SIAM J. MATRIX ANAL. APPL.
Vol. 42, No. 2, pp. 775–799

NONLINEARIZING TWO-PARAMETER EIGENVALUE PROBLEMS*

EMIL RINGH† AND ELIAS JARLEBRING†

Abstract. We investigate a technique to transform a linear two-parameter eigenvalue problem into a nonlinear eigenvalue problem (NEP). The transformation stems from an elimination of one of the equations in the two-parameter eigenvalue problem, by considering it as a (standard) generalized eigenvalue problem. We characterize the equivalence between the original and the nonlinearized problem theoretically and show how to use the transformation computationally. Special cases of the transformation can be interpreted as a reversed companion linearization for polynomial eigenvalue problems, as well as a reversed (less known) linearization technique for certain algebraic eigenvalue problems with square-root terms. Moreover, by exploiting the structure of the NEP we present algorithm specializations for NEP methods, although the technique also allows general solution methods for NEPs to be directly applied. The nonlinearization is illustrated in examples and simulations, with focus on problems where the eliminated equation is of much smaller size than the other two-parameter eigenvalue equation. This situation arises naturally in domain decomposition techniques. A general error analysis is also carried out under the assumption that a backward stable eigensolver is used to solve the eliminated problem, leading to the conclusion that the error is benign in this situation.

Key words. two-parameter eigenvalue problem, nonlinear eigenvalue problem, multiparameter eigenvalue problem, iterative algorithms, implicit function theorem

AMS subject classifications. 65F15, 15A18, 47J10, 65H17, 15A22, 15A69

DOI. 10.1137/19M1274316

1. Introduction.

This paper concerns the *two-parameter eigenvalue problem*: Determine nontrivial quadruplets $(\lambda, x, \mu, y) \in \mathbb{C} \times \mathbb{C}^n \times \mathbb{C} \times \mathbb{C}^m$ such that

$$0 = A_1 x + \lambda A_2 x + \mu A_3 x, \tag{1.1a}$$

$$0 = B_1 y + \lambda B_2 y + \mu B_3 y, \tag{1.1b}$$

where $A_1, A_2, A_3 \in \mathbb{C}^{n\times n}$ and $B_1, B_2, B_3 \in \mathbb{C}^{m\times m}$. More specifically, by nontrivial we mean that the eigenvectors should be nonzero, i.e., $y \neq 0$ and $x \neq 0$. We denote the corresponding functions $A(\lambda, \mu) := A_1 + \lambda A_2 + \mu A_3$ and $B(\lambda, \mu) := B_1 + \lambda B_2 + \mu B_3$. This problem has been extensively studied in the literature; see, e.g., the fundamental work of Atkinson [2] and the summary of recent developments below. We assume that $m \ll n$ and that A_1, A_2, and A_3 are large and sparse matrices, although several theoretical contributions of this paper are valid without this assumption.

The main idea of our approach can be described as follows. We view (1.1b) as a parameterized generalized linear eigenvalue problem, where λ is the parameter. Due to perturbation theory for eigenvalue problems, there is a family of continuous functions $\{g_i(\lambda)\}$ defined by the eigenvalues of (1.1b), where μ is the eigenvalue, of a *generalized eigenvalue problem* (GEP). More formally, for a fixed value of $\lambda \in \mathbb{C}$ the functions $g_i(\lambda) \in \mathbb{C}$ and $\phi_i(\lambda) \in \mathbb{C}^m$ can be defined as the solution to

$$0 = (B_1 + \lambda B_2 + g_i(\lambda) B_3)\phi_i(\lambda), \tag{1.2a}$$

$$1 = c^T \phi_i(\lambda) \tag{1.2b}$$

*Received by the editors July 11, 2019; accepted for publication (in revised form) by H. Fassbender February 9, 2021; published electronically June 8, 2021.
https://doi.org/10.1137/19M1274316

†Department of Mathematics, KTH Royal Institute of Technology, SeRC Swedish e-science research center, Lindstedtsvägen 25, SE-100 44 Stockholm, Sweden (eringh@kth.se, eliasj@kth.se).

for a given vector $c \in \mathbb{C}^m$. We explicitly introduced the normalization condition (1.2b) to uniquely define a corresponding eigenvector. The condition (1.2b) is not a restriction of generality except for the rare situation that the eigenvector is orthogonal to c. We prefer this condition over the standard Euclidean normalization since the right-hand side of (1.2b) is an analytic function.

By insertion of $\mu = g_i(\lambda)$ into (1.1a), we see that a solution to (1.1) will satisfy

$$M(\lambda)x = (A_1 + \lambda A_2 + g_i(\lambda)A_3)x = 0. \tag{1.3}$$

Note that we have now eliminated μ and (1.1b), at the cost of the introduction of a nonlinear function into the eigenvalue problem. The problem $M(\lambda)x = 0$ is called a *nonlinear eigenvalue problem* (NEP). In our setting it is rather a family of NEPs, since we have a different nonlinearity for each function $g_1, \ldots, g_m$. The study of NEPs is a mature field within numerical linear algebra, and there are considerable theoretical results, as well as algorithms and software for NEPs which aim to find a selection of solutions. Note that the NEP solvers in general only compute a subset of the eigenvalues, and therefore our approach is mainly for situations where particular (λ, μ)-values are of interest, e.g., close to a target.

We provide a theoretical characterization of the elimination procedure in section 2. The characterization shows that the functions are locally analytic (and not necessarily entire functions) everywhere except for certain points, which are explicitly described. Section 3 contains new methods for (1.1) derived from NEP methods designed for problems with local analyticity. Analysis of the conditioning of the structured perturbations corresponding to the elimination is provided in section 4. We provide software for the simulations, both for MATLAB and for Julia [6]. The Julia software is implemented using the data structures of the NEP-PACK software package [20], including adaption of theory for how to compute derivatives and projections. This provides new ways to solve (1.1) using the large number of NEP solvers available in NEP-PACK. Some contributions are also converse, i.e., we provide insight to NEPs based on the equivalence with two-parameter eigenvalue problems. For instance, in sections 2.2–2.3 we show how to transform certain NEPs with square-root nonlinearities to two-parameter eigenvalue problems. This in turn (using the operator determinants described below) allows us to transform the problem to a standard GEP, similar to companion linearization techniques for polynomial and rational eigenvalue problems.

We now summarize the NEP results relevant for our approach. For a broad overview see the summary papers [38, 30, 50, 10], as well as the benchmark collection [4] and software packages with NEP solvers [37, 12, 13, 20]. There are considerable theoretical works available for the NEP, in particular for polynomial eigenvalue problems. Techniques to transform polynomial NEPs to standard eigenvalue problems (known as linearization) have been completely characterized in a number of works, e.g., [27, 28] and [33]. We relate our approach to this type of linearization in section 2.2. In our derivation, we make explicit use of the implicit function theorem applied to the NEP. This has been done in the context of sensitivity analysis, leading to eigenvector-free formulas for conditioning [1]. There are a number of algorithms available for NEPs, of which many seem to be applicable to (1.3). More specifically, we characterize the specialization of residual inverse iteration [34], which forms the basis of more recent methods such as the nonlinear Arnoldi method [49]. We also show how the infinite Arnoldi method [23] can be adapted to (1.3).

In section 5.2 we illustrate how two-parameter eigenvalue problems of this type can arise by the separation of domains of a boundary value problem (BVP). The domains are decoupled in a way that the discretization leads to a two-parameter

eigenvalue problem. In this context, the elimination corresponds to an elimination of one of the domains. The elimination of an outer domain, in a way that directly leads to NEPs, by introduction of artificial boundary conditions is the origin of several standard NEPs in the literature, e.g., [44] and the electromagnetic cavity model in [48].

Relevant results for two-parameter eigenvalue problems can be summarized as follows. Many results for two-parameter eigenvalue problems are phrased in the more general setting of *multiparameter eigenvalue problems*. There are a number of recent efficient algorithms available, e.g., based on the Jacobi–Davidson approach [15, 17], including subspace methods in [16]. A number of generalizations of inverse iteration are derived in [36]. Our approach is based on an eigenvalue parameterization viewpoint. Eigenvalue parameterization and continuation techniques (but with an additional parameter) have been studied, e.g., in [35].

One of the most fundamental properties of two-parameter eigenvalue problems is the fact that solutions are given by the solution to a larger linear (generalized) eigenvalue problem. This is also often used in the numerical algorithms mentioned above and to our knowledge was first proposed as a numerical method in [40]. More precisely, we associate with (1.1) the *operator determinants*

$$\Delta_0 = B_2 \otimes A_3 - B_3 \otimes A_2, \tag{1.4}$$

$$\Delta_1 = B_3 \otimes A_1 - B_1 \otimes A_3, \tag{1.5}$$

$$\Delta_2 = B_1 \otimes A_2 - B_2 \otimes A_1, \tag{1.6}$$

where $\otimes$ denotes the Kronecker product. The solutions to (1.1) are (under certain assumptions) equivalent to the solutions to the two GEPs

$$\Delta_1 z = \lambda \Delta_0 z, \tag{1.7a}$$

$$\Delta_2 z = \mu \Delta_0 z, \tag{1.7b}$$

where $z = y \otimes x$. In practice, the application of a general purpose eigenvalue solver on one of the GEPs in (1.7) yields an accurate solution for small systems. The Sylvester-like structure of the operator determinants is exploited in [29] with applications in, e.g., detection of a Hopf bifurcation. The equivalence between (1.7) and (1.1) holds under nonsingularity assumption; in particular the problem is singular if A_3 and B_3 both are singular, A_2 and B_2 both are singular, A_2 and A_3 have intersecting null-spaces, or B_2 and B_3 have intersecting null-spaces. See [2] for a precise characterization and [24, 18] for more recent formulations.

The following matrix is often used in theory for eigenvalue multiplicity and eigenvalue conditioning and will be needed throughout the paper. We denote

$$C_0 := \begin{bmatrix} v^H A_2 x & v^H A_3 x \\ w^H B_2 y & w^H B_3 y \end{bmatrix}, \tag{1.8}$$

where v and w are left eigenvectors associated with (1.1a) and (1.1b), respectively. In particular, for an (algebraically) simple eigenvalue of the two-parameter eigenvalue problem (1.1), the matrix C_0 is nonsingular; see [24, Lemma 3], [15, Lemma 1.1], and [18, Lemma 1]. For a simple eigenvalue, the normwise condition number for the two-parameter eigenvalue problem is expressed as a special induced matrix norm of C_0^{-1}; see [18, section 4].

2. Nonlinearization.

2.1. Existence and equivalence. The elimination of the B-equation (1.1b) in the two-parameter eigenvalue problem can be explicitly characterized as we describe next. Note that when λ is viewed as a parameter, the second equation in the two-parameter eigenvalue problem is a GEP corresponding to the pencil $(-(B_1+\lambda B_2), B_3)$:

$$-(B_1+\lambda B_2)y = \mu B_3 y. \tag{2.1}$$

In the algorithms section, we will use GEP eigensolvers to compute μ. In order to describe when this GEP leads to an analytic well-defined parameterization we introduce the normalization $c^T y = 1$ for theoretical purposes.

The idea is based on viewing the GEP (2.1) and the normalization condition as a set of nonlinear equations in the variables y, λ, and μ. Conditions on the existence of a parameterization are in our first result expressed in terms of the partial Jacobian, with respect to the variables y and μ, of this nonlinear function. The Jacobian is

$$J(\lambda,\mu,y) := \begin{bmatrix} B(\lambda,\mu) & B_3 y \\ c^T & 0 \end{bmatrix}. \tag{2.2}$$

In the theorem that directly follows this lemma, we show how the singularity of the Jacobian is directly related to the multiplicity of the eigenvalue of the GEP (2.1).

