma > 0$ is a parameter [3, 23].

By the linearization proposed in section 3, the first step is to write the rational function $\lambda/(\lambda - \sigma)$ in a proper form. It results that REP (4.1) becomes

$$\left(A + e_n e_n^T - \lambda B + \frac{\sigma}{\lambda - \sigma} e_n e_n^T \right) x = 0. \tag{4.2}$$

Then it can be easily written in the realization form (3.4):

$$\left[A + e_n e_n^T - \lambda B - e_n \left(1 - \frac{\lambda}{\sigma} \right)^{-1} e_n^T \right] x = 0. \tag{4.3}$$

By defining the auxiliary vector

$$y = -\left(1 - \frac{\lambda}{\sigma} \right)^{-1} e_n^T x,$$

we have the LEP

$$(\mathcal{A} - \lambda\mathcal{B})z = 0, \tag{4.4}$$

where

$$\mathcal{A} = \begin{bmatrix} A + e_n e_n^T & e_n \\ e_n^T & 1 \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} B & \\ & 1/\sigma \end{bmatrix}, \quad z = \begin{bmatrix} x \\ y \end{bmatrix}.$$

The $(n+1) \times (n+1)$ matrices $\mathcal{A}$ and $\mathcal{B}$ have the same structure and property as the coefficient matrices A and B in the original REP (4.1). They are tridiagonal and symmetric positive definite.

An alternative way to solve REP (4.1) is to first transform it to a quadratic eigenvalue problem (QEP) by multiplying the linear factor $\lambda - \sigma$ on (4.1):

$$Q(\lambda)x = \left[\lambda^2 B - \lambda(A + \sigma B + e_n e_n^T) + \sigma A\right] x = 0. \tag{4.5}$$

Then convert the QEP (4.5) into an equivalent LEP by the linearization process. A symmetric linearization is

$$\left(\begin{bmatrix} -(A + \sigma B + e_n e_n^T) & \sigma A \\ \sigma A & \end{bmatrix} + \lambda \begin{bmatrix} B & \\ & -\sigma A \end{bmatrix} \right) \begin{bmatrix} \lambda x \\ x \end{bmatrix} = 0. \tag{4.6}$$

This is a generalized symmetric indefinite eigenvalue problem. If $Q(a) < 0$ for some scalar a, then, by letting $\lambda = \mu + a$, QEP (4.5) becomes

$$\left[\mu^2 B - \mu(A + (\sigma - 2a)B + e_n e_n^T) + Q(a)\right] x = 0.$$

Consequently, it can be linearized to a generalized symmetric definite eigenvalue problem:

$$\left(\begin{bmatrix} -\left(A + (\sigma - 2a)B + e_n e_n^T\right) & Q(a) \\ Q(a) & 0 \end{bmatrix} + \mu \begin{bmatrix} B & \\ & -Q(a) \end{bmatrix} \right) \begin{bmatrix} \mu x \\ x \end{bmatrix} = 0. \tag{4.7}$$

A practical issue is on the existence of the shift a and how to find it numerically; see recent work [7].

We note that the size of LEP (4.6) and LEP (4.7) is $2n$. On the other hand, the size of LEP (4.4) by the new linearization process is only $n + 1$.

4.2. QEP with low-rank stiffness matrix. Consider the QEP

$$(\lambda^2 M + \lambda D + K)x = 0. \tag{4.8}$$

If K is singular, then zero is an eigenvalue since $Kx = 0$ for some nonzero vector x. Let us consider how to compute nonzero eigenvalues of QEP (4.8) by exploiting the fact that K is singular and is of low rank. Let

$$K = LU^T$$

be the full-rank decomposition, where $L, U \in \mathbb{R}^{n \times r}$, r is the rank of K. By the linearization discussed in section 3, QEP (4.8) can be written in the REP form (1.1):

$$\left(\lambda M + D + \frac{1}{\lambda} LU^T\right) x = 0.$$

In the compact form (3.2), it becomes

$$\left[\lambda M + D - L(0 - \lambda I)^{-1}U^T\right] x = 0.$$

Note that the polynomial term $P(\lambda) = \lambda M + D$ is linear. Hence, by introducing the auxiliary vector

$$y = -(0 - \lambda I)^{-1}U^T x = \lambda^{-1}U^T x,$$

we have the LEP

$$\left(\lambda \begin{bmatrix} M & \\ & I \end{bmatrix} + \begin{bmatrix} D & L \\ -U^T & 0 \end{bmatrix}\right) \begin{bmatrix} x \\ y \end{bmatrix} = 0. \tag{4.9}$$

