Numer. Math. (2009) 114:355–372
DOI 10.1007/s00211-009-0259-x

Numerische
Mathematik

A block Newton method for nonlinear eigenvalue problems

Daniel Kressner

Received: 2 February 2009 / Revised: 20 July 2009 / Published online: 15 September 2009

Abstract We consider matrix eigenvalue problems that are nonlinear in the eigenvalue parameter. One of the most fundamental differences from the linear case is that distinct eigenvalues may have linearly dependent eigenvectors or even share the same eigenvector. This has been a severe hindrance in the development of general numerical schemes for computing several eigenvalues of a nonlinear eigenvalue problem, either simultaneously or subsequently. The purpose of this work is to show that the concept of invariant pairs offers a way of representing eigenvalues and eigenvectors that is insensitive to this phenomenon. To demonstrate the use of this concept in the development of numerical methods, we have developed a novel block Newton method for computing such invariant pairs. Algorithmic aspects of this method are considered and a few academic examples demonstrate its viability.

Mathematics Subject Classification (2000) Primary 65F15; Secondary 15A18 · 47A56

1 Introduction

Given a function $T : \Omega \to \mathbb{C}^{n\times n}$ holomorphic on an open set $\Omega \subseteq \mathbb{C}$, we consider the *nonlinear eigenvalue problem* of finding pairs $(x, \lambda) \in \mathbb{C}^n \times \Omega$ with $x \neq 0$ such that

$$T(\lambda)x = 0. \tag{1}$$

For any such pair (x, λ), we call x an *eigenvector* and λ an *eigenvalue*. This formulation includes linear eigenvalue problems, for which $T(\lambda) = A - \lambda I$ with $A \in \mathbb{C}^{n\times n}$, as well as polynomial eigenvalue problems, for which T is a matrix polynomial in λ.

D. Kressner (✉)
Seminar für Angewandte Mathematik, HG G 57.1, Rämistrasse 101, 8092 Zurich, Switzerland
e-mail: kressner@math.ethz.ch

To avoid degenerate situations, we assume that T is *regular*, i.e., $\det(T(\cdot)) \not\equiv 0$ on any of the components of Ω, throughout this paper. For a recent overview on the numerics and numerous applications of such nonlinear eigenvalue problems, we refer to [20].

In contrast to the linear case, there may be eigenvector/eigenvalue pairs (λ_1, x_1), …, (λ_k, x_k) of (1), for which the eigenvalues $\lambda_1, \dots, \lambda_k$ are pairwise distinct but $\{x_1, \dots, x_k\}$ is linearly dependent. This possibility is already evident from the fact that k can be larger than n. Another example [12] is given by

$$T(\lambda) = \begin{bmatrix} 0 & 12 \\ -2 & 14 \end{bmatrix} + \lambda \begin{bmatrix} -1 & -6 \\ 2 & -9 \end{bmatrix} + \lambda^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{2}$$

for which the eigenvalues 3 and 4 share the same eigenvector $\left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$. The occurrence of such linear dependencies is an annoyance when attempting to develop numerical methods for computing more than one eigenvalue of (1). For example, standard Newton methods [10,11] for the simultaneous computation of several eigenvalues crucially depend on the existence of a basis for the invariant subspace belonging to the eigenvalues of interest. In methods that determine several eigenvalues subsequently, such as Krylov subspace or Jacobi-Davidson methods [2], repeated convergence towards an eigenvalue is usually avoided by reorthogonalization against converged eigenvectors. If such an idea was directly applied to nonlinear eigenvalue problems, eigenvalues could be missed due to linear dependencies among eigenvectors.

In the case that the nonlinear eigenvalue problem admits a minimum–maximum characterization [26,31], its eigenvalues can be ordered and numbered. Voss and his co-authors [4,5,7,27–30] have developed Arnoldi-type and Jacobi-Davidson-type methods that employ this numbering as a safety scheme for avoiding repeated convergence towards the same eigenvalue. Unfortunately, for many applications such minimum–maximum characterizations do not exist or are difficult to verify.

In this work, we will propose a different approach for dealing with several eigenvalues, very much inspired by the work of Beyn and Thümmler [9] on continuation methods for quadratic eigenvalue problems. For this purpose, it will be more convenient to assume that the nonlinear eigenvalue problem (1) takes the form

$$(f_1(\lambda)A_1 + f_2(\lambda)A_2 + \cdots + f_m(\lambda)A_m)\, x = 0. \tag{3}$$

for holomorphic functions $f_1, \dots, f_m : \Omega \to \mathbb{C}$ and constant matrices $A_1, \dots, A_m \in \mathbb{C}^{n\times n}$. This is no restriction as we could turn (1) into (3) by choosing $m = n^2$, $f_{(i-1)n+j}(\lambda) = t_{ij}(\lambda)$ and $A_{(i-1)n+j} = e_i e_j^T$, with e_i and e_j denoting the ith and jth unit vectors of length n, respectively. However, many applications of nonlinear eigenvalue problems already come in the form (3) and such a reformulation is not needed. For example, in eigenvalue problems related to the stability of time-delay systems [21], the functions f_j are exponentials or polynomials. In applications related to vibrating mechanical structures [30], the functions f_j are rational and model different material properties.

The rest of this paper is organized as follows. In "Invariant pairs", the concept of invariant pairs for the nonlinear eigenvalue Problem (3) is introduced. We believe this to be the most suitable extension of an eigenvalue/eigenvector pair to several eigenvalues. Several useful properties are shown to substantiate this belief. In “A Newton method for simple invariant pairs”, a Newton method for computing such invariant pairs is developed, along with some algorithmic details and numerical experiments.

2 Invariant pairs

Definition 1 Let the eigenvalues of $S \in \mathbb{C}^{k\times k}$ be contained in Ω and let $X \in \mathbb{C}^{n\times k}$. Then $(X, S) \in \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}$ is called an *invariant pair* of the nonlinear eigenvalue problem (3) if

$$A_1 X f_1(S) + A_2 X f_2(S) + \cdots + A_m X f_m(S) = 0. \tag{4}$$

Note that the matrix functions $f_1(S), \ldots, f_m(S)$ are well defined under the given assumptions [16]. As an example, let (x_1, λ_1) and (x_2, λ_2) be eigenvector/eigenvalue pairs of (3). Then (X, S) with $X = [x_1, x_2]$ and $S = \operatorname{diag}(\lambda_1, \lambda_2)$ is an invariant pair.

To avoid trivial invariant pairs, such as $X = 0$, an additional property needs to be imposed. However, we have already seen that requiring X to have full column rank is not reasonable in the context of nonlinear eigenvalue problems. Instead, we use the concept of minimal invariant pairs from [6,9].

Definition 2 A pair $(X, S) \in \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}$ is called *minimal* if there is $l \in \mathbb{N}$ such that the matrix

$$V_l(X, S) = \begin{bmatrix} X \\ XS \\ \vdots \\ XS^{l-1} \end{bmatrix} \tag{5}$$

has rank k. The smallest such l is called the *minimality index* of (X, S).

