SIAM J. Sci. Comput.
Vol. 32, No. 6, pp. 3151–3169

A MULTILEVEL JACOBI–DAVIDSON METHOD FOR POLYNOMIAL PDE EIGENVALUE PROBLEMS ARISING IN PLASMA PHYSICS*

MARLIS HOCHBRUCK† AND DOMINIK LÖCHEL†

Abstract. The simulation of drift instabilities in the plasma edge leads to cubic polynomial PDE eigenvalue problems with parameter dependent coefficients. The aim is to determine the wave number which leads to the maximum growth rate of the amplitude of the wave. This requires the solution of a large number of PDE eigenvalue problems. Since we are only interested in a smooth eigenfunction corresponding to the eigenvalue with largest imaginary part, the Jacobi–Davidson method can be applied. Unfortunately, a naive implementation of this method is much too expensive for the large number of problems that have to be solved. In this paper we will present a multilevel approach for the construction of an appropriate initial search space. We will also discuss the efficient solution of the correction equation, and we will show how optimal scaling helps to accelerate the convergence.

Key words. polynomial eigenvalue problem, PDE eigenvalue problem, Jacobi–Davidson method, multilevel method, parameter dependent problem, optimal scaling, drift instabilities

AMS subject classification. 65L15

DOI. 10.1137/090774604

1. Introduction.

In this paper we consider polynomial eigenvalue problems of the form

$$p_0(\omega,\theta)\phi(\theta) + p_1(\omega,\theta)\frac{\partial}{\partial\theta}\phi(\theta) + p_2(\omega,\theta)\frac{\partial^2}{\partial\theta^2}\phi(\theta) = 0, \tag{1.1}$$

where

$$p_0(\omega,\theta) = \sum_{j=0}^{d_0} a_j(\theta)\omega^j, \quad p_1(\omega,\theta) = \sum_{j=0}^{d_1} b_j(\theta)\omega^j, \quad p_2(\omega,\theta) = \sum_{j=0}^{d_2} c_j(\theta)\omega^j. \tag{1.2}$$

Here, the coefficients a_j, b_j, c_j are smooth, complex valued functions of θ, which might depend on certain other parameters, and d_j denotes the degree of the polynomials p_j.

The application we are interested in is the study of instabilities in the plasma edge of a tokamak, which is a magnetic fusion device. One of the problems to make magnetic fusion efficient is to reduce energy losses resulting from the unavoidable transport of plasma towards the wall. These losses arise due to microinstabilities in the plasma edge; see [15, 18, 19, 20, 36] for details. In [15, 33] a model for the simulation of the so-called anomalous transport is derived, which—after several simplifications—leads to a cubic eigenvalue problem of the form (1.1) with $d_0 = 3$ and $d_1 = d_2 = 2$. The eigenvalue of interest is the one with maximum imaginary part, since it contributes most to the losses. All functions a_j, b_j, c_j are 2π-periodic in θ. The 2π-periodic eigenfunction ϕ corresponds to the envelope of the electric potential perturbation for a certain wave number K, which is one of the parameters entering the coefficient functions a_j, b_j, and c_j. To be more precise, the physicists aim at finding the wave number

*Received by the editors October 22, 2009; accepted for publication (in revised form) June 21, 2010; published electronically October 21, 2010. This work has been supported by the Deutsche Forschungsgemeinschaft through the research training group GRK 1203.
http://www.siam.org/journals/sisc/32-6/77460.html

†Fakultät für Mathematik, Karlsruher Institut für Technologie (KIT), Kaiserstraße 93, D–76133 Karlsruhe, Germany (marlis.hochbruck@kit.edu, dominik.loechel@kit.edu).

K which maximizes the energy loss. Solving this optimization problem requires the solution of many nonlinear eigenvalue problems.

We would like to emphasize that the proposed algorithm can be applied to general problems (1.1), where the desired eigenfunction is smooth. For instance, the well-known Mathieu equation fits into this class of problems (with $d_j = 2$, $j = 0, 1, 2$). One can also approximate eigenpairs with other properties than having largest imaginary part.

The aim of this paper is to present an efficient eigenvalue solver for polynomial PDE eigenvalue problems based on the Jacobi–Davidson method [28, 27]. It turns out that a naive implementation of this method is not efficient for solving a large number of parameter dependent problems, as it is required in our application. The contribution of this paper is twofold: first of all we propose accelerating the convergence by using a multilevel approach for finding a good initial search space. A similar idea was recently used in [21] for generalized eigenvalue problems, where p-hierachical finite element spaces are used to accelerate the convergence of the Jacobi–Davidson method in the preasymptotic convergence regime. It is mentioned in [21] that h-hierachical finite element spaces could also be used, although this was not investigated in detail. Clearly, a multilevel approach can also be used for the solution of the correction equation, as discussed in [7]. Both approaches can be easily combined.

In contrast to the application considered in [21], where the dominant eigenvalues are of interest, we are interested in the eigenpair corresponding to the eigenvalue of largest imaginary part. It turns out that it is not at all obvious to follow the correct eigenpair within a multilevel process for a cubic polynomial eigenvalue problem, where, in general, three eigenvalues correspond to one eigenfunction. Therefore, we carefully investigate the selection of the desired Ritz pair by introducing a suitable similarity measure which also allows us to follow eigenpairs corresponding to a sequence of parameters (such as the wave number K).

The paper is organized as follows. In section 2 we briefly discuss the spatial discretization. Moreover, we recall the standard companion linearization of polynomial eigenvalue problems. Section 3 gives a short summary of the Jacobi–Davidson method for polynomial eigenvalue problems. In section 4, we present our multilevel approach, where we use a hierarchy of grids to find a suitable initial search space which will then lead to fast convergence of the Jacobi–Davidson method. We will discuss the selection of the Ritz pair, the efficient solution of the correction equation, error control, and scaling. Finally, in section 5 we present some numerical examples to illustrate the performance of our method for the application of studying the instabilities in the plasma edge. Section 5.4 is devoted to parameter dependent problems, in particular to determine the wave number K which maximizes the growth rate.

2. Preliminaries.

To solve the eigenvalue problem (1.1), we first discretize it in space. Since the functions arising in (1.1) are 2π-periodic, we choose an even number N of grid points and define a grid

$$0 = \theta_1 < \theta_2 < \cdots < \theta_N < 2\pi, \qquad \theta_{N/2+1} = \pi.$$

For physical reasons, it is essential to include 0 and π in the grid. The discretization of the derivatives then leads to a cubic matrix eigenvalue polynomial

$$\begin{aligned} P(\omega) = {} & P_0(\omega) + P_1(\omega) D_1 + P_2(\omega) D_2 \\ =: {} & \omega^3 M_3 + \omega^2 M_2 + \omega M_1 + M_0. \end{aligned} \tag{2.1}$$

Here, the matrices

$$P_j(\omega) = \operatorname{diag}\bigl(p_j(\omega, \theta)\bigr)$$

are diagonal, and

$$\theta = [\theta_1, \dots, \theta_N]^T$$

denotes the vector of grid points. The spatial discretizations of the first and the second derivative are represented by D_1 and D_2, respectively. Since our application leads to one-dimensional problems, we consider using finite differences or a spectral method based on Fourier transformation; see, e.g., [34]. However, in two- or three-dimensional problems, *hp*-hierarchical finite element discretizations might be preferable.

2.1. Finite differences. We use a standard finite difference approximation

$$\frac{\partial}{\partial\theta}\phi(\theta_j) \approx q_j'(\theta_j), \qquad \frac{\partial^2}{\partial\theta^2}\phi(\theta_j) \approx q_j''(\theta_j), \tag{2.2}$$

where q_j denotes the unique interpolation polynomial of degree $2m < N$ interpolating

$$q_j(\theta_{j\pm k}) = \phi(\theta_{j\pm k}), \qquad k = 0, 1, \dots, m.$$

On an equidistant grid, the approximations are of order $2m$.

Let $\Phi = (\phi(\theta_j))_{j=1,\dots,N} \in \mathbb{C}^N$ be the vector containing the function ϕ evaluated on the grid. Then (2.2) can be written as

$$q_j'(\theta_j) = (D_1\Phi)_j, \qquad q_j''(\theta_j) = (D_2\Phi)_j$$

with matrices $D_{1,2}$ of upper and lower periodic bandwidth m, i.e., band matrices with additional nonzero elements in the lower left and upper right corner.

2.2. Spectral method. A second option is to use a spectral method. Here we use (2.2) with q being the trigonometric interpolation polynomial which interpolates $\phi(\theta_j) = q(\theta_j)$, $j = 1, \dots, N$. The differentiation matrices are given by

$$D_j = \mathcal{F}_N^{-1}\mathcal{D}_j\mathcal{F}_N,$$

where $\mathcal{F}_N$ denotes the discrete Fourier matrix and

$$\begin{aligned}\mathcal{D}_1 &= i\,\operatorname{diag}\,(0, 1, \dots, \tfrac{N}{2} - 1, 0, -\tfrac{N}{2} + 1, \dots, -1),\\ \mathcal{D}_2 &= -\operatorname{diag}\,(0, 1, \dots, \tfrac{N}{2} - 1, \tfrac{N}{2}, -\tfrac{N}{2} + 1, \dots, -1)^2.\end{aligned}$$

Note that spectral approximation leads to dense, complex differentiation matrices $D_{1,2}$. The benefit is that periodic boundary conditions are fulfilled automatically and that derivatives of a given function can be computed in $\mathcal{O}(N \log N)$ operations by fast Fourier transformation. Moreover, for smooth functions the accuracy is much higher than for finite differences.