LEMMA 2.1 (existence of implicit functions). *Let $\lambda \in \mathbb{C}$ be given. Suppose the pencil associated with the GEP* (2.1) *is regular. Let $(\mu_i, y_i) \in \mathbb{C}\times\mathbb{C}^m$ be an eigenpair of the GEP normalized such that $c^T y_i = 1$. Moreover, assume that $J(\lambda,\mu_i,y_i)$ as given by* (2.2) *is nonsingular. Then, there exist a domain $\Omega_i \subset \mathbb{C}$ and functions analytic in this domain $g_i : \Omega_i \to \mathbb{C}$ and $\phi_i : \Omega_i \to \mathbb{C}^m$ such that*

$$-(B_1 + sB_2)\phi_i(s) = g_i(s) B_3 \phi_i(s)$$

for all $s \in \Omega_i$, where Ω_i is a neighborhood of λ and $g_i(\lambda) = \mu_i$, $\phi_i(\lambda) = y_i$.

Proof. Consider the analytic function $f : \mathbb{C}^{m+2} \to \mathbb{C}^{m+1}$ given by

$$f(\lambda,\mu,y) := \begin{bmatrix} B(\lambda,\mu)y \\ c^T y - 1 \end{bmatrix}. \tag{2.3}$$

Then, as noted above, $J = \partial f/\partial(y,\mu)$. Since $f(\lambda,\mu_i,y_i) = 0$ and $J(\lambda,\mu_i,y_i)$ is nonsingular, the result follows from the complex implicit function theorem [8, Theorem I.7.6]. ∎

THEOREM 2.2 (J-singularity). *Let $\lambda \in \mathbb{C}$ be given. Assume that the pencil associated with the GEP* (2.1) *is regular. Let $(\mu_i, y_i) \in \mathbb{C}\times\mathbb{C}^m$ be an eigenpair of the GEP normalized such that $c^T y_i = 1$. Then $J(\lambda,\mu_i,y_i)$ defined in* (2.2) *is singular if and only if μ_i is a nonsimple eigenvalue of the GEP.*

Proof. We start by proving that $J(\lambda,\mu_i,y_i)$ being singular implies that μ_i is a nonsimple eigenvalue. Assume that $J(\lambda,\mu_i,y_i)$ is singular. Then there exists a nontrivial vector $\begin{bmatrix}\xi^T & \alpha\end{bmatrix}^T \in \mathbb{C}^{m+1}$ such that $J(\lambda,\mu_i,y_i)\begin{bmatrix}\xi^T & \alpha\end{bmatrix}^T = 0$. The first row gives

$$B(\lambda,\mu_i)\xi + B_3 y_i \alpha = 0, \tag{2.4}$$

and the second row gives

$$c^T \xi = 0. \tag{2.5}$$

The cases $\alpha = 0$ and $\alpha \neq 0$ are investigated separately. Assume that $\alpha = 0$; then $\xi \neq 0$, and thus (2.4) implies that ξ is an eigenvector to the GEP. However, (2.5) implies that ξ is not a scaling of y_i; hence, μ_i is not simple. Assume that $\alpha \neq 0$. Note that since the pencil is regular and $\mu_i \in \mathbb{C}$ we have that $B_3 y_i \neq 0$. Then by rescaling (2.4) with $1/\alpha$ we see that there exists a Jordan chain of length at least two; hence, μ_i is not simple.

To prove the converse, assume that μ_i is a nonsimple eigenvalue (semisimple or nonsemisimple). Choose a vector $u \in \mathbb{C}^m$ such that in the semisimple case u is a second eigenvector to μ_i with any normalization, and in the nonsemisimple case u is the second vector of a Jordan chain of length at least two, corresponding to μ_i. Let $\xi := u - (c^T u) y_i$, and note that $\xi \neq 0$ in the semisimple case. By inserting the definition of ξ into equations (2.4) and (2.5) and utilizing the definition of u, we can see that the vectors $[\xi^T \quad 0]^T$ and $[\xi^T \quad 1]^T$ are nontrivial vectors in the nullspace of $J(\lambda, \mu_i, y_i)$ for the semisimple and nonsemisimple case, respectively. Hence, $J(\lambda, \mu_i, y_i)$ is singular. In the latter case, $Bu + B_3 y_i = 0$ since that is the Jordan chain defining the chosen u. ∎

Under the same conditions that the implicit functions exist we have the following equivalence between the solutions to the NEP (1.3) and the solutions to the two-parameter eigenvalue problem (1.1).

THEOREM 2.3 (equivalence). *Assume the quadruplet* $(\lambda, x, \mu, y) \in \mathbb{C} \times \mathbb{C}^n \times \mathbb{C} \times \mathbb{C}^m$ *is such that* $c^T y = 1$*, the pencil associated with the GEP* (2.1) *is regular, and* $J(\lambda, \mu, y)$ *defined in* (2.2) *is nonsingular. Then,* (λ, x, μ, y) *is a solution to* (1.1) *if and only if* (λ, x) *is a solution to the NEP* (1.3) *for one pair of functions* $(g_i(\lambda), \phi_i(\lambda)) = (\mu, y)$ *which satisfies* (1.2)*, where* g_i *and* ϕ_i *are the functions defined in Lemma* 2.1.

Proof. To prove the forward implication direction suppose (λ, x, μ, y) is a solution to (1.1). From Lemma 2.1, there are functions g and ϕ such that $g(\lambda) = \mu$ and $\phi(\lambda) = y$. Therefore, (1.3) is satisfied for that pair $(g(\lambda), \phi(\lambda))$.

To prove the backward implication direction suppose (λ, x) is a solution to (1.3) for a given pair $(g(\lambda), \phi(\lambda))$. Then $(\lambda, x, \mu, y) = (\lambda, x, g(\lambda), \phi(\lambda))$ is a solution to (1.1). More precisely, (1.1a) is satisfied since (1.3) is, and (1.1b) is satisfied due to (1.2). ∎

The theorems above can be further interpreted as follows. A direct consequence of Lemma 2.1 and Theorem 2.2 is that if the pencil is regular, then the simple eigenvalues of the GEP (2.1) are analytic in a region around the point λ. Hence, in this sense, there exists a nonlinearization. We now further discuss the assumptions in the theory.

Remark 2.4 (theory assumptions). Note that the problem (2.1) is a GEP, whose properties are independent of normalization, and for every eigenpair of the GEP there exists a vector c not orthogonal to the eigenvector. The assumption $c^T y \neq 0$ is therefore not a restriction of generality.

If eigenvalues of the GEP have multiplicity greater than one, the theory does not predict an analytic nonlinearization. Moreover, from perturbation theory we know that there are algebraic multivalued functions which can have branch-point singularities. Hence, at such a point, there are corresponding nonlinear functions and a (multivalued) nonlinearization exists in this sense. We have restricted the theory to simple eigenvalues for simplicity.

The theorems are based on the assumption that the pencil is regular for a fixed $\lambda \in \mathbb{C}$. The pencil can be singular, e.g., if $B_1 + \lambda B_2$ and B_3 have intersecting nullspaces. If the pencil is singular, then for each μ there exists a nonzero vector $y \in \mathbb{C}^m$ such that $(B_1 + \lambda B_2 + \mu B_3) y = 0$. Since all values of μ satisfy the GEP, the equation

does not define μ as a function of λ. The assumption is thus a limitation of the method since it cannot directly be applied to these eigenvalues. However, situations with a singular pencil may often be approached directly, and it is possible that a whole set of eigenvalues for the two-parameter eigenvalue problem can be found by solving (1.1a) for μ while keeping λ fixed. The situation can be exemplified by the extreme case where $B_3 = 0$. Then values of λ can be determined by (1.1b), independently of μ, and the latter can then be found by solving (1.1a) with the corresponding fixed λ-values. In this case the λ-values are exactly those that make the pencil $(-(B_1 + \lambda B_2), 0)$ singular.

2.2. Nonlinearizations leading to quadratic eigenvalue problems. We first illustrate the theory in the previous section with an implicitly defined function which can be derived explicitly. Consider the two-parameter eigenvalue problem

$$0 = A_1 x + \lambda A_2 x + \mu A_3 x, \tag{2.6a}$$

$$0 = \left(\begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix} + \lambda \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \mu \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix} \right) y \tag{2.6b}$$

for general matrices A_1, A_2, and A_3. The second row in (2.6b) implies that the elements in the vector $y^T = \begin{bmatrix} y_1 & y_2 \end{bmatrix}$ are related by $y_2 = \lambda y_1$. The first row in (2.6b) becomes $\lambda^2 y_1 - \mu y_1 = 0$. Hence, since $y_1 \neq 0$, we have $\mu = \lambda^2$, and (2.6a) becomes

$$0 = A_1 x + \lambda A_2 x + \lambda^2 A_3 x. \tag{2.7}$$

This problem is commonly known as the quadratic eigenvalue problem, which has been extensively studied in the literature [45]. The example shows that the two-parameter eigenvalue problem (2.6) can be nonlinearized to a quadratic eigenvalue problem. Moreover, the determinant operator equation (1.7a) leads to the equation

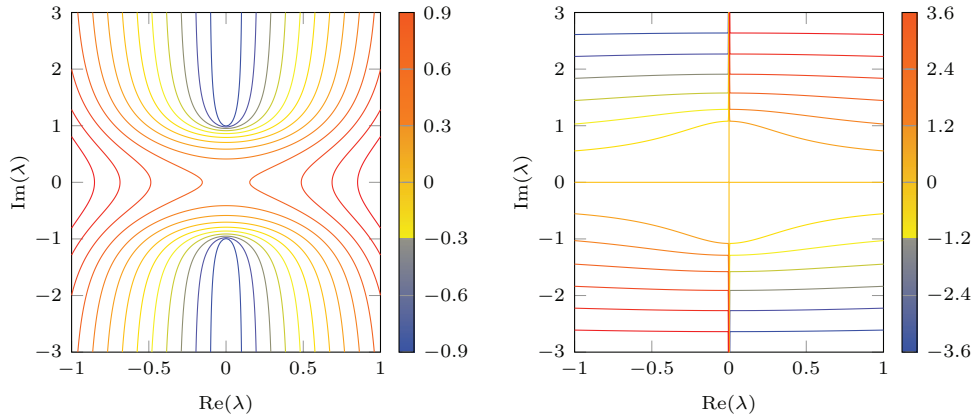
$$\begin{bmatrix} -A_1 & 0 \\ 0 & A_3 \end{bmatrix} z = \lambda \begin{bmatrix} A_2 & A_3 \\ A_3 & 0 \end{bmatrix} z,$$

which is a particular companion linearization of (2.7). (It is in fact a symmetry preserving linearization [45, section 3.4].) Many of the linearizations of polynomial eigenvalue problems given in [28] can be obtained in a similar fashion. Since the second equation (1.1b) can be expressed as $\det(B(\lambda, \mu)) = 0$, which is a bivariate polynomial, this example is consistent with the bivariate viewpoint of companion linearizations in [33]. Some higher-degree polynomials can be constructed analogously to above, e.g., the polynomial eigenvalue problem $A_1 + \lambda A_2 + \lambda^m A_3$. However, the general higher-degree polynomial eigenvalue problem does not seem to fit into the class of two-parameter eigenvalue problems.