Example 3 For the example (2), the pair (X, S) with $X = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ and $S = \operatorname{diag}(3, 4)$ is invariant and minimal with minimality index 2.

It has been shown in [6, Theorem 3] that any non-minimal pair can be turned into a minimal one in the following sense. If $V_l(X, S)$ has rank $\tilde{k} < k$ then there is a minimal pair $(\widetilde{X}, \widetilde{S}) \in \mathbb{C}^{n\times \tilde{k}} \times \mathbb{C}^{\tilde{k}\times \tilde{k}}$ such that $\operatorname{span} \widetilde{X} = \operatorname{span} X$ and $\operatorname{span} V_l(\widetilde{X}, \widetilde{S}) = \operatorname{span} V_l(X, S)$. The following Lemma reveals the connection of minimal invariant pairs to the nonlinear eigenvalue problem (3).

Lemma 4 *Let $(X, S) \in \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}$ be a minimal invariant pair of (3). Then the following statements hold.*

1. *For any invertible matrix $Z \in \mathbb{C}^{k\times k}$, $(XZ, Z^{-1}SZ)$ is also a minimal invariant pair of (3).*
2. *The eigenvalues of S are eigenvalues of (3).*

Proof 1. Using $f_j(Z^{-1}SZ) = Z^{-1}f_j(S)Z$, the relation (4) can be written as

$$\begin{aligned} &A_1XZf_1(Z^{-1}SZ)Z^{-1} + A_2XZf_2(Z^{-1}SZ)Z^{-1} + \cdots \\ &\quad + A_mXZf_m(Z^{-1}SZ)Z^{-1} = 0, \end{aligned}$$

which is equivalent to

$$A_1XZf_1(Z^{-1}SZ) + A_2XZf_2(Z^{-1}SZ) + \cdots + A_mXZf_m(Z^{-1}SZ) = 0, \quad (6)$$

and shows that $(XZ, Z^{-1}SZ)$ is an invariant pair. Its minimality follows from

$$V_l(XZ, Z^{-1}SZ) = V_l(X, S)Z.$$

2. By the Schur decomposition, we can choose Z orthogonal such that $\widetilde{S} = Z^{-1}SZ$ is upper triangular with any eigenvalue λ of S appearing in the $(1, 1)$ position of $\widetilde{S}$. Setting $x = XZe_1$, the first column of $V_l(Z^{-1}SZ, XZ)$ has the entries $x, x\lambda, \ldots, x\lambda^{l-1}$. Hence, $x \neq 0$ since otherwise $V_l(Z^{-1}SZ, XZ)$ would be rank deficient for any l. Moreover,

$$XZf_j(Z^{-1}SZ)e_1 = f_j(\lambda)x$$

and thus the first column of (6) implies that (x, λ) is an eigenvector/eigenvalue pair. □

Let us briefly discuss the practical consequences of Lemma 4. Once a minimal invariant pair is computed we can extract the corresponding eigenvalues of $T(\cdot)$ by computing the eigenvalues of S. Moreover, if S admits a diagonalization $Z^{-1}SZ$ then the columns of XZ contain the corresponding eigenvectors.

The following lemma shows that for checking minimality, it is sufficient to check the rank of $V_k(X, S)$.

Lemma 5 *If a pair $(X, S) \in \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}$ is minimal then its minimality index cannot exceed k.*

Proof Since (X, S) is minimal, there is $l \in \mathbb{N}$ such that $\operatorname{rank}(V_l(X, S)) = k$. For $l \leq k$ there is nothing to prove. For $l > k$, the Cayley-Hamilton theorem yields the existence of coefficients $\alpha_{ij} \in \mathbb{C}$ such that

$$XS^{k+i} = \alpha_{i0}X + \alpha_{i1}XS + \cdots + \alpha_{i,k-1}XS^{k-1}, \quad i \geq 0.$$

Hence, there is a square invertible matrix W such that

$$W\, V_l(X, S) = \begin{bmatrix} V_k(X, S) \\ 0 \end{bmatrix},$$

implying $\operatorname{rank}(V_k(X, S)) = k$. □

2.1 Relation to polynomial eigenvalue problems

Given a minimal invariant pair $(X, S) \in \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}$, the nonlinear eigenvalue problem (3) can be locally transformed into a polynomial eigenvalue problem. To see this, we choose $p_j \in \Pi_k$, where Π_k denotes all polynomials of degree at most k, as the Hermite interpolating polynomial of f_j at the spectrum of S [16]. Then $f_j(S) = p_j(S)$ and (4) can be written as

$$A_1 X p_1(S) + A_2 X p_2(S) + \cdots + A_m X p_m(S) = 0.$$

Hence, (X, S) is a minimal invariant pair for the polynomial eigenvalue problem

$$(p_1(\lambda)A_1 + p_2(\lambda)A_2 + \cdots + p_m(\lambda)A_m)\, x = 0. \tag{7}$$

In particular, Lemma 4.2 implies that the eigenvalues of S are eigenvalues of (7). Note, however, that the converse is not true: from the kn eigenvalues of (7) only k can be expected to solve the original nonlinear eigenvalue problem. Nevertheless, (7) allows us to show that minimal invariant pairs can be easily constructed in the case of pairwise distinct eigenvalues.

Lemma 6 *Let $(x_1, \lambda_1), \ldots, (x_k, \lambda_k)$ be eigenvector/eigenvalue pairs for the nonlinear eigenvalue problem* (3)*, with $\lambda_i \neq \lambda_j$ for $i \neq j$. Then the invariant pair*

$$(X, S) = ([x_1, \ldots, x_k], \operatorname{diag}(\lambda_1, \ldots, \lambda_k))$$

is minimal.

Proof The construction above shows that (X, S) is an invariant pair for the polynomial eigenvalue problem (7). Without loss of generality, we may assume that (7) is regular. If this is not the case, (7) can always be replaced by a regular higher order interpolation that also incorporates derivative of f_j, see the proof of Theorem 10.

By the companion linearization [14] of (7), there is a matrix pencil $C_{\mathcal{A}} - \lambda C_{\mathcal{B}} \in \mathbb{C}^{kn\times kn}$ such that $(v_1, \lambda_1), \ldots, (v_k, \lambda_k)$ with $v_j = [x_j^T, \lambda_j x_j^T, \ldots, \lambda_j^{k-1} x_j^T]^T$ are eigenvector/eigenvalue pairs of $C_{\mathcal{A}} - \lambda C_{\mathcal{B}}$. Eigenvectors belonging to pairwise distinct eigenvalues of matrix pencils are linearly independent and hence $V_k(X, S) = [v_1, \ldots, v_k]$ has rank k, concluding the proof. □

2.2 Simple invariant pairs

Relation (1) immediately implies that λ is an eigenvalue if and only if $\det(T(\lambda)) = 0$. The algebraic multiplicity of λ is defined as usual.

Definition 7 Let λ be an eigenvalue of a holomorphic function $T : \Omega \to \mathbb{C}^{n\times n}$. Then the *algebraic multiplicity* of λ, denoted by $\operatorname{alg}_T(\lambda)$, is the smallest integer j such that $\frac{\partial^j}{\partial\lambda^j} \det(T(\lambda)) \neq 0$.

