A Spectral Newton-Schur Algorithm for the Solution of Symmetric Generalized Eigenvalue Problems

Vassilis Kalantzis

February 2020

EPrint ID: 2020.1

IBM Research
Thomas J. Watson Research Center

Electronic Transactions on Numerical Analysis.
Volume 52, pp. 132–153, 2020.
Copyright © 2020, Kent State University.
ISSN 1068–9613.
DOI: 10.1553/etna_vol52s132

A SPECTRAL NEWTON-SCHUR ALGORITHM FOR THE SOLUTION OF SYMMETRIC GENERALIZED EIGENVALUE PROBLEMS[*]

VASSILIS KALANTZIS[†‡]

Abstract. This paper proposes a numerical algorithm based on spectral Schur complements to compute a few eigenvalues and the associated eigenvectors of symmetric matrix pencils. The proposed scheme follows an algebraic domain decomposition viewpoint and transforms the generalized eigenvalue problem into one of computing roots of scalar functions. These scalar functions are defined so that their roots are equal to the eigenvalues of the original pencil, and these roots are computed by Newton's method. We describe the theoretical aspects of the proposed scheme and demonstrate its performance on a few test problems.

Key words. spectral Schur complements, domain decomposition, symmetric generalized eigenvalue problem, Newton's method

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

1. Introduction.

This paper considers the computation of a few eigenvalue-eigenvector pairs (eigenpairs) of eigenvalue problems of the form $Ax = \lambda Mx$, where the matrices A and M are assumed sparse and real symmetric and M is also positive-definite (SPD). Throughout the rest of this paper we will use the notation (A, M) to refer to the eigenvalue problem with the pencil $A - \lambda M$.

The numerical technique presented in this paper combines domain decomposition with Spectral Schur Complements (SSC) [4, 5, 9, 10, 11, 12, 15, 17, 18, 20]. In particular, we assume that the adjacency graph of the given pencil is already partitioned into a set of non-overlapping subdomains. The eigenvalue problem $Ax = \lambda Mx$ can then be split into two disjoint parts: a) one associated with interior variables of each subdomain and b) one associated with the corresponding interface variables. Once the part of the solution associated with interface variables is obtained, the corresponding part associated with interior variables is computed independently in each subdomain [23, 25]. The restriction of the original eigenvalue problem onto the region defined by the interface variables is also an eigenvalue problem, albeit of a nonlinear nature. The dimension of this interface eigenvalue problem is equal to the total number of interface variables of the discretized domain.

The algorithm proposed in this paper recasts the interface eigenvalue problem into one of computing roots of scalar functions. These scalar functions are chosen so that their roots are equal to the eigenvalues of the pencil (A, M). The root-finding problem is then solved by Newton's method. To define the scalar functions of interest, we consider a parameterization of the eigenvalues of a first-order approximation of the nonlinear matrix-valued function associated with interface variables. In contrast to schemes which extract eigenpairs of the pencil (A, M) via Rayleigh-Ritz projections onto low-dimensional subspaces, e.g., Krylov subspace methods [21] or (generalized) Davidson subspace methods [24], the proposed algorithm is based on single-vector iterations, and each sought eigenpair is computed independently of each other.

The method proposed in this paper shares similarities with the technique presented in [11], where the concept of SSC root-finding approaches was first introduced. However, there are some notable differences. First, the technique presented in [11] considers only standard

[*] Received January 4, 2019. Accepted November 12, 2019. Published online on February 11, 2020. Recommended by Qiang Ye.

[†] Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, USA (kalan019@umn.edu).

[‡] IBM Research, Thomas J. Watson Research Center, Yorktown Heights, NY 10598, USA (vkal@ibm.com).

eigenvalue problems while this paper focuses explicitly on generalized eigenvalue problems. Second, the root-finding approach presented in [11] is applied to a scalar function defined by a zeroth-order approximation of the nonlinear matrix-valued function associated with the interface variables. In contrast, in this paper we exploit a first-order approximation instead. This choice requires a separate theoretical analysis while it also leads to faster convergence of the root-finding scheme.

The structure of this paper is as follows: Section 2 presents the general framework of Schur complement eigenvalue solvers. Section 3 discusses the proposed approach and its associated theoretical details. Section 4 discusses practical details. Section 5 presents numerical experiments performed on a set of pencils and compares the scheme proposed in this paper against the root-finding scheme presented in [11]. Extensions to the computation of additional eigenpairs are also provided. Finally, Section 6 contains our concluding remarks.

2. The Schur complement viewpoint.

Let A and M be partitioned in the following 2×2 block form:

$$A = \begin{bmatrix} B & E \\ E^T & C \end{bmatrix}, \qquad M = \begin{bmatrix} M_B & M_E \\ M_E^T & M_C \end{bmatrix}, \tag{2.1}$$

where B and M_B are square matrices of size $d \times d$, E and M_E are rectangular matrices of size $d \times s$, C and M_C are square matrices of size $s \times s$, and $n = d + s$.

DEFINITION 2.1. *For any matrix pencil* (L, C) *we define*

$$\Lambda(L, C) := \{\lambda \mid \det [L - \lambda C] = 0\},$$

where the det *operator denotes the determinant of the input matrix.*

Let λ be some eigenvalue of the pencil (A, M). Similarly to (2.1) we consider the following 2×1 vector partitioning of the corresponding eigenvector x:

$$x = \begin{bmatrix} u \\ y \end{bmatrix},$$

where $u \in \mathbb{R}^d$ and $y \in \mathbb{R}^s$. We can rewrite $(A - \lambda M)x = 0$ as

$$\begin{aligned} (B - \lambda M_B)u + (E - \lambda M_E)y &= 0, \\ (E - \lambda M_E)^T u + (C - \lambda M_C)y &= 0, \end{aligned} \tag{2.2}$$

where substituting

$$u = (B - \lambda M_B)^{-1}(E - \lambda M_E)y \tag{2.3}$$

in the second equation in (2.2) leads to the nonlinear eigenvalue problem

$$S(\lambda)y = 0, \tag{2.4}$$

where $S(\sigma) : \mathbb{R} \to \mathbb{R}^{s \times s}$ is a univariate, non-linear matrix-valued function defined as

$$S(\sigma) = C - \sigma M_C - (E - \sigma M_E)^T (B - \sigma M_B)^{-1}(E - \sigma M_E).$$

Combining (2.3) and (2.4) shows that an eigenpair (λ, x) of the pencil (A, M) can be computed by the following procedure:

1. Compute a scalar-vector pair (λ, y) that satisfies (2.4).
2. Solve the linear system $(B - \lambda M_B)u = -(E - \lambda M_E)y$ to retrieve u.

Unless mentioned otherwise, throughout the rest of this paper we will assume that:

(a) the sought eigenvalues λ are simple, and

(b) $\lambda \notin \Lambda(B, M_B)$.

Extensions to the cases where λ is either a multiple eigenvalue of the pencil (A, M) or an eigenvalue of the pencil (B, M_B) (or both) are possible, and we will comment where deemed necessary.

3. The method of Mixed Linear Approximations.

Consider the following $s \times s$ parameterized (symmetric) generalized eigenvalue problem stemming from a first-order approximation of $S(\lambda)$ around a real scalar $\sigma \in \mathbb{R}$:

$$S(\sigma)\hat{y}_j(\sigma) = -\theta_j(\sigma)S'(\sigma)\hat{y}_j(\sigma),$$

where $S'(\sigma)$ denotes the first derivative of $S(\sigma)$, $\theta_j(\sigma)$ denotes the jth eigenvalue of the pencil $(S(\sigma), -S'(\sigma))$, and $\hat{y}_j(\sigma)$ denotes the corresponding eigenvector.

DEFINITION 3.1. *For any $\sigma \in \mathbb{R}$, we define the following matrix-valued functions:*

$$\begin{aligned} B_\sigma &= B - \sigma M_B, \\ E_\sigma &= E - \sigma M_E, \\ C_\sigma &= C - \sigma M_C. \end{aligned}$$

PROPOSITION 3.2. *For any $\sigma \in \mathbb{R}$, the first derivative of $S(\sigma)$, $S'(\sigma)$, is given by the formula*

$$S'(\sigma) = \frac{dS(\sigma)}{d\sigma} = -M_C - E_\sigma^T B_\sigma^{-1} M_B B_\sigma^{-1} E_\sigma + E_\sigma^T B_\sigma^{-1} M_E + M_E^T B_\sigma^{-1} E_\sigma. \tag{3.1}$$

Moreover, $-S'(\sigma)$ (when defined) is SPD.

Proof. Let $\omega \in \mathbb{R}$. We can write

$$\begin{aligned} S(\sigma+\omega) &= C_\sigma - \omega M_C - (E_\sigma - \omega M_E)^T (B_\sigma - \omega M_B)^{-1} (E_\sigma - \omega M_E) \\ &= C_{\sigma+\omega} - E_\sigma^T B_{\sigma+\omega}^{-1} E_\sigma + \omega\left(E_\sigma^T B_{\sigma+\omega}^{-1} M_E + M_E^T B_{\sigma+\omega}^{-1} E_\sigma - \omega M_E^T B_{\sigma+\omega}^{-1} M_E\right). \end{aligned}$$

Taking the derivative of $S(\sigma+\omega)$ with respect to ω gives

$$S'(\sigma+\omega) = -M_C + E_\sigma^T B_{\sigma+\omega}^{-1} M_E + M_E^T B_{\sigma+\omega}^{-1} E_\sigma - E_\sigma^T \left[B_{\sigma+\omega}^{-1}\right]' E_\sigma.$$

The result in (3.1) follows by setting $S'(\sigma) = S'(\sigma+\omega)_{\omega=0}$ and noticing that

$$\left[B_{\sigma+\omega}^{-1}\right]'_{\omega=0} = B_\sigma^{-1} M_B B_\sigma^{-1}.$$

To prove the second item, let the matrix $U_\sigma \in \mathbb{R}^{n\times n}$ be defined as

$$U_\sigma = \begin{bmatrix} I & -B_\sigma^{-1}E_\sigma \\ 0 & I \end{bmatrix}.$$

Forming $U_\sigma^T M U_\sigma$ explicitly gives

$$\begin{aligned} U_\sigma^T M U_\sigma &= \begin{bmatrix} I & 0 \\ E_\sigma^T B_\sigma^{-1} & I \end{bmatrix} \begin{bmatrix} M_B & M_E \\ M_E^T & M_C \end{bmatrix} \begin{bmatrix} I & B_\sigma^{-1}E_\sigma \\ 0 & I \end{bmatrix} \\ &= \begin{bmatrix} M_B & M_E - M_B B_\sigma^{-1} E_\sigma \\ M_E^T - E_\sigma^T B_\sigma^{-1} M_B & -S'(\sigma) \end{bmatrix}. \end{aligned}$$

Recalling that M is SPD (and thus $U_\sigma^T M U_\sigma$ is also SPD) concludes the proof. □

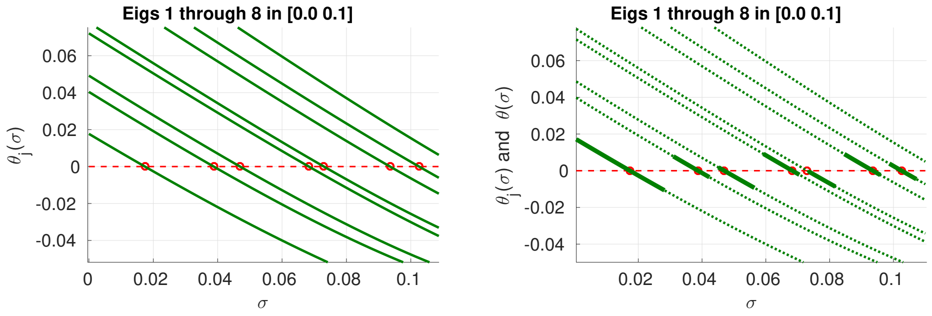

FIG. 3.1. *Visualization of $\theta_j(\sigma)$ and $\theta(\sigma) \equiv \theta_{\kappa_\sigma}(\sigma)$. The red circles along the real axis denote eigenvalues of the pencil (A, M). Left: eigenbranches $\theta_j(\sigma),\ j = 1, \ldots, 8$. Right: eigenbranches $\theta_j(\sigma),\ j = 1, \ldots, 8$, (dotted lines) and $\theta_{\kappa_\sigma}(\sigma) \equiv \theta(\sigma)$ (solid line). Each eigenvalue of (A, M) is a root of (at least) one eigenbranch, and thus a root of $\theta(\sigma)$ as well.*

3.1. Eigenbranches.

DEFINITION 3.3. *The scalar function $\theta_j(\sigma) : \mathbb{R} \to \mathbb{R}$,*

$$\theta_j(\sigma) = -\frac{\hat{y}_j^T(\sigma) S(\sigma) \hat{y}_j(\sigma)}{\hat{y}_j^T(\sigma) S'(\sigma) \hat{y}_j(\sigma)} \tag{3.2}$$

will be referred to as the jth eigenbranch *of $(S(\sigma), -S'(\sigma)),\ j = 1, \ldots, s$. Consider now the following integer:*

$$\kappa_\sigma := \arg\min_{1 \le j \le s} |\theta_j(\sigma)|. \tag{3.3}$$

We then define the functions $\theta(\sigma) : \mathbb{R} \to \mathbb{R}$ and $\hat{y}(\sigma) : \mathbb{R} \to \mathbb{R}^s$ as

$$\theta(\sigma) := \theta_{\kappa_\sigma}(\sigma),$$
$$\hat{y}(\sigma) := \hat{y}_{\kappa_\sigma}(\sigma),$$

i.e., $(\theta(\sigma), \hat{y}(\sigma))$ denotes the eigenpair associated with the eigenvalue of smallest magnitude of the pencil $(S(\sigma), -S'(\sigma))$.