2.3. Nonlinearization leading to algebraic functions. The previous example can be modified in such a way that it leads to algebraic functions, which is also the generic situation. Nontrivial solutions to (1.1b) satisfy $\det(B(\lambda, \mu)) = 0$, which is a bivariate polynomial. Therefore, the functions $g_i(\lambda)$ are roots of a polynomial, where the coefficients are polynomials in λ, i.e., g_i are algebraic functions. The generic situation can be seen from the case where $m = 2$:

$$0 = (A_1 + \lambda A_2 + \mu A_3)x, \tag{2.8a}$$

$$0 = \left(\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \lambda \begin{bmatrix} 0 & e \\ f & 0 \end{bmatrix} + \mu \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) y. \tag{2.8b}$$



(a) $\mathrm{Re}(g_+(\lambda))$ (b) $\mathrm{Im}(g_+(\lambda))$

FIG. 2.1. *The square-root nonlinearity illustrated in the example in section* 2.3*, with* $a = 3$*,* $b = 2$*,* $c = -1$*,* $d = -2$*,* $e = 2$*, and* $f = 1$*. We observe a square-root singularity at* $\lambda = \pm\sqrt{-3/2}$ *which are the roots of* $p(\lambda)$*.*

We obtain that μ is the root of a polynomial, where the coefficients depend on λ, i.e.,

$$0 = (\mu + a)(\mu + d) - (c + \lambda f)(b + \lambda e).$$

The explicit solutions to this quadratic equation are given by

$$\mu = g_\pm(\lambda) = -\frac{a+d}{2} \pm \sqrt{\frac{(a+d)^2}{4} - ad + (b + \lambda e)(c + \lambda f)}.$$

We see by insertion of $\mu = g_\pm$ into (2.8a) that the nonlinearization of (2.8) is an NEP with an algebraic nonlinearity. The function g_+ is illustrated in Figure 2.1.

Several general conclusions can be made from this example. Note that the variables a, b, c, d, e, f can be used for fitting of any function $\sqrt{p(\lambda)}$ where p is a polynomial of degree two. Therefore, we can now reverse the nonlinearization, and for the trivial case $a = d = 0$ we directly obtain the following characterization.

LEMMA 2.5 (two-parameterization of an algebraic NEP). *Suppose* $p(\lambda) = (b + \lambda e)(c + \lambda f)$ *is given, and let* $a = d = 0$*. If* (λ, x) *is a solution to the NEP*

$$(A_1 + \lambda A_2 + \sqrt{p(\lambda)}A_3)x, \tag{2.9}$$

then (λ, x, μ, y) *satisfies the two-parameter eigenvalue problem equation* (2.8) *with* $\mu := \sqrt{p(\lambda)}$ *and*

$$y := \begin{bmatrix} \sqrt{b + \lambda e} \\ -\sqrt{c + \lambda f} \end{bmatrix}.$$

A further consequence of the lemma is that problems of the type (2.9) can be linearized to a GEP using the determinant operators (1.7). More precisely, the combination of Lemma 2.5 and (1.7) shows that (2.9) can be solved by computing solutions to

$$\begin{bmatrix} A_1 & -bA_3 \\ -cA_3 & A_1 \end{bmatrix} z = \lambda \begin{bmatrix} -A_2 & eA_3 \\ fA_3 & -A_2 \end{bmatrix} z.$$

The fact that algebraic NEPs can be linearized was already pointed out in the conference presentation [39], using techniques not involving two-parameter eigenvalue problems.

Also note that the functions $g_i(\lambda)$ have branch-point singularities. This is the generic situation, and we can therefore never expect that the nonlinearizations are entire functions in general. The singularities restrict the performance of many methods, as we will see in the simulations. The implications of singularities in practice are well known in quantum chemistry, where parameterized eigenvalue problems are a fundamental tool and the singularities are referred to as intruder states [11, Chapter 14]. In that context, methods for computing the closest singularity (which limits the performance of the method) are given in [21, 25].

3. Algorithm specializations.

3.1. Derivative based algorithms. Many NEP algorithms are based on derivatives of M. We will now illustrate how to efficiently and reliably access the derivatives of the NEP stemming from a nonlinearization of a two-parameter eigenvalue problem. As a representative first situation we consider the augmented Newton method; see [38, 47]. It can be derived by an elimination of the correction equation in Newton's method and leads to separate eigenvalue and eigenvector update formulas expressed as

$$x_{k+1} = \alpha_k M(\lambda_k)^{-1} M'(\lambda_k) x_k, \tag{3.1a}$$

$$\alpha_k^{-1} = d^T M(\lambda_k)^{-1} M'(\lambda_k) x_k, \tag{3.1b}$$

and $\lambda_{k+1} = \lambda_k - \alpha_k$, where $d \in \mathbb{C}^n$ is a normalization vector. In an implementation, one takes advantage of the fact that the same linear system appears twice and only needs to be computed once. The iteration has appeared in many variations with different names, e.g., inverse iteration [41] and Newton's method [46].

In order to apply (3.1) we clearly need the derivative of M defined in (1.3), which can be obtained directly if we can compute the derivative of the implicitly defined function g_i. Note that the functions $g_i(\lambda)$ (as well as the auxiliary vector $\phi_i(\lambda)$) can be evaluated by solving the GEP (2.1) and normalizing according to $c^T y_i = 1$. Since the functions are analytic in general, their respective derivatives exist. They can be computed according to the following result, which gives a recursion that can compute the kth derivative by solving k linear systems of dimension $(m+1) \times (m+1)$. The adaption of the theorem and (3.1) into an algorithm results in Algorithm 1.

THEOREM 3.1 (explicit recursive form for derivatives). *Let $\lambda \in \mathbb{C}$ be given. Assume that the pencil associated with the GEP (2.1) is regular and that $(\mu_i, y_i) \in \mathbb{C} \times \mathbb{C}^m$ is a solution to the GEP with y_i normalized as $c^T y_i = 1$. Moreover, assume that $J(\lambda, \mu_i, y_i)$ is invertible, where J is defined in (2.2). Let g_i and ϕ_i be the functions defined in Lemma 2.1; then the kth derivative, $k = 1, 2, \ldots$, of g_i and ϕ_i are given by*

$$\begin{bmatrix} \phi_i^{(k)}(\lambda) \\ g_i^{(k)}(\lambda) \end{bmatrix} = J(\lambda, \mu_i, y_i)^{-1} \begin{bmatrix} -b_k \\ 0 \end{bmatrix}, \tag{3.2}$$

where

$$b_k = kB_2\phi_i^{(k-1)}(\lambda) + \sum_{j=1}^{k-1} \binom{k}{j} g_i^{(k-j)}(\lambda) B_3 \phi_i^{(j)}(\lambda).$$

Proof. We again consider the analytic function f given by (2.3). By Lemma 2.1 we know that g_i and ϕ_i are analytic around λ and that $f(\lambda, g_i(\lambda), \phi_i(\lambda)) = 0$ in a neighborhood of λ. Taking the kth implicit derivative with respect to λ gives

$$0 = \frac{d^k}{d\lambda^k}\begin{bmatrix} B_1\phi_i(\lambda) \\ c^T\phi_i(\lambda) - 1 \end{bmatrix} + \frac{d^k}{d\lambda^k}\begin{bmatrix} \lambda B_2\phi_i(\lambda) \\ 0 \end{bmatrix} + \frac{d^k}{d\lambda^k}\begin{bmatrix} g_i(\lambda)B_3\phi_i(\lambda) \\ 0 \end{bmatrix}.$$

The first term is found directly as

$$\frac{d^k}{d\lambda^k}\begin{bmatrix} B_1\phi_i(\lambda) \\ c^T\phi_i(\lambda) - 1 \end{bmatrix} = \begin{bmatrix} B_1\phi_i^{(k)}(\lambda) \\ c^T\phi_i^{(k)}(\lambda) \end{bmatrix}.$$

The second and third term can be calculated, by using the Leibniz derivation rule for products, to be

$$\frac{d^k}{d\lambda^k}\begin{bmatrix} \lambda B_2\phi_i(\lambda) \\ 0 \end{bmatrix} = \begin{bmatrix} \lambda B_2\phi_i^{(k)}(\lambda) \\ 0 \end{bmatrix} + \binom{k}{k-1}\begin{bmatrix} B_2\phi_i^{(k-1)}(\lambda) \\ 0 \end{bmatrix}$$

and

$$\frac{d^k}{d\lambda^k}\begin{bmatrix} g_i(\lambda)B_3\phi_i(\lambda) \\ 0 \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{k-1}\binom{k}{j}g_i^{(k-j)}(\lambda)B_3\phi_i^{(j)}(\lambda) \\ 0 \end{bmatrix} + \begin{bmatrix} g_i^{(k)}(\lambda)B_3\phi_i(\lambda) \\ 0 \end{bmatrix} + \begin{bmatrix} g_i(\lambda)B_3\phi_i^{(k)}(\lambda) \\ 0 \end{bmatrix}.$$

We emphasize the recursion: All derivatives up to order $k-1$ can be considered known since these do not depend on the higher derivatives. Collecting the known terms in the right-hand side gives the result. ∎

Remark 3.2. As a special case of Theorem 3.1, for $k = 1$, we find that $g_i'(\lambda) = -\frac{w_i^H B_2 y_i}{w_i^H B_3 y_i}$ where w_i is the corresponding left eigenvector to the eigenpair (μ_i, y_i). It follows from multiplying the first block-row of equation system (3.2) from the left with w_i^H. The result is a special case of well-known perturbation analysis for GEPs [14, Theorem 2.5]. In our case $g'(\lambda)$ is the perturbation of the eigenvalue μ with respect to λ in the GEP (2.1), more precisely, a perturbation of the matrix $-(B_1 + \lambda B_2)$ with the structured perturbation εB_2.

Specifically, the closed form of $g_i'(\lambda)$ means that the derivative of the NEP (1.3) can be written in closed form as

$$M'(\lambda) = A_2 - \frac{w_i^H B_2 y_i}{w_i^H B_3 y_i} A_3.$$

For methods only requiring the first derivative of $M(\lambda)$, the above expression can be used instead of (3.2). However, that requires the computations of the left eigenvector of the GEP. We will need the expression for theoretical purposes in section 4.

The family of methods in [23, 22, 32] (flavors of the infinite Arnoldi method) also requires derivative information. These methods require computation of quantities such as

$$\begin{aligned} z_0 &= M(\sigma)^{-1}(M'(\sigma)x_1 + \cdots + M^{(p)}(\sigma)x_p) \\ &= M(\sigma)^{-1}(A_1x_1 + A_2\sum_{j=1}^{p} g^{(j)}(\sigma)x_j), \end{aligned}$$

Algorithm 1: Augmented Newton method for nonlinearized two-parameter eigenvalue problem

input: Starting values $\lambda_0 \in \mathbb{C}$ and $x_0 \in \mathbb{C}^n$
output: Approximations of eigenpairs of (1.3)

for $k = 1, 2, 3, \ldots,$ **do**
  Compute $g_i(\lambda_k) := \mu$ from the GEP (2.1) with c-normalized eigenvector $y \in \mathbb{C}^m$
  if $\|A(\lambda_k, \mu)x_k\| \leq TOL$ **then**
    break
  end
  Compute $g_i'(\lambda_k)$ by computing $b_1 = B_2 y$ and solving the linear system of equations (3.2)
  Compute $u = M(\lambda_k)^{-1} M'(\lambda_k) x_k$ by using the results in steps 2–5
  Compute $\alpha_k = (d^T u)^{-1}$
  Compute $x_{k+1} = \alpha_k y$
  Compute $\lambda_{k+1} = \lambda_k - \alpha_k^{-1}$
end

where $x_1, \ldots, x_p$ are given vectors. The computation requires higher derivatives of g_i. However, σ is unchanged throughout the iteration, and therefore the matrix in the linear system for derivative computation (3.2) is unchanged. Hence, all needed derivatives can be computed by solving an additional linear system. If $m \ll n$, this will in general not be computationally demanding. We also note that these fixed-shift methods choose a branch g_i in the initial solution of the GEP (2.1) and then stay on that branch.