Note that $\det(T(\cdot))$ is also holomorphic in the open set Ω and thus – by basic complex analysis results – the algebraic multiplicity of λ is finite unless $\det(T(\cdot)) \equiv 0$ in some component of Ω. However, in the latter case T is not regular, contradicting the assumption made in the introduction.

The following definition defines an invariant pair to be simple if it includes eigenvalues in their full multiplicity.

Definition 8 An invariant pair (X, S) for a regular, holomorphic function $T : \Omega \to \mathbb{C}^{n\times n}$ is called simple if (X, S) is minimal and the algebraic multiplicities of the eigenvalues of S are identical to the algebraic multiplicities of the corresponding eigenvalues of T.

In the following, it will be shown that simple invariant pairs are well posed objects in the sense of being regular solutions to a nonlinear matrix equation. For this purpose, we introduce the nonlinear matrix operator

$$\begin{aligned} \mathbb{T} : \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}_{\Omega} &\to \mathbb{C}^{n\times k}, \\ (X, S) &\mapsto A_1 X f_1(S) + A_2 X f_2(S) + \cdots + A_m X f_m(S), \end{aligned} \tag{8}$$

associated with $T(\lambda) = f_1(\lambda)A_1 + f_2(\lambda)A_2 + \cdots + f_m(\lambda)A_m$. Here, $\mathbb{C}^{k\times k}_{\Omega}$ denotes the set of $k \times k$ matrices with eigenvalues in Ω. By definition, see (4), an invariant pair (X, S) satisfies $\mathbb{T}(X, S) = 0$. But this relation is clearly not sufficient to characterize (X, S) and we need to add some normalization. For this purpose, choose l such that the matrix $V_l(X, S)$, see (5), has rank k. Define and partition the matrix

$$W = \begin{bmatrix} W_0 \\ W_1 \\ \vdots \\ W_{l-1} \end{bmatrix} := V_l(X, S)\left(V_l(X, S)^H V_l(X, S)\right)^{-1} \in \mathbb{C}^{nk\times k},$$

with $W_j \in \mathbb{C}^{n\times k}$. Thus, $\mathbb{V}(X, S) = 0$ for the operator

$$\begin{aligned} \mathbb{V} : \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}_{\Omega} &\to \mathbb{C}^{k\times k}, \\ (X, S) &\mapsto W^H V_l(X, S) - I_k. \end{aligned} \tag{9}$$

Note that W is considered constant and not treated as a function of X and S.

In the following it will be investigated whether the two conditions $\mathbb{T}(X, S) = 0$, $\mathbb{V}(X, S) = 0$ are sufficient to characterize (X, S), at least locally. For this purpose, we compute the Fréchet derivatives of $\mathbb{T}$ and $\mathbb{V}$ at (X, S):

$$\mathbb{D}\mathbb{T} : (\triangle X, \triangle S) \mapsto \mathbb{T}(\triangle X, S) + \sum_{j=1}^{m} A_j X\,[\mathbb{D} f_j(S)](\triangle S), \tag{10}$$

$$\mathbb{D}\mathbb{V} : (\triangle X, \triangle S) \mapsto W_0^H \triangle X + \sum_{j=1}^{l-1} W_j^H (\triangle X S^j + X\,\mathbb{D}S^j(\triangle S)). \tag{11}$$

Here, $[\mathbb{D}f_j(S)]$ denotes the Fréchet derivative of f_j at S. Note that the Fréchet derivative $\mathbb{D}S^j$ of the map $S \mapsto S^j$ can be written as

$$\mathbb{D}S^j : \triangle S \mapsto \sum_{i=0}^{j} S^i \triangle S\, S^{j-i-1}.$$

The following example illustrates the definitions above.

Example 9 Consider $T(\lambda) = \lambda I - A_0 - A_1 e^{-\lambda\tau}$ with $A_0, A_1 \in \mathbb{C}^{n\times n}$ and $\tau \in \mathbb{R}$. The associated nonlinear eigenvalue problem arises from the stability study of a linear delay differential equation (DDE) with a single delay [21]. For $k = l = 2$, the operators $\mathbb{T}$ and $\mathbb{V}$ take the form

$$\begin{aligned}\mathbb{T}(X, S) &= XS - A_0X - A_1Xe^{-\tau S},\\ \mathbb{V}(X, S) &= W_0^H X + W_1^H XS - I_2.\end{aligned}$$

The corresponding Fréchet derivatives at (X, S) are given by

$$\begin{aligned}\mathbb{DT}(\triangle X, \triangle S) &= \mathbb{T}(\triangle X, S) + X\,\triangle S - A_1X\,\mathbb{D}e^{-\tau S}(\triangle S),\\ \mathbb{DV}(\triangle X, \triangle S) &= W_0^H \triangle X + W_1^H (\triangle X\, S + X\,\triangle S).\end{aligned}$$

Note that $\mathbb{D}e^{-\tau S}$, the Fréchet derivative of the exponential matrix function, can be computed using methods described in [16,18,22], see also “A Newton method for simple invariant pairs” below.

The following theorem is the main result of this section and proves that simple invariant pairs are well posed.

Theorem 10 *Let (X, S) be a minimal invariant pair for the nonlinear eigenvalue problem (3). Then (X, S) is simple if and only if the associated linear matrix operator*

$$\begin{aligned}\mathbb{L} : \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k} &\to \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}\\ (\triangle X, \triangle S) &\to (\mathbb{DT}(\triangle X, \triangle S),\ \mathbb{DV}(\triangle X, \triangle S))\,,\end{aligned}$$

with $\mathbb{DT}$ and $\mathbb{DV}$ defined in (10)–(11), is invertible.

Proof The basic idea of the proof is to replace the holomorphic functions $f_1, \ldots, f_m$ by polynomials and apply results from [6]. Complex interpolation theory [13] shows that there are polynomials p_i, $i = 1, \ldots, m$, of degree not larger than ℓ, with some integer $\ell > k$, such that

$$p_i^{(j)}(\lambda_s) = f_i^{(j)}(\lambda_s), \quad j = 0, \ldots, 2k - 1,$$

for every eigenvalue λ_s of S. In particular, this implies $f_i(S) = p_i(S)$ for $i = 1, \ldots, m$ and hence (X, S) is a minimal invariant pair for $P(\lambda) = A_1 p_1(\lambda) + \cdots + A_m p_m(\lambda)$.