Figure 3.1 visualizes the first few eigenbranches of a discretized 2D Laplacian operator. The discussion in Section 2 tells us that a scalar $\zeta \notin \Lambda(B, M_B)$ is an eigenvalue of the pencil (A, M) if and only if there exists an index $1 \le j \le s$ such that $\theta_j(\zeta) = 0$.[1] We can see that each eigenvalue of the pencil (A, M) is a root of (at least) one eigenbranch $\theta_j(\sigma),\ 1 \le j \le s$, and thus also a root of $\theta(\sigma)$. Thus, the problem of computing an eigenpair (λ, x) of the pencil (A, M) is equivalent to that of computing the root λ of the scalar function $\theta(\sigma)$.

The rest of this section presents a root-finding approach to compute the eigenpair (λ, x) of the pencil (A, M) under the assumption that some user-given initial approximation of λ is available.

3.2. Formulation of a Newton-based procedure. In this paper we consider Newton's method as our root-finding algorithm [14]. While $\theta(\sigma)$ is not differentiable across its entire

[1] This does not exclude the possibility that $\theta_j(\zeta)$ has roots which are not eigenvalues of the pencil (A, M). Indeed, as we show later on, each eigenvalue of the pencil (B, M_B) is typically a root of some $\theta_j(\sigma)$.

domain of definition, Newton's method can still be applied if the integer κ_σ defined in (3.3) remains fixed. Thus, it suffices to show analyticity of the eigenbranches.

PROPOSITION 3.4. *The eigenbranches* $\theta_j(\sigma),\ j=1,\dots,s$, *are analytic functions of* $\sigma\in\mathbb{R}$. *The derivative of each eigenbranch is given by the formula*

$$\theta_j'(\sigma)=\frac{d\theta_j(\sigma)}{d\sigma}=-\frac{\hat{y}_j^T(\sigma)\left[S'(\sigma)+\theta_j(\sigma)S''(\sigma)\right]\hat{y}_j(\sigma)}{\hat{y}_j^T(\sigma)S'(\sigma)\hat{y}_j(\sigma)}, \tag{3.4}$$

where

$$\begin{aligned}S''(\sigma)=&-2E_\sigma^TB_\sigma^{-1}M_BB_\sigma^{-1}M_BB_\sigma^{-1}E_\sigma-2M_E^TB_\sigma^{-1}M_E+2M_E^TB_\sigma^{-1}M_BB_\sigma^{-1}E_\sigma\\&+2E_\sigma^TB_\sigma^{-1}M_BB_\sigma^{-1}M_E\end{aligned}$$

denotes the second derivative of the matrix-valued function $S(\sigma)$.

Proof. Differentiating $S(\sigma)\hat{y}_j(\sigma)=-\theta_j(\sigma)S'(\sigma)\hat{y}_j(\sigma)$ with respect to σ leads to

$$S'(\sigma)\hat{y}_j(\sigma)+S(\sigma)\hat{y}_j'(\sigma)=-\theta_j(\sigma)\left[S''(\sigma)\hat{y}_j(\sigma)+S'(\sigma)\hat{y}_j'(\sigma)\right]-\theta_j'(\sigma)S'(\sigma)\hat{y}_j(\sigma),$$

where $\hat{y}_j'(\sigma)=\dfrac{d\hat{y}_j(\sigma)}{d\sigma}$. After collecting terms we get

$$\left[S(\sigma)+\theta_j(\sigma)S'(\sigma)\right]\hat{y}_j'(\sigma)=-\left[S'(\sigma)+\theta_j(\sigma)S''(\sigma)\right]\hat{y}_j(\sigma)-\theta_j'(\sigma)S'(\sigma)\hat{y}_j(\sigma).$$

Multiplying by $\hat{y}_j^T(\sigma)$ from the left and noticing that $\hat{y}_j^T(\sigma)\left[S(\sigma)+\theta_j(\sigma)S'(\sigma)\right]=0$ leads to (3.4), which can be further simplified to

$$\begin{aligned}\theta_j'(\sigma)&=-1-\theta_j(\sigma)\frac{\hat{y}_j^T(\sigma)S''(\sigma)\hat{y}_j(\sigma)}{\hat{y}_j^T(\sigma)S'(\sigma)\hat{y}_j(\sigma)},\\&=-1+\theta_j(\sigma)\hat{y}_j^T(\sigma)S''(\sigma)\hat{y}_j(\sigma)\quad\left(\text{if } \hat{y}_j^T(\sigma)S'(\sigma)\hat{y}_j(\sigma)=-1\right).\end{aligned}$$

Differentiating $S(\sigma+\omega)$ twice leads to

$$S''(\sigma+\omega)=-E_\sigma^T\left[B_{\sigma+\omega}^{-1}\right]''E_\sigma-2\left[M_E^T\left(B_{\sigma+\omega}^{-1}M_E-\left[B_{\sigma+\omega}^{-1}\right]'E_\sigma\right)-E_\sigma^T\left[B_{\sigma+\omega}^{-1}\right]'M_E\right].$$

The expression of the second derivative then follows by setting $S''(\sigma)=S''(\sigma+\omega)_{\omega=0}$ and noticing that in addition to $\left[B_{\sigma+\omega}^{-1}\right]'_{\omega=0}=B_\sigma^{-1}M_BB_\sigma^{-1}$, we also have

$$\left[B_{\sigma+\omega}^{-1}\right]''_{\omega=0}=2B_\sigma^{-1}M_BB_\sigma^{-1}M_BB_\sigma^{-1}.$$

Regarding the analyticity of the eigenbranches, recall that for any $\sigma\notin\Lambda(B,M_B)$ both $S(\sigma)$ and $-S'(\sigma)$ are analytic matrix-functions while $-S'(\sigma)$ is also SPD. Both $\theta_j(\sigma)$ and $\hat{y}_j(\sigma)$ then are analytic functions of σ with these results also holding for multiple (semi-simple) eigenvalues of the pencil $(S(\sigma),-S'(\sigma))$; see [2]. When $\sigma\in\Lambda(B,M_B)$, the pencil $(S(\sigma),-S'(\sigma))$ is not formally defined but the eigenpairs are, and the above results still apply.
□

REMARK 3.5. When $M=I$, we have $M_B=M_C=I,\ M_E=0$, and the two leading derivatives of the matrix-valued function $S(\sigma)$ are equal to $S'(\sigma)=-I-E^T(B-\sigma I)^{-2}E$ and $S''(\sigma)=-2E^T(B-\sigma I)^{-3}E$, respectively.

Algorithm 1 MLA.

```
//- - $S(\sigma) \ = C_\sigma - E_\sigma^T B_\sigma^{-1} E_\sigma$ - -//
0.  Given tol $\in \mathbb{R}$ and an initial eigenvalue approximation $\sigma \in \mathbb{R}$.
1.  Do while (true):
2.      Solve $S(\sigma)\hat{y}(\sigma) + \theta(\sigma)S'(\sigma)\hat{y}(\sigma) = 0$ for the eigenvalue $\theta(\sigma)$
        of smallest magnitude and the associated eigenvector $\hat{y}(\sigma)$.
3.      If $|\theta(\sigma)| \le$ tol; Break; EndIf;
4.      Compute $\theta'(\sigma) = -1 - \theta(\sigma)\dfrac{\hat{y}^T(\sigma)S''(\sigma)\hat{y}(\sigma)}{\hat{y}^T(\sigma)S'(\sigma)\hat{y}(\sigma)}$.
5.      Update $\sigma = \sigma - \dfrac{\theta(\sigma)}{\theta'(\sigma)}$.
6.  EndDo
7.  Return $\sigma$ and $\hat{x}(\sigma) = \left[-(B_\sigma^{-1}E_\sigma\hat{y}(\sigma))^T, \hat{y}^T(\sigma)\right]^T$.
```

3.2.1. The algorithm.

Algorithm 1 sketches the main steps of Newton's method applied to the scalar function $\theta(\sigma)$. For any initial value of σ, Algorithm 1 computes the eigenpair $(\theta(\sigma)$,$\hat{y}(\sigma))$ of the pencil $(S(\sigma), -S'(\sigma))$ and evaluates the derivative $\theta'(\sigma)$ to update σ. The procedure terminates as soon as $|\theta(\sigma)|$ becomes smaller than a threshold tolerance tol $\in \mathbb{R}$. Throughout the rest of this paper we will refer to Algorithm 1 as the method of Mixed Linear Approximations (MLA).

The convergence of MLA can be monitored by computing the residual norm of the approximate eigenpair $(\sigma, \hat{x}(\sigma))$, i.e., by computing the quantity $\|A\hat{x}(\sigma) - \sigma M\hat{x}(\sigma)\|$. Alternatively, we can set for a less accurate criterion and only monitor $|\theta(\sigma)|$ since

$$\|A\hat{x}(\sigma) - \sigma M\hat{x}(\sigma)\|_2 = \left\| \begin{bmatrix} B_\sigma & E_\sigma \\ E_\sigma^T & C_\sigma \end{bmatrix} \begin{bmatrix} -B_\sigma^{-1}E_\sigma\hat{y}(\sigma) \\ \hat{y}(\sigma) \end{bmatrix} \right\|_2 = \left\| \begin{bmatrix} 0 \\ -\theta(\sigma)S'(\sigma)\hat{y}(\sigma) \end{bmatrix} \right\|_2$$
$$\le |\theta(\sigma)| \ \|\hat{y}(\sigma)\|_2 \ \lambda_{\max}\left(-S'(\sigma)\right),$$

where $\lambda_{\max}(.)$ denotes the largest eigenvalue of $-S'(\sigma)$ and is bounded from above for any $\sigma \notin \Lambda(B, M_B)$. When λ is multiple (with multiplicity up to s), the bottom $s \times 1$ components of the associated eigenvectors can be captured by computing all eigenvectors associated with the (multiple) zero eigenvalue of the pencil $(S(\lambda), -S'(\lambda))$.

Since MLA is essentially Newton's method applied to the scalar function $\theta(\sigma)$, we expect the former to converge quadratically; at least if a sufficiently accurate initial approximation is provided [14]. In practice, the observed convergence rate is higher due to the fact that the pencil $(S(\sigma), -S'(\sigma))$ also becomes an increasingly better approximation of the pencil $(S(\lambda), -S'(\lambda))$. Moreover, as we show in Section 4.1, the shape of the eigenbranches can be very close to that of a linear function even when the initial value of σ is far from the sought eigenvalue λ, in which case we expect MLA to converge rapidly.