If $L = K, U = I$, then LEP (4.9) is a linearization of the QEP in the first companion form. If $L = -I, U = -K^T$, then LEP (4.9) is a linearization in the second companion form [12]. If K has rank $r < n$, the linearization (4.9) is not a linearization of QEP (4.8) under the standard definition of linearization of the PEPs [6]. However, by Theorem 3.1, we can conclude that the nonzero eigenvalues of QEP (4.8) and LEP (4.9) are the same. LEP (4.9) is a trimmed linearization of the QEP. The order of LEP (4.9) $n + r$ could be significantly smaller than the size $2n$ of the linearization in the companion forms.

If M is singular or more general, the leading coefficient matrix A_d is singular in the matrix polynomial, then the PEP $P(\lambda) = 0$ has infinite eigenvalues and/or corresponding singular structure. It is discussed in [4] how to exploit the structure of the zero blocks in the coefficient matrices to obtain a trimmed linearization by (partially) deflating those infinite eigenvalues and/or singular structures. In the linearization (4.9), by using a full-rank factorization of the stiffness matrix K, we derived a trimmed linearization such that those zero eigenvalues are explicitly deflated.

4.3. Vibration of a fluid-solid structure. Let us consider an REP arising from the simulation of mechanical vibrations of fluid-solid structures [16, 17, 25]. It is of the form

$$\left(A - \lambda B + \sum_{i=1}^{k} \frac{\lambda}{\lambda - \sigma_i} E_i\right) x = 0, \tag{4.10}$$

where the poles $\sigma_i, i = 1, 2, \ldots, k$ are positive, the matrices A and B are symmetric positive definite, and

$$E_i = C_i C_i^T.$$

$C_i \in \mathbb{R}^{n \times r_i}$ has rank r_i for $i = 1, 2, \ldots, k$.

By the linearization proposed in section 3, we first write the rational terms of (4.10) in the proper form

$$\left(A + \sum_{i=1}^{k} C_i C_i^T - \lambda B - \sum_{i=1}^{k} \frac{\sigma_i}{\sigma_i - \lambda} C_i C_i^T\right) x = 0. \tag{4.11}$$

Let

$$C = \begin{bmatrix} C_1 & C_2 & \cdots & C_k \end{bmatrix}, \quad \Sigma = \operatorname{diag}(\sigma_1 I_{r_1}, \ldots, \sigma_k I_{r_k}),$$

where I_{r_i} is the r_i-by-r_i identity matrix. Then (4.11) can be written as

$$\left[A + CC^T - \lambda B - C(I - \lambda\Sigma^{-1})^{-1}C^T\right] x = 0.$$

By introducing the variable $y = -(I - \lambda\Sigma^{-1})^{-1}C^T x$, we have the following LEP:

$$(\mathcal{A} - \lambda\mathcal{B}) \begin{bmatrix} x \\ y \end{bmatrix} = 0, \tag{4.12}$$

where

$$\mathcal{A} = \begin{bmatrix} A + CC^T & C \\ C^T & I \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} B & \\ & \Sigma^{-1} \end{bmatrix}$$

are of the size $n + \sum_{i=1}^{k} r_i$. Note that the matrix $\mathcal{A}$ is symmetric, and $\mathcal{B}$ is symmetric positive definite. The LEP (4.12) is a generalized symmetric definite eigenvalue problem, which can be essentially solved by a symmetric eigensolver, such as implicitly restarted Lanczos algorithm [13] or the thick-restart Lanczos method [28].

Mazurenko and Voss [16] addressed the question to determine the number of eigenvalues of REP (4.10) in a given interval (α, β) by using the fact that the eigenvalues of (4.10) can be characterized as minimax values of a Rayleigh functional [16, p. 610]. By using the linearization (4.12), the question can be answered through computing the inertias of the symmetric matrix pencil $\mathcal{A} - \lambda\mathcal{B}$. First, we have the following proposition.