Moreover,

$$\frac{\partial^j}{\partial\lambda^j}\det(T(\lambda_s)) = \frac{\partial^j}{\partial\lambda^j}\det(P(\lambda_s)), \quad j = 0, \ldots, 2k-1, \tag{12}$$

showing that the algebraic multiplicities of T for the eigenvalues of S match those of P, unless the algebraic multiplicity exceeds $2k-1$ (in which case (X, S) cannot be simple, neither for T nor for P). By definition, (12) thus proves that (X, S) is a *simple* invariant pair for T if and only if it is a *simple* invariant pair for P. In [6, Theorem 7], the latter condition is shown to be equivalent to the condition that (X, S) is a regular solution to the matrix equations

$$\mathbb{P}(X, S) = A_1 X p_1(S) + \cdots + A_m X p_1(S) = 0, \quad \mathbb{T}(X, S) = \widetilde{W}^H V_\ell(X, S) = 0. \tag{13}$$

A minor complication is that in general $\ell > l$, but this can be easily fixed by defining $\widetilde{W}$ as the matrix obtained from appending W with $(\ell - l)n$ zero rows. In turn, (X, S) is a *simple* invariant pair if and only if the linear matrix operator

$$\begin{aligned} \widetilde{\mathbb{L}} : \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k} &\to \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k} \\ (\triangle X, \triangle S) &\to (\mathbb{DP}(\triangle X, \triangle S),\ \mathbb{DV}(\triangle X, \triangle S)), \end{aligned}$$

is invertible, where

$$\mathbb{DP} : (\triangle X, \triangle S) \mapsto \mathbb{T}(\triangle X, S) + \sum_{j=1}^{m} A_j X\, [\mathbb{D}p_j(S)](\triangle S).$$

Using (12) we obtain from the results in [18] that

$$\begin{aligned} \begin{bmatrix} f_j(S) & [\mathbb{D}f_j(S)](\triangle S) \\ 0 & f_j(S) \end{bmatrix} &= f_j\left(\begin{bmatrix} S & \triangle S \\ 0 & S \end{bmatrix}\right) = p_j\left(\begin{bmatrix} S & \triangle S \\ 0 & S \end{bmatrix}\right) \\ &= \begin{bmatrix} p_j(S) & [\mathbb{D}p_j(S)](\triangle S) \\ 0 & p_j(S) \end{bmatrix} \end{aligned}$$

for $j = 1, \ldots, m$. Thus $[\mathbb{D}f_j(S)](\triangle S) = [\mathbb{D}p_j(S)](\triangle S)$, which implies $\widetilde{\mathbb{L}} = \mathbb{L}$ and concludes the proof. □

By the implicit function theorem for holomorphic functions [17], Theorem 10 reveals that the entries of a simple invariant pair (X, S) vary analytically under analytic changes of T. This shows that (X, S) is well posed and reasonable to compute numerically.

3 A Newton method for simple invariant pairs

In this section, we show how the theoretical framework developed in the previous section can be turned into a numerical algorithm for computing simple invariant pairs.

The aim is to compute a solution $(X, S) \in \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}$ to the nonlinear matrix equations

$$\mathbb{T}(X, S) = 0, \quad \mathbb{V}(X, S) = 0. \tag{14}$$

with $\mathbb{T}(X, S) = A_1 X f_1(S) + \cdots + A_m X f_m(S)$ and $\mathbb{V}(X, S) = W^H V_l(X, S) - I_k$ for some fixed $W \in \mathbb{C}^{ln\times k}$ of full column rank. By Lemma 4, the eigenvalues of S are eigenvalues of $T(\cdot)$ and if there is an invertible Z such that $Z^{-1}SZ$ is diagonal then XZ contains the corresponding eigenvectors.

Formally, the Newton method for solving (14) can be written as

$$(X_{p+1}, S_{p+1}) = (X_p, S_p) - \mathbb{L}_p^{-1}\left(\mathbb{T}(X_p, S_p), \mathbb{V}(X_p, S_p)\right) \tag{15}$$

where $\mathbb{L}_p$ is the Jacobian of (14) at (X_p, S_p). We have

$$\mathbb{L}_p(\triangle X, \triangle S) = \left(\mathbb{DT}_p(\triangle X, \triangle S), \mathbb{DV}_p(\triangle X, \triangle S)\right)$$

with $\mathbb{DT}_p$ and $\mathbb{DV}_p$ defined as in (10)–(11) but with X, S replaced by X_p, S_p. Theorem 10 implies that the iteration (15) converges *locally* quadratically to a simple invariant pair.

Remark 11 In principle, W can be chosen arbitrarily as long as $W^H V$ is invertible for any basis V of span $(V_l(X, S))$. Equivalently, the largest principal angle between span(W) and span $(V(X, S))$ must be less than $\pi/2$ [25]. To avoid an ill-conditioned basis in the course of the iteration, it is common practice in Newton methods for eigenvalue computation [1] to choose an orthonormal basis for W in each iteration. In our setting, this corresponds to computing a compact QR decomposition $V_l(X_p, S_p) = WR$ with $W^H W = I_k$ and $R \in \mathbb{C}^{k\times k}$ invertible. To preserve the relation $W^H V_l(X_p, S_p) = I_k$, we then have to replace $X_p \leftarrow X_p R^{-1}$ and $S_p \leftarrow R S_p R^{-1}$ accordingly.

Algorithm 1 Newton method for computing invariant pairs
Input: Initial pair $(X_0, S_0) \in \mathbb{C}^{n\times k} \times \mathbb{C}^{k\times k}$ such that $V_l(X_0, S_0)^H V_l(X_0, S_0) = I_k$.
Output: Approximate solution (X_{p+1}, S_{p+1}) to (14).

```
$p \leftarrow 0$, $W \leftarrow V_l(X_0, S_0)$
repeat
  Res $\leftarrow \mathbb{T}(X_p, S_p)$
  Solve linear matrix equation $\mathbb{L}_p(\triangle X, \triangle S) = (\text{Res}, 0)$.
  $\widetilde{X}_{p+1} \leftarrow X_p - \triangle X, \quad \widetilde{S}_{p+1} \leftarrow S_p - \triangle S$
  Compute compact QR decomposition $V_l\left(\widetilde{X}_{p+1}, \widetilde{S}_{p+1}\right) = WR$.
  $X_{p+1} \leftarrow \widetilde{X}_{p+1} R^{-1}, \quad S_{p+1} \leftarrow R \widetilde{S}_{p+1} R^{-1}$
until convergence
```

3.1 Setting up the initial pair (X_0, S_0)

In many applications the approximate location of the eigenvalues of interest might be known, but usually little or no information is available for the eigenvectors. To compensate for this imbalance, a variant of inverse iteration should be applied *before* starting Algorithm 1.

Algorithm 2 Inverse iteration for stetting up initial pair (X_0, S_0)

```
Input: An initial matrix $S_0 \in \mathbb{C}^{k\times k}$.
Output: An initial matrix $X_0 \in \mathbb{C}^{n\times k}$.
1: Choose random matrix $X_0 \in \mathbb{C}^{n\times k}$.
2: for $p \leftarrow 1, 2, 3$ do
3:     Compute solution $Y$ to linear matrix equation $\mathbb{T}(Y, S_0) = X_0$.
4:     Compute compact QR decomposition $V_l(Y, S_0) = WR$.
5:     Update $X_0 \leftarrow Y_k R^{-1}, \quad S_0 \leftarrow RS_0R^{-1}$
6: end for
```

The linear matrix equation to be solved in Step 2 is a special case of the matrix equation in Step 1 of Algorithm 1, whose efficient solution is discussed in the next section. Note that choosing 3 as the number of loops to be performed in Algorithm 2 is a heuristics; for the examples reported below we did not observe any significant benefit from iterating further.