2.3. Linearization. In the Jacobi–Davidson algorithm and in our multilevel approach we introduce later on, polynomial eigenvalue equations of the same form as in (2.1) but of sufficiently small dimension need to be solved. The standard approach for solving such problems is to use an equivalent linearized formulation, e.g.,

$$\left(\omega\begin{bmatrix} M_3 & & \\ & I & \\ & & I\end{bmatrix} + \begin{bmatrix} M_2 & M_1 & M_0\\ -I & & \\ & -I & \end{bmatrix}\right)\begin{bmatrix}\omega^2\Phi\\ \omega\ \Phi\\ \Phi\end{bmatrix} = 0 \tag{2.3}$$

or

$$\left(\omega \begin{bmatrix} M_3 & M_2 & M_1 \\ & I & \\ & & I \end{bmatrix} + \begin{bmatrix} & & M_0 \\ -I & & \\ & -I & \end{bmatrix}\right) \begin{bmatrix} \omega^2\Phi \\ \omega\ \Phi \\ \Phi \end{bmatrix} = 0, \tag{2.4}$$

which are generalized eigenvalue problems of triple dimension. These linearizations are called companion form linearizations [8, 9, 10]. We will discuss other linearizations in section 4.7 and present numerical examples in section 5.3. The standard algorithm for solving (2.3) is the QZ algorithm [1, 6, 12]. However, it requires at least $O(N^3)$ floating point operations and is therefore inefficient for large scale problems but well suited for small problems.

3. Jacobi–Davidson method.

Since we are interested in very few eigenpairs of the cubic eigenvalue problem (2.1), a Ritz–Galerkin method seems appropriate. There are several Krylov subspace methods like Lanczos- and Arnoldi-type methods which are very efficient in finding eigenvalues in a user-selected part of the spectrum. However, the shift and invert strategies require suitable preconditioners in each iteration step. For our application we suggest applying the Jacobi–Davidson-scheme, which is a Ritz–Galerkin method with a particularly clever strategy to extend the search space. In contrast to Lanczos- and Arnoldi-type methods, the augmentation is done via a Newton step which requires only an approximative solution of a linear system; see [28, 27]. Due to its Newton like behavior, global convergence of the Jacobi–Davidson method cannot be ensured. However, if a good initial approximation of an eigenpair is available, then locally quadratic convergence can be expected. Thus the Jacobi–Davidson method will be the choice for our multilevel approach.

3.1. Basic algorithm.

To keep this paper self-contained, we briefly summarize the Jacobi–Davidson algorithm. Let $V \in \mathbb{C}^{N,k}$ contain an orthonormal basis of the current search space on a grid G. We want to determine a Ritz vector $u = Vy$, $y \in \mathbb{C}^k$ and a Ritz value ν satisfying the Galerkin condition

$$P(\nu)Vy \perp V.$$

This leads to the k-dimensional polynomial eigenvalue problem

$$V^H P(\nu)Vy = 0. \tag{3.1}$$

Since $k \ll N$, this eigenvalue problem can easily be solved by the QZ algorithm after linearization.

An optimal extension to the search space V would be to take an orthogonal correction $t \perp V$ such that $u + t$ is an exact eigenvector to the sought eigenvalue ω,

$$P(\omega)(u + t) = 0, \quad t \perp u.$$

However, since ω is unknown, we replace it by its current approximation ν. This leads to the correction equation

$$\left(I - \frac{P'(\nu)uu^H}{u^H P'(\nu)u}\right) P(\nu) \left(I - \frac{uu^H}{u^H u}\right) t = -r, \qquad t \perp u; \tag{3.2}$$

see [29] and references given there. A nice interpretation is that this corresponds to one Newton step applied to

$$F(\lambda, x) := \begin{pmatrix} P(\lambda)x \\ x^H x - 1 \end{pmatrix} = 0 \tag{3.3}$$

with starting values (ν, u). The Jacobi–Davidson method is summarized in Algorithm 1. It has been shown that it converges locally quadratically if the correction equation (3.2) is solved exactly; cf. [23, 24, 27]. If the correction equation is only solved approximately, then one can still expect fast convergence [35].

Algorithm 1 Jacobi–Davidson algorithm.

```
Require: matrix polynomial $P(\omega) = \sum_{j=0}^{3} \omega^l M_j$, $M_j \in \mathbb{C}^{N \times N}$
 1: choose an initial search space $V = [v^1, \dots, v^k] \in \mathbb{C}^{N \times k}$, $1 \le k \ll N$
 2: loop
 3:   orthonormalize $V$
 4:   compute $W_j := M_j V$ and $H_j := V^H W_j$ for $j = 0, \dots, 3$
 5:   compute desired eigenpair(s) $(\nu, y)$ of projected equation
          $\widetilde{P}(\nu) := \sum_{j=0}^{3} \nu^j H_j y = 0, \quad \|y\| = 1$
 6:   compute Ritz vector in original space $u := Vy$
 7:   compute residual $r := P(\nu)u$
 8:   if $\|r\| <$ tolerance then
 9:     stop
10:   end if
11:   compute $w := P'(\nu)u = \sum_{j=1}^{3} j\nu^{j-1} M_j u$
12:   find $t$ such that the correction equation (3.2) is solved approximately
13:   expand search space $V := [V, t]$
14: end loop
```

Algorithm 1 leaves a lot of freedom in various steps, namely, the choice of the initial subspace (line 1), the selection of the Ritz pair (line 5), and the solution of the correction equation (line 12). The efficiency of the method for a particular application strongly depends on these steps. We postpone the first two aspects to section 4 and start with the solution of the correction equation.

3.2. Solution of correction equation. Assume that we are given an exact inverse $Q(\nu)$ of the matrix polynomial $P(\nu)$, i.e., $Q(\nu)P(\nu) = I$. Then the Newton step is equivalent to

$$t = \frac{u^H Q(\nu) r}{u^H Q(\nu) P'(\nu) u} Q(\nu) P'(\nu) u - Q(\nu) r = \frac{u^H z}{u^H s} s - z. \tag{3.4}$$

Computing t requires the solution of two linear systems, $P(\nu)z = r$ and $P(\nu)s = P'(\nu)u$; cf. [29, 35]. The advantage of representing t by the so-called one-step approximation (3.4) is that the orthogonality condition in (3.2) is eliminated. Moreover, there is no need to compute t exactly, since it is only used to expand the search space. Due to the robustness of the search space expansion [35], we recommend determining an approximative inverse $Q_f(\nu)$ such that $Q_f(\nu)P_f(\nu) = I$, where P_f is the polynomial resulting from a finite difference approximation of (1.1) even if the spectral method is used to determine P. Since $P_f(\nu)$ is a periodic band matrix, its LU decomposition is sparse, and the linear systems can be solved with $\mathcal{O}(N)$ operations only.

The solution $t \perp u$ is added to the search space V. At the beginning of the next cycle, the columns of the search space V are reorthogonalized because t is not necessarily perpendicular to the whole search space.

4. A multilevel Jacobi–Davidson method.

In this section we discuss an efficient implementation of the Jacobi–Davidson method. We first assume that all parameters of the problem (such as the wave number K) are fixed. The Jacobi–Davidson algorithm can be interpreted as an accelerated Newton method, and thus global convergence cannot be assured. A good initial guess of the desired eigenpair enables us to enter the regime of locally quadratic convergence. Recently, in [21], Nickel and Dyczij-Edlinger used a p-hierarchical finite element method, where the polynomial degree of the finite element basis is increased while the approximation of the eigenpair improves. Here we discuss the construction of an appropriate initial vector using a multilevel approach.

We would like to note that multilevel or multigrid methods can also be employed in the Jacobi–Davidson algorithm for solving the linear system of the correction equation [7]. However, in our one-dimensional problem (1.1), multigrid methods do not excel preconditioning with the finite differences derivative approximation as described in section 3.2, but this changes in two or three space dimensions. In any case, the most efficient solver available for a particular application should be used for the solution of the correction equation.

4.1. A multilevel approach.

The main idea of our algorithm is to use a nested sequence $G_\ell \subset G_{\ell+1}$ of spatial grids

$$G_\ell = \{\theta_{j,\ell},\ j = 1, \dots, N_\ell\}, \qquad \ell = 0, \dots, L,$$

where $N_{l+1} = 2N_\ell$ and

$$\theta_{j,\ell} = (j-1)\,\frac{h_0}{2^\ell}, \quad j = 1, \dots, N_\ell.$$

Here, $h_0 = 2\pi/N_0$ denotes the mesh width of the coarsest grid.