Remark 3.3. There are in general m different branches of the nonlinearization. Each branch corresponds to a different eigenvalue, $g_i(\lambda) := \mu_i$, $i = 1, \ldots, m$, of the GEP (2.1). In a method, at each evaluation of $g_i(\lambda)$, a branch needs to be chosen. Practical ways to choose a branch are the following:

(i) closest to the previous value, i.e., $\arg\min_i\{|g_i(\lambda_k) - g_i(\lambda_{k-1})|\}$;
(ii) closest to a specific target value μ_*, i.e., $\arg\min_i\{|g_i(\lambda_k) - \mu_*|\}$;
(iii) to minimize the residual norm of the NEP (1.3), i.e., $\arg\min_i\{\|(A_1 + \lambda_k A_2 + g_i(\lambda_k)A_3)x_k\|\}$

We use option 3.3 in the simulations in section 5.

Note that the way the iterate λ_k is updated depends on the NEP algorithm and is in that sense independent of how the branch $g_i(\lambda_k)$ is chosen. Hence, the resulting method can either aim for eigenvalues (λ, μ) close to some joint target (λ_*, μ_*) or close to some target λ_*.

3.2. Projection methods. Many NEP algorithms require the computation of a projected problem

$$W^T M(\lambda) V z = 0, \tag{3.3}$$

where $V, W \in \mathbb{C}^{n \times p}$ are orthogonal matrices. The problem (3.3) is again an NEP but of smaller size. This can be viewed as a Petrov–Galerkin projection of the spaces spanned by the columns of V and W. The projection is sometimes called subspace acceleration (or the nonlinear Rayleigh–Ritz procedure), since it is often used to

improve properties of a more basic algorithm, e.g., the nonlinear Arnoldi method [49], Jacobi–Davidson methods [7, 5], block preconditioned harmonic projection methods [51], the infinite Lanczos method [31], and many more.

In order to give access to these methods, we need to provide a way to solve (3.3) for our nonlinearized problem. Fortunately, the projected problem stemming from the nonlinearized two-parameter eigenvalue problem, i.e.,

$$(W^T A_1 V + \lambda W^T A_2 V + g_i(\lambda) W^T A_3 V) z = 0, \tag{3.4}$$

has a structure which suggests straightforward methods for the projected problem. This is because the projected NEP has the same structure as the nonlinearized two-parameter eigenvalue problem and can therefore be lifted back to a two-parameter eigenvalue problem, but now of much smaller size. We can then use general methods for two-parameter eigenvalue problems. This is directly observed from the fact that (3.4) is the nonlinearization of a two-parameter eigenvalue problem with projected A-matrices. It is made more precise in the following result.

COROLLARY 3.4 (projected nonlinearized problem). *Assume the quadruplet* $(\lambda, z, \mu, y) \in \mathbb{C} \times \mathbb{C}^p \times \mathbb{C} \times \mathbb{C}^m$ *is such that* $c^T y = 1$*, the pencil associated with the GEP* (2.1) *is regular, and* $J(\lambda, \mu, y)$ *defined in* (2.2) *is nonsingular. Then,* (λ, z, μ, y) *is a solution to the two-parameter eigenvalue problem*

$$0 = W^T A_1 V z + \lambda W^T A_2 V z + \mu W^T A_3 V z, \tag{3.5a}$$

$$0 = B_1 y + \lambda B_2 y + \mu B_3 y \tag{3.5b}$$

if and only if (λ, z) *is a solution to* (3.4) *for one pair of functions* $(g(\lambda), \phi(\lambda)) = (\mu, y)$ *which satisfies* (1.2).

Proof. This follows directly from the application of Theorem 2.3 on the projected problem (3.5) and the NEP (3.4). ∎

If the projection space is small, $p \ll n$, and $m \ll n$, then we may even solve the two-parameter eigenvalue problem using the operator determinant eigenvalue equations (1.7) or [15, Algorithm 2.3].

The situation $p = 1$ implies that the projected problem is a scalar problem and reduces to the so-called Rayleigh functional. There are several methods based on the Rayleigh functional, e.g., residual inverse iteration [34], and variational principle based approaches such as [43] and references therein. The fact that the projected problem is scalar and linear allows us to eliminate one of the variables, and we find that the other variable is a solution to the GEP. The following corollary specifies the formulas more precisely when μ is eliminated, and the adaption of the result into the residual inverse iteration is given in Algorithm 2.

COROLLARY 3.5. *Assume that* $w^T A_3 v \neq 0$*. A solution,* λ*,* $\mu \in \mathbb{C}$ *and* $y \in \mathbb{C}^m$*, to the projected NEP* (3.4) *with* $p = 1$ *can be characterized as follows. The tuple* (λ, y) *is a solution to the GEP*

$$((w^T A_3 v) B_1 - (w^T A_1 v) B_3) y = \lambda ((w^T A_2 v) B_3 - (w^T A_3 v) B_2) y, \tag{3.6}$$

and μ *is given by*

$$\mu = -\frac{w^T A_1 v + \lambda w^T A_2 v}{w^T A_3 v}. \tag{3.7}$$

Proof. This is derived from a special case of Corollary 3.4 where $p = 1$. Assuming that $w^T A_3 v \neq 0$, the relation (3.5a) with $W = w$ and $V = v$ can be solved for μ resulting in the relation (3.7). By inserting this relation into (3.5b) we obtain the GEP (3.6). ∎

Algorithm 2: Resinv for nonlinearized two-parameter eigenvalue problem

input: Approximate eigenvector $x_0 \in \mathbb{C}^n$, shift $\sigma \in \mathbb{C}$, right Rayleigh functional vector $w \in \mathbb{C}^n$

output: Approximations of eigenpairs of (1.3)

1 Compute $M(\sigma)$ and factorize
while *not converged* **do**
2 Compute $\lambda_{k+1} = \lambda$ by solving the GEP (3.6) for $v = x_k$
3 Compute μ from (3.7) with $v = x_k$
4 Compute $z := M(\lambda_{k+1})x_k = A_0 x_k + \lambda_{k+1} A_1 x_k + \mu A_2 x_k$
5 Compute correction $u_{k+1} = x_k - M(\sigma)^{-1} z$ using the factorization computed in step 1
6 Normalize $x_{k+1} = u_{k+1}/\|u_{k+1}\|$
end

Remark 3.6. In Corollary 3.5 we have assumed that $w^T A_3 v \neq 0$. There is an analogous formula that can be used when $w^T A_2 v \neq 0$, and λ is eliminated. Then (μ, y) is a solution to the GEP

$$((w^T A_2 v)B_1 - (w^T A_1 v)B_2)y = \mu((w^T A_3 v)B_2 - (w^T A_2 v)B_3)y,$$

and λ is given by

$$\lambda = -\frac{w^T A_1 v + \mu w^T A_3 v}{w^T A_2 v}.$$

For completeness we also consider the assumption $w^T A_2 v = w^T A_3 v = 0$. There are two cases: First, if $w^T A_1 v \neq 0$, then there is no solution to the projected problem (3.4). Second, if $w^T A_1 v = 0$, then (3.4) is satisfied for any value λ. Hence, for any fixed λ, (μ, y) can be taken as any solution to the GEP (3.5b).

Remark 3.7. In step 2 of Algorithm 2 a specific value has to be selected for λ_{k+1}, and there are in general m different values to choose from. The situation is inherent to the algorithm, and the literature suggests to choose the value closest to the previous iterate, i.e., $\arg\min_i\{|\lambda_{k+1}^{(i)} - \lambda_k|\}$; see, e.g., [34, 10]. This is the strategy used in the simulations in section 5.

4. Conditioning and accuracy.

In order to characterize when the elimination procedure works well, we now analyze how the technique behaves subject to perturbations. As a consequence of this we can directly conclude how backward stable computation of g influences the accuracy (section 4.2).[1]

4.1. Conditioning as an NEP.

Standard results for the condition number of NEPs can be used to analyze perturbations with respect to the A-matrices. More precisely, for $\lambda \in \mathbb{C}$ we define

[1] For notational convenience the i index on g_i is dropped in this section.

$$\kappa_A(\lambda) := \limsup_{\varepsilon\to 0}\left\{\frac{|\Delta\lambda|}{\varepsilon} : \|\Delta A_j\| \le \varepsilon\alpha_j, j = 1,2,3\right\},$$

where α_j are scalars for $j = 1,2,3$, and $\Delta\lambda$ is such that

$$0 = (A_1 + \Delta A_1 + (\lambda+\Delta\lambda)(A_2+\Delta A_2) + g(\lambda+\Delta\lambda)(A_3+\Delta A_3))(x+\Delta x), \tag{4.1}$$

where additionally we require that $\|\Delta x\| \to 0$ and $|\Delta\lambda| \to 0$ as $\varepsilon \to 0$; cf. [14, page 499] Then we know (see, e.g., [1]) that

$$\kappa_A(\lambda) = \|v\|\|x\|\frac{\alpha_1 + |\lambda|\alpha_2 + |g(\lambda)|\alpha_3}{|v^H M'(\lambda)x|}, \tag{4.2}$$

where v, x are the corresponding left and right eigenvectors. In the following we will establish how this formula is modified when we also consider perturbations in the B-matrices. Note that this implies that the function g is also perturbed and we cannot directly use the standard result. We therefore define, for $\lambda \in \mathbb{C}$, the condition number

$$\kappa(\lambda) := \limsup_{\varepsilon\to 0}\left\{\frac{|\Delta\lambda|}{\varepsilon} : \|\Delta A_j\| \le \varepsilon\alpha_j, j = 1,2,3, \text{ and } \|\Delta B_j\| \le \varepsilon\beta_j, j = 1,2,3\right\},$$

where β_j are scalars for $j = 1,2,3$, and $\Delta\lambda$ fulfills (4.1) but with a perturbed g, i.e., $\mu + \Delta\mu = g(\lambda+\Delta\lambda)$, such that

$$0 = (B_1 + \Delta B_1 + (\lambda+\Delta\lambda)(B_2+\Delta B_2) + (\mu+\Delta\mu)(B_3+\Delta B_3))\,(y+\Delta y), \tag{4.3a}$$

$$1 = c^T(y+\Delta y), \tag{4.3b}$$

where additionally we require that $\|\Delta y\| \to 0$ and $|\Delta\mu| \to 0$ as $\varepsilon \to 0$. The definitions can be used both for *absolute* and *relative condition* numbers by setting $\alpha_j = \beta_j = 1$ or $\alpha_j = \|A_j\|$, $\beta_j = \|B_j\|$ for $j = 1,2,3$, respectively.

As an intermediate step we first consider the perturbation of $\mu \in \mathbb{C}$ subject to perturbations in the B-matrices and fixed perturbations in $\lambda \in \mathbb{C}$ by analyzing

$$\kappa_g(\lambda) := \limsup_{\varepsilon\to 0}\left\{\frac{|\Delta\mu|}{\varepsilon} : |\Delta\lambda| \le \varepsilon\gamma \text{ and } \|\Delta B_j\| \le \varepsilon\beta_j, j = 1,2,3\right\},$$

where γ is a scalar and $\Delta\mu$ satisfies (4.3) for a given λ . The following result shows that κ_g can be expressed as a sum of perturbations associated with the B-matrices and perturbations associated with λ.