PROPOSITION 4.1. *If A and B are positive definite and the poles satisfy $\sigma_i > 0$, then all eigenvalues of REP* (4.10) *are real and positive.*

Proof. Since the pencil $\mathcal{A} - \lambda\mathcal{B}$ in (4.12) is a symmetric definite pencil, all eigenvalues are real. Furthermore, by the fact that $\mathcal{A}$ is the sum of two positive semidefinite matrices, namely

$$\mathcal{A} = \begin{bmatrix} A & \\ & 0 \end{bmatrix} + \begin{bmatrix} C \\ I \end{bmatrix} \begin{bmatrix} C^T & I \end{bmatrix} \geq 0,$$

all eigenvalues of the pencil $\mathcal{A} - \lambda\mathcal{B}$ are nonnegative. Therefore, we just need to show that zero is not an eigenvalue of the pencil $\mathcal{A} - \lambda\mathcal{B}$. By contradiction, let $z = \begin{bmatrix} x^T & y^T \end{bmatrix}^T \neq 0$ be an eigenvector corresponding to the zero eigenvalue. Then by $(\mathcal{A} - 0 \cdot \mathcal{B})z = \mathcal{A}z = 0$, we have

$$\begin{aligned} (A + CC^T)x + Cy &= 0, \\ C^T x + y &= 0. \end{aligned}$$

Note that A is positive definite; we have $x = 0$ and $y = 0$. This is a contradiction. Therefore all eigenvalues of LEP (4.12) are positive. By Theorem 3.1, we conclude that all eigenvalues of REP (4.10) are positive. ∎

Based on Theorem 3.1 and Proposition 4.1, we conclude that the number of eigenvalues of REP (4.10) in the interval (α, β) is given by

$$\kappa = \ell - \ell_0, \tag{4.13}$$

where ℓ is the number of eigenvalues of $\mathcal{A} - \lambda\mathcal{B}$ in the interval (α, β), $\ell_0 = \sum_{\sigma_i \in (\alpha,\beta)} \ell_i$, and ℓ_i is the number of zero eigenvalues of $\mathcal{A} - \sigma_i\mathcal{B}$. The quantities ℓ and ℓ_0 can be computed using Sylvester's law of inertia for the real symmetric matrices $\mathcal{A} - \tau\mathcal{B}$ for $\tau = \alpha, \beta$ and poles $\sigma_i \in (\alpha, \beta)$.

4.4. Damped vibration of a structure. This is an REP arising from the free vibrations of a structure if one uses a viscoelastic constitutive relation to describe the behavior of a material [17]. The REP is of the form

$$\left(\lambda^2 M + K - \sum_{i=1}^{k} \frac{1}{1+b_i\lambda}\Delta G_i\right) x = 0, \tag{4.14}$$

where the mass and stiffness matrices M and K are symmetric positive definite, b_j are relaxation parameters over the k regions, and ΔG_j is an assemblage of element stiffness matrices over the region with the distinct relaxation parameters.

We consider the case where $\Delta G_i = L_i L_i^T$ and $L_i \in \mathbb{R}^{n\times r_i}$. By defining

$$L = [L_1, L_2, \ldots, L_k], \quad D = \operatorname{diag}(b_1 I_{r_1}, b_2 I_{r_2}, \ldots, b_k I_{r_k}),$$

REP (4.14) can be written in the form (3.2):

$$(\lambda^2 M + K - L(I+\lambda D)^{-1} L^T)x = 0.$$

By linearizing the second-order matrix polynomial term $\lambda^2 M + K$ in a symmetric form, we derive the following symmetric LEP:

$$\left(\begin{bmatrix} -M & & \\ & K & L \\ & L^T & I \end{bmatrix} + \lambda \begin{bmatrix} & M & \\ M & & \\ & & D \end{bmatrix}\right) \begin{bmatrix} \lambda x \\ x \\ y \end{bmatrix} = 0,$$

where the auxiliary vector $y = -(I+\lambda D)^{-1} L^T x$. The size of the LEP is $2n + r_1 + r_2 + \cdots + r_k$.