We start from the coarsest grid G_0 and compute an approximation to the desired eigenpair. The number N_0 of grid points of G_0 is chosen such that the dimension of the linearized problem is so small that the QZ algorithm is efficient. Then we correct this approximation on the sequence G_ℓ of refined grids.

4.2. Refining the approximation.

Assume that we already have an approximation to the desired eigenpair on grid $G_{\ell-1}$ and want to compute the solution on grid G_ℓ. We first prolongate the eigenfunction from $G_{\ell-1}$ to the next finer grid G_ℓ and call the interpolated function Φ the reference eigenfunction. The interpolation is either done by cubic spline interpolation with periodic boundary conditions in the case of finite difference discretizations or by trigonometric interpolation for the spectral method. The reference eigenfunction Φ is used as the basis of the initial search space V_ℓ on grid G_ℓ. The eigenvalue approximation together with the prolongated eigenfunction becomes the reference eigenpair (ω, Φ). On grid G_ℓ, we select the Ritz pair that is closest to the reference eigenpair; see [31] for a different application using similarity of data.

Cubic eigenvalue problems of dimension N_ℓ have $3N_\ell$ eigenpairs. Thus the eigenvectors are linearly dependent, and the eigenvalues are not necessarily distinct. This requires a suitable similarity measure which takes eigenvalues and eigenvectors into account. The similarity between the Ritz value ν and the reference eigenvalue ω is measured via the relative distance

$$\operatorname{sim}(\nu, \omega) = \exp\left(-\frac{|\nu - \omega|}{|\nu| + |\omega|}\right), \tag{4.1}$$

while the similarity between the eigenfunctions is measured as usual via the angle between the vectors,

$$\operatorname{sim}(u,\Phi) = \left|u^H\Phi\right|, \qquad \|\Phi\|_2 = \|u\|_2 = 1. \tag{4.2}$$

Both similarity measures take values in the interval $[0,1]$, where one is attained if the eigenpairs are identical. Values close to zero arise if the eigenvectors are almost orthogonal and the eigenvalues have large distance.

We suggest combining both similarity measures and select the Ritz pair (ν, u), which maximizes

$$\operatorname{sim}((\nu,u),(\omega,\Phi)) := \operatorname{sim}(\nu,\omega)\operatorname{sim}(u,\Phi). \tag{4.3}$$

Remark. It might be tempting to approximate and refine several eigenpairs simultaneously; see [4, 13, 17]. However, we found that in our physical application the initial search space constructed by the multilevel procedure leads to quadratic convergence behavior from the very beginning of the Jacobi–Davidson iteration. Thus the block version turned out to be less efficient here. Nevertheless, the techniques proposed in this paper naturally generalize to block methods.

4.3. Projection of eigenvalue equation. In the Jacobi–Davidson Algorithm 1, the k-dimensional eigenvalue problem (3.1) has to be solved. This requires the computation of the projected problem. The multiplication from the right can be computed in $\mathcal{O}(kqN)$ operations if the discretization is done by finite differences and in $\mathcal{O}(kN\log N)$ operations via fast Fourier transformation if the spectral method is used. To achieve this, the multiplication with V has to be done separately for each matrix coefficient M_j of the matrix polynomial (2.1). The multiplication from the left can be computed in $\mathcal{O}(k^2N)$ operations only. Thus, the most expensive operation is to apply the discrete derivative operators to the basis of the search space (i.e., the columns of V). Fortunately, this has to be done only once per iteration.

4.4. Error control. Since it is well known that residual bounds are not reliable error indicators, we suggest stopping the iteration based on the forward error. The forward error can be approximately bounded by the product of the backward error and the condition number; see [8, 10] and [32]. The normwise variant provides sharp estimates but is computationally expensive, since it involves the solution of standard eigenvalue problems. The componentwise variant is applicable but invariant under scaling [2].

Let

$$\Delta P(\lambda) := \sum_{j=0}^{d} \lambda^j \Delta M_j$$

be the perturbation polynomial, and let $(\widetilde{\lambda},\widetilde{x})$ be an approximative eigenpair and $r := P(\widetilde{\lambda})\widetilde{x}$ its residual. In [32] the normwise backward error $\eta(\widetilde{\lambda},\widetilde{x})$ is defined as

$$\eta(\widetilde{\lambda},\widetilde{x}) := \min\left\{\varepsilon \;:\; \big(P(\widetilde{\lambda}) + \Delta P(\widetilde{\lambda})\big)\widetilde{x} = 0,\ \|\Delta M_j\|_2 \le \varepsilon\|E_j\|_2,\ j = 0,\dots,d\right\},$$

where d denotes the degree of the matrix polynomial ($d = 3$ in our application) and E_j are arbitrary matrices that represent the tolerances against which the perturbations

ΔM_j to M_j will be measured. It was shown in [32, Theorem 1] that

$$\eta(\widetilde{\lambda}, \widetilde{x}) = \frac{\|r\|_2}{\widetilde{\alpha}\|\widetilde{x}\|_2}, \qquad \widetilde{\alpha} = \sum_{j=0}^{d} |\widetilde{\lambda}|^j \|E_j\|_2. \tag{4.4}$$

For an exact eigenpair(λ, x), the normwise condition number is defined as

$$\kappa(\lambda, P) := \lim_{\varepsilon \to 0} \sup \left\{ \frac{|\Delta\lambda|}{\varepsilon|\lambda|} \ : \ \big(P(\lambda + \Delta\lambda) + \Delta P(\lambda + \Delta\lambda)\big)(x + \Delta x) = 0, \right.$$
$$\left. \|\Delta M_j\|_2 \le \varepsilon \|E_j\|_2,\ j = 0, \dots, d \right\}.$$

Let y be a left eigenvector corresponding to λ. From [32, Theorem 5] we have

$$\kappa(\lambda, P) = \frac{\alpha\|y\|_2\|x\|_2}{|\lambda||y^H P'(\lambda)x|}, \qquad \alpha = \sum_{j=0}^{d} |\lambda|^j \|E_j\|_2. \tag{4.5}$$

If the condition number $\kappa(\lambda, P)$ is approximated by using the approximate eigentriple $(\widetilde{y}, \widetilde{\lambda}, \widetilde{x})$ instead of the exact one in (4.5), we have $\widetilde{\alpha} = \alpha$, and the forward error estimate becomes

$$\eta(\widetilde{\lambda}, \widetilde{x})\kappa(\widetilde{\lambda}, P) = \frac{\|r\|_2\|\widetilde{y}\|_2}{|\widetilde{\lambda}||\widetilde{y}^H P'(\widetilde{\lambda})\widetilde{x}|}. \tag{4.6}$$

The iteration is stopped if the forward error estimate (4.6) is below a given threshold, which will be called the relative accuracy of the eigenvalue in the following.

4.5. Optimal scaling. Ultimately, we are interested in computing eigenvalues as accurate as possible. Hence, we would like to find a scaling that leads to small forward errors. The physical equations are scaled such that $\|M_j\|_\infty \approx 1$. In our application, the diagonal matrices are well conditioned but the condition of the discrete derivative matrices grows quadratically with the number N of grid points. Besides the general scaling of the physical equations, the optimal scaling depends on the specific eigentriple; see [2] and references given there. An optimal scaling aims at an eigenvalue with eigenvectors whose components are of equal magnitude; see [2, Theorem 3.3]. To be more precise, let $(\Phi_{\mathrm{l}}, \omega, \Phi_{\mathrm{r}})$ be the desired eigentriple of P with all components of Φ_{l} and Φ_{r} being different from zero and $\omega \neq 0$. Then define optimally scaled vectors

$$\widehat{\Phi}_{\mathrm{r,l}} = S_{\mathrm{r,l}}^{-1}\Phi_{\mathrm{r,l}}, \qquad S_{\mathrm{r,l}} = \operatorname{diag}(|\Phi_{\mathrm{r,l}}|).$$

Since linearization of a polynomial eigenvalue problem leads to eigenvectors with components multiplied by powers of the eigenvalue ω, it is also necessary to scale the eigenvalue. Thus we define the scaled eigenvalue $\widehat{\omega}$ via

$$\omega = |\omega|\widehat{\omega}$$

and obtain the scaled polynomial eigenvalue problem

$$\widehat{P}(\widehat{\omega})\widehat{\Phi}_{\mathrm{r}} := S_{\mathrm{l}} P(|\omega|\widehat{\omega}) S_{\mathrm{r}} \widehat{\Phi}_{\mathrm{r}} = 0.$$

A good approximation $(\omega, \Phi_{\mathrm{r}})$ leads to $|\widehat{\omega}| \approx 1$ and $|\widehat{\Phi}_r| \approx (1, \ldots, 1)^T$ so that the entries of the eigenvector of the linearized system are of modulus close to one. This gives a nearly optimal scaling. However, the left eigenvector of the linearized system is not optimally scaled by companion form linearizations. Therefore we will discuss block-symmetric linearizations in section 4.7.