LEMMA 4.1. *Let $\lambda \in \mathbb{C}$ be given. Suppose the pencil associated with the GEP* (2.1) *is regular and that $g(\lambda) = \mu \in \mathbb{C}$ is a simple eigenvalue of the GEP with w and y being corresponding left and right eigenvectors, respectively. Then,*

$$\kappa_g(\lambda) = \kappa_{g,B}(\lambda) + \kappa_{g,\lambda}(\lambda),$$

where

$$\kappa_{g,B}(\lambda) = \|w\|\|y\|\frac{\beta_1 + |\lambda|\beta_2 + |g(\lambda)|\beta_3}{|w^H B_3 y|} \qquad \textit{and} \qquad \kappa_{g,\lambda}(\lambda) = \gamma\frac{|w^H B_2 y|}{|w^H B_3 y|}.$$

Proof. Since μ is a simple eigenvalue of the GEP (2.1), the eigenvalue and eigenvector are analytic, and therefore $\Delta y = \mathcal{O}(\varepsilon)$ when all the perturbations are $\mathcal{O}(\varepsilon)$. Moreover, since μ is a simple finite eigenvalue, then $w^H B_3 y \neq 0$. By collecting all the higher order terms the perturbed GEP (4.3a) can thus be written as

$$(\Delta B_1 + \lambda\Delta B_2 + \Delta\lambda B_2 + \mu\Delta B_3 + \Delta\mu B_3)\, y + B(\lambda,\mu)\Delta y = \mathcal{O}(\varepsilon^2).$$

Multiplying with w^H from the left, solving for $\Delta\mu$, and dividing with ε gives that

$$\frac{\Delta\mu}{\varepsilon} = -\frac{w^H\Delta B_1 y + \lambda w^H\Delta B_2 y + \Delta\lambda w^H B_2 y + \mu w^H\Delta B_3 y}{\varepsilon w^H B_3 y} + \mathcal{O}(\varepsilon). \tag{4.4}$$

An upper bound is thus found as

$$\frac{\Delta\mu}{\varepsilon} \leq \|w\|\|y\|\frac{\beta_1 + |\lambda|\beta_2 + |\mu|\beta_3}{|w^H B_3 y|} + \gamma\frac{|w^H B_2 y|}{|w^H B_3 y|} + \mathcal{O}(\varepsilon).$$

It remains to show that the bound can be attained. This follows from considering $\hat{B} = wy^H/\|w\|\|y\|$ and inserting

$$\Delta B_1 := -\varepsilon\beta_1\hat{B}, \qquad \Delta B_2 := -\varepsilon\frac{\overline{\lambda}}{|\lambda|}\beta_2\hat{B},$$
$$\Delta B_3 := -\varepsilon\frac{\overline{g(\lambda)}}{|g(\lambda)|}\beta_3\hat{B}, \qquad \Delta\lambda := -\varepsilon\frac{\overline{w^H B_2 y}}{|w^H B_2 y|}\frac{|w^H B_3 y|}{\overline{w^H B_3 y}}\gamma$$

into (4.4). ∎

Using the intermediate result we can now show that the condition number $\kappa(\lambda)$ is the sum of the standard condition number of NEPs and a term representing perturbations in g generated by perturbations in the B-matrices, i.e., $\kappa_{g,B}(\lambda)$.

THEOREM 4.2. *Let $\lambda \in \mathbb{C}$ be a simple eigenvalue of the NEP (1.3) with v and x being corresponding left and right eigenvectors, respectively. Moreover, for this λ, suppose the pencil associated with the GEP (2.1) is regular and that $g(\lambda) = \mu \in \mathbb{C}$ is a simple eigenvalue of the GEP with w and y being corresponding left and right eigenvectors, respectively.*[2] *Then,*

$$\kappa(\lambda) = \kappa_A(\lambda) + \kappa_{g,B}(\lambda)\frac{|v^H A_3 x|}{|v^H M'(\lambda) x|},$$

where $\kappa_A(\lambda)$ is given by (4.2).

Proof. Recall the assumptions that the NEP (1.3), i.e., M, is analytic, that λ is a simple eigenvalue of the NEP, and that μ is a simple eigenvalue of the GEP (2.1). Hence, the eigenvalues and eigenvectors are analytic, and therefore $\Delta x = \mathcal{O}(\varepsilon)$ when all the perturbations are $\mathcal{O}(\varepsilon)$. Moreover, we note that it also implies that $v^H M'(\lambda)x \neq 0$ and $w^H B_3 y \neq 0$. By using that $g(\lambda+\Delta\lambda) = g(\lambda) + \Delta\mu$ and collecting all the higher order terms, the perturbed NEP (4.1) can therefore be written as

$$(\Delta A_1 + \lambda\Delta A_2 + \Delta\lambda A_2 + g(\lambda)\Delta A_3 + \Delta\mu A_3)x + M(\lambda)\Delta x = \mathcal{O}(\varepsilon^2).$$

Multiplying with v^H from the left, expanding $\Delta\mu$ according to (4.4), solving for $\Delta\lambda$, and dividing with ε gives that

$$\frac{\Delta\lambda}{\varepsilon} = -\frac{v^H\Delta A_1 x + \lambda v^H\Delta A_2 x + g(\lambda)v^H\Delta A_3 x + \theta_{g,B}(\lambda)v^H A_3 x}{\varepsilon v^H\left(A_2 - \frac{w^H B_2 y}{w^H B_3 y}A_3\right)x} + \mathcal{O}(\varepsilon), \tag{4.5}$$

[2] This corresponds to (λ,μ) being a simple eigenvalue to the two-parameter eigenvalue problem.

where $\theta_{g,B}(\lambda) := -(w^H\Delta B_1 y + \lambda w^H \Delta B_2 y + g(\lambda) w^H \Delta B_3 y)/(w^H B_3 y)$. Based on Remark 3.2 we observe that the denominator of (4.5) is equal to $\varepsilon v^H M'(\lambda) x$. An upper bound is therefore

$$\frac{\Delta\lambda}{\varepsilon} \leq \|v\|\|x\| \frac{\alpha_1 + |\lambda|\alpha_2 + |g(\lambda)|\alpha_3}{|v^H M'(\lambda) x|} + \kappa_{g,B}(\lambda) \frac{|v^H A_3 x|}{|v^H M'(\lambda) x|} + \mathcal{O}(\varepsilon).$$

It remains to show that the bound can be attained. Similar to the proof of Lemma 4.1, this follows from considering $\hat{B} = wy^H/\|w\|\|y\|$ and $\hat{A} = vx^H/\|v\|\|x\|$ and inserting

$$\Delta B_1 := \varepsilon\beta_1\hat{B}, \qquad \Delta B_2 := \varepsilon\frac{\overline{\lambda}}{|\lambda|}\beta_2\hat{B}, \qquad \Delta B_3 := \varepsilon\frac{\overline{g(\lambda)}}{|g(\lambda)|}\beta_3\hat{B},$$

$$\Delta A_1 := -\varepsilon\alpha_1\hat{A}, \qquad \Delta A_2 := -\varepsilon\frac{\overline{\lambda}}{|\lambda|}\alpha_2\hat{A}, \qquad \Delta A_3 := -\varepsilon\frac{\overline{g(\lambda)}}{|g(\lambda)|}\alpha_3\hat{A}$$

into (4.5). ∎

4.2. Backward stable computation of g. The nonlinearization is based on solving a GEP to evaluate the function $g(\lambda)$. We analyze the effects on the accuracy in the computed λ when the GEP is solved numerically with a backward stable method. The analysis assumes the two triplets $(\lambda, x, v) \in \mathbb{C}\times\mathbb{C}^n\times\mathbb{C}^n$ and $(\mu, y, w) \in \mathbb{C}\times\mathbb{C}^m\times\mathbb{C}^m$ are such that λ is a simple eigenvalue of the NEP (1.3), the pencil associated with the GEP (2.1) is regular, μ is a simple eigenvalue of the GEP, and v, w and x, y are corresponding left and right eigenvectors, respectively.

From the assumption that the GEP (2.1) is solved by a backward stable method we know that μ can be characterized as the exact solution to a nearby problem. More precisely, if we define $C_1 := (B_1 + \lambda B_2)$ and $C_2 := B_3$, then μ solves

$$(C_1 + \Delta C_1)y = \mu(C_2 + \Delta C_2)y,$$

where the perturbations ΔC_1 and ΔC_2 are proportional to the errors in our GEP solver. Specifically, there are nonnegative $\beta_1, \beta_3 \in \mathbb{R}$ such that $\|\Delta C_1\| = \beta_1\varepsilon$ and $\|\Delta C_2\| = \beta_3\varepsilon$. Thus, the perturbation in g is precisely captured by $\kappa_{g,B}(\lambda)$ from Lemma 4.1, with $\beta_2 = 0$ and β_1 and β_3 given above, i.e., by the specific choice of GEP solver. Hence, by application of Theorem 4.2 with $\alpha_j = 0$ for $j = 1, 2, 3$ we can conclude that the forward error in λ, induced by the inexact but backward stable computation of $g(\lambda)$, is bounded by

$$|\Delta\lambda| \leq \|w\|\|y\| \frac{\beta_1 + |g(\lambda)|\beta_3}{|w^H B_3 y|} \frac{|v^H A_3 x|}{|v^H M'(\lambda) x|}\varepsilon + \mathcal{O}(\varepsilon^2). \tag{4.6}$$

Without loss of generality we now assume that $\|x\| = \|v\| = \|y\| = \|w\| = 1$.

The upper bound (4.6) is related to the condition number for multiparameter eigenvalue problems as follows. As mentioned in the introduction, the condition number for the two-parameter eigenvalue problem can be directly expressed with the inverse of C_0 defined in (1.8). First note that our assumptions imply that C_0 is invertible.

LEMMA 4.3. *Under the conditions of Theorem* 4.2 *the matrix C_0 is nonsingular, where C_0 is defined in* (1.8).

Proof. By using the expression for $M'(\lambda)$ from Remark 3.2 we thus have

$$(w^H B_3 y)(v^H M'(\lambda) x) = (v^H A_2 x)(w^H B_3 y) - (v^H A_3 x)(w^H B_2 y) = \det(C_0). \tag{4.7}$$

Since the eigenvalues λ and μ are simple we know that $w^H B_3 y \neq 0$ and that $v^H M'(\lambda) x \neq 0$. Hence, $\det(C_0) \neq 0$. ∎

From (4.7) we can conclude that the bound (4.6) on $|\Delta\lambda|$ can be written as

$$|\Delta\lambda| \leq (\beta_1 + |g(\lambda)|\beta_3)\,\frac{|v^H A_3 x|}{|\det(C_0)|}\varepsilon + \mathcal{O}(\varepsilon^2). \tag{4.8}$$

Moreover, for a nonsingular C_0 it is shown in [18, Theorem 6] that the condition number of the two-parameter eigenvalue is

$$K = \|C_0^{-1}\|_\theta,$$

where the θ-norm, i.e., $\|\cdot\|_\theta$, is an induced norm defined in [18, equation (5)].[3] In our case we can explicitly bound the condition number by using bounds following directly from the definition of the θ-norm:

$$\|C_0^{-1}\|_\theta = \frac{1}{|\det(C_0)|}\left\|\begin{bmatrix} w^H B_3 y & -v^H A_3 x \\ -w^H B_2 y & v^H A_2 x \end{bmatrix}\right\|_\theta \geq \frac{1}{|\det(C_0)|}\left\|\begin{bmatrix} 0 & -v^H A_3 x \\ 0 & 0 \end{bmatrix}\right\|_\theta = \frac{|v^H A_3 x||\theta_2|}{|\det(C_0)|}.$$

The parameter θ_2 is the second component of the θ-vector used in the definition of the θ-norm. Hence, the bound in (4.8) can be further bounded by

$$|\Delta\lambda| \leq K\frac{\beta_1 + |g(\lambda)|\beta_3}{|\theta_2|}\varepsilon + \mathcal{O}(\varepsilon^2). \tag{4.9}$$

The typical choices of θ corresponding to the absolute, respectively, relative condition number of the two-parameter eigenvalue problem are $|\theta_2| = 1 + |\lambda| + |g(\lambda)|$ and $|\theta_2| = \|B_1\| + |\lambda|\|B_2\| + |g(\lambda)|\|B_3\|$. From the bounds in (4.9) we therefore conclude that: the error generated by a backward stable method is benign for well-conditioned two-parameter eigenvalue problems.