5. Numerical examples. In this section, we present two numerical examples to show computational efficiency of the proposed linearization process of REP (1.1) in sections 3 and 4. We do not compare the proposed approach with a general-purpose nonlinear eigensolver, such as Newton's method [19], nonlinear Arnoldi method [26], or preconditioned iterative methods [23]. Instead, we compare the extra cost of solving REP (1.1) over the problem of solving the PEP $P(\lambda)x = 0$ without the rational terms in (1.1). All numerical experiments were run in MATLAB 7.0.1 on a Pentium IV PC with 2.6GHz CPU and 1GB of core memory.

Example 1. We present numerical results for REP (4.1) arising from vibration analysis of a loaded elastic string discussed in section 4.1. This REP is included in the collection of nonlinear eigenvalue problems (NLEVP) [3]. If the NLEVP is included in MATLAB , then the matrices A, B, and E can be generated by calling

```
coeffs = nlevp('loaded_string',n);
A = coeffs{1};  B = coeffs{2}; E = coeffs{3};
```

where n is the size of the REP. As in [23], the pole σ is set to be 1. The interested eigenvalues are the few smallest ones in the interval $\lambda \in (1, +\infty)$.

The following table records the 10 computed smallest eigenvalues and the corresponding residual norms of the trimmed LEP (4.4) with the size $n = 100$ by MATLAB function `eig`:

i	$\widehat{\lambda}_i$	residual norm
1	0.457318488953671	$5.58e-013$
2	4.48217654587198	$5.96e-013$
3	24.2235731125539	$6.69e-013$
4	63.7238211419405	$9.40e-013$
5	123.031221067605	$8.63e-013$
6	202.200899143561	$9.56e-013$
7	301.310162794155	$1.09e-012$
8	420.456563106511	$1.01e-012$
9	559.757586307048	$7.12e-013$
10	719.350660116386	$9.15e-013$

The residual norm $\|R(\widehat{\lambda})\widehat{x}\|_2/\|\widehat{x}\|_2$ is used to measure the precision of a computed eigenpair $(\widehat{\lambda}, \widehat{x})$ of REP (1.1), the same as in [17, Algorithm 5].

We note that the first eigenvalue $\widehat{\lambda}_1 < 1$, which is not of practical interest according to [23]. Eigenvalues $\widehat{\lambda}_2$ to $\widehat{\lambda}_6$ match all significant digits of the computed eigenvalues by a preconditioned iterative method reported in [23].

The following table reports the CPU elapsed time of solving the trimmed LEP (4.4) for different sizes n. The solver used is MATLAB dense eigensolver `eig`. For comparison, we also report the CPU elapsed time of solving QEP (4.5) by using MATLAB function `polyeig` and the symmetric LEP (4.7) of size $2n$ by using `eig`. In this particular case, it is known that $Q(a) < 0$ for $a = 2$. Therefore, the symmetric LEP (4.7) is a generalized symmetric definite eigenproblem.

	solver	$n = 200$	$n = 400$	$n = 600$	$n = 800$
Trimmed LEP (4.4)	`eig`	0.0156	0.1406	1.0625	3.5469
QEP (4.5)	`polyeig`	0.7500	6.8594	24.5313	84.9063
Full sym-LEP (4.7)	`eig`	0.0781	0.5938	2.0781	5.4219
$A - \lambda B$	`eig`	0.0156	0.1406	1.0469	3.5313

It is clear that the trimmed LEP (4.4) is the most efficient scheme to solve REP (4.1). Although one can efficiently exploit the positive definiteness in the generalized symmetric definite eigenproblem (4.7), it is still slower than the trimmed LEP (4.4) due to the fact that the size of (4.7) is doubled to $2n$. From the table, we also see that the brute-force approach to convert REP (4.1) into QEP (4.5) and then solve it via a companion form linearization is the most expensive.

Note that the matrix pair $(\mathcal{A}, \mathcal{B})$ in the trimmed linearization (4.4) is symmetric tridiagonal and positive definite, the same properties as the matrices A and B. Therefore, from the last row of the previous table we see that the CPU elapsed times are essentially the same for solving the REP via the proposed linearization and solving the eigenvalue problem of the symmetric definite pencil $A - \lambda B$ only.