Unfortunately, the optimal scaling requires the unknown eigenpair. However, our multilevel approach can also be used to provide a nearly optimal scaling (see Algorithm 2). On the coarsest grid G_0, the spatial grid size is quite large. Thus the derivative matrices $D_{1,2}$ are well conditioned, and the QZ algorithm gives enough accuracy. On subsequently finer grids G_ℓ, $\ell \geq 1$, we interpolate the approximative eigenfunction from grid $G_{\ell-1}$ and use this approximation for scaling.

The effect of scaling in the multilevel Jacobi–Davidson algorithm will be discussed in section 5.3 below.

ALGORITHM 2 MULTILEVEL JACOBI–DAVIDSON APPROACH INCLUDING SCALING.

```
solve $P(\omega)\Phi_{\mathrm{r}} = 0$ on the coarsest grid $G_0$
select the desired eigenpair $(\omega, \Phi_{\mathrm{r}})$
**for** $\ell = 1, \ldots, L$ **do**
    prolongate $\Phi_{\mathrm{r}}$ to the next finer grid $G_\ell$
    compute $\Phi_{\mathrm{l}}$
    scale $P$: $\widehat{P}(\widehat{\omega}) := \mathrm{diag}(|\Phi_{\mathrm{l}}|)P(|\omega|\widehat{\omega})\,\mathrm{diag}(|\Phi_{\mathrm{r}}|)$
    solve $\widehat{P}(\widehat{\omega})\Psi = 0$,    $\omega \leftarrow |\omega|\widehat{\omega}$,    $\Phi_{\mathrm{r}} \leftarrow \mathrm{diag}(|\Phi_{\mathrm{r}}|)\Psi$
**end for**
```

4.6. Two-sided Rayleigh quotient iteration. It is well known that the Rayleigh quotient iteration of standard eigenvalue equations $Ax = \lambda x$ converges cubically for normal matrices and quadratically for nonnormal matrices [11]. However, Ostrowski's two-sided Rayleigh quotient iteration [22]

$$\nu(u_{\mathrm{l}}, u_{\mathrm{r}}) := \frac{u_{\mathrm{l}}^H A u_{\mathrm{r}}}{u_{\mathrm{l}}^H u_{\mathrm{r}}},$$

where u_{l} and u_{r} are approximations to the left and right eigenvectors, respectively, converges cubically if the approximated eigenvalue is simple. A good overview of Rayleigh quotient iteration can be found in [26]. In the case where left and right eigenvectors are required for scaling or for error control, the projection of the eigenvalue problem in the Jacobi–Davidson cycle can be modified analogously to the two-sided Rayleigh quotient iteration [11]. Let V_{l} and V_{r} be the search spaces which are initialized by an approximation of the left and the right eigenvector, respectively. The projected eigenvalue equation then reads

$$y_{\mathrm{l}}^H V_{\mathrm{l}}^H P(\nu_{\mathrm{l}}) V_{\mathrm{r}} = 0 \quad \text{and} \quad V_{\mathrm{l}}^H P(\nu_{\mathrm{r}}) V_{\mathrm{r}} y_{\mathrm{r}} = 0.$$

The search space V_{r} is expanded as described in section 3.2, and V_{l} is expanded analogously. Here, we have two eigenvalue approximations ν_{l} and ν_{r}, whose difference tends to zero if the method converges.

4.7. Linearization. The conditioning and the backward error of different linearizations have been studied in [8, 9, 10]. In particular, it was shown that a whole class of block symmetric linearizations of symmetric matrix polynomials exists.

For our problem, we suggest using one of the following block symmetric linearizations

$$\left(\omega \begin{bmatrix} M_3 & & \\ & -M_1 & -M_0 \\ & -M_0 & \end{bmatrix} + \begin{bmatrix} M_2 & M_1 & M_0 \\ M_1 & M_0 & \\ M_0 & & \end{bmatrix}\right) \begin{bmatrix} \omega^2 \Phi \\ \omega\ \Phi \\ \Phi \end{bmatrix} = 0, \tag{4.7}$$

or

$$\left(\omega \begin{bmatrix} & & M_3 \\ & M_3 & M_2 \\ M_3 & M_2 & M_1 \end{bmatrix} + \begin{bmatrix} & -M_3 & \\ -M_3 & -M_2 & \\ & & M_0 \end{bmatrix}\right) \begin{bmatrix} \omega^2 \Phi \\ \omega\ \Phi \\ \Phi \end{bmatrix} = 0, \tag{4.8}$$

because they inherit the same simple structure of the right eigenvector as the companion forms (2.3) and (2.4). The main advantage of this structure is that a scaling of Φ of the original polynomial eigenvalue problem P automatically yields the same scaling for the linearized problem. In the complex symmetric case, left and right eigenvectors coincide except for complex conjugation. In fact, in [10] it is shown that the condition number of the linearization (4.7) or (4.8) is not much worse than the one of P for eigenvalues of modulus larger than one or smaller than one, respectively, if the coefficient matrices are balanced.

5. Numerical examples.

In this section we present some numerical examples illustrating the performance of our new method for the physical application. First we give some more information about the physical problem, which is necessary to state the equations. However, it is beyond the scope of this paper to explain the physical background in full detail. We refer to [14, 15, 33] for further reading.

5.1. Details on the physical problem.

The model derived in [15, 33] leads to the rational eigenvalue problem

$$\frac{\partial^2 \phi}{\partial \theta^2} + \frac{\big(\omega\,\sigma_1 + \sigma_2 + i\,\sigma_3\big)}{(\sigma_7 + \omega\,\sigma_8)} \big(\sigma_4 + \omega\,\sigma_5 + \omega^2\,\sigma_6\big)\phi = 0, \tag{5.1}$$

where the dependence of the coefficients σ_j on the wave number K is as follows:

$$\begin{aligned} &\sigma_1 = \beta + z\,\gamma_3\,\mu\,K^2, && \sigma_2 = -2\,\beta\,K, && \sigma_3 = z\,\gamma_3\,C\,K^2, \\ &\sigma_4 = 2\,\frac{\gamma_2}{\gamma_3}\omega_{\mathrm{B}}, && \sigma_5 = z\,K - 2\,z\,\gamma_2\,\omega_{\mathrm{B}}\,K, && \sigma_6 = z, \\ &\sigma_7 = -z\,\gamma_1^3\,K, && \sigma_8 = z\,\gamma_1^3\,\big(1 + 2\,z\,\gamma_3\,K^2\big). && \end{aligned} \tag{5.2}$$

Multiplication of (5.1) by its denominator leads to the polynomial eigenvalue problem

$$(\sigma_7 + \omega\,\sigma_8)\,\frac{\partial^2 \phi}{\partial \theta^2} + \big(\omega\,\sigma_1 + \sigma_2 + i\,\sigma_3\big)\big(\sigma_4 + \omega\,\sigma_5 + \omega^2\,\sigma_6\big)\phi = 0, \tag{5.3}$$

which is of the form (1.1) with $b_0 = b_1 = 0$. Note that the left eigenvector can be obtained from the eigenvalue and the right eigenvector by

$$\Phi_{\mathrm{l}} = \overline{\mathrm{diag}(\sigma_7 + \omega\sigma_8)^{-1}\Phi_{\mathrm{r}}}.$$

We would like to note that there are methods to solve rational or more general nonlinear eigenvalue problems directly; see, for instance, [16, 25, 26, 30, 35].

In the simplest case of a toroidal device with concentric flux surfaces we have $\gamma_1 = \gamma_3 = 1$ and $\gamma_2 = \cos(\theta)$. Averaging the remaining plasma parameters over the poloidal angle θ leads to a Mathieu equation [33]. There are more interesting cases where the plasma shape is elongated, i.e., where the poloidal cross section becomes an ellipse.

Depending on the model assumptions, a relation between K and the radial wave number z is to be taken into account, e.g.,

$$z = 2\ \max\left\{1, \frac{1}{2} + \left(\frac{\sigma_9}{K}\right)^2\right\} \tag{5.4}$$

for some constant σ_9. This means that the coefficients are continuous but not differentiable functions of the wave number K. It is shown that this does not affect the functionality of the algorithm presented. The parameters β, ω_{B}, and C are functions of certain plasma parameters such as temperature, density, etc., which also depend on the angle θ. Therefore, we have $\sigma_j = \sigma_j(\theta)$, $j = 1, \ldots, 9$. For more details on the model and the application we refer to [14, 15, 33].