3.2 Solving the linear system

The most expensive part of Algorithm 1 is certainly the solution of the linear matrix equation in Step 1. In principal, by using Kronecker products to replace the involved matrix products, this is equivalent to a linear system of order $(nk+k^2)$. However, even if this system can be cheaply set up, its solution still requires $\mathcal{O}(k^3(n+k)^3)$ flops, which is only acceptable as long as k is not significantly larger than 1. Fortunately, ideas from [8,9] can be extended to reduce the cost significantly. For notational convenience, we drop the index p and consider the solution of a linear system of matrix equations

$$\mathbb{T}(\triangle X, S) + \sum_{j=1}^{m} A_j X\, [\mathbb{D} f_j(S)](\triangle S) = R_T, \tag{16}$$

$$W_0^H \triangle X + \sum_{j=1}^{l-1} W_j^H (\triangle X S^j + X\, \mathbb{D} S^j(\triangle S)) = R_V, \tag{17}$$

where $(\triangle X, \triangle S)$ is unknown. Moreover, by a suitable normalization of X, S, W we can assume w.l.o.g. that S is in (complex) Schur form. Because of the triangular structure of S, the equations (16)–(17) simplify considerably for the first columns of $\triangle X$ and $\triangle S$. To see this, we will make use of the following technical result.

Lemma 12 *Let f be holomorphic on an open set $\Omega \subseteq \mathbb{C}$ containing the spectrum of the upper triangular matrix $S \in \mathbb{C}^{k\times k}$. Then there is an upper triangular matrix $[\mathbb{D} f(S)]_{11} \in \mathbb{C}^{k\times k}$ such that*

$$[\mathbb{D} f(S)]_{11}\, C e_1 = [\mathbb{D} f(S)](C)\, e_1 \tag{18}$$

for any matrix $C \in \mathbb{C}^{k\times k}$. Moreover, we have the relation

$$f\left(\begin{bmatrix} S & I_k \\ 0 & s_{11} I_k \end{bmatrix}\right) = \begin{bmatrix} f(S) & [\mathbb{D} f(S)]_{11} \\ 0 & f(s_{11}) I_k \end{bmatrix}. \tag{19}$$

Proof Since $f : \Omega \to \mathbb{C}$ and its derivative are considered for a fixed matrix S, we can assume w.l.o.g. (e.g., by replacing f by a Hermite interpolant) that Ω is connected. Let Γ be a closed contour that is contained in Ω and encircles the eigenvalues of S. Then [16],

$$\begin{aligned} [\mathbb{D} f(S)](C)\, e_1 &= \frac{1}{2\pi \mathrm{i}} \int_\Gamma f(z)(zI - S)^{-1} C (zI - S)^{-1} e_1 \, \mathrm{d}z \\ &= \frac{1}{2\pi \mathrm{i}} \int_\Gamma \frac{f(z)}{z - s_{11}} (zI - S)^{-1} C e_1 \, \mathrm{d}z, \end{aligned}$$

where we used that S is upper triangular. Hence,

$$[\mathbb{D} f(S)]_{11} = \frac{1}{2\pi \mathrm{i}} \int_\Gamma \frac{f(z)}{z - s_{11}} (zI - S)^{-1} \, \mathrm{d}z,$$

which is clearly upper triangular. The equality (19) follows from

$$\begin{aligned} f\left(\begin{bmatrix} S & I_k \\ 0 & s_{11} I_k \end{bmatrix}\right) &= \frac{1}{2\pi \mathrm{i}} \int_\Gamma f(z) \begin{bmatrix} zI - S & -I_k \\ 0 & (z - s_{11}) I_k \end{bmatrix}^{-1} \mathrm{d}z \\ &= \frac{1}{2\pi \mathrm{i}} \int_\Gamma f(z) \begin{bmatrix} (zI - S)^{-1} & (z - S)^{-1}(z - s_{11})^{-1} \\ 0 & (z - s_{11})^{-1} I_k \end{bmatrix} \mathrm{d}z \\ &= \begin{bmatrix} f(S) & [\mathbb{D} f(S)]_{11} \\ 0 & f(s_{11}) I_k \end{bmatrix}, \end{aligned}$$

which concludes the proof. □

Using Lemma 12, post-multiplying (16)–(17) by e_1 yields the following linear system of order nk:

$$\begin{bmatrix} T(s_{11}) & \sum\limits_{j=1}^{m} A_j X [\mathbb{D} f_j(S)]_{11} \\ \sum\limits_{j=0}^{l-1} s_{11}^j W_j^H & \sum\limits_{j=1}^{l-1} W_j^H X [\mathbb{D} S^j]_{11} \end{bmatrix} \begin{bmatrix} \triangle x_1 \\ \triangle s_1 \end{bmatrix} = \begin{bmatrix} R_T e_1 \\ R_V e_1 \end{bmatrix}, \tag{20}$$

where $\triangle x_1 = \triangle X\, e_1$ and $\triangle s_1 = \triangle S\, e_1$. While (19) offers an appropriate way to compute $[\mathbb{D} f_j(S)]_{11}$, the matrices $[\mathbb{D} S^j]_{11}$ can be computed more efficiently by making use of the recursion

$$\mathbb{D}S^1(\triangle S) = \triangle S, \quad \mathbb{D}S^j(\triangle S) = (\mathbb{D}S^{j-1}(\triangle S))S + S^{j-1}\triangle S, \quad j \geq 2,$$

implying

$$[\mathbb{D}S^1]_{11} = I_k, \quad [\mathbb{D}S^j]_{11} = s_{11}[\mathbb{D}S^{j-1}]_{11} + S^{j-1}, \quad j \geq 2.$$

Before continuing this process for the next columns of $\triangle X$ and $\triangle S$, we need to update the right hand sides of (16)–(17) after $\triangle x_1$ and $\triangle s_1$ have been computed from (20) (this bears resemblance to the forward substitution process for solving lower triangular systems). For this purpose, partition

$$\triangle X = [\triangle x_1, \triangle X_2], \quad \triangle S = [\triangle s_1, \triangle S_2], \quad R_T = [R_T e_1, R_{T2}], \quad R_V = [R_V e_1, R_{V2}],$$

and

$$S = \begin{bmatrix} s_{11} & s_{12} \\ 0 & S_{22} \end{bmatrix}, \quad f(S) = \begin{bmatrix} f(s_{11}) & [f(S)]_{12} \\ 0 & f(S_{22}) \end{bmatrix}$$

for some function f. Inserted into (16)–(17), we obtain the following linear matrix equation for the pair $(\triangle X_2, \triangle S_2) \in \mathbb{C}^{n\times(k-1)} \times \mathbb{C}^{k\times(k-1)}$:

$$\mathbb{T}(\triangle X_2, S_{22}) + \sum_{j=1}^{m} A_j X\,[\mathbb{D}f_j(S)]\,([0, \triangle S_2]) \begin{bmatrix} 0 \\ I_{k-1} \end{bmatrix} = \widetilde{R}_{T2}, \tag{21}$$

$$W_0^H \triangle X_2 + \sum_{j=1}^{l-1} W_j^H \left(\triangle X_2 S_{22}^j + X\,\mathbb{D}S^j\,([0, \triangle S_2]) \begin{bmatrix} 0 \\ I_{k-1} \end{bmatrix} \right) = \widetilde{R}_{V2}, \tag{22}$$

with updated right-hand sides

$$\widetilde{R}_{T2} = R_{T2} - \sum_{j=1}^{m} A_j \left(\triangle x_1 [f_j(S)]_{12} + X[\mathbb{D}f_j(S)]([\triangle s_1, 0]) \begin{bmatrix} 0 \\ I_{k-1} \end{bmatrix} \right),$$

$$\widetilde{R}_{V2} = R_{V2} - \sum_{j=1}^{l-1} W_j^H \left(\triangle x_1 [S^j]_{12} + X\,\mathbb{D}S^j\,([\triangle s_1, 0]) \begin{bmatrix} 0 \\ I_{k-1} \end{bmatrix} \right).$$