Example 2. This is a numerical example for the REP (4.10) discussed in section 4.3. The size of matrices A and B is $n = 36,046$. The number of nonzeros of A is $nnz = 255,088$. B has the same sparsity as A. There are nine rational terms, $k = 9$. The matrices C_i in rational terms have two dense columns and are of rank $r_i = 2$. The pole $\sigma_i = i$ for $i = 1, 2, \ldots, k$. Our aim is to compute all eigenvalues in the interval $(\alpha, \beta) = (1, 2)$, i.e., between the first and second poles.

As we discussed in section 4, the linearization of REP (4.10) leads to the LEP $(\mathcal{A} - \lambda\mathcal{B})z = 0$, where $\mathcal{A}$ and $\mathcal{B}$ are defined as in (4.12) with the size $n + r_1 d_1 + \cdots + r_k d_k = n + 2 \times 9 = n + 18$.

By the expression (4.13), we can conclude that there are 8 eigenvalues in the interval. To apply the expression (4.13), we need to know the inertias of the matrices $\mathcal{A} - \tau\mathcal{B}$ for $\tau = \alpha, \beta$. These can be computed using the $\mathrm{LDL}^{\mathrm{T}}$ decomposition of the

matrix $\mathcal{A}-\tau\mathcal{B}$. Since C_i are dense, the explicit computation of the LDL$^{\mathrm{T}}$ decomposition of $\mathcal{A}-\tau\mathcal{B}$ is too expensive. Instead, we can first perform the following congruence transformation:

$$\left[\begin{array}{c|c} \mathcal{A}-\tau\mathcal{B} & \\ \hline & -I \end{array}\right] = \mathcal{L}_1 \left[\begin{array}{cc|c} A-\tau B & & C \\ & -\tau\Sigma^{-1} & I \\ \hline C^T & I & -I \end{array}\right] \mathcal{L}_1^T \tag{5.1}$$

$$= \mathcal{L}_1\mathcal{L}_2 \left[\begin{array}{ccc} A-\tau B & & \\ & -\tau\Sigma^{-1} & \\ & & F \end{array}\right] \mathcal{L}_2^T\mathcal{L}_1^T,$$

where $F = -I - C^T(A-\tau B)^{-1}C + \Sigma/\tau$,

$$\mathcal{L}_1 = \left[\begin{array}{cc|c} I & & C \\ & I & I \\ \hline & & I \end{array}\right], \quad \mathcal{L}_2 = \left[\begin{array}{cc|c} I & & \\ & I & \\ \hline C^T(A-\tau B)^{-1} & -\tau^{-1}\Sigma & I \end{array}\right].$$

Under these congruence transformations, we see that the inertias of the matrices $\mathcal{A}-\tau\mathcal{B}$ can be computed from the inertias of the matrices $A-\tau B$, $-\tau^{-1}\Sigma$ and F. Therefore, we only need to compute the LDL$^{\mathrm{T}}$ decomposition of the sparse symmetric matrix $A-\tau B$. In MATLAB , the LDL$^{\mathrm{T}}$ decomposition of a sparse symmetric matrix is computed by the function `ldlsparse`, which is based on [5].

Let us turn to compute the 8 eigenvalues in the interval (α,β). Many large-scale eigensolvers exist; see [2] and references therein. Specifically, for the symmetric LEP (4.12), one can use the Lanczos method [28], Jacobi-Davidson methods [22], and subspace iteration [20, 24]. We here apply MATLAB 's sparse eigensolver `eigs`. The function `eigs` is based on the implicitly restarted Arnoldi algorithm [13]. The following parameters are used for applying the function `eigs` with the shift-and-invert spectral transformation:

```
tau = 1.5;           % the shift
num = 8;             % number of wanted eigenvalues
opts.issym = true;
opts.isreal = true;
opts.disp = 1;
opts.tol = 1.e-13;   % residual bound
opts.p = 4*num;      % number of Lanczos basis vectors
```

Furthermore, we need to provide an external linear solver for the linear system

$$\left(\left[\begin{array}{cc} A-\tau B & \\ & -\tau\Sigma^{-1} \end{array}\right] + \left[\begin{array}{c} C \\ I \end{array}\right][C^T\ I]\right)\left[\begin{array}{c} x_1 \\ x_2 \end{array}\right] = \left[\begin{array}{c} b_1 \\ b_2 \end{array}\right]. \tag{5.2}$$

The following procedure is such a solver to compute the solution vectors x_1 and x_2 based on the LDL$^{\mathrm{T}}$ decomposition $A-\tau B = LDL^T$ and the Sherman-Morrison-Woodbury formula[1]:

1. $C := L^{-1}C$;
2. $b_1 := L^{-1}b_1$;

[1] $(A+BB^T)^{-1} = A^{-1} - A^{-1}B(I+B^TA^{-1}B)^{-1}B^TA^{-1}$.