3.2.2. Computational cost.

Computing the eigenpair $(\theta(\sigma), \hat{y}(\sigma))$ in Step 2 represents the major computational procedure of MLA. We will refer to the eigenvalue solver used to perform Step 2 in MLA as the *inner eigenvalue solver*. By default, inverse iteration is chosen as our inner eigenvalue solver [8]. The main kernel then is the solution of linear systems of the form $S(\sigma)w = -S'(\sigma)b$, which in practice can be accomplished by a preconditioned Krylov subspace iterative solver.

The matrix $S(\sigma)$ does not have to be formed explicitly, i.e., we only need to be able to multiply a dense vector with the matrices C_σ and E_σ, as well as to solve a linear system with

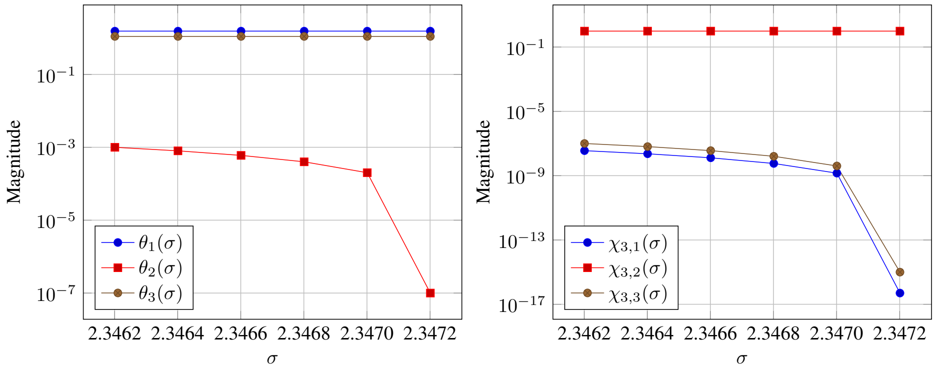

FIG. 3.2. *Magnitude of $\theta_j(\sigma)$ and $\chi_{3,j}(\sigma)$ as $\sigma \to \lambda_3$.*

B_σ. To make the above computations practical, we consider matrix partitionings such that both B_σ and E_σ are block-diagonal. Then, each time we update σ in MLA, we only need to factorize the block-diagonal matrix B_σ, and this can be done in parallel.

3.3. Characterization of the eigenbranches as $\boldsymbol{\sigma}$ approaches an eigenvalue of $(\boldsymbol{A}, \boldsymbol{M})$. We now consider the behavior of the eigenbranches as σ approaches a simple eigenvalue of (A, M).

THEOREM 3.6. *Let $\lambda_\kappa \notin \Lambda(B, M_B),\ 1 \leq \kappa \leq n$, denote a simple eigenvalue of the pencil (A, M) with an associated eigenvector $x^{(\kappa)} = \left[\left(u^{(\kappa)}\right)^T, \left(y^{(\kappa)}\right)^T\right]^T$. Then,*

$$\operatorname{span}\left\{\lim_{\sigma \to \lambda_\kappa} \hat{y}(\sigma)\right\} \equiv \operatorname{span}\left\{\hat{y}(\lambda_\kappa)\right\} \equiv \operatorname{span}\left\{y^{(\kappa)}\right\}.$$

Moreover, for any $j \neq \lim_{\sigma \to \lambda_\kappa} \kappa_\sigma$, we have

$$\lim_{\sigma \to \lambda_\kappa} \theta_j(\sigma) = \lim_{\sigma \to \lambda_\kappa} \left(\frac{\hat{y}_j^T(\sigma)\hat{y}_j(\sigma)}{\sum_{i=1,\ i\neq\kappa}^{n} \dfrac{\left(\hat{y}_j^T(\sigma)y^{(i)}\right)\left(\hat{y}_j^T(\sigma)S'(\sigma)y^{(i)}\right)}{\lambda_i - \sigma}} \right), \tag{3.5}$$

where $1 \leq \kappa_\sigma \leq s$ denotes the index of the eigenvalue of smallest magnitude of the pencil $(S(\sigma), -S'(\sigma))$, i.e., $(\theta_{\kappa_\sigma}(\sigma), \hat{y}_{\kappa_\sigma}(\sigma)) \equiv (\theta(\sigma), \hat{y}(\sigma))$.

Theorem 3.6 suggests that as σ converges towards the sought (simple) eigenvalue of (A, M), a shift-and-invert-based inner eigenvalue solver will require fewer iterations to compute $(\theta(\sigma), \hat{y}(\sigma))$. We will verify this statement experimentally in Section 5. Moreover, the results shown in Theorem 3.6 can be extended to the case where λ_κ is a semi-simple eigenvalue of (A, M) with a multiplicity up to s.

EXAMPLE 3.7.

$$A = \begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 3 & 1 & 1 \\ 0 & 1 & 2 & 0 \\ 1 & 1 & 0 & 2 \end{bmatrix}, \qquad M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and consider $d = 1,\ s = 3$. Figure 3.2 displays a plot of the magnitude of the eigenbranches $\theta_j(\sigma)$, $j = 1, 2, 3$ (left subfigure), as well as the magnitude of the scalar function $\chi_{3,j}(\sigma)$ as

$\sigma \to \lambda_3$ ($\lambda_3 = 2.3473$ is simple). As expected, there exists exactly one eigenbranch, in this example $\theta_2(\sigma)$, for which $\lim_{\sigma\to\lambda_3} \theta_2(\sigma) = 0$. In agreement with Theorem 3.6, we also have $\lim_{\sigma\to\lambda_3} \chi_{2,1}(\sigma) \neq 0$ and $\lim_{\sigma\to\lambda_3} \chi_{3,j}(\sigma) = 0$ for any $j \neq 2$.

3.4. Characterization of the eigenbranches as $\boldsymbol{\sigma}$ approaches an eigenvalue of $\boldsymbol{(B, M_B)}$. Let the ηth eigenvalue-eigenvector pair of (B, M_B) be denoted as (δ_η, v_η), $\eta = 1, \ldots, d$. Setting

$$w_\eta = E_\sigma^T v_\eta, \qquad f_\eta = M_E^T v_\eta, \tag{3.6}$$

and making use of the equation

$$B_\sigma^{-1} = \sum_{\eta=1}^{d} \frac{v_\eta v_\eta^T}{\delta_\eta - \sigma},$$

we can rewrite

$$S(\sigma) = C_\sigma - \sum_{\eta=1}^{d} \frac{w_\eta w_\eta^T}{\delta_\eta - \sigma}, \tag{3.7}$$

and

$$-S'(\sigma) = M_C + \sum_{\eta=1}^{d} \frac{w_\eta w_\eta^T}{(\delta_\eta - \sigma)^2} - \sum_{\eta=1}^{d} \frac{w_\eta f_\eta^T}{\delta_\eta - \sigma} - \sum_{\eta=1}^{d} \frac{f_\eta w_\eta^T}{\delta_\eta - \sigma}. \tag{3.8}$$

While $S(\sigma)$ and $S'(\sigma)$ can not be formally defined when $\sigma \in \Lambda(B, M_B)$, the eigenbranches remain well-defined regardless of how close σ approximates an eigenvalue of the pencil (B, M_B).

THEOREM 3.8. *Let $\delta_k \notin \Lambda(A, M)$, $1 \le k \le d$, be a simple eigenvalue of (B, M_B), and define for any $\eta = 1, \ldots, d$,*

$$\epsilon_{j,\eta}(\sigma) = w_\eta^T \hat{y}_j(\sigma), \qquad \phi_{j,\eta}(\sigma) = f_\eta^T \hat{y}_j(\sigma), \tag{3.9}$$

where w_η and f_η are defined in (3.6). Then:

- *If $\lim_{\sigma\to\delta_k} \epsilon_{j,k}(\sigma) \neq 0$, then $\lim_{\sigma\to\delta_k} \theta_j(\sigma) = 0$.*
- *If $\lim_{\sigma\to\delta_k} \epsilon_{j,k}(\sigma) = 0$, then*

$$\lim_{\sigma\to\delta_k} \theta_j(\sigma) = \frac{\hat{y}_j^T(\sigma) C_\sigma \hat{y}_j(\sigma) - \sum_{\eta=1,\eta\neq k}^{d} \dfrac{\epsilon_{j,\eta}^2(\sigma)}{\delta_\eta - \sigma}}{\hat{y}_j^T(\sigma) M_c \hat{y}_j(\sigma) + \sum_{\eta=1,\eta\neq k}^{d} \left[\dfrac{\epsilon_{j,\eta}^2(\sigma)}{(\delta_\eta - \sigma)^2} - \dfrac{2\epsilon_{j,\eta}(\sigma)\phi_{j,\eta}(\sigma)}{\delta_\eta - \sigma}\right]}.$$

Proof. Making use of equations (3.7) and (3.8), each eigenbranch $\theta_j(\sigma)$, $j = 1, \ldots, s$, can be written as

$$\theta_j(\sigma) = \frac{\hat{y}_j^T(\sigma) C_\sigma \hat{y}_j(\sigma) - \sum_{\eta=1}^{d} \dfrac{\epsilon_{j,\eta}^2(\sigma)}{\delta_\eta - \sigma}}{\hat{y}_j^T(\sigma) M_c \hat{y}_j(\sigma) + \sum_{\eta=1}^{d} \dfrac{\epsilon_{j,\eta}^2(\sigma)}{(\delta_\eta - \sigma)^2} - \sum_{\eta=1}^{d} \dfrac{2\epsilon_{j,\eta}(\sigma)\phi_{j,\eta}(\sigma)}{\delta_\eta - \sigma}}.$$

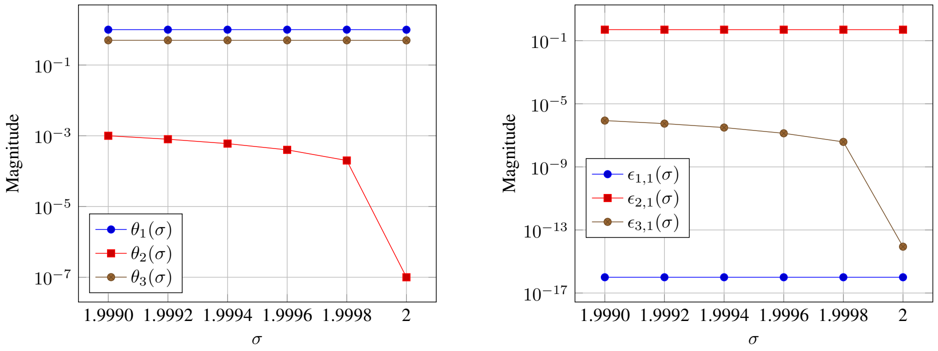

FIG. 3.3. *Numerical values of $\theta_j(\sigma)$ and $\epsilon_{j,1}(\sigma)$ as $\sigma \to \delta_1$.*

Multiplying both the numerator and denominator by $(\delta_k - \sigma)^2$ gives
(3.10)

$$\theta_j(\sigma) = \frac{(\delta_k - \sigma)^2 \left(\hat{y}_j^T(\sigma) C_\sigma \hat{y}_j(\sigma) - \sum_{\eta=1,\eta\neq k}^d \dfrac{\epsilon_{j,\eta}^2(\sigma)}{\delta_\eta - \sigma} \right) - (\delta_k - \sigma)\epsilon_{j,\kappa}^2(\sigma)}{(\delta_k - \sigma)^2 \left(\hat{y}_j^T(\sigma) M_c \hat{y}_j(\sigma) + \sum_{\eta=1,\eta\neq k}^d \left[\dfrac{\epsilon_{j,\eta}^2(\sigma)}{(\delta_\eta - \sigma)^2} - \dfrac{2\epsilon_{j,\eta}(\sigma)\phi_{j,\eta}(\sigma)}{\delta_\eta - \sigma} \right] \right) + \gamma(j,k,\sigma)},$$

where

$$\gamma(j,k,\sigma) = \epsilon_{j,k}^2(\sigma) - 2(\delta_k - \sigma)\epsilon_{j,k}(\sigma)\phi_{j,k}(\sigma).$$

As $\sigma \to \delta_k$, the numerator in (3.10) approaches zero while the denominator approaches $\lim_{\sigma\to\delta_k} \gamma(j,k,\sigma) = \lim_{\sigma\to\delta_k} \epsilon_{j,k}^2(\sigma)$. Therefore, when $\lim_{\sigma\to\delta_k} \epsilon_{j,k}(\sigma) \neq 0$, we have

$$\lim_{\sigma\to\delta_k} \theta_j(\sigma) = \frac{0}{\lim_{\sigma\to\delta_k} \epsilon_{j,k}^2(\sigma)} = 0. \qquad \square$$

COROLLARY 3.9. *Let $w_k \neq 0$. Then, there exists at least one integer $1 \leq j \leq s$ such that* $\lim_{\sigma\to\delta_k} \theta_j(\sigma) = 0$.