The first columns of the solutions to (21)–(22) can be computed similarly as for the original transformed equations (16)–(17). We refrain from providing the algorithmic details here and refer to Appendix 4 for a MATLAB implementation of the sketched forward substitution process. If $k \ll n$ and $l \ll n$ then the cost of the overall algorithm is dominated by the solution of k linear systems of the form (20). Since each of these systems has order $n + k$, the overall cost is $\mathcal{O}(k(n + k)^3)$ flops, which compares well with the $\mathcal{O}(k^3(n+k)^3)$ flops needed by the Kronecker product formulation. Moreover, if the matrices A_j are sparse then (20) is a bordered sparse system and a sparse direct solver, possibly adapted to such bordered matrices [3], could be used.

3.3 Improving global convergence

In an attempt to improve the global convergence of Algorithm 1, we have implemented a simple Armijo rule based on the residual norm

$$\|\mathbb{T}(X, S)\|_F = \|A_1 X f_1(S) + \cdots + A_m X f_m(S)\|_F. \tag{23}$$

More specifically from a discrete set $\{2^{-3}, 2^{-2}, 2^{-1}, 1\}$ of step sizes we choose the largest step size τ such that

$$\|\mathbb{T}(X_\tau, S_\tau)\|_F \leq (1 - 10^{-4}\tau)\|\mathbb{T}(X_p, S_p)\|_F, \tag{24}$$

where (X_τ, S_τ) is obtained from applying the orthogonalization steps 1–1 of Algorithm 1 to $(X_p + \tau\triangle X,\ S_p + \tau\triangle X)$. If (24) cannot be fulfilled we choose τ as small as possible, i.e., $\tau = 2^{-3}$. The next iterate is obtained as $(X_{p+1}, S_{p+1}) \leftarrow (X_\tau, S_\tau)$.

3.4 Application 1

We continue Example 3; computing eigenvalues for

$$T(\lambda) = \lambda I - A_0 - A_1 e^{-\lambda\tau} \tag{25}$$

with $A_0, A_1 \in \mathbb{C}^{n\times n}$ and a delay $\tau > 0$. For the stability analysis of the corresponding DDE $\dot{x}(t) = A_0 x(t) + A_1 x(t - \tau)$, it is of interest to compute eigenvalues with large real part. To obtain an initial guess, we approximate $T(\lambda)$ by a polynomial

$$T(\lambda) \approx P(\lambda) := \lambda I - A_0 - A_1 \sum_{i=0}^{\ell} \frac{1}{i!}(-\lambda\tau)^i. \tag{26}$$

and compute the k eigenvalues $\lambda_1, \ldots, \lambda_k$ of P that have largest real part. We then choose $S_0 = \operatorname{diag}(\lambda_1, \ldots, \lambda_k)$ and compute X_0 with Algorithm 2.

Example 13 [21, Sec. 2.4.1] Consider (25) for the matrices

$$A_0 = \begin{bmatrix} -5 & 1 \\ 2 & -6 \end{bmatrix}, \quad A_1 = \begin{bmatrix} -2 & 1 \\ 4 & -1 \end{bmatrix}$$

and $\tau = 1$. We aim at computing an invariant pair for 5 eigenvalues. In this case, the minimality index is at least 3.

Figure 1 displays the numerical results obtained from running the block Newton method proposed in this paper for this example with $k = 5$ and $l = 3$. For the initial approximation (26), we have chosen $\ell = 4$. Initially, three eigenvalues are well and

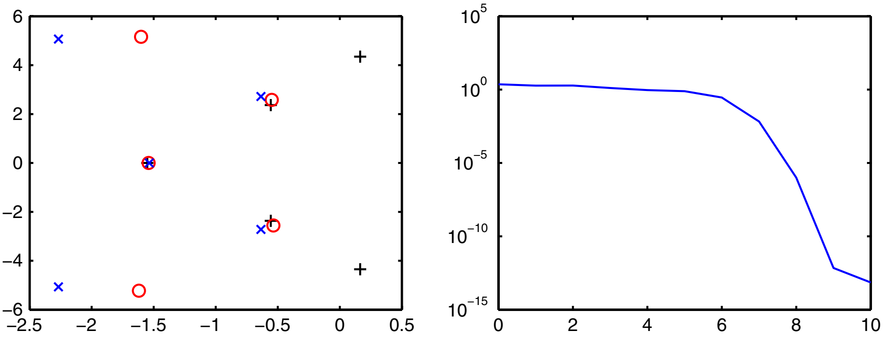

Fig. 1 Numerical results for Example 13. *Left plot:* Location of eigenvalue approximations initially (*plus*), after 3 Iterations (*circle*), and at convergence after 12 iterations (*cross*). *Right plot:* Residual norm (23) in the course of the Newton iteration

two eigenvalues are poorly approximated. During the first 2 iterations the step size is at the allowed minimum 2^{-3} before it successively increases to 1 at the sixth step, after which quadratic convergence sets in. Comparing with the results [21], it turns out that the converged eigenvalues are in fact the ones with largest real part. Note that the condition number of the Jacobian is 9.2×10^5 at convergence, which could explain the poor transient behavior of the Newton method.

Example 14 [21, Sec. 2.4.2] The experiments from Example 13 are repeated for the matrices

$$A_0 = \begin{bmatrix} -0.8498 & 0.1479 & 44.37 \\ 0.003756 & -0.2805 & -229.2 \\ -0.1754 & 0.02296 & -0.3608 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0.28 & 0 & 0 \\ 0 & -0.28 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

and $\tau = 1$, which has – according to [15] – applications in the stability analysis of a semiconductor laser subject to external feedback. This time, $\ell = 2$ in the approximation (26), and $k = 4, l = 2$. The step size is either 2^{-2} or 2^{-1} during the first 5 iterations and settles at 1 at the sixth iteration see Fig. 2.

The condition number of the Jacobian is 2.2×10^6 at convergence.