5. Simulations.

5.1. Random example. We generate an example similar to the example in [15], but with $m \ll n$. More precisely, we let

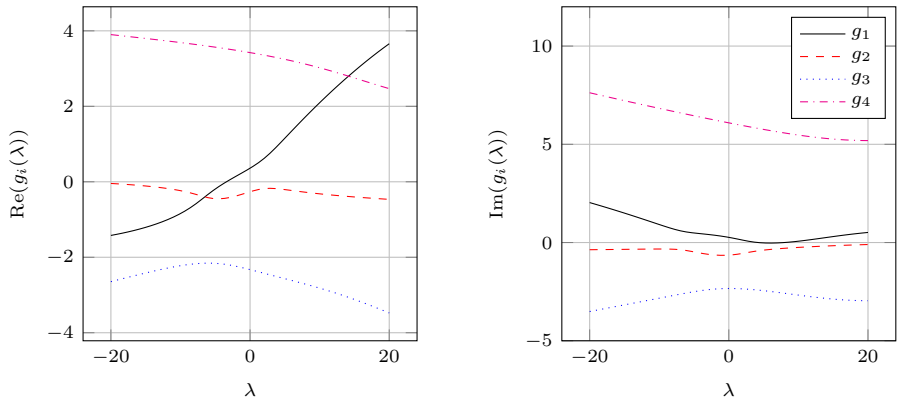
$$A_i = \alpha_i V_{A_i} F_i U_{A_i}, \quad B_i = \beta_i V_{B_i} G_i U_{B_i}, \quad i = 1, 2, 3,$$

where $n = 5000$ and $m = 20$. The matrices V_{A_i}, U_{A_i}, V_{B_i}, U_{B_i} have randomly normal distributed elements, and F_i, G_i are diagonal matrices with randomly normal distributed diagonal elements. The scalars α_i and β_i were selected such that the eigenvalues closest to the origin were of order of magnitude one in modulus ($\alpha_1 = \beta_1 = 1$, $\alpha_2 = \beta_2 = 1/500$, $\alpha_3 = \beta_3 = 1/50$). The simulations were carried out using the Julia language [6] (version 1.1.0), but implementations of the algorithms are available online for both Julia and MATLAB.[4]

Since $m = 20$, we in general obtain 20 different functions $g_1, \ldots, g_{20}$, which we order by magnitude in the origin, each corresponding to a different NEP. Some of the nonlinear functions g_i are visualized in Figure 5.1. The solutions closest to the

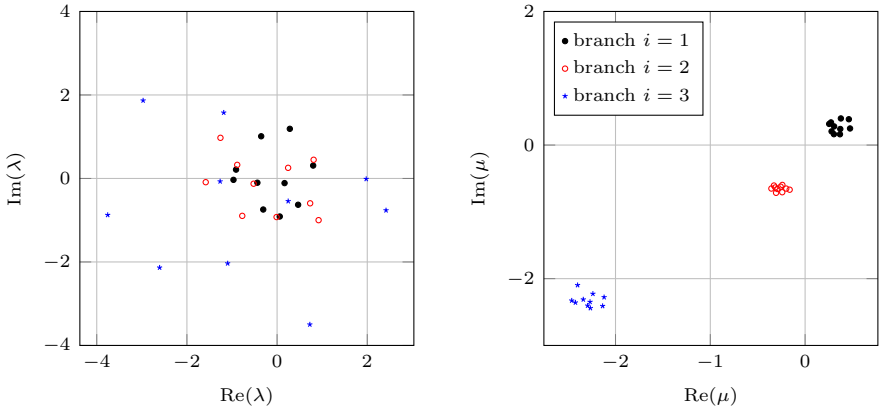
[3] For a given vector $\theta \in \mathbb{R}^n$ with positive entries, the θ-norm of a matrix $C \in \mathbb{C}^{n\times n}$ is defined as $\|C\|_\theta := \max\{\|Cz\| : z \in \mathbb{C}^n, |z_k| = \theta_k \text{ for } k = 1, 2, \ldots, n\}$.

[4] The matrices and the simulations are provided online for reproducibility: http://www.math.kth.se/~eliasj/src/nonlinearization. The simulations were carried out using Ubuntu Linux, Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, 16 GB of RAM.



(a) Real (b) Imag

FIG. 5.1. *The functions $g_i(\lambda)$, $i = 1, \ldots, 4$, closest to the origin, for $\lambda \in [-20, 20]$. All functions are analytic in the considered interval.*

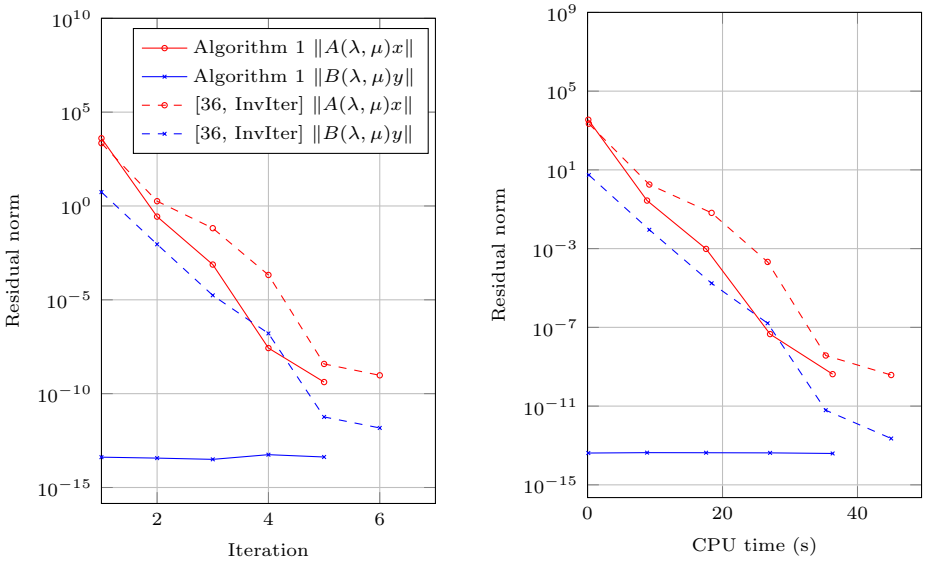


(a) Solutions λ (b) Solutions μ

FIG. 5.2. *Solutions corresponding to g_i where $i = 1, 2, 3$.*

origin, for the NEPs corresponding to the functions g_1, g_2, g_3, are given in Figure 5.2. The solutions are computed with the tensor infinite Arnoldi method. In theory, if the equivalence as described in Theorem 2.3 holds in the desired point, the solution of the two-parameter eigenvalue problem closest to the origin could be found by computing solutions for all the 20 NEPs corresponding to $g_1, \ldots, g_{20}$.

We illustrate our algorithms and compare with several other single-vector state-of-the-art algorithms in [36]. As starting values we use $\lambda_0 = 0.15 + 0.1i$ and $\mu_0 = 35 + 0.25i$ and a starting vector with an elementwise absolute error (from a nearby solution) less than 0.05. The iteration history of Algorithm 1, in terms of residual norm (with eigenvectors normalized with respect to the 2-norm), is given in Figure 5.3. We observe an asymptotic fast convergence for Algorithm 1, which is expected since the solution point is analytic and simple. The error is measured at step 3 in Algorithm 1 which implies that by construction, the error in the B-equation is (numerically) zero. This



(a) Residual norm vs. iteration

(b) Residual norm vs. CPU time

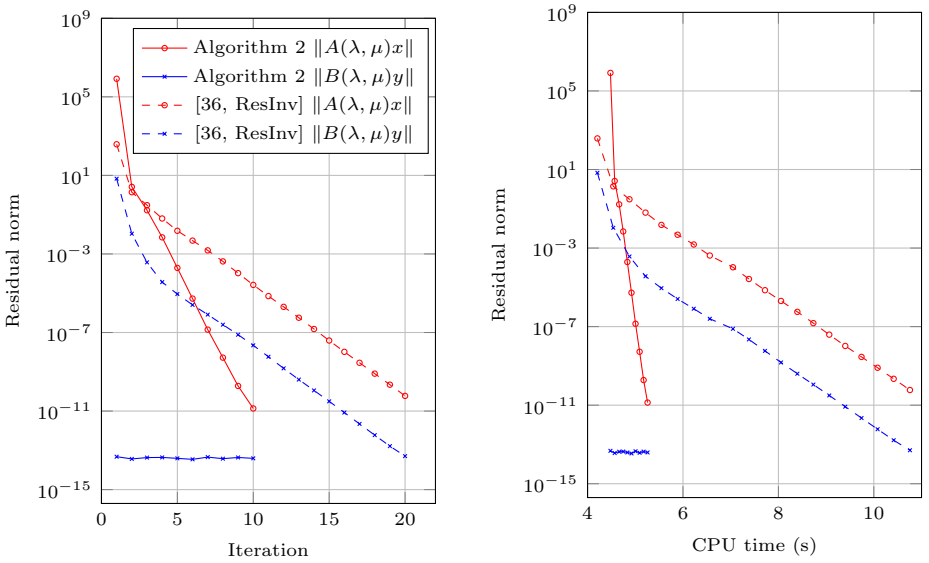
FIG. 5.3. *Visualization of the convergence of Algorithm* 1 *and* [36, Algorithm 1 (InvIter)] *for the problem in section* 5.1.

is a property of the elimination in our approach. We compare (with the same starting values) with the inverse iteration Newton approach proposed in [36]. Note that this method is designed for more general problems and not specifically our situation where $m \ll n$ and also multiparameter nonlinear problems.[5] In our implementation of [36, InvIter] we use two LU factorizations to handle the multiple linear solves per iteration. For the above example that was faster in terms of execution time compared to direct solves, although that might not be the case for a larger and sparse problem. The comparison between the two algorithms as a function of iteration is inconclusive, as can be seen in Figure 5.3(a). However, in terms of CPU time Algorithm 1 is somewhat faster, as can seen in Figure 5.3(b).

The convergence of our adaption of residual inverse iteration (Algorithm 2) initiated in the same way (except the starting vector is chosen as a vector of ones) is illustrated in Figure 5.4. We clearly see the expected linear convergence, since it is equivalent to residual inverse iteration for NEPs and the convergence theory in [34, section 3–4] is directly applicable. We compare with a proposed generalization of residual inverse iteration [36, InvIter], again noting that it has a much wider applicability domain than our approach. In this case, our method has a smaller convergence factor, intuitively motivated by the fact that we solve the B-equation exactly.

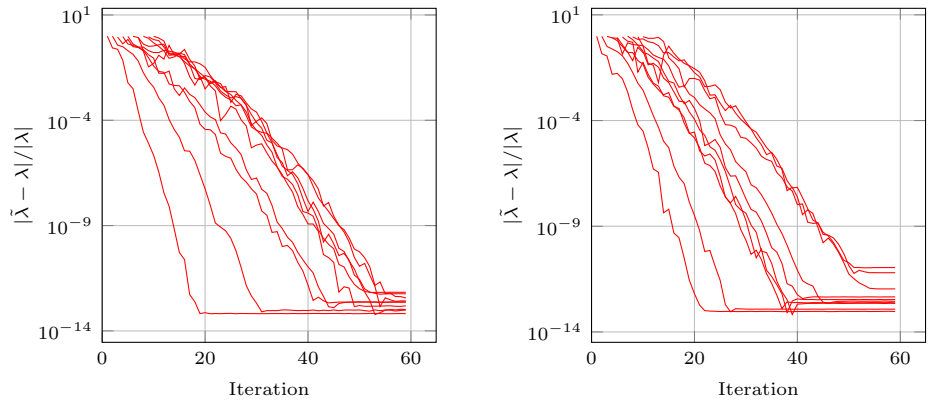
The problem can also be solved with the tensor infinite Arnoldi method [22]. More specifically, we use the implementation of the method available in the Julia package NEP-PACK [20] (version 1.0.2). By directly using Theorem 3.1 we can compute the 60 first derivatives. The convergence of the first ten eigenvalues are visualized in Figure 5.5 for two branches. The solutions are visualized in Figure 5.2.