Proof. The eigenvectors of $(S(\sigma), -S'(\sigma))$ are linearly independent and thus span the entire s-dimensional space. Since w_k is nonzero, there exists at least one eigenvector $\hat{y}_j(\sigma)$ of $(S(\sigma), -S'(\sigma))$ such that $\lim_{\sigma\to\delta_k} \epsilon_{j,k}(\sigma) = \lim_{\sigma\to\delta_k} w_k^T \hat{y}_j(\sigma) \neq 0$, and thus by Theorem 3.8, $\lim_{\sigma\to\delta_k} \theta_j(\sigma) = 0$. $\square$

EXAMPLE 3.10. Consider the same matrices A and M as in Example 3.7, and let $d = 1$, $s = 3$. Figure 3.3 reports the values of $\theta_j(\sigma),\ j = 1, 2, 3$, and $\epsilon_{j,1}(\sigma)$ as $\sigma \to \delta_1$ ($\delta_1 = 2.0$ is simple). As predicted by Theorem 3.8, only the eigenbranch for which $\lim_{\sigma\to\delta_1} \epsilon_{j,1}(\sigma) \neq 0$ (in this example $\theta_2(\sigma)$) shifts to zero. On the other hand, $\lim_{\sigma\to\delta_1} \epsilon_{j,1}(\sigma) = 0$ for any $j \neq 2$.

DEFINITION 3.11. *Let $r \in \mathbb{R}^s \setminus \{0\}$. We define the* generalized Rayleigh quotient*:*

$$\begin{aligned} \pi(\sigma, r) &= \frac{r^T S(\sigma) r}{r^T[-S'(\sigma)]r} \\ &= \frac{r^T C_\sigma r - \sum_{\eta=1}^d \dfrac{(r^T w_\eta)^2}{\delta_\eta - \sigma}}{r^T M_C r + \sum_{\eta=1}^d \dfrac{(r^T w_\eta)^2}{(\delta_\eta - \sigma)^2} - 2\sum_{\eta=1}^d \dfrac{(r^T w_\eta)(r^T f_\eta)}{\delta_\eta - \sigma}}. \end{aligned}$$

The eigenvalues of the pencil $(S(\sigma), -S'(\sigma))$ can be then characterized by a generalization of the Courant-Fischer minimax theorem as

$$\theta_j(\sigma) = \min_{\dim(\mathcal{U})=j} \max_{r\in\mathcal{U},\ r\neq 0} \pi(\sigma, r).$$

THEOREM 3.12. *Let $\delta_k \notin \Lambda(A, M)$, $1 \leq k \leq d$, be a simple eigenvalue of (B, M_B) such that $w_k \neq 0$. Then:*

- *There exists exactly one index $1 \leq \zeta \leq s$ such that $\lim_{\sigma\to\delta_k} \theta_\zeta(\sigma) = 0$.*
- *For any eigenpair $(\theta_j(\sigma), \hat{y}_j(\sigma))$ that satisfies $\lim_{\sigma\to\delta_k} \theta_j(\sigma) \neq 0$, we have $\lim_{\sigma\to\delta_k} \hat{y}_j^T(\sigma) w_k = 0$.*

Proof. Let $\mathcal{I} = [\delta_k - \sigma_0, \delta_k) \cup (\delta_k, \delta_k + \sigma_0]$ for some positive $\sigma_0 \in \mathbb{R}$ such that $\mathcal{I}$ contains no other eigenvalues of (B, M_B) or any eigenvalues of (A, M), and let $\sigma \in \mathcal{I}$. By Corollary 3.9, we know that there exists at least one integer $1 \leq \zeta \leq s$ such that $\lim_{\sigma\to\delta_k} \theta_\zeta(\sigma) = 0$. Let $1 \leq \psi \leq s$ be the first such integer, i.e.,

$$\lim_{\sigma\to\delta_k} \theta_\psi(\sigma) = \lim_{\sigma\to\delta_k} \left(\min_{\dim(\mathcal{U})=\psi} \max_{r\in\mathcal{U}\setminus\{0\}} \pi(\sigma, r) \right) = 0.$$

When $\psi \equiv s$, the pencil $\lim_{\sigma\to\delta_k}(S(\sigma), -S'(\sigma))$ has exactly one zero eigenvalue. Let now $\psi < s$, and define the subspaces

$$\mathcal{V} = \text{span}\left(\left\{v | v^T w_k = 0\right\}\right) \quad \text{and} \quad \mathcal{Z} = \text{span}\left(\left\{v | v^T \left[-S'(\sigma)\right] \hat{y}_j(\sigma) = 0,\ j = 1, \ldots, \psi\right\}\right).$$

Note that dim($\mathcal{Z}$)= $s - \psi$ and dim($\mathcal{V}$)= $s - 1$ and the dimension of the subspace $\mathcal{Z} \cap \mathcal{V}$ will be larger than or equal to one as long as $w_k \notin \mathcal{Z}$. The latter is always true as the opposite implies that $S(\sigma)$ is singular. Thus, for any $\mathcal{U}$ such that dim($\mathcal{U}$)=$\psi + 1$, we have dim($\mathcal{U} \cap \mathcal{Z} \cap \mathcal{V}) \geq 1$, and thus

$$\lim_{\sigma\to\delta_k} \theta_{\psi+1}(\sigma) = \lim_{\sigma\to\delta_k} \left(\min_{\dim(\mathcal{U})=\psi+1} \max_{r\in\mathcal{U}\setminus\{0\}} \pi(\sigma, r) \right) > 0.$$

To prove the second item, recall the variables $\epsilon_{j,\eta}(\sigma) = w_\eta^T \hat{y}_j(\sigma)$ and $\phi_{j,\eta}(\sigma) = f_\eta^T \hat{y}_j(\sigma)$, $\eta = 1, \ldots, d$, defined in (3.9). Setting

$$\zeta_{j,k}(\sigma) = (\delta_k - \sigma)^2 \left(\hat{y}_j^T(\sigma) C_\sigma \hat{y}_j(\sigma) - \sum_{\eta=1,\eta\neq k}^{d} \frac{\epsilon_{j,\eta}^2(\sigma)}{\delta_\eta - \sigma} \right)$$

and

$$\omega_{j,k}(\sigma) = (\delta_k - \sigma)^2 \left(\hat{y}_j^T(\sigma) M_c \hat{y}_j(\sigma) + \sum_{\eta=1,\eta\neq k}^{d} \left[\frac{\epsilon_{j,\eta}^2(\sigma)}{(\delta_\eta - \sigma)^2} - \frac{2\epsilon_{j,\eta}(\sigma)\phi_{j,\eta}(\sigma)}{\delta_\eta - \sigma} \right] \right),$$

we can rewrite (3.10) as

$$\lim_{\sigma\to\delta_k} \theta_j(\sigma) = \lim_{\sigma\to\delta_k} \left[\frac{\zeta_{j,k}(\sigma) - (\delta_k - \sigma)\epsilon_{j,\kappa}^2(\sigma)}{\omega_{j,k}(\sigma) + (\epsilon_{j,k}^2(\sigma) - 2(\delta_k - \sigma)\epsilon_{j,k}(\sigma)\phi_{j,k}(\sigma))} \right]. \tag{3.11}$$

Then, for any $\lim_{\sigma\to\delta_k} \theta_j(\sigma) \neq 0$, rearranging[2] terms in (3.11) leads to

$$\lim_{\sigma\to\delta_k} \epsilon_{j,k}^2(\sigma) = \lim_{\sigma\to\delta_k} \left[2(\delta_k - \sigma)\epsilon_{j,k}(\sigma)\phi_{j,k}(\sigma) - \omega_{j,k}(\sigma) + \frac{\zeta_{j,k}(\sigma) - (\delta_k - \sigma)\epsilon_{j,\kappa}^2(\sigma)}{\theta_j(\sigma)} \right]. \tag{3.12}$$

[2] Note that the limit of the denominator is assumed non-zero and thus $\lim_{\sigma\to\delta_k} \dfrac{a}{b} = \dfrac{\lim_{\sigma\to\delta_k} a}{\lim_{\sigma\to\delta_k} b}$.

Since both $\lim_{\sigma\to\delta_k}\omega_{j,k}(\sigma)\to 0$ and $\lim_{\sigma\to\delta_k}\zeta_{j,k}(\sigma)\to 0$, the right-hand side in (3.12) converges to zero, and thus $\lim_{\sigma\to\delta_k}\epsilon_{j,k}^2(\sigma)=\lim_{\sigma\to\delta_k}\left(w_k^T\hat{y}_j(\sigma)\right)^2=0$. □

3.4.1. Characterization of the non-zero eigenvalues. Our interest now turns on determining to what values the nonzero eigenvalues of $\lim_{\sigma\to\delta_k}(S(\sigma),-S'(\sigma))$ converge to.

We define $\hat{w}_\eta=w_\eta/\|w_\eta\|$, $\eta=1,\dots,d$, where w_η is defined in (3.6), and the corresponding orthogonal projector

$$P_\eta = I-\hat{w}_\eta\hat{w}_\eta. \tag{3.13}$$

Moreover, let $1\le k\le d$, and assume that $\delta_k\notin\Lambda(A,M)$ is a simple eigenvalue of (B,M_B). We define the matrices

$$S_k(\sigma)=C_\sigma-\sum_{\eta=1,\eta\neq k}^{d}\frac{w_\eta w_\eta^T}{\delta_\eta-\sigma},$$

and

$$-S_k'(\sigma)=M_C+\sum_{\eta=1,\eta\neq k}^{d}\frac{w_\eta w_\eta^T}{(\delta_\eta-\sigma)^2}-\sum_{\eta=1,\eta\neq k}^{d}\frac{w_\eta f_\eta^T}{\delta_\eta-\sigma}-\sum_{\eta=1,\eta\neq k}^{d}\frac{f_\eta w_\eta^T}{\delta_\eta-\sigma}$$

and the operators

$$S_{k,|}(\sigma) \;=\; \left[P_kS_k(\sigma)P_k\right]_{|{w_k}^\perp}, \qquad -S_{k,|}'(\sigma) \;=-\left[P_kS_k'(\sigma)P_k\right]_{|{w_k}^\perp},$$

where $\left[P_kS_k(\sigma)P_k\right]_{|{w_k}^\perp}$ and $\left[-P_kS_k'(\sigma)P_k\right]_{|{w_k}^\perp}$ are $(s-1)\times(s-1)$ matrices that denote the restriction of $P_kS_k(\sigma)P_k$ and $-P_kS_k'(\sigma)P_k$ to the subspace orthogonal to $\hat{w}_k$, respectively.

LEMMA 3.13. *Let P_k be defined as in (3.13). Then, the eigenvalues of the pencil $(S_{k,|}(\sigma),-S_{k,|}'(\sigma))$ are identical to the determined eigenvalues of the pencil $(P_kS_k(\sigma)P_k, -P_kS_k'(\sigma)P_k)$.*

Proof. See Lemma 4 in [3]. □

We can now prove the following theorem.

THEOREM 3.14. *Let $\delta_k\notin\Lambda(A,M)$, $1\le k\le d$, be a simple eigenvalue of (B,M_B), and let $\tau_1(\delta_k)\le\ldots\le\tau_{s-1}(\delta_k)$ denote the eigenvalues of the pencil $(S_{k,|}(\delta_k),-S_{k,|}'(\delta_k))$. Then*

$$\lim_{\sigma\to\delta_k}\theta_j(\sigma)=\tau_i(\delta_k),$$

where $i\in\{1,2,\dots,s-1\}$ and $j\in\{1,2,\dots,s\}-\{\zeta\}$ with ζ determined by Lemma 3.12.