3. $e = C^T D^{-1} b_1 - \Sigma b_2/\tau$;
4. $e := (I + C^T D^{-1} C - \Sigma/\tau)^{-1} e$;
5. $x_1 = L^{-T} D^{-1}(b_1 - Ce), x_2 = -\Sigma(b_2 - e)/\tau$.

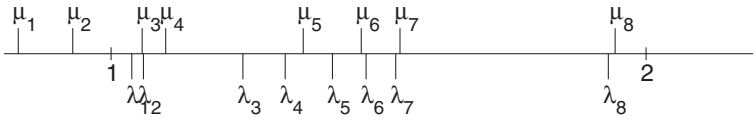
The linear system (5.2) needs to be solved repeatedly for different right-hand sides. For computational efficiency, the matrix $C := L^{-1}C$ in step 1 and the matrix $(I + C^T D^{-1} C - \Sigma/\tau)^{-1}$ in step 4 are computed only once and stored before calling `eigs`.

The CPU elapsed time is displayed in the following table, where "`ldlsparse`" is the time for computing the decomposition $A - \tau B = LDL^T$; "preprocessing" is for computing the matrices $C := L^{-1}C$ and $(I + C^T D^{-1} C - \Sigma/\tau)^{-1}$, and the assemblage of the matrix $\mathcal{B} = \mathrm{diag}(B, \Sigma^{-1})$.

	$\mathcal{A} - \lambda\mathcal{B}$	$A - \lambda B$
`ldlsparse`	0.64	0.64
preprocessing	0.49	0
`eigs`	7.14	6.66
Total	8.27	7.30

The residuals for all 8 computed eigenpairs are less than 5.5×10^{-13}. For comparison, in the third column of the previous table, we also record the CPU time to solve only the eigenvalue problem of the linear term $A - \lambda B$ of REP (4.10) using the same parameters for calling `eigs`. As we can see, it only takes about 13.3% extra time to solve the full REP over the time to solve the simple eigenvalue problem of the pencil $A - \lambda B$.

The computed 8 eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_8$ of REP (4.10) in the interval $(1, 2)$ are plotted below, along with the computed 8 eigenvalues $\mu_1, \mu_2, \ldots, \mu_8$ of the linear pencil $A - \mu B$ closest to the shift $\tau = 1.5$.



We observe that there are only 6 eigenvalues of the pencil $A - \lambda B$ in the interval $(1, 2)$. Among them, μ_6, μ_7, μ_8 are good approximations of $\lambda_6, \lambda_7, \lambda_8$, respectively. Suppose we treat REP (4.10) as a general NEP, and we apply a general-purpose nonlinear eigensolver for large-scale problems, say, Newton's method, the successive linear approximation method, or the nonlinear Arnoldi method [19, 26]. Then with the initial approximations μ_6, μ_7, μ_8, we can expect the convergence to the eigenvalues $\lambda_6, \lambda_7, \lambda_8$. However, any of these NEP eigensolvers would be difficult to converge the rest of five eigenvalues with the initial approximations $\mu_1, \mu_2, \ldots, \mu_5$.

Acknowledgments. We are grateful to Heinrich Voss for providing the data for Example 2 in section 5. We thank Françoise Tisseur for her numerous comments. We also thank the referees for their comments that improved the quality of the paper.

REFERENCES

[1] A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems*, Philadelphia, SIAM, 2005.
[2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, ed., *Templates for the Solution of Algebraic Eigenvalue Problems*, SIAM, Philadelphia, 2000.
[3] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder, and F. Tisseur, *NLEVP: A Collection of Nonlinear Eigenvalue Problems.* Technical report MIMS EPrint 2008.40,

School of Mathematics, The University of Manchester, 2008. http://www.manchester.ac.uk/mims/eprints.

[4] R. Byers, V. Mehrmann, and H. Xu, *Trimmed linearizations for structured matrix polynomials*, Linear Algebra Appl., 429 (2008), pp. 2373–2400.