[5]For multiparameter linear problems [36, InvIter] is equivalent to the tensor Rayleigh quotient iteration from [35, 16].



(a) Residual norm vs. iteration (b) Residual norm vs. CPU time

FIG. 5.4. *Visualization of the convergence of Algorithm* 2 *and* [36, Algorithm 2 (ResIter)] *for the problem in section* 5.1.



(a) Branch 1: $M(\lambda) = A_0 + \lambda A_1 + g_1(\lambda)A_2$ (b) Branch 2: $M(\lambda) = A_0 + \lambda A_1 + g_2(\lambda)A_2$

FIG. 5.5. *Visualization of convergence of the tensor infinite Arnoldi method for the problem in section* 5.1 *for* g_1 *and* g_2. *The error is measured as relative error where the computed value* λ *is compared to a precomputed reference solution* $\tilde{\lambda}$.

5.2. Domain decomposition example. We consider a BVP-eigenvalue problem, which we separate into two domains in such a way that it leads to a two-parameter eigenvalue problem. Similar techniques and analysis are found in, e.g., [9], [3, Chapter 2], and [19, Experiment 4], where it is common to force the solution to have roots within the considered interval.

Consider the Helmholtz eigenvalue problem defined in the domain $x \in [x_0, x_2]$,

$$u''(x) + \kappa^2(x)u(x) = \lambda u(x) \text{ for } x \in [x_0, x_2], \tag{5.1a}$$

$$u(x_0) = 0, \tag{5.1b}$$

$$u'(x_2) = 0 \tag{5.1c}$$

with a wavenumber κ which is discontinuous in one part of the domain and smooth in another, as in Figure 5.6. We take a point x_1 such that κ is smooth for $x > x_1$, assume that the solution is nonzero in the interface point $x = x_1$, and define

$$\mu := \frac{u'(x_1)}{u(x_1)}.$$

This means we have two separate BVPs for the two domains:

$$u_1''(x) + \kappa^2(x)u_1(x) = \lambda u_1(x), \quad x_0 \le x \le x_1, \tag{5.2a}$$

$$u_1(x_0) = 0, \tag{5.2b}$$

$$u_1'(x_1) - \mu u_1(x_1) = 0 \tag{5.2c}$$

and

$$u_2''(x) + \kappa^2(x)u_2(x) = \lambda u_2(x), \quad x_1 \le x \le x_2, \tag{5.3a}$$

$$u_2'(x_1) - \mu u_2(x_1) = 0, \tag{5.3b}$$

$$u_2'(x_2) = 0. \tag{5.3c}$$

These are standard linear BVPs (with Robin boundary conditions), and the uniqueness of these BVPs implies an equivalence with the original BVP (5.1). See [26] and references therein for domain decomposition methods for PDE eigenvalue problems.

The wavenumber is given as in Figure 5.6, i.e., it is discontinuous at several points in $[x_0, x_1]$ and with a high frequency decaying sine curve in $[x_1, x_2]$, representing an inhomogeneous periodic medium. We invoke different discretizations in the two domains, for the following reasons. Since κ is discontinuous in $[x_0, x_1]$, spectral discretization in that domain will not be considerably faster than a finite difference approximation. We therefore use a uniform second order finite difference for (5.2) to obtain sparse matrices and one sided second order finite different scheme for the

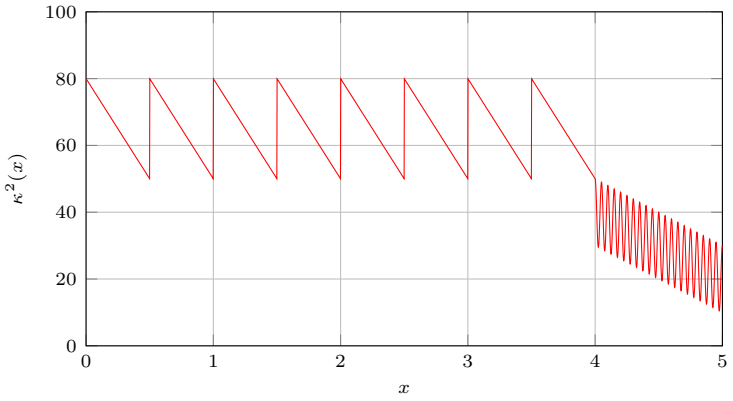


FIG. 5.6. *The wavenumber for the example in section* 5.2. *The wavenumber is sinusoidal with high frequency in the interval* $[4, 5]$, *and discontinuous in* $\frac{1}{2}, \frac{2}{2}, \frac{3}{2}, \ldots, \frac{7}{2}$.

boundary condition. A spectral discretization is used for $[x_1, x_2]$ where the wavenumber is smooth. Since μ appears linearly in the boundary condition, the discretization leads to a two-parameter eigenvalue problem of the type (1.1). In our setting A_1, A_2, A_3 are large and sparse, and B_1, B_2, B_3 are full matrices of smaller size. We use the discretization parameters such that $n = 10^6$ and $m = 30$ and $x_0 = 0$, $x_1 = 4$, and $x_2 = 5$. In order to make the measurement of error easier, we use left diagonal scaling of the problem such that the diagonal elements of $A(1.0, 1.0)$ and $B(1.0, 1.0)$ are equal to one.

The eigenvalues and some corresponding eigenfunctions are plotted in Figure 5.7 and Figure 5.8. In this one-dimensional case, the structure of the problem implies that B_3 is a rank-one matrix. Hence, the GEP (2.1) only has one finite solution. The nonlinear function g_1 of this problem is given in Figure 5.9. Clearly the function has

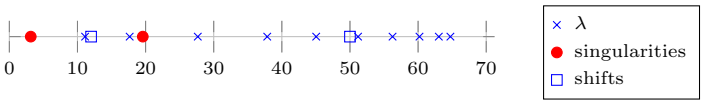


FIG. 5.7. *Computed eigenvalues, singularities, and the shifts used in the infinite Arnoldi method.*

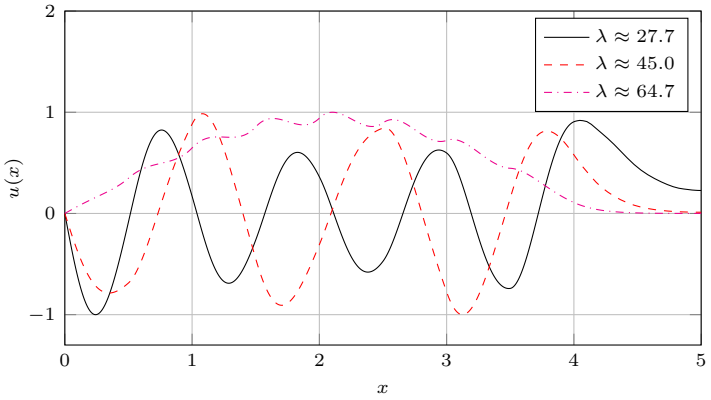


FIG. 5.8. *Some computed eigenfunctions of* (5.1)

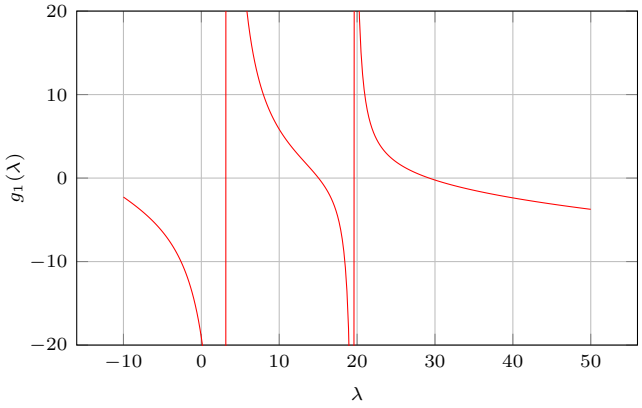
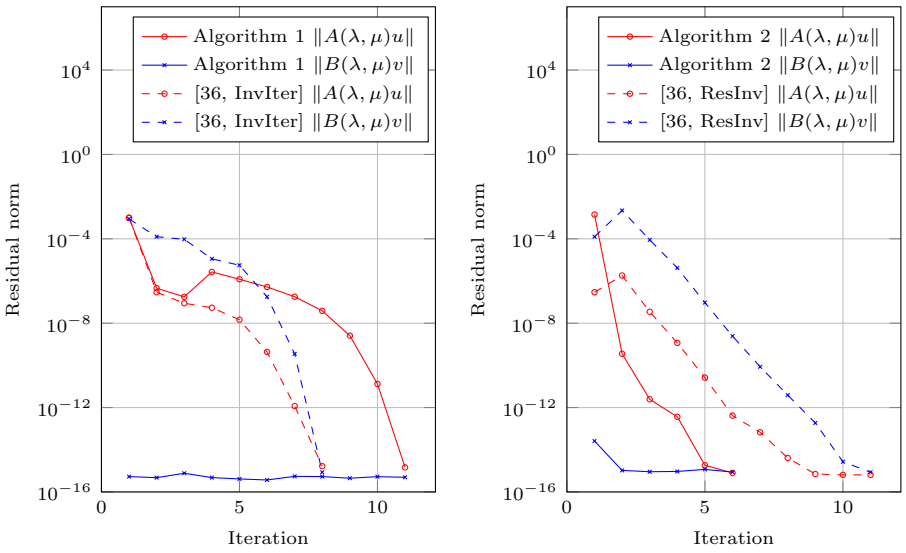
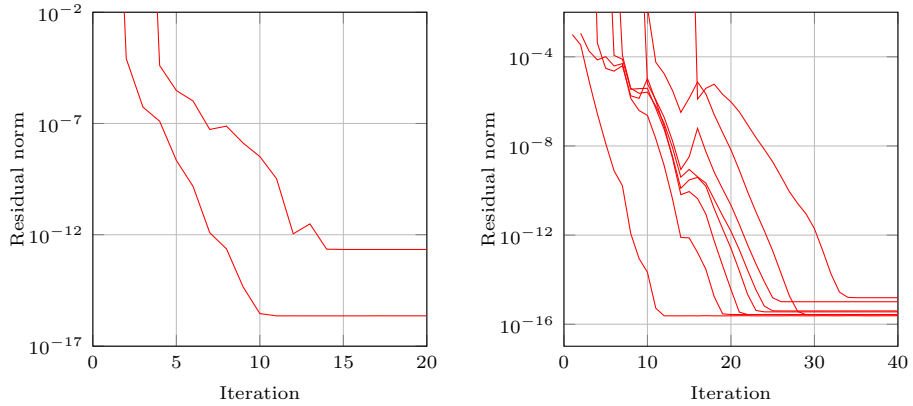


FIG. 5.9. *The nonlinear function g_1 in the simulation in section* 5.2.



(a) Newton-type algorithms (b) Fixed-shift methods

FIG. 5.10. *Visualization of the convergence of the proposed algorithms and two algorithms in* [36] *applied to the domain decomposition example in section* 5.2.



(a) Shift $\sigma = 12.0$ (b) Shift $\sigma = 50.0$

FIG. 5.11. *Convergence history for two different shifts.*

singularities for some real λ-values. The convergence of Algorithm 1 and Algorithm 2 is again compared to [36] in Figure 5.10. We again conclude that both our approaches are competitive, although not always faster in terms of iterations, but our approach is generally faster in terms of CPU time. We note that the closed-form solution of g_1 is not exploited in these simulations. The algorithms are initiated with approximate rounded eigenvectors and eigenvalues close to the eigenvalue $\lambda \approx 18$. We note that our methods do not require a starting value for μ (in contrast to [36]) which is an attractive feature from an application point of view, since the value $\mu = u'(x_1)/u(x_1)$ is an artificially introduced parameter and may not be easy to estimate.