Proof. Multiplying both sides of $S(\sigma)\hat{y}_j(\sigma)=\theta_j\left[-S'(\sigma)\right]\hat{y}_j(\sigma)$ by P_k from the left and taking advantage of the identity

$$\hat{y}_j(\sigma)=P_k\hat{y}_j(\sigma)+\left(\hat{w}_k^T\hat{y}_j(\sigma)\right)\hat{w}_k$$

gives

$$\begin{gathered}P_kS(\sigma)\hat{y}_j(\sigma)=\theta_j\left[-P_kS'(\sigma)\right]\hat{y}_j(\sigma)\\ P_kS(\sigma)\left[P_k\hat{y}_j(\sigma)+\left(\hat{w}_k^T\hat{y}_j(\sigma)\right)\hat{w}_k\right]=\theta_j\left[-P_kS'(\sigma)\right]\left[P_k\hat{y}_j(\sigma)+\left(\hat{w}_k^T\hat{y}_j(\sigma)\right)\hat{w}_k\right].\end{gathered}$$

Reordering terms and noticing that

$$P_kS(\sigma)P_k=P_kS_k(\sigma)P_k,\qquad P_kS'(\sigma)P_k=P_kS_k'(\sigma)P_k,$$

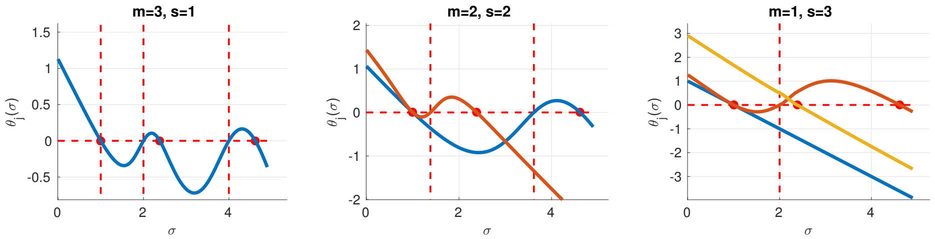

FIG. 4.1. *Eigenbranches of the pencil $(S(\sigma), -S'(\sigma))$ for different values of $d \equiv m$ and s when A is as in* (4.1) *and $M = I$. The red circles along the real axis denote eigenvalues of the pencil (A, I). The dashed vertical lines denote eigenvalues of matrix $B = A(1:m, 1:m)$.*

leads to

$$P_k\left[S_k(\sigma)+\theta_j(\sigma)S_k'(\sigma)\right]P_k\hat{y}_j(\sigma) = -P_k\left[S_k(\sigma)+\theta_j(\sigma)S_k'(\sigma)\right]\left(\hat{w}_k^T\hat{y}_j(\sigma)\right)\hat{w}_k. \tag{3.14}$$

The right-hand side of (3.14) simply expresses the residual of the approximate eigenpair $(\theta_j(\sigma), \hat{y}_j(\sigma))$ with respect to the matrix pencil $(P_kS_k(\sigma)P_k,\ -P_kS_k'(\sigma)P_k)$, which is now well-defined as $\sigma \to \delta_k$ and which converges to $(P_kS_k(\delta_k)P_k, -P_kS_k'(\delta_k)P_k)$. The latter, by Lemma 3.13, is a trivial extension of the pencil $(S_{k,|}(\delta_k), -S_{k,|}'(\delta_k))$. To finalize the proof, recall that by Lemma 3.12 we have $\lim_{\sigma\to\delta_k}\hat{w}_k^T\hat{y}_j(\sigma) = 0$ for any $\lim_{\sigma\to\delta_k}\theta_j(\sigma) \neq 0$. Thus, each nonzero $\lim_{\sigma\to\delta_k}\theta_j(\sigma)$ converges to the corresponding eigenvalue of the pencil $(S_{k,|}(\delta_k), -S_{k,|}'(\delta_k))$. □

The result in Theorem 3.14 can be easily extended to the case where δ_k is semi-simple.

4. Practical aspects of MLA.

DEFINITION 4.1. *We define the "eigenpoles" of an eigenbranch*[3] *$\theta_j(\sigma)$, $1 \le j \le s$, to be those real scalars $\delta \in \Lambda(B, M_B)$ that satisfy $\lim_{\sigma\to\delta}\theta_j(\sigma) = 0$. Similarly, we define the "regular roots" of an eigenbranch $\theta_j(\sigma)$ to be those real scalars $\lambda \in \Lambda(A, M)$ that satisfy $\theta_j(\lambda) = 0$.*

Figure 4.1 displays the eigenbranches of a toy pencil (A, M) where

$$A = \begin{bmatrix} B & E \\ E^T & C \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 3 & 1 & 1 \\ 0 & 1 & 2 & 0 \\ 1 & 1 & 0 & 2 \end{bmatrix}, \tag{4.1}$$

and $M = I$. The dimension d of the pencil (B, M_B) was varied. When $d = n - 1 = 3$, there exists only one eigenbranch, and this eigenbranch crosses the real axis at all real points $\sigma \in \Lambda(B, M_B) \cup \Lambda(A, M)$. Note that although $\lambda_1 = \lambda_2 = 1$, only one copy of this semi-simple eigenvalue can be computed. On the other hand, when $d = n - 2 = 2$, we have $s = 2$, and both eigenbranches cross the real axis at $\sigma = \lambda_1 = \lambda_2$. More generally, MLA can compute the correct multiplicity of a semi-simple eigenvalue of (A, M) only if the latter is less than or equal to s.

4.1. The impact of the location of the eigenvalues of (B, M_B).

Ideally, we would like the shape of the eigenbranch $\theta_j(\sigma)$ which satisfies $\lim_{\sigma\to\lambda}\theta_j(\sigma) = 0$ to be as close to that of a linear function as possible.

Let us first consider the case where $\sigma \approx \lambda$, i.e., σ lies in a small neighborhood around λ. In this scenario, we expect the shape of the eigenbranch $\theta_j(\sigma)$ to be close to linear since

[3] See also the definition of an "eigenpole" in [1].

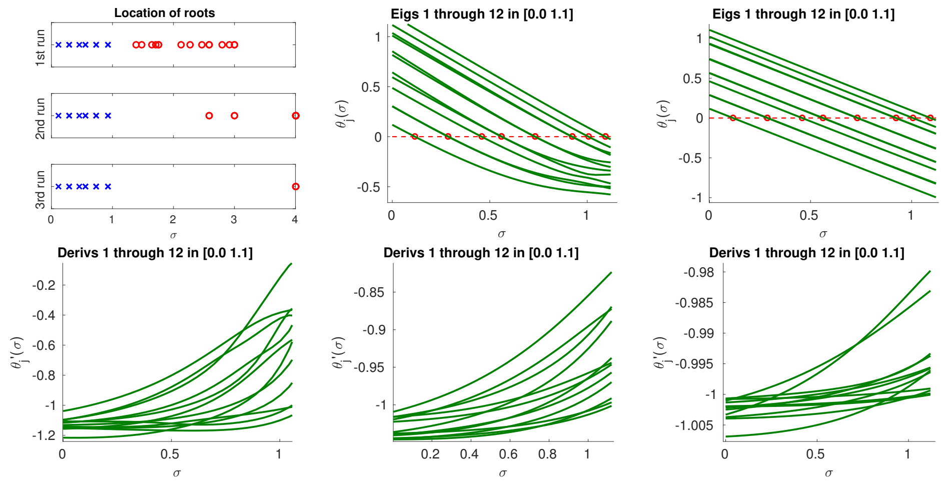

FIG. 4.2. *Top row (left subfigure): the eigenvalues of* (B, M_B) *located inside the interval* $[0.0, 4.0]$ *(marked by "○") for three different arrangements. The eigenvalues* $\lambda_1, \ldots, \lambda_{12}$ *of* (A, M) *(marked by "x") are also shown. Top row (middle and right subfigures): plot of* $\theta_j(\sigma)$, $1 \leq j \leq 12$, *for the first and second arrangement of the algebraically smallest eigenvalues of* (B, M_B). *Bottom row: plot of* $\theta'_j(\sigma)$, $1 \leq j \leq 12$, *for the first (left), second (middle), and third (right) arrangement of the eigenvalues of* (B, M_B).

$\theta_j(\sigma) \approx 0$ and thus $\theta'_j(\sigma) = -1 + O(\theta(\sigma)) \approx -1$ is almost constant. But what can we tell about the shape of $\theta_j(\sigma)$ when σ is located further away from λ? While a detailed analysis lies outside the scope of this paper, in general the further σ lies from the eigenpoles of $\theta_j(\sigma)$, the closer to linear the shape of $\theta_j(\sigma)$ is. This observation implies that, ideally, the sought eigenvalue λ should lie as far as possible from the eigenpoles of the eigenbranch which satisfies $\lim_{\sigma \to \lambda} \theta_j(\sigma) = 0$.

EXAMPLE 4.2. We consider the computation of the eigenvalues $\lambda_1, \ldots, \lambda_{12}$ of a SPD pencil (A, M) generated by a five-point finite difference discretization of the Laplace operator on the unit plane (M is chosen as the identity matrix). Each eigenvalue λ_j, $j = 1, \ldots, 12$, is a regular root of an eigenbranch $\theta_j(\sigma)$, and we focus on the shape of the eigenbranches $\theta_j(\sigma)$, $j = 1, \ldots, 12$, as their eigenpoles located the closest to $[\lambda_1, \lambda_{12}]$ are progressively shifted away. More specifically, we consider three different arrangements of the eigenpoles (we only show those located within the interval $[0.0, 4.0]$), where each arrangement was formed by progressively increasing the number of interface variables of the pencil (A, M) (thus reducing the value of d). The exact location of the eigenvalues $\lambda_1, \ldots, \lambda_{12}$ (some of these eigenvalues were semi-simple) as well as the location of the algebraically smallest eigenpoles of the eigenbranches $\theta_j(\sigma)$, $j = 1, \ldots, 12$, are shown in Figure 4.2. In the same figure we also display the eigenbranches $\theta_j(\sigma)$, $1 \leq j \leq 12$, for the first and second arrangement of the eigenpoles. It is easy to verify that for the second arrangement of the eigenpoles, the shape of the eigenbranches is closer to linear compared to that for the first arrangement. The latter is shown more clearly in the bottom row subfigures in Figure 4.2, where we plot

$$\theta'_j(\sigma) = -1 - \theta_j(\sigma)\frac{\hat{y}_j^T(\sigma)S''(\sigma)\hat{y}_j(\sigma)}{\hat{y}_j^T(\sigma)S'(\sigma)\hat{y}_j(\sigma)}, \qquad j = 1, \ldots, 12,$$

for the first (left), second (middle), and third (right) arrangement of the eigenpoles.

While it is impossible to determine what eigenvalues of (B, M_B) are the eigenpoles of a given eigenbranch $\theta_j(\sigma)$, it is possible to reduce the overall number of eigenpoles by reducing the size of (B, M_B). For example, the eigenvalues of the pencil (B, M_B) in Figure 4.1 are

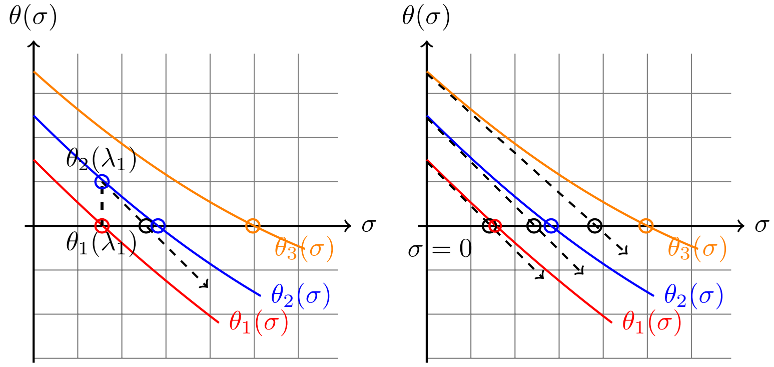

FIG. 5.1. *Black circles denote eigenvalue approximations. Left: computing an initial approximation of* λ_2 *as soon as* λ_1 *is computed. Right: computing initial approximations of multiple eigenvalues using the eigenpairs of the pencil* $(S(0), -S'(0))$.

equal to $\{1, 2, 4\}$, $\{1.38, 3.61\}$, and $\{2\}$, when the size of the matrices B and M_B is set to $d = 3$, $d = 2$, and $d = 1$, respectively. Therefore, choosing the dimension d of the pencil (B, M_B) to be smaller increases the likelihood that a regular root $\lambda \in \Lambda(A, M)$ of an eigenbranch $\theta_j(\sigma)$ will be located relatively further away from the eigenpoles of the latter.