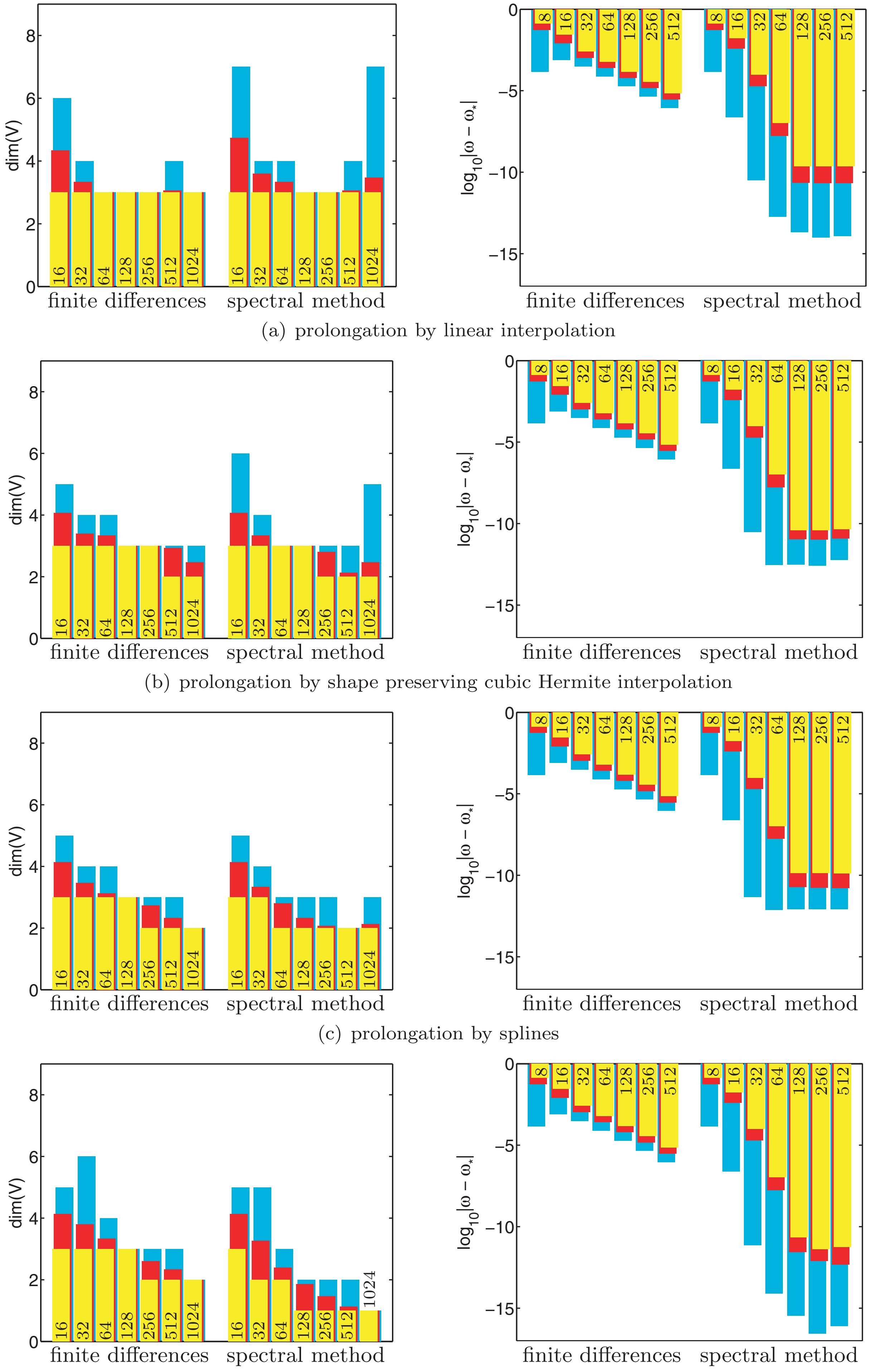
5.2. Comparison of spatial discretizations. In the first experiment we analyze the impact of the spatial discretization along with the interpolation that is applied for prolongating to the next finer grid. The discretization is either realized with finite differences of order four (corresponding to the choice $q = 2$ in section 2.1) or by using the spectral method as discussed in section 2.2. For the prolongation we use linear interpolation, cubic Hermite interpolation [5], cubic spline interpolation, or trigonometric interpolation. This gives a total of eight combinations.

We take 15 cubic eigenvalue equations of the form (1.1) that are obtained from different physical configurations which lead to different metric coefficients γ_j, $j = 1, 2, 3$, in (5.2), (5.3). For each of the combinations, we start on a grid G_0 with $N_0 = 8$ grid points, compute the 24 eigenpairs by the QZ algorithm, and select the eigenpair of physical interest. This eigenpair is improved on finer grids by the multilevel Jacobi–Davidson algorithm as described in section 4. The iteration is stopped if the forward error estimate is below a tolerance of 10^{-5}. The correction equation is always solved by using a finite difference approximation as discussed in section 3.2.

Statistics on the dimension $k = \dim V$ of the search space at the end of the Jacobi–Davidson cycle and the accuracy of the eigenvalue approximation with respect to the one on the finest grid are shown in Figure 5.1. Subfigures (a) to (d) are computed with different types of prolongation, and within each plot, each group of bars belongs to the specified differentiation method. The bars show the minimal, average, and maximal value of the final search space dimension (left pictures) or the accuracy of the eigenvalue approximation (right pictures). Here, ω_* denotes the solution on the finest grid.

The most obvious fact is a higher accuracy of the spectral method even on medium sized grids. On the grid of $N = 128$ points the spectral method almost reaches the final level of accuracy, which depends on the type of prolongation. The best final accuracy is achieved for trigonometric interpolation. If finite differences are used, the prolongation does not affect the accuracy of the intermediate eigenvalue approximations. However, the prolongation method has a strong impact on the number of Jacobi–Davidson cycles. The final search space is smallest if spline or trigonometric interpolation is used, but cubic and linear interpolation are quite good as well.

The situation is different for the spectral method, where the trigonometric interpolation saves one dimension on the finer grids. This is in accordance with the large gain of accuracy of the eigenvalue approximations and can be explained by the



(a) prolongation by linear interpolation

(b) prolongation by shape preserving cubic Hermite interpolation

(c) prolongation by splines

(d) prolongation by trigonometric interpolation

FIG. 5.1. *Dimension of search space and accuracy of eigenvalue approximation at several levels of the multilevel Jacobi–Davidson algorithm. More information is in the text.*

fact that the high frequencies in the eigenvector have minor impact. The accuracy of the eigenvalue approximation, which is much better than the desired tolerance, can be explained by the fact that the stopping criterion is matched at first iteration ($\dim V = 1$) already.

5.3. Effect of linearization and scaling. In the next experiment, we study the impact of different linearizations and scaling. We choose two test equations of the form (1.1) and refer to them as examples 1 and 2 in the following. The equations belong to a physical scenario where the plasma profiles are highly inhomogeneous and yield strongly varying coefficients a_j, c_j with $b_j = 0$. The desired eigenfunctions of both examples are shown in Figure 5.2(a). Obviously, at least locally, a high spatial resolution is required to approximate the derivatives of these functions accurately.

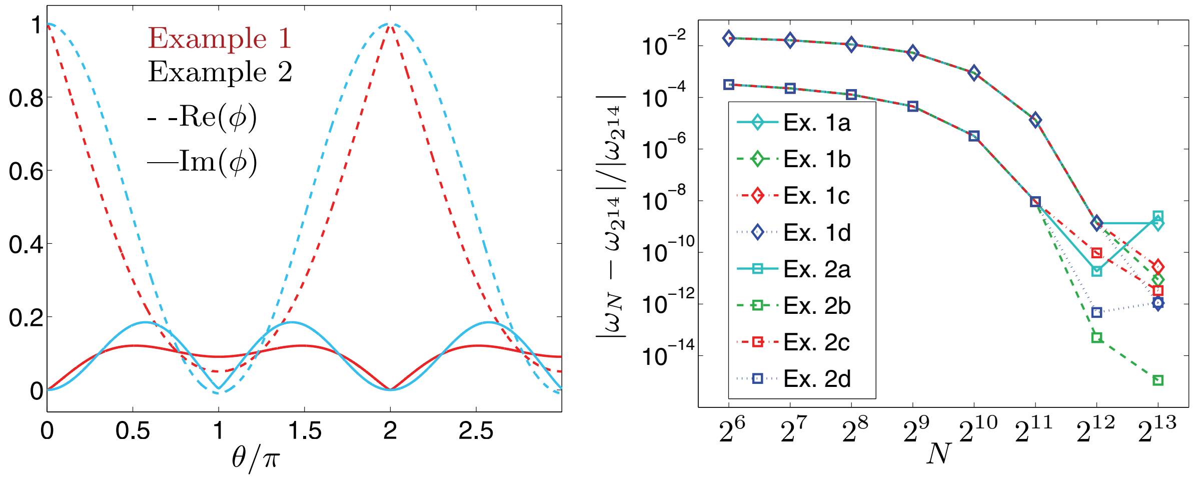
As in section 5.2, we approximate the derivatives with the spectral method, except for the correction equation, where finite differences of fourth order are applied. The initial approximation is computed by the QZ algorithm on a grid G_0 of $N_0 = 64$ grid points. This coarse grid approximation is improved by our multilevel Jacobi–Davidson method on subsequently finer grids G_ℓ of $N_\ell = 2^{\ell+6}$ grid points, $\ell = 1, \ldots, 8$. The prolongatation is done by trigonometric interpolation.

The Jacobi–Davidson cycle is iterated until the normwise forward error estimate (4.6) is below 10^{-7}. We stop the iteration if this accuracy is not achieved within a certain maximum number of iterations. For illustration, we set the limit to 64 iterations, but, in practice, one should use a smaller number, since the algorithm is expected to converge locally quadratically. The experiments below show that a proper scaling and linearization lead to convergence within 16 steps at most.