3.5 Application 2

As a second application, we consider a simple boundary eigenvalue problem, which was also considered in [24]. Find $\lambda > \kappa$ and a nonzero function $u : [0, 1] \to \mathbb{R}$ such that

$$-u''(y) = \lambda u(y), \quad u(0) = 0, \quad -u'(1) = f(\lambda)u(1), \tag{27}$$

where $f(\lambda) = \frac{\kappa M \lambda}{\lambda - \kappa}$ and $\kappa = \frac{K}{M}$ for given positive numbers K, M. The Eq. (27) describes the eigenvibrations of a string with a load of mass M attached by an elastic

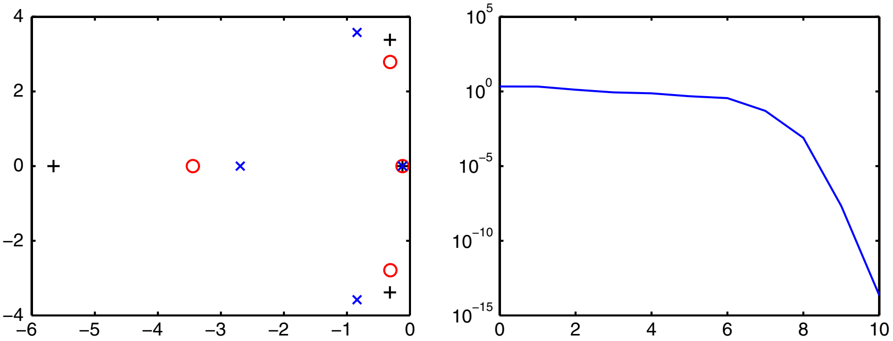

Fig. 2 Numerical results for Example 14. *Left plot:* Location of eigenvalue approximations initially (*plus*), after 3 Iterations (*circle*), and at convergence after 12 iterations (*cross*). *Right plot:* Residual norm (23) in the course of the Newton iteration

spring of stiffness K. In practice, two- or three-dimensional variants of (27) are used to model mechanical structures with elastically attached loads.

A finite element discretization of (27) with linear elements on subintervals of length $h = 1/n$ leads to the nonlinear matrix eigenvalue problem

$$(A_1 + f(\lambda)e_n e_n^T - \lambda A_3)x = 0, \tag{28}$$

where

$$A_1 = \frac{1}{h}\begin{bmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & 2 & -1 \\ & & -1 & 1 \end{bmatrix}, \quad A_3 = \frac{h}{6}\begin{bmatrix} 4 & 1 & & \\ 1 & \ddots & \ddots & \\ & \ddots & 4 & 1 \\ & & 1 & 2 \end{bmatrix}.$$

Note that (28) could be turned into a polynomial eigenvalue problem by multiplying with $\lambda - \kappa$ but this introduces the erroneous eigenvalue κ. Moreover, in more realistic problems with a large number of different springs, this strategy would lead to a polynomial of high degree and massive numerical cancellation can be expected when forming this polynomial.

Example 15 We consider (28) for $n = 100$ and $M = K = \kappa = 1$. We apply the Newton method with $k = 5$, $l = 1$. As initial pair, we choose $S_0 = \text{diag}(2, 2, 2, 2, 2)$ and a random matrix $X_0 \in \mathbb{R}^{n\times 5}$. Figure 3 reveals convergence despite this poor choice of initial eigenvalues. Also, quadratic convergence almost immediately sets in. We have repeated this experiment for $n = 400$ and obtained a rather similar convergence behavior.

For reference, we provide the first 10 decimal digits of the computed eigenvalues:

n	λ_1	λ_2	λ_3	λ_4	λ_5
100	4.4821765459	24.223573113	63.723821142	123.03122107	202.20089914
400	4.4820338110	24.219005847	63.692138408	122.91317036	201.88234012

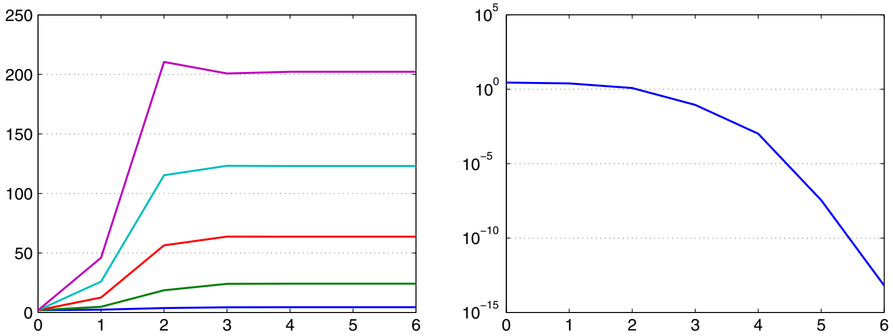

Fig. 3 Numerical results for Example 15. *Left plot:* Obtained eigenvalue approximations in the course of the Newton iteration. *Right plot:* Residual norm (23) in the course of the Newton iteration

These values compare well with the analytically computed values from [24] and reveal that the Newton method has in fact converged to the 5 smallest eigenvalues.

4 Conclusions

When little is known about a nonlinear eigenvalue problem at hand, the concept of invariant pairs proposed in this paper offers a robust way of representing several eigenvalues and eigenvectors simultaneously. We are not aware of any competitive alternative representation in the literature. To compute such invariant pairs, we have developed a block Newton method and described some algorithmic details, mainly to maintain a reasonable computational cost. However, it should be emphasized that our block Newton method inherits the disadvantages of similar methods [11] for solving linear eigenvalue problems: Its global convergence may be erratic and already a single slowly converging eigenvalue contained in S will hinder the convergence of the entire pair. To a certain extent, this is avoided in single-vector methods such as Jacobi-Davidson [7,23], Arnoldi [27], and preconditioned inverse iteration [24]. A logical next step of future research is to employ invariant pairs in single-vector methods for safely locking and purging converged eigenpairs, similar to the work by Meerbergen [19] on the quadratic eigenvalue problem.

Acknowledgments The author thanks the referees for carefully checking the paper and providing helpful comments. Discussions with Volker Mehrmann, TU Berlin, on the subject of this paper as well as joint work [6] with Timo Betcke, University of Reading, have provided an important source of inspiration.