5. Numerical experiments.

Our experiments were performed in a Matlab environment (version R2018a) using 64-bit arithmetic (double precision) on a single core of a computing system equipped with an Intel Haswell E5-2680v3 processor and 32 GB of system memory.

5.1. Computing more than one eigenpairs.

MLA can be extended to the computation of more than one eigenpairs. The only requirement is the computation of an initial approximation of each sought eigenvalue. We now briefly describe two such generalizations:

(a) The sought eigenpairs are computed sequentially, one after the other. As soon as a scalar σ is signaled as an accurate approximation of some eigenvalue λ, we proceed to compute the smallest (or second smallest if $\theta(\sigma) > 0$) positive eigenvalue of the pencil $(S(\sigma), -S'(\sigma))$. We then set this eigenvalue and corresponding eigenvector as the new eigenpair $(\theta(\sigma), \hat{y}(\sigma))$ in MLA. A similar technique can be also found in [11]. An example is presented in the left subfigure of Figure 5.1, where after we have computed the eigenvalue λ_1, the initial approximation of the eigenvalue λ_2 is computed as $\lambda_1 - \theta_2(\lambda_1)/\theta_2'(\lambda_1)$, where $\theta_2(\lambda_1)$ is the smallest positive eigenvalue of the pencil $(S(\lambda_1), -S'(\lambda_1))$.

(b) The sought eigenpairs are computed independently, and the roots of multiple eigenbranches are chased simultaneously. To achieve this, *we set the same initial approximation* σ *for all sought eigenvalues*. We then compute multiple eigenpairs of the pencil $(S(\sigma), -S'(\sigma))$. These eigenpairs are then used to generate initial approximations of the eigenvalues near σ. An example is presented in the right subfigure of Figure 5.1 where we set $\sigma = 0$ and update the initial eigenvalue approximation of the eigenvalues λ_i, $i = 1, 2, 3$, as $\sigma - \theta_i(\sigma)/\theta_i'(\sigma) =: 0 - \theta_i(0)/\theta_i'(0)$, respectively.

The second approach can take advantage of parallel computing since the roots of different eigenbranches can be computed independently. In addition, it allows MLA to compute eigenvalues of the pencil (A, M) which lie further away from σ. Indeed, the only requirement to compute an eigenvalue λ is to be able to "jump" on the correct eigenbranch. Depending on how close the shape of the corresponding eigenbranch is to that of a linear function, the value of σ need not be close to λ.

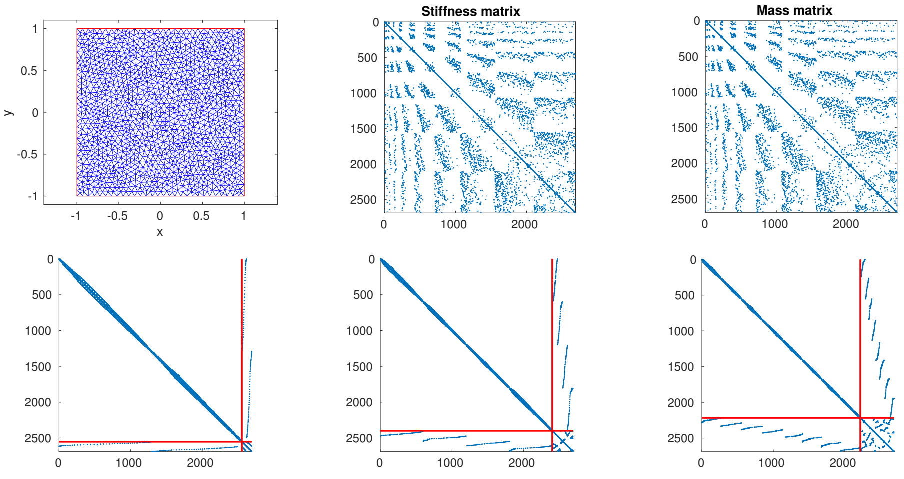

FIG. 5.2. *From left to right, top row: the finite element mesh, the sparsity pattern of* A*, and the sparsity pattern of* M*. From left to right, bottom row: the sparsity pattern of the pencil* (A, M) *after setting* $p = 2$*,* $p = 4$*, and* $p = 8$*, respectively.*

5.2. Domain decomposition. We reorder the matrices A and M such that the submatrices B_σ, E_σ, and M_E are block-diagonal. To achieve this, we apply the METIS graph partitioner [13] to partition the adjacency graph of the matrix $|A| + |M|$ into p non-overlapping partitions. We then permute the rows/columns of the matrices A and M so that variables/equations associated with interior variables are listed before those associated with interface variables. We then get the following block-pattern (we kept the same notation for the reordered version of matrices A and M):

$$A = \begin{bmatrix} B_1 & & & & E_1 \\ & B_2 & & & E_2 \\ & & \ddots & & \vdots \\ & & & B_p & E_p \\ E_1^T & E_2^T & \dots & E_p^T & C \end{bmatrix}, \ M = \begin{bmatrix} M_B^{(1)} & & & & M_E^{(1)} \\ & M_B^{(2)} & & & M_E^{(2)} \\ & & \ddots & & \vdots \\ & & & M_B^{(p)} & M_E^{(p)} \\ \left(M_E^{(1)}\right)^T & \left(M_E^{(2)}\right)^T & \dots & \left(M_E^{(p)}\right)^T & M_C \end{bmatrix},$$

where the matrices B_i and $M_B^{(i)}$ are square matrices of size $d_i \times d_i$ and the matrices E_i and $M_E^{(i)}$ are rectangular matrices of size $d_i \times s_i$. The variables d_i and s_i denote the number of interior and interface nodes located in the ith subdomain of the adjacency graph of $|A| + |M|$, respectively. In addition, the matrices C and M_C are of size $s \times s$, where $s = \sum_{i=1}^{p} s_i$. Figure 5.2 displays the sparsity pattern of the matrices A and M after a domain decomposition reordering of a pencil (A, M) stemming from a finite element discretization of the Laplace operator on the $[-1, 1] \times [-1, 1]$ plane using linear elements with target maximum mesh edge length equal to $h = 0.05$.

Higher values of p lead to cheaper linear system solves with the matrix B_σ and increase the amount of parallelism during computations with interior variables. On the other hand, increasing p naturally leads to an increase in the number of interface variables, thus increasing the dimension of the Schur complement eigenvalue problem. In this paper we do not consider these trade-offs, although higher values of p lead to fewer eigenpoles which in practice leads to faster convergence in MLA.

5.3. Details on the experimental framework. Each approximate eigenpair $(\tilde{\lambda}, \tilde{x})$ will be signaled as a sufficiently accurate approximation of an eigenpair (λ, x) as soon as the corresponding residual norm satisfies the inequality $\|A\tilde{x} - \tilde{\lambda}M\tilde{x}\| \leq$ 1e-8 $\times \|\tilde{x}^T M\tilde{x}\|$. On the other hand, each approximation $(\tilde{\theta}, \tilde{y})$ of the eigenpair $(\theta(\sigma), \hat{y}(\sigma))$ will be considered sufficiently accurate as soon as the residual norm satisfies[4] $\|S(\sigma)\tilde{y} + \tilde{\theta}S'(\sigma)\tilde{y}\| \leq$ 1e-10 $\times \|\tilde{y}^T S'(\sigma)\tilde{y}\|$. Unless mentioned otherwise, each eigenpair $(\theta(\sigma), \hat{y}(\sigma))$ will be computed by an (inexact) inverse iteration [19].

5.4. Numerical demonstration. We consider the computation of the $n_{ev} = 20$ algebraically smallest eigenvalues of three different pencils using the second mechanism discussed in Section 5.1. The first two pencils are of the standard form ($M = I$), and the matrix A represents a finite difference approximation of the Laplace operator on a regular grid with homogeneous Dirichlet boundary conditions on the entire boundary of the domain $[0, 1] \times [0, 1]$ and $[0, 1] \times [0, 1] \times [0, 1]$, respectively. Their mesh sizes are chosen as a) $n_x = 253$, $n_y = 148$, and $n_z = 1$, and b) $n_x = 53$, $n_y = 48$, and $n_z = 15$, and thus the dimension of the matrix A is equal to $n = 37\,444$ and $n = 38\,160$, respectively. We finally consider the pencil “Kuu/Muu" listed in Table 5.2. In all cases, the number of subdomains p is set such that $n/s \geq 10$.

The initial approximation of all n_{ev} sought eigenvalues is set equal to $\sigma := 0$. *MLA is then applied to the computation of each eigenvalue independently of each other*. The only deviation from Algorithm 1 is that the first evaluation of MLA during the computation of the eigenvalue λ_i replaces $(\theta(\sigma), \hat{y}(\sigma))$ with the eigenpair $(\theta_i(\sigma), \hat{y}_i(\sigma))$ so that MLA gets onto the right eigenbranch. Figure 5.3 displays the n_{ev} sought eigenvalues of the above pencils together with their approximation by MLA. The corresponding residual norms at each iteration of MLA are also shown (different dashed curves correspond to different eigenpairs of the pencil (A, M)). Notice that especially for the pencil “Kuu/Muu", the zero initial approximations were actually quite far from all sought eigenvalues (e.g., $\lambda_{20} = 194.42$).

5.4.1. Comparisons against Krylov subspace eigenvalue solvers. A reasonable question is how MLA compares with variants of the Lanczos method [16, 22]. In practice, such a comparison is not trivial since MLA works with pencils of the form $(S(\sigma), -S'(\sigma))$ while the Lanczos method works with the pencil (A, M). In addition, Lanczos is a subspace method, e.g., it extracts approximate eigenpairs from a (Krylov) subspace, while MLA is a single-vector iteration. Additionally, MLA is based on the concept of domain decomposition, and thus a proper distributed memory implementation is required to fully exploit its advantages. Nonetheless, MLA might be a competitive approach with respect to the Lanczos method even for sequential architectures when: a) the number of sought eigenvalues n_{ev} is small and b) a shift-and-invert approach is not feasible due to excessive fill-in in the associated triangular factors.

Recall the 253 × 148 finite difference discretization of the 2D Dirichlet eigenvalue problem discussed above, and consider the computation of its $n_{ev} = 2$ (algebraically) smallest eigenvalues by Matlab’s Implicitly Restarted Arnoldi routine “`eigs`" (IRA) equipped with a subspace of size forty, e.g., the maximum subspace dimension in IRA is twenty times the number of sought eigenvalues (the default choice is only two times the number of sought eigenvalues). Without shift-and-invert, applying IRA directly to the pencil (A, M) requires 82 cycles to approximate the two sought eigenpairs up to the required accuracy. In contrast, applying IRA to the pencil $(S(0), -S'(0))$ in order to generate initial approximations in MLA only requires 24 cycles. Compared to applying IRA directly to the discretized Laplacian, the saving in orthogonalization costs achieved by MLA is proportional to the ratio $82n/24s$, and

[4] In practice, the inner eigenvalue problem can be solved even less accurately (see [7] for more details on the convergence of inexact Newton methods).