We apply the tensor infinite Arnoldi method also for this problem. Since this family of methods is based on a power series expansion, one can only expect to be able to compute eigenvalues on the same side of the singularities as the shift. We therefore run the algorithm several times for different shifts and select the shifts far away from the singularities, as described in Figure 5.7. The convergence of the two runs is illustrated in Figure 5.11. Note that the convergence corresponding to one eigenvalue for the shift $\sigma = 12$ stagnates. This is because the eigenvalue is close to the singularity, and therefore difficult to compute, as can be seen in Figure 5.7.

6. Conclusions and outlook. We have presented a general framework to approach two-parameter eigenvalue problems, by nonlinearization to NEPs. Several steps in this technique seem to be generalizable (but beyond the scope of the paper), e.g., to general multiparameter eigenvalue problems essentially by successive application of the elimination. One such elimination leads to a nonlinear two-parameter eigenvalue problem as considered, e.g., in [36].

Our paper uses the assumption $m \ll n$ and that A_1, A_2, and A_3 are large and sparse. We made this assumption mostly for convenience, since this allows us to apply a general purpose method for the parameterized eigenvalue problem (2.1). If, on the other hand, we wish to solve two-parameter eigenvalue problems where these assumptions are not satisfied, the ideas may still be useful. The GEP (2.1) may, for instance, be approached with structured algorithms (exploiting sparsity, low-rank properties, and symmetry) or iterative methods for the GEP, where early termination is coupled with the NEP solver.

The generated nonlinear functions g_i are algebraic functions and can therefore contain singularities (e.g., branch-point singularities as characterized in section 2). These can be problematic in the numerical method, and therefore it would be useful with transformations that remove singularities. Linearization which does not lead to singularities has been established for rational eigenvalue problems [42].

The problem in section 5.2 is such that we obtain one large and sparse parameterized matrix $A(\lambda, \mu)$ which is coupled with a small and dense system. The setting matches the assumptions of the paper and is a representative example of cases where the behavior is different in the two physical domains. The example may be generalizable to other coupled physical systems where the modeling in one domain leads to a much smaller matrix, e.g., using domain decomposition with more physical dimensions. Note, however, that the presented methods seem mostly computationally attractive if the discretization of one domain is much smaller. If we apply the same technique to domains of equal size, other generic two-parameter eigenvalue methods (such as those in [36]) may be more effective.

Acknowledgment. We thank Olof Runborg, KTH Royal Institute of Technology, for discussions regarding the Helmholtz eigenvalue problem.

REFERENCES

[1] R. Alam and S. Safique Ahmad, *Sensitivity analysis of nonlinear eigenproblems*, SIAM J. Matrix Anal. Appl., 40 (2019), pp. 672–695.

[2] F. Atkinson, *Multiparameter Eigenvalue Problems. Volume* I*: Matrices and Compact Operators*, Academic Press, New York, 1972.

[3] F. V. Atkinson and A. B. Mingarelli, *Multiparameter Eigenvalue Problems: Sturm-Liouville Theory*, CRC Press, Boca Raton, FL, 2011.

[4] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder, and F. Tisseur, *NLEVP: A collection of nonlinear eigenvalue problems*, ACM Trans. Math. Software, 39 (2013), pp. 7:1–7:28.
[5] T. Betcke and H. Voss, *A Jacobi–Davidson type projection method for nonlinear eigenvalue problems*, Future Generation Comput. Syst., 20 (2004), pp. 363–372.
[6] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, *Julia: A fresh approach to numerical computing*, SIAM Rev., 59 (2017), pp. 65–98.
[7] C. Effenberger, *Robust successive computation of eigenpairs for nonlinear eigenvalue problems*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 1231–1256.
[8] K. Fritzsche and H. Grauert, *From Holomorphic Functions to Complex Manifolds*, Grad. Texts. Math. 213, Springer, New York, 2002.
[9] M. Greguš, F. Neuman, and F. M. Arscott, *Three-point boundary value problems in differential equations*, J. Lond. Math. Soc. (2), 2 (1971), pp. 429–436.
[10] S. Güttel and F. Tisseur, *The nonlinear eigenvalue problem*, Acta Numer., 26 (2017), pp. 1–94.
[11] T. Helgaker, P. Jørgensen, and J. Olsen, *Molecular Electronic-Structure Theory*, Wiley, Hobokan, NJ, 2000.
[12] V. Hernandez, J. E. Roman, and V. Vidal, *SLEPc: Scalable library for eigenvalue problem computations*, in High performance computing for computational science - VEPCAR 2002, Lect. Notes Comput. Sci. 2565, Springer, Berlin, 2003 pp. 377–391.
[13] V. Hernandez, J. E. Roman, and V. Vidal, *SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems*, ACM Trans. Math. Software, 31 (2005), pp. 351–362.
[14] D. J. Higham and N. J. Higham, *Structured backward error and condition of generalized eigenvalue problems*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 493–512.
[15] M. E. Hochstenbach, T. Košir, and B. Plestenjak, *A Jacobi–Davidson type method for the two-parameter eigenvalue problem*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 477–497.
[16] M. E. Hochstenbach, K. Meerbergen, E. Mengi, and B. Plestenjak, *Subspace methods for three-parameter eigenvalue problems*, Numer. Linear Algebra Appl., (2019), e2240.
[17] M. E. Hochstenbach and B. Plestenjak, *A Jacobi–Davidson type method for a right definite two-parameter eigenvalue problem*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 392–410.
[18] M. E. Hochstenbach and B. Plestenjak, *Backward error, condition numbers, and pseudospectra for the multiparameter eigenvalue problem*, Linear Algebra Appl., 375 (2003), pp. 63–81.
[19] M. E. Hochstenbach and B. Plestenjak, *Computing several eigenvalues of nonlinear eigenvalue problems by selection*, Calcolo, 57 (2020), pp. 1–25.
[20] E. Jarlebring, M. Bennedich, G. Mele, E. Ringh, and P. Upadhyaya, *NEP-PACK: A Julia Package for Nonlinear Eigenproblems*, 2018, https://github.com/nep-pack.
[21] E. Jarlebring, S. Kvaal, and W. Michiels, *Computing all pairs (λ, μ) such that λ is a double eigenvalue of $A + \mu B$.*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 902–927.
[22] E. Jarlebring, G. Mele, and O. Runborg, *The waveguide eigenvalue problem and the tensor infinite Arnoldi method*, SIAM J. Sci. Comput., 39 (2017), pp. A1062–A1088.
[23] E. Jarlebring, W. Michiels, and K. Meerbergen, *A linear eigenvalue algorithm for the nonlinear eigenvalue problem*, Numer. Math., 122 (2012), pp. 169–195.
[24] T. Košir, *Finite-dimensional multiparameter spectral theory: The nonderogatory case*, Linear Algebra Appl., 212 (1994), pp. 45–70.
[25] S. Kvaal, E. Jarlebring, and W. Michiels, *Computing singularities of perturbation series*, Phys. Rev. A, 83 (2011), 032505.
[26] S. H. Lui, *Domain decomposition methods for eigenvalue problems*, J. Comput. Appl. Math., 117 (2000), pp. 17–34.
[27] S. Mackey, N. Mackey, C. Mehl, and V. Mehrmann, *Structured polynomial eigenvalue problems: Good vibrations from good linearizations*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 1029–1051.
[28] S. Mackey, N. Mackey, C. Mehl, and V. Mehrmann, *Vector spaces of linearizations for matrix polynomials*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 971–1004.
[29] K. Meerbergen and B. Plestenjak, *A Sylvester–Arnoldi type method for the generalized eigenvalue problem with two-by-two operator determinants*, Numer. Linear Algebra Appl., 22 (2015), pp. 1131–1146.
[30] V. Mehrmann and H. Voss, *Nonlinear eigenvalue problems: A challenge for modern eigenvalue methods*, GAMM-Mitt., 27 (2004), pp. 121–152.
[31] G. Mele, *The Infinite Lanczos Method for Symmetric Nonlinear Eigenvalue Problems*, tech. report, KTH Royal Institute of Technology, 2018.
[32] G. Mele and E. Jarlebring, *On restarting the tensor infinite Arnoldi method*, BIT, 58 (2018), pp. 133–162.

[33] Y. Nakatsukasa, V. Noferini, and A. Townsend, *Vector spaces of linearizations for matrix polynomials: A bivariate polynomial approach*, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 1–29.
[34] A. Neumaier, *Residual inverse iteration for the nonlinear eigenvalue problem*, SIAM J. Numer. Anal., 22 (1985), pp. 914–923.
[35] B. Plestenjak, *A continuation method for a right definite two-parameter eigenvalue problem*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1163–1184.
[36] B. Plestenjak, *Numerical methods for nonlinear two-parameter eigenvalue problems*, BIT, 56 (2016), pp. 241–262.
[37] J. E. Roman, C. Campos, E. Romero, and A. Tomas, *SLEPc Users Manual*, tech. report DSIC-II/24/02 - Revision 3.10, D. Sistemes Informàtics i Computació, Universitat Politècnica de València, 2018.
[38] A. Ruhe, *Algorithms for the nonlinear eigenvalue problem*, SIAM J. Numer. Anal., 10 (1973), pp. 674–689.
[39] M. Shao, *Conquering algebraic nonlinearity in nonlinear eigenvalue problems*, in Proceedings of the SIAM Conference Applied Linear Algebra, Hong Kong, Joint work with Z. Bai, W. Gao, and X. Huang, 2018.
[40] T. Slivnik and G. Tomšič, *A numerical method for the solution of two-parameter eigenvalue problems*, J. Comput. Appl. Math., 15 (1986), pp. 109–115.
[41] A. Spence and C. Poulton, *Photonic band structure calculations using nonlinear eigenvalue techniques*, J. Comput. Phys., 204 (2005), pp. 65–81.
[42] Y. Su and Z. Bai, *Solving rational eigenvalue problems via linearization*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 201–216.
[43] D. Szyld and F. Xue, *Preconditioned eigensolvers for large-scale nonlinear Hermitian eigenproblems with variational characterizations.* I. *Extreme eigenvalues*, Math. Comp., 85 (2016), pp. 2887–2918.
[44] J. Tausch and J. Butler, *Floquet multipliers of periodic waveguides via Dirichlet-to-Neumann maps*, J. Comput. Phys., 159 (2000), pp. 90–102.
[45] F. Tisseur and K. Meerbergen, *The quadratic eigenvalue problem*, SIAM Rev., 43 (2001), pp. 235–286.
[46] G. Unger, *Convergence orders of iterative methods for nonlinear eigenvalue problems*, in Advanced Finite Element Methods and Applications, Springer, New York, 2013, pp. 217–237.
[47] H. Unger, *Nichtlineare Behandlung von Eigenwertaufgaben*, Z. Angew. Math. Mech., 30 (1950), pp. 281–282. English translation, http://www.math.tu-dresden.de/~schwetli/Unger.html.
[48] R. Van Beeumen, O. Marques, E. G. Ng, C. Yang, Z. Bai, L. Ge, O. Kononenko, Z. Li, C.-K. Ng, and L. Xiao, *Computing resonant modes of accelerator cavities by solving nonlinear eigenvalue problems via rational approximation*, J. Comput. Phys., 374 (2018), pp. 1031–1043.
[49] H. Voss, *An Arnoldi method for nonlinear eigenvalue problems*, BIT, 44 (2004), pp. 387–401.
[50] H. Voss, *Chapter. Nonlinear eigenvalue problems* in Handbook in Linear Algebra, CRC Press, Boca Raton, FL, 2012.
[51] F. Xue, *A block preconditioned harmonic projection method for large-scale nonlinear eigenvalue problems*, SIAM J. Sci. Comput., 40 (2018), pp. A1809–A1835.