In Figure 5.2(b), the accuracy of the eigenvalue on intermediate grids is illustrated. For both examples 1 and 2 we consider the following combinations of two linearizations with the nearly optimal scaling strategy of section 4.5:

	without scaling	with scaling
linearization (2.3)	(i)	(ii)
linearization (4.7)	(iii)	(iv)

All examples are calculated with one-sided Rayleigh quotients. On the coarsest grid,



(a) Eigenfunctions

(b) Precision of eigenvalue approximation

FIG. 5.2. *(a) Real and imaginary part of the eigenfunction of Examples* 1 *and* 2*. (b) Accuracy of eigenvalue with respect to the level of resolution given by the number* N *of grid points. The tolerance of forward error is set to* 10^{-7}.

Table 5.1

Average computational time t of the QZ algorithm including linearization for a cubic eigenvalue equation on several resolution levels indicated by the number N of grid points.

N	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}
t/min	0.00007	0.00038	0.0025	0.019	0.17	1.7	13	103

Table 5.2

Average computational time t of one Jacobi–Davidson cycle in the multilevel Jacobi–Davidson algorithm. We estimated the cost for the final cycle as half because one can save the solution of the correction equation for it.

N	2^7	2^8	2^9	2^{10}	2^{11}	2^{12}	2^{13}	2^{14}
t/sec	0.01	0.02	0.05	0.07	0.10	0.18	0.27	0.32

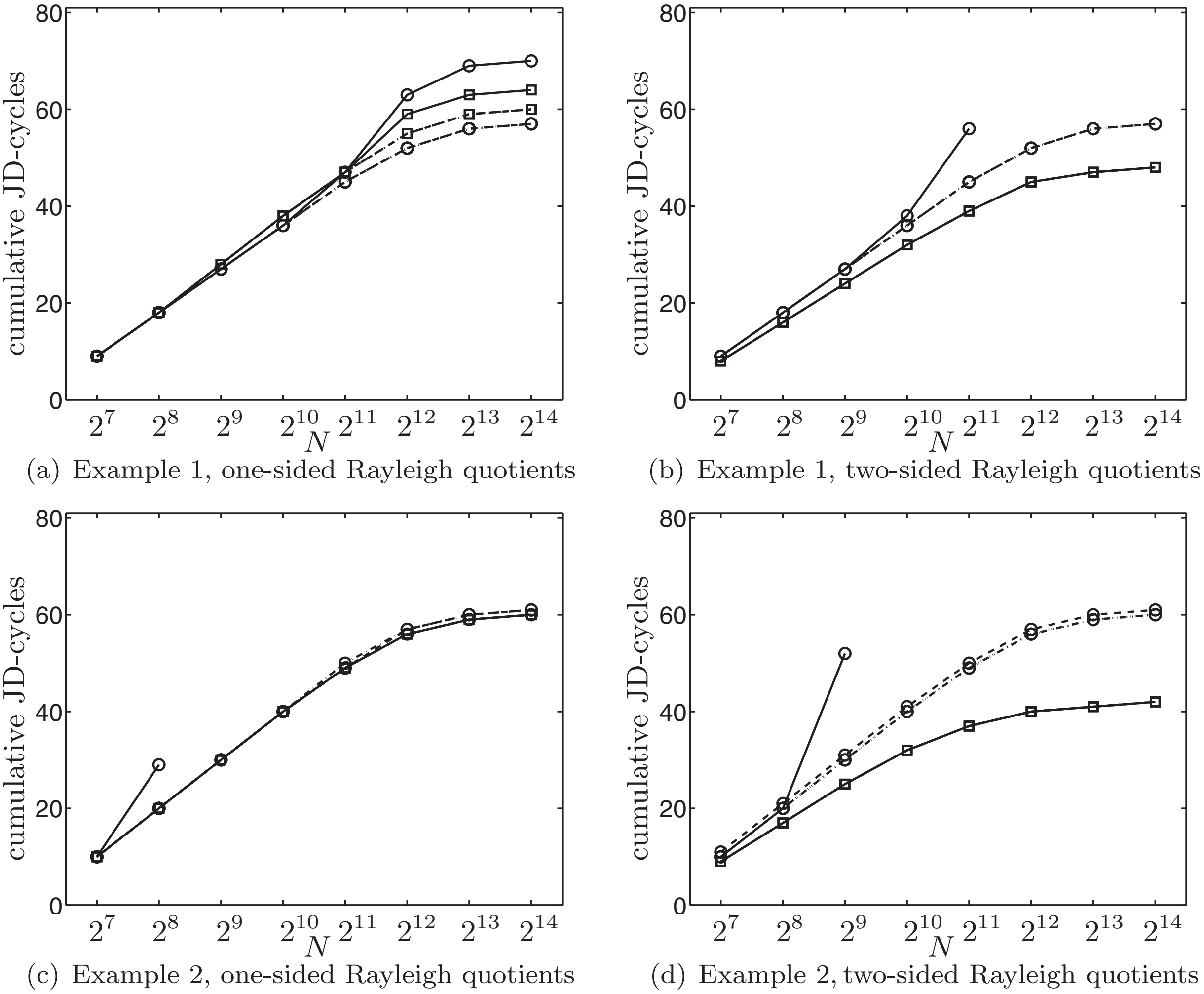
the relative accuracy $|\omega_N - \omega_{2^{14}}|/|\omega_{2^{14}}|$ is 10^{-1} and 10^{-3}, respectively, and it decreases marginally until 2^9 grid points, where superlinear convergence starts. The accuracy improves monotonically except for the cases where a Jacobi–Davidson cycle does not converge until the dimension of the search space reaches the maximum number. For grids with at least 2^{12} points, the accuracy is much better than the desired accuracy of 10^{-7}. This is due to the fact that here the Jacobi–Davidson procedure is in the regime of quadratic convergence.

In additional numerical experiments we observed that both examples can be calculated by starting on a very coarse grid of eight grid points only. However, it is more reliable to start with a finer grid. 64 grid points turned out to be a good compromise between computational time and reliability to find an approximation of the desired eigenpair. Table 5.1 shows the average computational time of the QZ algorithm to compute the eigenvalues of a complex cubic eigenvalue equation. To be more precise, the time measurements include the linearization and the solution of the linearized system by the QZ algorithm (but not the initialization of the matrices). The four types of linearizations (2.3), (2.4), (4.7), and (4.8) have been tested without noteworthy difference in computational time.

For comparison, the average time of one Jacobi–Davidson cycle is given in Table 5.2. In the final iteration, the projected system is solved, but since the correction equation does not need to be solved, we estimated the cost for the final iteration as half the cost of the remaining iterations. Doubling the number of grid points increases the computational time of the QZ algorithm by about a factor of ten, while the computational time of the Jacobi–Davidson cycle increases by a factor of about two. Therefore, the aim is to reduce the number of Jacobi–Davidson cycles in total and in particular on fine grids.

The number of inner cycles that the multilevel Jacobi–Davidson approach takes to achieve the desired accuracy of 10^{-7} is shown in Figure 5.3. On the coarse grids, the final dimension of the search space is $\dim V = 9 \pm 1$. Without scaling, in example 2 (circles in Figures 5.3(c) and 5.3(d)) the companion linearization (2.3) does not reach the desired accuracy within 64 Jacobi–Davidson cycles, and this leads to the large slopes of the cut-off curves. In general the slopes of the curves and so the numbers of Jacobi–Davidson cycles is almost constant until $N = 2^{11}$. On higher resolutions the slopes reduce to one. This is in accordance with the fact that the final accuracy of the eigenvalue is almost reached (cf. Figure 5.2(a)).

Without scaling, the results with linearization (2.3) are the worst. This is in accordance with the theory in [10], where this linearizations is only recommended for eigenvalues with modulus greater than one. However, the eigenvalues in our example



(a) Example 1, one-sided Rayleigh quotients

(b) Example 1, two-sided Rayleigh quotients

(c) Example 2, one-sided Rayleigh quotients

(d) Example 2, two-sided Rayleigh quotients

FIG. 5.3. *Cumulative number of Jacobi–Davidson cycles in the multilevel Jacobi–Davidson Algorithm* 2, *where the linearizations are* (2.3) *(solid),* (2.4) *(dashed),* (4.7) *(dash-dotted), and* (4.8) *(dotted). The calculation is done without scaling (circles) and with the scaling described in section* 4.5 *(squares).*

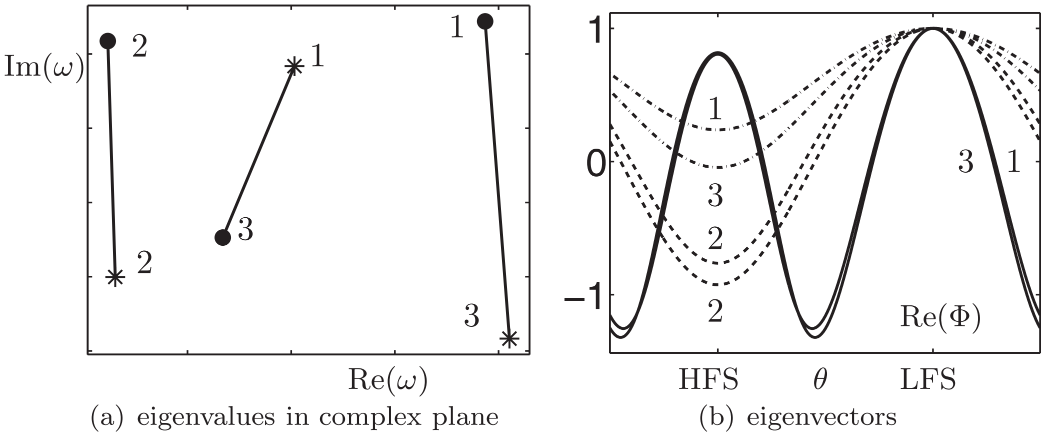
are of modulus less than one. The curves of the other three linearizations are very close. If we compare the computations with scaling (squares) and without scaling (circles), we obviously see a reduction of the dimensions of the search spaces if scaling is applied. This reduction is significant for two-sided Rayleigh quotients where the differences due to the choice of the linearization becomes negligible.