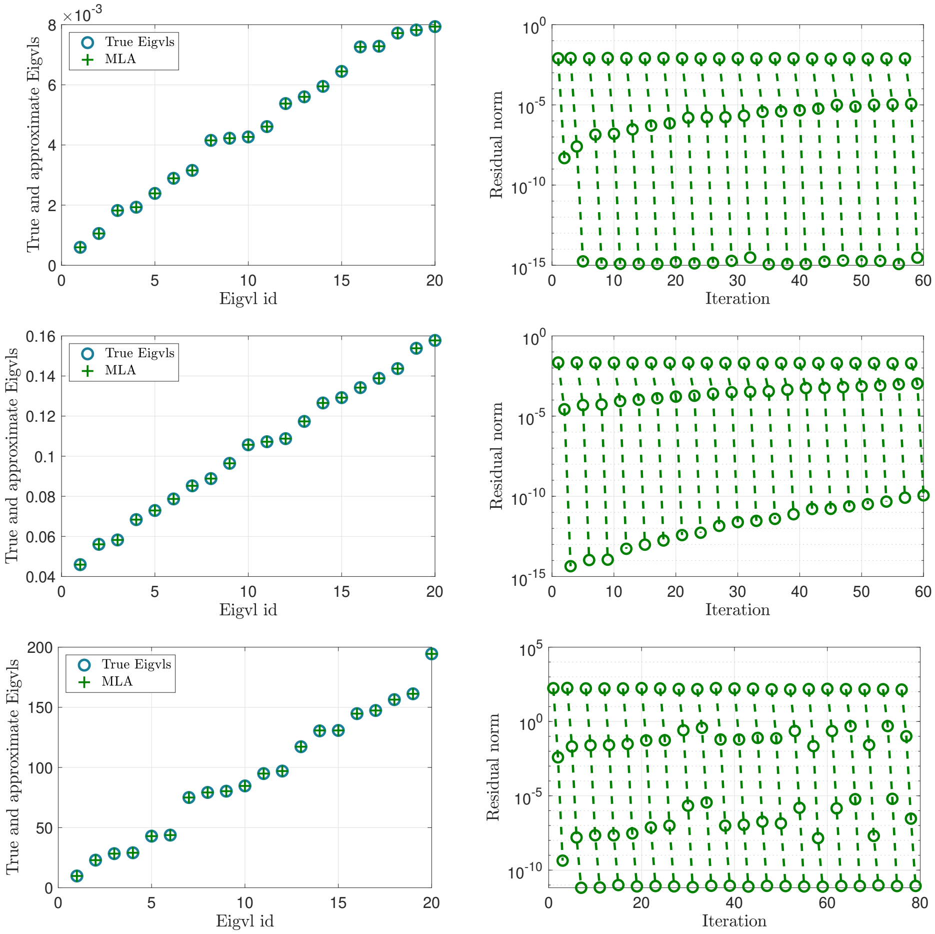

FIG. 5.3. *Left: the n_{ev} sought eigenvalues and their approximation by MLA. Right: the residual norms of the n_{ev} sought eigenpairs at each iteration of MLA. Top figures: $n_x = 253$, $n_y = 148$, and $n_z = 1$. Middle figures: $n_x = 53$, $n_y = 48$, and $n_z = 15$. Bottom figures: "{K/M}uu".*

in this particular example, we had $n = 37\,444$ and $s = 2724$.[5] In summary, MLA managed to compute the two smallest eigenpairs of the 253×148 finite difference discretization of the 2D Dirichlet eigenvalue problem without shift-and-invert in about half of the time required by IRA applied directly to the discretized Dirichlet eigenvalue problem. MLA was faster than IRA even when seeking up to five smallest eigenpairs but lost ground to IRA as the number of sought eigenpairs increased due to the ability of IRA to take advantage of high-dimensional Krylov subspaces. In contrast, the cost of MLA is approximately linear with respect to the value of n_{ev}. A wall-clock time comparison between IRA and MLA is provided in the left subfigure of Figure 5.4.

In agreement with the earlier discussion in the beginning of this section, enabling shift-and-invert leads to faster convergence for IRA. Indeed, computing the $n_{ev} = 2$ (algebraically) smallest eigenvalues of 253×148 finite difference discretization by IRA using a subspace of

[5] Similarly, memory savings were of the order $O(n/s)$ in favor of MLA.

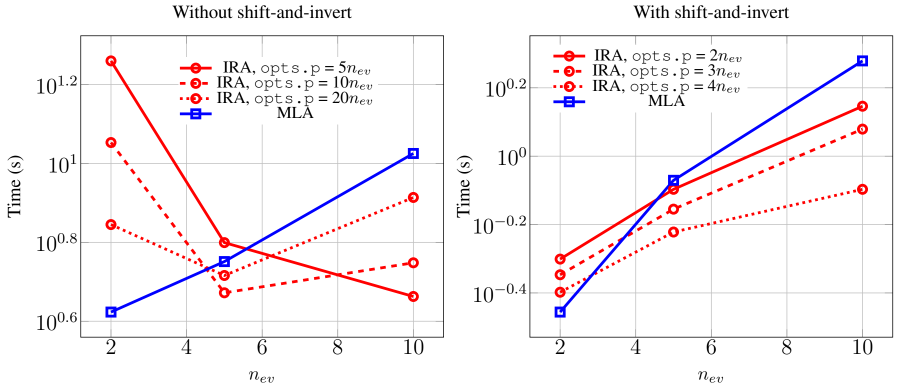

FIG. 5.4. *Wall-clock times to compute the n_{ev} algebraically smallest eigenvalues and associated eigenvectors for the 253×148 five-point finite difference approximation of the Laplace operator on a regular grid with homogeneous Dirichlet boundary conditions. The variable "*`opts.p`*" denotes the maximum Krylov subspace dimension before IRA restarts.*

size $2n_{ev}$ now requires only 7 cycles. A wall-clock time comparison of MLA[6] and IRA, this time with shift-and-invert, is given in the right subfigure of Figure 5.4. Note that "`opts.p`" can be chosen to be much smaller in the case of shift-and-invert. In summary, when shift-and-invert was enabled, MLA was slower than IRA for any $n_{ev} > 2$. Nonetheless, MLA can still be competitive to shift-invert Lanczos in cases where a distributed memory environment is available and the sought eigenvalues are clustered.

The above experiment is by no means exhaustive as both IRA and MLA can take advantage of several modifications and the test matrix used was not large. However, it sheds light in which scenarios MLA might be a competitive approach with respect to Krylov subspace eigenvalue solvers.

5.5. Comparisons against other root-finding eigenvalue solvers.

We now compare the numerical performance of MLA with that of the "Branch Hopping" root-finding technique presented in [11]. This scheme essentially coincides with MLA applied to the pencil $(S(\sigma), I)$ and will be abbreviated as "BrH".

TABLE 5.1
n: size of the matrices A and M, $nnz(.)$: number of nonzero entries. The integers under the label "s" denote the number of interface variables for the default choice of $p = 16$ subdomains.

#	Matrix pencil	n	$nnz(A)/n$	$nnz(M)/n$	s	Application
1.	nos5	468	11.1	1.0	251	Structural
2.	nos3	960	16.5	1.0	310	Structural
3.	bcsst{k,m}27	1,224	45.9	45.9	418	Structural
4.	FEmesh	2,689	6.9	6.8	287	Model problem
5.	FDmesh	5,000	6.6	1.0	433	Model problem
6.	{K,M}uu	7,102	47.9	24.0	911	Structural
7.	Dubcova1	16,129	15.7	1.0	1226	2D/3D

[6] The shift-and-invert variant of MLA uses shift-and-invert only during the computation of eigenpairs of the pencil $(S(\sigma), -S'(\sigma))$.

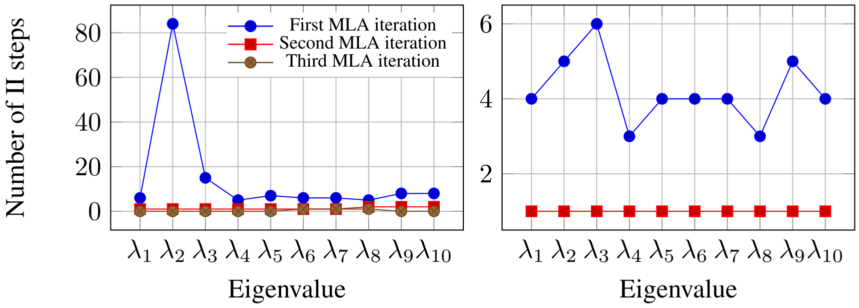

FIG. 5.5. *Total number of steps required by the inverse iteration (II) at each individual iteration of MLA when computing the ten algebraically smallest eigenvalues of the matrices "nos3". The initial eigenvalue approximation σ_0 for each sought eigenvalue λ was set to $\sigma_0 = \lambda(1 + 1e\text{-}2)$ (left) and $\sigma_0 = \lambda(1 + 1e\text{-}3)$ (right).*

We tested MLA and BrH on a few pencils listed in Table 5.1. Without loss of generality we consider the computation of the few algebraically smallest eigenvalues and associated eigenvectors. All matrix pencils but 4) and 5) can be found in the Suite Sparse Matrix Collection [6]. The matrix "FDmesh" is set as in the previous section except that now $n_x = 100$ and $n_y = 50$, respectively. The matrix pencil "FEmesh" represents the pencil shown in Figure 5.2. The matrix pencils 1), 2), and 5) are of the standard form.

TABLE 5.2
Total number of iterations required by MLA and BrH to compute the ten algebraically smallest eigenvalues and the associated eigenvectors of the matrix pencils listed in Table 5.1. The numbers inside the parentheses denotes the number of times each method misconverged.

Partitions	$p = 8$		$p = 16$		$p = 32$	
Matrix	MLA	BrH	MLA	BrH	MLA	BrH
nos5	28	29	21	28	20	26
nos3	27 (1)	30 (1)	27	30	23	27
bcsst{k,m}27	38	39	32	35	28	34
FEmesh	30	43	30	43	25	35
FDmesh	29 (1)	33 (2)	25	29	20	27
{K,M}uu	29 (1)	38 (1)	27	34	26	30
Dubcova1	30 (1)	35 (1)	27	30	21	28

Table 5.2 reports the total number of iterations required by MLA and BrH to compute the ten algebraically smallest eigenvalues and the associated eigenvectors of the matrix pencils listed in Table 5.1 as p varies. It is easy to notice that MLA is faster than BrH, requiring two to three Newton steps per sought eigenpair. Moreover, in agreement with the discussion in Section 4.1, increasing the number of partitions leads to faster convergence.

Figure 5.5 displays the total number of steps required by the inverse iteration to compute the eigenvalue $\theta(\sigma)$ at each iteration of MLA, when the latter is applied to the computation of the ten algebraically smallest eigenvalues and the associated eigenvectors of the matrix "nos3". As the iteration index of MLA increases, the number of inverse iteration steps per iteration decreases since the sought simple eigenvalue (root) λ of (A, M), $\theta(\sigma)$ converges to zero while, at the same time, the rest of the eigenvalues of pencil $(S(\sigma), -S'(\sigma))$ converge to nonzero values (recall that all sought eigenvalues were simple).

6. Conclusion. This paper proposed a root-finding scheme based on domain decomposition to compute eigenpairs of symmetric generalized eigenvalue problems. The proposed scheme, termed as Mixed Linear Approximations (MLA), recasts the original eigenvalue problem into one of computing roots of scalar functions defined by the eigenvalue of smallest magnitude of a generalized eigenvalue problem stemming from a first-order approximation of the non-linear matrix-valued function associated with the interface variables of the adjacency graph of the original pencil. To solve this root-finding problem, MLA considers Newton's iteration. We discussed several theoretical and implementation details and demonstrated by experiments that MLA can converge faster than the root-finding eigenvalue solver in [11].

Several practical details remain to be considered. For example, in our experiments we primarily considered inverse iteration as the inner eigenvalue solver in MLA, however, other approaches such as a (generalized) Davidson method [24] are possible. A comparison against Lanczos methods in a distributed-memory setting is also an idea worth exploring in the future. Furthermore, the trade-off between performing the Newton updates and the linear system solutions inexactly and their effects on the convergence of MLA is the subject of ongoing research. Finally, another interesting research direction would be the adaptation of MLA to the solution of symmetric (Hermitian) nonlinear eigenvalue problems.