Appendix A: Matlab code

The following MATLAB function implements the forward substitution method sketched in Sect. 3.2.

```
function [dX,dS] = nlevp_newtonstep( A, f, X, S, W, RT, RV )
% Computes the solution (dX,DS) to the linearized system in one
  step of
% the block Newton method for a nonlinear eigenvalue problem (NLEVP).
```

```
%
% Input: A - 3d-array containing the matrices A_j of the NLEVP
%        f - handle to a function f(j,M) that returns f_j(M)
%            for any square matrix M
%    (X,S) - current iterate. S is assumed upper triangular.
%        W - 3d-array containing the normalization matrices W_j
%  (RT,RV) - right-hand side
%
% Output: Solution (dX,dS) to the linearized equation in one step
%        of the block Newton method for the NLEVP.
n = size(A,1); m = size(A,3); k = size(X,2); l = size(W,3);
dX = zeros(n,k); dS = zeros(k,k);
% Precompute all required powers and functions of S.
fS = zeros(k,k,m); pS = zeros(k,k,l-1); pS(:,:,1) = S;
for j = 1:m, fS(:,:,j) = feval(f,j,S); end
for j = 2:l-1, pS(:,:,j) = pS(:,:,j-1)*S; end
% Main loop for computing the ith columns of dX and dS
for i = 1:k,
    % Set up and solve linear system
    s = S(i,i);
    T11 = zeros(n); for j = 1:m, T11 = T11 + A(:,:,j)*feval(f,j,s);
    end
    T12 = zeros(n,k);
    for j = 1:m,
        DF = feval(f,j,[S, eye(k);zeros(k) s*eye(k) ]);
        T12 = T12 + A(:,:,j)*X*DF(1:k,k+1:2*k);
    end
    T21 = W(:,:,1)’; for j = 2:l, T21 = T21 + s^(j-1) * W(:,:,j)’; end
    DS = eye(k); T22 = zeros(k);
    for j = 2:l, T22 = T22 + W(:,:,j)’*X*DS; DS = s*DS + pS(:,:,j-1);
    end
    sol = [T11 T12; T21 T22] \ [RT(:,i);RV(:,i)];
    dX(:,i) = sol(1:n); dS(:,i) = sol(n+1:end);
    % Update right-hand side
    Z = zeros(k); Z(:,i) = dS(:,i); DS = Z;
    for j = 1:m,
        DF = feval(f,j,[S, Z;zeros(k) S ]);
        RT(:,i+1:k) = RT(:,i+1:k) - A(:,:,j) * ( dX(:,i)*fS(i,i+1:k,j)
           + X*DF(1:k,k+i+1:2*k) );
    end
    for j = 2:l,
        RV(:,i+1:k) = RV(:,i+1:k) - W(:,:,j)’ * ( dX(:,i)*pS(i,i+1:
             k,j-1) + X*DS(:,i+1:k) );
        DS = DS*S + pS(:,:,j-1)*DS;
    end
end
```

References

1. Absil, P.-A., Mahony, R., Sepulchre, R., Van Dooren, P.: A Grassmann–Rayleigh quotient iteration for computing invariant subspaces. SIAM Rev. **44**(1), 57–73 (2002)
2. Bai, Z., Demmel, J.W., Dongarra, J.J., Ruhe, A., van der Vorst, H (ed.).: Templates for the Solution of Algebraic Eigenvalue Problems: Software, Environments, and Tools. SIAM, Philadelphia (2000)
3. Benzi, M., Golub, G.H., Liesen, J.: Numerical solution of saddle point problems. Acta Numer. **14**, 1–137 (2005)

4. Betcke, M., Voss, H.: Stationary Schrödinger equations governing electronic states of quantum dots in the presence of spin-orbit splitting. Appl. Math. **52**(3), 267–284 (2007)
5. Betcke, M.M., Voss, H.: Restarting projection methods for rational eigenproblems arising in fluid-solid vibrations. Math. Model. Anal. **13**(2), 171–182 (2008)
6. Betcke, T., Kressner, D.: Perturbation, computation and refinement of invariant pairs for matrix polynomials. Technical report 2009-21, seminar for applied mathematics, ETH Zurich, July 2009
7. Betcke, T., Voss, H.: A Jacobi-Davidson type projection method for nonlinear eigenvalue problems. Future Gener. Comput. Syst. **20**, 363–372 (2004)
8. Beyn, W.-J., Kleß, W., Thümmler, V.: Continuation of low-dimensional invariant subspaces in dynamical systems of large dimension. In: Ergodic theory, analysis, and efficient simulation of dynamical systems, pp. 47–72. Springer, Berlin (2001)
9. Beyn, W.-J., Thümmler, V.: Continuation of invariant subspaces for parameterized quadratic eigenvalue problems. Technical report, University of Bielefeld, Department of Mathematics (2008)
10. Chatelin, F.: Simultaneous Newton's iteration for the eigenproblem. In: Defect correction methods (Oberwolfach, 1983), volume **5** of Comput. Suppl., pp. 67–74. Springer, Vienna (1984)
11. Demmel, J.W.: Three methods for refining estimates of invariant subspaces. Computing **38**, 43–57 (1987)
12. Dennis, J.E. Jr., Traub, J.F., Weber, R.P.: The algebraic theory of matrix polynomials. SIAM J. Numer. Anal. **13**(6), 831–845 (1976)
13. Gaier, D.: Vorlesungen über Approximation im Komplexen. Birkhäuser Verlag, Basel (1980)
14. Gohberg, I., Lancaster, P., Rodman, L.: Matrix Polynomials. Academic Press. [Harcourt Brace Jovanovich Publishers], New York, Computer Science and Applied Mathematics (1982)
15. Green, K., Wagenknecht, T.: Pseudospectra and delay differential equations. J. Comput. Appl. Math. **196**(2), 567–578 (2006)
16. Higham, N.J.: Functions of Matrices. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2008)
17. Krantz, S.G.: Function Theory of Several Complex Variables. Wiley, New York (1982)
18. Mathias, R.: A chain rule for matrix functions and applications. SIAM J. Matrix Anal. Appl. **17**(3), 610–620 (1996)
19. Meerbergen, K.: Locking and restarting quadratic eigenvalue solvers. SIAM J. Sci. Comput. **22**(5), 1814–1839 (2000)
20. Mehrmann, V., Voss, H.: Nonlinear eigenvalue problems: a challenge for modern eigenvalue methods. GAMM Mitteilungen **27** (2004)
21. Michiels, W., Niculescu, S.-I.: Stability and Stabilization of Time-delay Systems, vol. 12 of Advances in Design and Control. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2007)
22. Najfeld, I., Havel, T.F.: Derivatives of the matrix exponential and their computation. Adv. Appl. Math. **16**, 321–375 (1995)
23. Sleijpen, G.L.G., Booten, A.G.L., Fokkema, D.R., Van der Vorst, H.A.: Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems. BIT **36**(3), 595–633 (1996)
24. Solovev, S.I.: Preconditioned iterative methods for a class of nonlinear eigenvalue problems. Linear Algebra Appl. **415**(1), 210–229 (2006)
25. Stewart, G.W., Sun, J.-G.: Matrix Perturbation Theory. Academic Press, New York (1990)
26. Voss, H.: A maxmin principle for nonlinear eigenvalue problems with application to a rational spectral problem in fluid-solid vibration. Appl. Math. **48**(6), 607–622 (2003)
27. Voss, H.: An Arnoldi method for nonlinear eigenvalue problems. BIT **44**(2), 387–401 (2004)
28. Voss, H.: A Jacobi-Davidson method for nonlinear eigenproblems. In: Computational Science—ICCS 2004. Part II, vol. 3037 of Lecture Notes in Computer Science, pp. 34–41. Springer, Berlin (2004)
29. Voss, H.: Iterative projection methods for computing relevant energy states of a quantum dot. J. Comput. Phys. **217**(2), 824–833 (2006)
30. Voss, H.: A Jacobi-Davidson method for nonlinear and nonsymmetric eigenproblems. Comput. Struct. **85**(17/18), 1284–1292 (2007)
31. Voss, H., Werner, B.: A minimax principle for nonlinear eigenvalue problems with applications to nonoverdamped systems. Math. Methods Appl. Sci. **4**(3), 415–424 (1982)