The total computation time of the multilevel Jacobi–Davidson algorithm is about five seconds for the scaled examples and thus significantly smaller than the QZ algorithm, which already takes almost two hours on a grid of only 2^{10} points. To summarize, the most efficient variant of the multilevel Jacobi–Davidson algorithm is scaling combined with the two-sided Rayleigh quotients.

5.4. Parameter dependent problems. So far, we fixed all parameters in the eigenvalue problems (1.1) and (5.3). However, as discussed in the introduction, we would like to determine the wave number K which solves the following optimization problem:

$$K_{\max} = \underset{K}{\operatorname{argmax}} \max_{\omega \text{ eigenvalue of } P^{[K]}} \operatorname{Im}(\omega).$$

Here we write $P^{[K]}$ to emphasize the dependence of the coefficients of the polynomial on the wave number K. The eigenvalues of $P^{[K]}$ are the solutions of $\det(P^{[K]}(\omega)) = 0$;



$$\mathrm{sim}(\omega_\star, \omega_\bullet) = \begin{bmatrix} 0.99 & 0.99 & \boxed{1.00} \\ 0.97 & \boxed{1.00} & 1.00 \\ \boxed{1.00} & 0.96 & 0.98 \end{bmatrix}, \quad \mathrm{sim}(\Phi_\star, \Phi_\bullet) = \begin{bmatrix} 0.47 & 0.87 & \boxed{0.99} \\ 0.28 & \boxed{0.99} & 0.87 \\ \boxed{1.00} & 0.34 & 0.45 \end{bmatrix},$$

$$S = \begin{bmatrix} 0.47 & 0.86 & \boxed{0.99} \\ 0.27 & \boxed{0.99} & 0.87 \\ \boxed{1.00} & 0.33 & 0.44 \end{bmatrix}$$

FIG. 5.4. *Assignment of eigenpairs into modes. At two wave numbers K_1 (stars) and K_2 (circles) the three eigenpairs with maximal* Im(ω) *are computed. The eigenvalues are plotted in the complex plane at the left side, and the phase normalized real part of each eigenfunction is plotted on the right. In the similarity matrices* (4.1) *and* (4.2) *the assigned components are indicated by boxes.*

hence the eigenvalues are continuous functions of K. We thus write $(\omega(K), \Phi(K))$ and call this pair a mode. For each mode, $\omega(K)$ is a continuous curve in the complex plane.

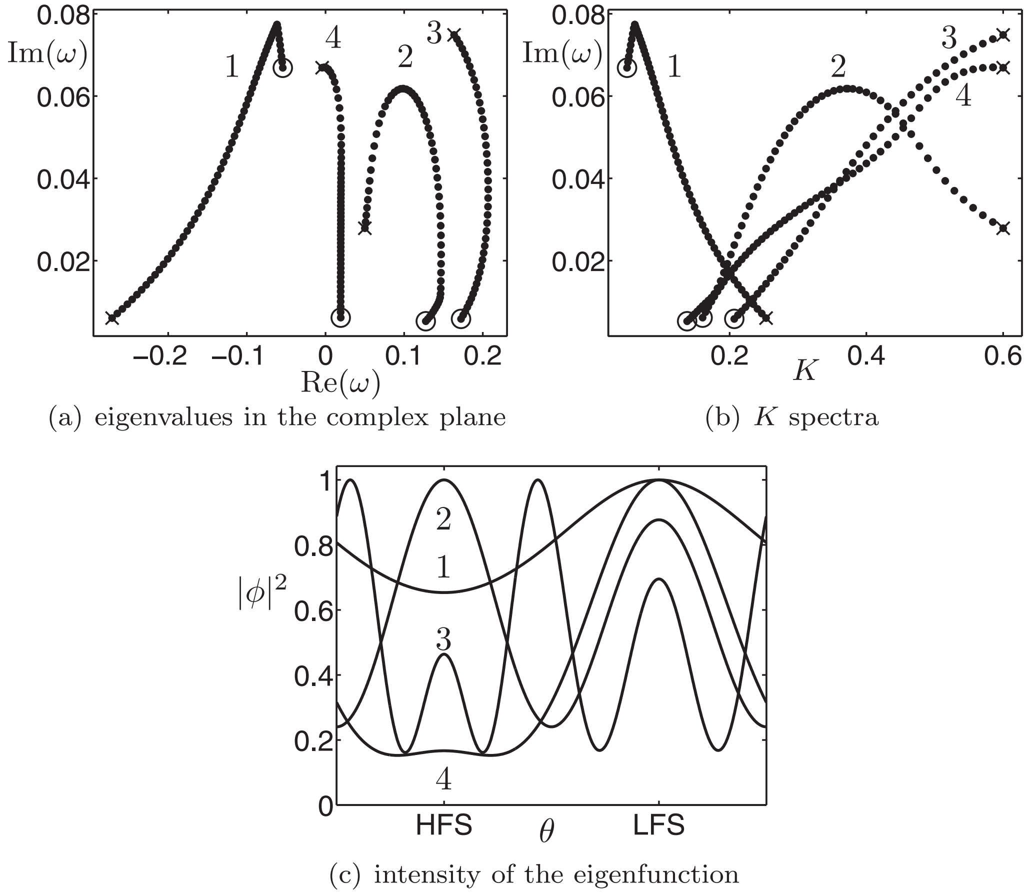
For the solution of the optimization problem we compute the eigenpairs for a discrete set of wave numbers K_j. For each value K_j we obtain a set of eigenpairs, and we have to determine which eigenpairs belong to the same mode. Assume that the sequence $\{K_j\}$ is monotonically increasing. Given a subset $(\omega_l(K_j), \Phi_l(K_j))$ of eigenpairs corresponding to K_j, we have to assign each of them to an eigenpair $(\omega_l(K_{j+1}), \Phi_l(K_{j+1}))$ corresponding to K_{j+1}. If the subset of eigenpairs is a proper subset of all the eigenpairs, there may exist eigenpairs without a valid partner.

We assume that K_j and K_{j+1} are sufficiently close so that eigenpairs of the same mode are closer to each other than to any other eigenpair. In order to measure the distance we use the similarity measure (4.3). For simplicity, we set $j = 1$. We define the similarity matrix $S = (s_{k,l})$ by

$$s_{k,l} := \mathrm{sim}((\omega_k(K_1), \Phi_k(K_1)), (\omega_l(K_2), \Phi_l(K_2))).$$

By definition of the similarity measure we have $0 \leq s_{k,l} \leq 1$, where 1 indicates that the modes coincide. A value of 0 is attained for orthogonal eigenvectors. To assign the mode we choose a threshold $\tau \in [0, 1]$. If the largest entry $s_{m,n}$ of S satisfies $s_{m,n} \geq \tau$, we assign the eigenpair $(\omega_m(K_1), \Phi_m(K_1))$ to $(\omega_n(K_2), \Phi_n(K_2))$. Then we remove the mth row and the nth column from S and repeat the procedure until all the eigenpairs at one wave number are assigned or the largest element in S is below the threshold τ.

We illustrate this procedure in Figure 5.4. The asterisks represents K_1 and the full circles represent K_2. For each wave number we compute the three most dominant eigenpairs. The eigenvalues are displayed in the complex plane in the left picture,



(a) eigenvalues in the complex plane

(b) K spectra

(c) intensity of the eigenfunction

FIG. 5.5. *Tracing eigenpairs as a function of the wave number K.*

where the numbering is by decreasing value of $\mathrm{Im}(\omega)$. The real part of the phase normalized eigenvector Φ is displayed in the right graph. The eigenvalue and eigenvector similarity matrices are shown below the plots. Their elementwise product yields the similarity matrix S. The procedure above carries out the indicated assignment.