Acknowledgments. The majority of the work featured in this manuscript was carried out while the author was with the University of Minnesota and was finalized under the support of the International Business Machines Corporation through its Herman H. Goldstine Postdoctoral Fellowship program. The author is thankful to the referees and associate editor for their time and comments which led to considerable improvements in the material presented in this paper. In addition, the author would like to acknowledge Michele Benzi, Jie Chen, and Yousef Saad for their suggestions and comments on earlier versions of this paper.

Appendix A. Proof of Theorem 3.6.

For any $\sigma \notin \{\Lambda(A,M) \cup \Lambda(B,M_B)\}$ we can write

$$(A - \sigma M)^{-1} = \begin{bmatrix} B_\sigma^{-1} + B_\sigma^{-1} E_\sigma S(\sigma)^{-1} E_\sigma^T B_\sigma^{-1} & -B_\sigma^{-1} E_\sigma S(\sigma)^{-1} \\ -S(\sigma)^{-1} E_\sigma^T B_\sigma^{-1} & S(\sigma)^{-1} \end{bmatrix}. \tag{A.1}$$

In addition, we have

$$\begin{aligned} (A - \sigma M)^{-1} &= \sum_{i=1}^{n} \frac{1}{\lambda_i - \sigma} x^{(i)} \left(x^{(i)}\right)^T \\ &= \sum_{i=1}^{n} \frac{1}{\lambda_i - \sigma} \begin{bmatrix} u^{(i)} \left(u^{(i)}\right)^T & u^{(i)} \left(y^{(i)}\right)^T \\ y^{(i)} \left(u^{(i)}\right)^T & y^{(i)} \left(y^{(i)}\right)^T \end{bmatrix}. \end{aligned} \tag{A.2}$$

Equating the (2,2) blocks of the right-hand sides in (A.1) and (A.2) gives

$$S(\sigma)^{-1} = \sum_{i=1}^{n} \frac{1}{\lambda_i - \sigma} y^{(i)} \left(y^{(i)}\right)^T.$$

Writing $S(\sigma)\hat{y}_j(\sigma) = \theta_j(\sigma)[-S'(\sigma)]\hat{y}_j(\sigma)$ as

$$\frac{1}{\theta_j(\sigma)} \hat{y}_j(\sigma) = -S(\sigma)^{-1} S'(\sigma) \hat{y}_j(\sigma) = \sum_{i=1}^{n} \frac{1}{\lambda_i - \sigma} y^{(i)} \left(y^{(i)}\right)^T [-S'(\sigma)]\, \hat{y}_j(\sigma) \tag{A.3}$$

and defining the scalar function

$$\chi_{i,j}(\sigma) = -\left(y^{(i)}\right)^T S'(\sigma) \hat{y}_j(\sigma),$$

for all $i = 1, \dots, n$ and $j = 1, \dots, s$, we can express (A.3) as

$$\text{(A.4)} \qquad \frac{1}{\theta_j(\sigma)}\hat{y}_j(\sigma) = \frac{\chi_{\kappa,j}(\sigma)}{\lambda_\kappa - \sigma}y^{(\kappa)} + \sum_{i=1,\ i\neq\kappa}^{n} \frac{\chi_{i,j}(\sigma)}{\lambda_i - \sigma}y^{(i)}.$$

Now, let $j \equiv \kappa_\sigma$, and multiply both sides in (A.4) by $(\lambda_\kappa - \sigma)/\chi_{\kappa,\kappa_\sigma}$. Taking limits leads to

$$\text{(A.5)} \qquad \begin{aligned} \lim_{\sigma\to\lambda_\kappa}\left(\frac{\lambda_\kappa - \sigma}{\chi_{\kappa,\kappa_\sigma}(\sigma)\theta(\sigma)}\hat{y}(\sigma)\right) &= y^{(\kappa)} + \lim_{\sigma\to\lambda_\kappa}\left(\sum_{i=1,\ i\neq\kappa}^{n}\frac{\chi_{i,\kappa_\sigma}(\sigma)(\lambda_\kappa - \sigma)}{\chi_{\kappa,\kappa_\sigma}(\sigma)(\lambda_i - \sigma)}y^{(i)}\right) \\ &= y^{(\kappa)}, \end{aligned}$$

where the second equality follows directly from the fact that λ_κ is simple.

Since $\lim_{\sigma\to\lambda_\kappa}\theta(\sigma) \to 0$, the limit of each entry of the vector on the left-hand side of (A.5) is of the form $0/0$. Let ℓ_η denote the ηth, $1 \leq \eta \leq s$, entry of the vector $\hat{y}(\sigma)$. Applying l'Hôspital's rule (where we differentiate with respect to σ) gives

$$\text{(A.6)} \qquad \lim_{\sigma\to\lambda_\kappa}\left(\frac{(\lambda_\kappa - \sigma)\ell_\eta}{\chi_{\kappa,\kappa_\sigma}(\sigma)\theta(\sigma)}\right) = \lim_{\sigma\to\lambda_\kappa}\left(\frac{[(\lambda_\kappa - \sigma)\,\ell_\eta]'}{[\chi_{\kappa,\kappa_\sigma}(\sigma)\theta(\sigma)]'}\right) = \lim_{\sigma\to\lambda_\kappa}\left(\frac{\ell_\eta}{\chi_{\kappa,\kappa_\sigma}(\sigma)}\right),$$

where we used $\lim_{\sigma\to\lambda_\kappa}\theta(\sigma) = \theta(\lambda_k) = 0$ and $\lim_{\sigma\to\lambda_\kappa}\theta'(\sigma) = \theta'(\lambda_k) = -1$. Since both $\chi_{\kappa,\kappa_\sigma}(\sigma)$ and $\theta'(\sigma)$ are non-zero, the above limit is well-defined. Applying (A.6) to all entries of $\hat{y}(\sigma)$ finally gives

$$\lim_{\sigma\to\lambda_\kappa}\left(\frac{1}{\chi_{\kappa,\kappa_\sigma}(\sigma)}\hat{y}(\sigma)\right) = \frac{1}{\chi_{\kappa,\kappa_{\lambda_\kappa}}(\lambda_\kappa)}\hat{y}(\lambda_\kappa) = y^{(\kappa)}.$$

To prove the second item, first notice that the eigenvectors of $(S(\sigma), -S'(\sigma))$ are $S'(\sigma)$-orthogonal, and thus $\hat{y}_j^T(\sigma)S'(\sigma)\hat{y}_{\kappa_\sigma}(\sigma) = 0,\ j \neq \kappa_\sigma$. Recalling the fact that $\text{span}\{\lim_{\sigma\to\lambda_\kappa}\hat{y}_{\kappa_\sigma}(\sigma)\} \equiv \text{span}\left\{y^{(\kappa)}\right\}$ leads to $\lim_{\sigma\to\lambda_\kappa}(\hat{y}_j^T(\sigma)S'(\sigma)y^{(\kappa)}(\sigma)) = 0$ for any $j \neq \kappa_\sigma$. Thus, (A.4) is simplified to

$$\text{(A.7)} \qquad \lim_{\sigma\to\lambda_\kappa}\frac{1}{\theta_j(\sigma)}\hat{y}_j(\sigma) = \lim_{\sigma\to\lambda_\kappa}\left(\sum_{i=1,\ i\neq\kappa}^{n}\frac{\chi_{i,j}(\sigma)}{\lambda_i - \sigma}y^{(i)}\right),$$

for any $(\theta_j(\sigma), \hat{y}_j(\sigma))$, $j = 1, \dots, s$, $j \neq \kappa_\sigma$. Multiplying (A.7) by $\hat{y}_j^T(\sigma)$ from the left and rearranging terms leads to (3.5).

REFERENCES

[1] R. Alam and N. Behera, *Linearizations for rational matrix functions and Rosenbrock system polynomials*, SIAM J. Matrix Anal. Appl., 37 (2016), pp. 354–380.

[2] A. L. Andrew and R. C. E. Tan, *Computation of derivatives of repeated eigenvalues and the corresponding eigenvectors of symmetric matrix pencils*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 78–100.

[3] H. Avron, E. Ng, and S. Toledo, *A generalized Courant-Fischer minimax theorem*, Tech. Rep., Lawrence Berkeley National Laboratory, Berkeley, 2008.

[4] C. Bekas and Y. Saad, *Computation of smallest eigenvalues using spectral Schur complements*, SIAM J. Sci. Comput., 27 (2006), pp. 458–481.

[5] J. K. Bennighof and R. B. Lehoucq, *An automated multilevel substructuring method for eigenspace computation in linear elastodynamics*, SIAM J. Sci. Comput, 25 (2004), pp. 2084–2106.

[6] T. A. Davis and Y. Hu, *The University of Florida sparse matrix collection*, ACM Trans. Math. Softw., 38 (2011),Art. 1, 25 pages.

[7] S. C. EISENSTAT AND H. F. WALKER, *Choosing the forcing terms in an inexact Newton method*, SIAM J. Sci. Comput., 17 (1996), pp. 16–32.

[8] I. C. F. IPSEN, *Computing an eigenvector with inverse iteration*, SIAM Review, 39 (1997), pp. 254–291.

[9] V. KALANTZIS, *Domain Decomposition Algorithms for the Solution of Sparse Symmetric Generalized Eigenvalue Problems*, PhD. Thesis, Computer Science and Engineering Department, University of Minnesota, Minneapolis, 2018.

[10] V. KALANTZIS, J. KESTYN, E. POLIZZI, AND Y. SAAD, *Domain decomposition approaches for accelerating contour integration eigenvalue solvers for symmetric eigenvalue problems*, Numer. Linear Algebra Appl., 25 (2018), Art. e2154, 19 pages.

[11] V. KALANTZIS, R. LI, AND Y. SAAD, *Spectral Schur complement techniques for symmetric eigenvalue problems*, Electron. Trans. Numer. Anal. 45 (2016), pp. 305–329.
http://etna.ricam.oeaw.ac.at/vol.45.2016/pp305-329.dir/pp305-329.pdf

[12] V. KALANTZIS, Y. XI, AND Y. SAAD, *Beyond automated multilevel substructuring: domain decomposition with rational filtering*, SIAM J. Sci Comput., 40 (2018), pp. C477–C502.

[13] G. KARYPIS AND V. KUMAR, *A fast and high quality multilevel scheme for partitioning irregular graphs*, SIAM J. Sci. Comput., 20 (1998), pp. 359–392.

[14] C. KELLEY, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, 1995.

[15] J. KESTYN, V. KALANTZIS, E. POLIZZI, AND Y. SAAD, *PFEAST: A high performance sparse eigenvalue solver using distributed-memory linear solvers*, in Proceedings of the ACM/IEEE Supercomputing Conference (SC16), IEEE Conference Proceedings, Los Alamitos, 2016, pp. 178–189

[16] R. B. LEHOUCQ, D. C. SORENSEN, AND C. YANG, *ARPACK Users' Guide*, SIAM, Philadelphia, 1998.

[17] S. LUI, *Kron's method for symmetric eigenvalue problems*, J. Comput. Appl. Math., 98 (1998), pp. 35–48.

[18] ———, *Domain decomposition methods for eigenvalue problems*, J. Comput. Appl. Math., 117 (2000), pp. 17–34.

[19] B. PARLETT, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, 1998.

[20] B. PHILIPPE AND Y. SAAD, *On correction equations and domain decomposition for computing invariant subspaces*, Comput. Methods Appl. Mech. Engrg., 196 (2007), pp. 1471–1483.

[21] Y. SAAD, *Numerical Methods for Large Eigenvalue Problems*, SIAM, Philadelphia, 2011.

[22] H. D. SIMON, *The Lanczos algorithm with partial reorthogonalization*, Math. Comp, 42 (1984), pp. 115–142.

[23] B. F. SMITH, P. E. BJØRSTAD, AND W. D. GROPP, *Domain Decomposition*, Cambridge University Press, Cambridge, 1996.

[24] A. STATHOPOULOS AND J. R. MCCOMBS, *PRIMME: PReconditioned Iterative MultiMethod Eigensolver—methods and software description*, ACM Trans. Math. Softw., 37 (2010), Art. 21, 30 pages.

[25] A. TOSELLI AND O. WIDLUND, *Domain Decomposition Methods—Algorithms and Theory*, Springer, Berlin, 2005.