[5] T. A. Davis, *Algorithm* 849*: A concise sparse Cholesky factorization package*, ACM Trans. Math. Software, 31 (2005), pp. 587–591. http://www.cise.ufl.edu/research/sparse/ldl/.

[6] I. Gohberg, P. Lancaster, and L. Rodman, *Matrix Polynomials*, Academic Press, New York, 1982.

[7] C.-H. Guo, N. J. Higham, and F. Tisseur, *Detecting and solving hyperbolic quadratic eigenvalue problems*, SIAM J. Matrix Anal. Appl., 30 (2009), pp. 1593–1613.

[8] N. J. Higham, D. S. Mackey, N. Mackey, and F. Tisseur, *Symmetric linearizations for matrix polynomials*, SIAM J. Matrix Anal. Appl., 29 (2006), pp. 143–159.

[9] N. J. Higham, D. S. Mackey, and F. Tisseur, *The conditioning of linearizations of matrix polynomials*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 1005–1028.

[10] T.-M. Hwang, W.-W. Lin, J.-L. Liu, and W. Wang, *Jacobi-Davidson methods for cubic eigenvalue problems*, Numer. Linear Algebra Appl., 12 (2005), pp. 605–624.

[11] T.-M. Hwang, W.-W. Lin, W.-C. Wang, and W. Wang, *Numerical simulation of three dimensional quantum dot*, J. Comput. Phys., 196 (2004), pp. 208–232.

[12] P. Lancaster and M. Tismenetsky, *The Theory of Matrices*, 2nd ed., Academic Press, London, 1985.

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

[14] D. S. Mackey, N. Mackey, C. Mehl, and V. Mehrmann, *Structured polynomial eigenvalue problems: Good vibrations from good linearization*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 1029–1051.

[15] D. S. Mackey, N. Mackey, C. Mehl, and V. Mehrmann, *Vector spaces of linearizations for matrix polynomials*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 971–1004.

[16] L. Mazurenko and H. Voss, *Low rank rational perturbations of linear symmetric eigenproblems*, Z. Angew. Math. Mech., 86 (2006), pp. 606–616.

[17] V. Mehrmann and H. Voss, *Nonlinear eigenvalue problems: A challenge for modern eigenvalue methods*, GAMM-Reports, 27 (2004), pp. 121–152.

[18] M. J. O'Sullivan and M. A. Saunders, *Sparse rank-revealing LU factorization (via threshold complete pivoting and threshold rook pivoting)*, presented at Householder Symposium XV on Numerical Linear Algebra, 2002. Available at http://www.stanford.edu/group/SOL/talks.html.

[19] A. Ruhe, *Algorithms for the nonlinear eigenvalue problem*, SIAM J. Numer. Anal., 10 (1973), pp. 674–689.

[20] H. Rutishauser, *Simultaneous iteration method for symmetric matrices*, Numer. Math., 16 (1970), pp. 205–223.

[21] B. De Schutter, *Minimal state-space realization in linear system theory: An overview*, J. Comput. Appl. Math., 121 (2000), pp. 331–354.

[22] G. L. G. Sleijpen and H. A. Van der Vorst, *A Jacobi–Davidson iteration method for linear eigenvalue problems*, SIAM Rev., 42 (2000), pp. 267–293.

[23] S. I. Solov'Ëv, *Preconditioned iterative methods for a class of nonlinear eigenvalue problems*, Linear Algebra Appl., 415 (2006), pp. 210–229.

[24] G. W. Stewart, *Matrix Algorithms. Volume* II*: Eigensystems*, SIAM, Philadelphia, 2001.

[25] H. Voss, *A rational spectral problem in fluid-solid vibration*, Electron. Trans. Numer. Anal., 16 (2003), pp. 94–106.

[26] H. Voss, *An Arnoldi method for nonlinear eigenvalue problems*, BIT, 44 (2004), pp. 387–401.

[27] H. Voss, *Iterative projection methods for computing relevant energy states of a quantum dot*, J. Comput. Phys., 217 (2006), pp. 824–833.

[28] K. Wu and H. Simon, *Thick-restart Lanczos method for large symmetric eigenvalue problems*, SIAM J. Matrix Anal. Appl., 22 (2000), pp. 602–616.