For our next experiment, we assigned the modes for 99 wave numbers. The results are presented in Figure 5.5. Figure 5.5(a) shows four eigenvalue curves in the complex plane. Figure 5.5(b) shows the growth rate ($\mathrm{Im}(\omega)$) as a function of the wave number K. The coefficient functions in (1.1) are continuous, but the derivatives of these functions emerge a jump at $K \approx 0.1$ due to (5.4). This results in the kink of the first eigenvalue curve at $-0.07 + 0.078\mathrm{i}$. In Figure 5.5(c) the intensity $|\Phi|^2$ of the four eigenfunctions are plotted for the wave number $K_{\max}$ where the growth rate of the mode is maximal.

The huge number of wave numbers has been taken for visualization only. In general, 10 to 20 wave numbers are sufficient to assign the modes correctly. From these results, the maximum is found by a prediction correction strategy via polynomial interpolation.

A more challenging situation occurs if eigenvalues cross each other. To illustrate this phenomenon, we consider the cubic polynomial

$$P^{[K]}(\omega) := \begin{pmatrix} (\omega - \mathrm{i}K)(\omega^2 - 1) & 0 \\ 0 & (\omega - \mathrm{i}K^2)(\omega^2 - 4) \end{pmatrix}$$

with eigenvalues $\{\mathrm{i}K, \mathrm{i}K^2, \pm 1, \pm 2\}$. At $K = 1$ the curves of $\mathrm{i}K$ and $\mathrm{i}K^2$ cross each other, but the eigenvectors are orthogonal. Since our similarity measure (4.3) involves eigenvalues and eigenvectors, it is able to distinguish these eigenpairs.

6. Conclusion and outlook.

In this paper we presented a multilevel Jacobi–Davidson algorithm for parameter dependent polynomial PDE eigenvalue problems. The multilevel approach provides a good initial search space for the Jacobi–Davidson method, and it also helps to scale the problem in a nearly optimal way. In our application from plasma physics, this algorithm reduced the computational time from hours (QZ) to seconds.

The fast solution of the eigenvalue problem enabled us to compute plasma parameters self-consistently [14, 15] and to analyze the impact of different geometries on the energy losses. Since such simulations require the solution of 10^3 to 10^5 eigenvalue equations, an efficient numerical solver is indispensable.

Here, we presented the method for cubic eigenvalues problems only, but since the algorithm is independent of the degree of the matrix polynomial, it can be generalized in an obvious way. The essential requirement to successfully apply the multilevel Jacobi–Davidson algorithm is that the desired eigenfunction is spatially smooth enough to be well approximated on a sequence of nested grids.

We would like to note that the original problem is given as a rational eigenvalue problem. Since the application we presented here could be rewritten as a well conditioned polynomial eigenvalue problem, we presented the multilevel approach for such problems only. However, for other geometries, it turns out that the rational formulation is better conditioned than the polynomial one. Thus we intend to generalize our techniques to the solution of rational problems [3, 30].

Acknowledgment. We gratefully acknowledge the intensive collaboration with Mikhail Tokar and Dirk Reiser from the Institute for Energy Research at Research Center Jülich. We also thank Timo Betcke for fruitful discussions and the anonymous referees for constructive criticism.

REFERENCES

[1] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.

[2] T. Betcke, *Optimal scaling of generalized and polynomial eigenvalue problems*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1320–1338.

[3] T. Betcke and H. Voss, *A Jacobi-Davidson-type projection method for nonlinear eigenvalue problems*, Future Gener. Comput. Syst., 20 (2004), pp. 363–372.

[4] W. J. Beyn and V. Thümmler, *Continuation of invariant subspaces for parameterized quadratic eigenvalue problems*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 1361–1381.

[5] F. N. Fritsch and R. E. Carlson, *Monotone piecewise cubic interpolation*, SIAM J. Numer. Anal., 17 (1980), pp. 238–246.

[6] G. Golub and C. van Loan, *Matrix Computations*, Johns Hopkins, University Press, Baltimore, MD, 1996.

[7] V. Heuveline and C. Bertsch, *On multigrid methods for the eigenvalue computation of nonselfadjoint elliptic operators*, East-West J. Numer. Math., 8 (2000), pp. 275–297.

[8] N. J. Higham, R.-C. Li, and F. Tisseur, *Backward error of polynomial eigenproblems solved by linearization*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 1218–1241.

[9] 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.

[10] 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.

[11] M. E. Hochstenbach and G. L. G. Sleijpen, *Two-sided and alternating Jacobi-Davidson*, Linear Algebra Appl., 358 (2003), pp. 145–172.

[12] D. Kressner, *Numerical Methods for General and Structured Eigenvalue Problems*, Springer-Verlag, Berlin, 2005.

[13] D. Kressner, *A block Newton method for nonlinear eigenvalue problems*, Numer. Math., 114 (2009), pp. 355–372.

[14] D. Löchel, *Numerical Methods for Eigenvalue Problems in the Description of Drift Instabilities in the Plasma Edge*, Ph.D. thesis, Mathematisches Institut, Heinrich-Heine Universität Düsseldorf, Lehrstuhl für Angewandte Mathematik, Düsseldorf, Germany, 2009.

[15] D. Löchel, M. Z. Tokar, M. Hochbruck, and D. Reiser, *Effect of poloidal inhomogeneity in plasma parameters on edge anomalous transport*, Phys. Plasmas, 16 (2009), article 044508.

[16] M. Markiewicz and H. Voss, *A local restart procedure for iterative projection methods for nonlinear symmetric eigenproblems*, in Proceedings of Algoritmy 2005, 17th Conference on Scientific Computing, Vysoké Tatry–Podbanské, Slovakia, 2005, pp. 212–221.

[17] K. Meerbergen, *Locking and restarting quadratic eigenvalue solvers*, SIAM J. Sci. Comput., 22 (2001), pp. 1814–1839.

[18] J. R. Myra, D. A. D'Ippolito, and J. P. Goedbloed, *Generalized ballooning and sheath instabilities in the scrape-off layer of divertor tokamaks*, Phys. Plasmas, 4 (1997), pp. 1330–1341.

[19] J. R. Myra, D. A. D'Ippolito, X. Q. Xu, and R. H. Cohen, *Resistive modes in the edge and scrape-off layer of diverted tokamaks*, Phys. Plasmas, 7 (2000), pp. 4622–4631.

[20] J. R. Myra, D. A. D'Ippolito, X. Q. Xu, and R. H. Cohen, *Resistive x-point modes in tokamak boundary plasmas*, Phys. Plasmas, 7 (2000), pp. 2290–2293.

[21] P. Nickel and R. Dyczij-Edlinger, *An improved Jacobi-Davidson method with multi-level startup procedure*, Transactions Magnetics, 45 (2009), pp. 1372–1375.

[22] A. M. Ostrowski, *On the convergence of the Rayleigh quotient iteration for the computation of the characteristic roots and vectors.* III, Arch. Ration. Mech. Anal., 1 (1957), pp. 325–340.

[23] A. M. Ostrowski, *On the convergence of the Rayleigh quotient iteration for the computation of characteristic roots and vectors*, Arch. Ration. Mech. Anal., 3 (1959), pp. 472–481.

[24] B. Parlett, *The Rayleigh quotient iteration and some generalizations*, Math. Comp., 28 (1974), pp. 679–693.

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

[26] K. Schreiber, *Nonlinear Eigenvalue Problems: Newton-type Methods and Nonlinear Rayleigh Functionals*, Ph.D. thesis, Fakultät II–Mathematik und Naturwissenschaften, TU Berlin, Berlin, 2008.

[27] G. Sleijpen, A. Booten, D. Fokkema, and H. van der Vorst, *Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems*, BIT, 36 (1996), pp. 595–633.

[28] G. L. G. Sleijpen, and H. van der Vorst, *A Jacobi–Davidson iteration method for linear eigenvalue problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 401–425.

[29] G. Sleijpen, H. van der Vorst, and E. Meijerink, *Efficient expansion of subspaces in the Jacobi-Davidson method for standard and generalized eigenproblems*, Electron. Trans. Numer. Anal., 7 (1998), pp. 75–89.

[30] Y. Su and Z. Bai, *Solving Rational Eigenvalue Problems via Linearization*, Technical report UCD/CS-2008-13, School of Mathematical Sciences, Fudan University, Shanghai, China, Department of Computer Science and Mathematics, University of California, Davis, CA, 2008.

[31] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, Addison-Wesley, Reading, MA, 2006.

[32] F. Tisseur, *Backward error and condition of polynomial eigenvalue problems*, Linear Algebra Appl., 309 (2000), pp. 339–361.

[33] M. Z. Tokar, F. Kelly, and X. Loozen, *Role of thermal instabilities and anomalous transport in threshold of detachment and multifacetted asymmetric radiation from the edge (MARFE)*, Phys. Plasmas, 12 (2005), article 052510.

[34] L. Trefethen, *Spectral Methods in MATLAB*, SIAM, Philadelphia, 2002.

[35] H. Voss, *A new justification of the Jacobi-Davidson method for large eigenproblems*, Linear Algebra Appl., 424 (2007), pp. 448–455.

[36] J. Weiland, *Analytical eigenvalue solution for η_i modes of general modewidth*, Phys. Plasmas, 11 (2004), pp. 3238–3241.