# An SVD-approach to Jacobi-Davidson solution of nonlinear Helmholtz eigenvalue problems

M. A. Botchev[*] G. L. G. Sleijpen[†] A. Sopaheluwakan[‡]

February 17, 2009

*To Henk van der Vorst*
*for all his seminal contributions to Numerical Mathematics*
*and his support to his colleagues and our community*

## Abstract

Numerical solution of the Helmholtz equation in an infinite domain often involves restriction of the domain to a bounded computational window where a numerical solution method is applied. On the boundary of the computational window artificial transparent boundary conditions are posed, for example, widely used perfectly matched layers (PMLs) or absorbing boundary conditions (ABCs). Recently proposed transparent-influx boundary conditions (TIBCs) resolve a number of drawbacks typically attributed to PMLs and ABCs, such as introduction of spurious solutions and the inability to have a tight computational window. Unlike the PMLs or ABCs, the TIBCs lead to a nonlinear dependence of the boundary integral operator on the frequency. Thus, a nonlinear Helmholtz eigenvalue problem arises.

This paper presents an approach for solving such nonlinear eigenproblems which is based on a truncated singular value decomposition (SVD) polynomial approximation of the nonlinearity and subsequent solution of the obtained approximate polynomial eigenproblem with the Jacobi-Davidson method. The suggested truncated SVD polynomial approximation seems to be of interest on its own. It can be applied in combination with existing eigensolvers to the problems where the nonlinearity is expensive to evaluate or not explicitly given.

*2000 Mathematics Subject Classification:* 65F15, 65F30, 35J05
*Keywords:* truncated SVD, Helmholtz equation, nonlinear eigenvalue problems, Jacobi-Davidson method, transparent boundary conditions, proper orthogonal decomposition (POD), principal component approximation (PCA)

[*] Corresponding author. Dept. Applied Mathematics, University of Twente, P.O.Box 216, 7500 AE Enschede, `mbotchev@na-net.ornl.gov`. This author's research was supported in part by the Dutch government through the national program BSIK: knowledge and research capacity, in the ICT project BRICKS (`http://www.bsik-bricks.nl`), theme MSV1.

[†] Mathematical Institute, Utrecht University, P.O.Box 80.010, 3508 TA Utrecht, the Netherlands, `sleijpen@math.uu.nl`.

[‡] LabMath-Indonesia, Jl.Anatomi No.19, Bandung 40191, Indonesia, `a.sopaheluwakan@math.utwente.nl`, `ardhasena@labmath-indonesia.or.id`.

# 1 Introduction

This paper deals with nonlinear, nonpolynomial eigenvalue problems stemming from discretized Helmholtz problems. Posed in an unbounded domain (in this paper $\mathbb{R}^2$), the Helmholtz equation reads

$$\Delta E + \omega^2 n^2(x,z)E = 0, \qquad (x,z) \in \mathbb{R}^2, \tag{1}$$

where $\Delta$ is the Laplacian, $E(x,z)$ is the unknown field (in the case of the Transverse Electric (TE) formulation and the harmonic time dependence $e^{-i\omega t}$, $E$ defines the only nonzero component of the electric field as $\vec{E} = (0, e^{-i\omega t}E(x,z), 0)$), $\omega$ is the unknown eigenfrequency and $n(x,z)$ is the refraction index. The refraction index is space-dependent to account for different materials of which the domain consists. Solving problem (1) numerically, we restrict the infinite domain to a bounded domain $\Omega$ encompassing the device. On the boundary $\partial\Omega$ special artificial boundary conditions then have to be posed which should guarantee transparency of the boundary for outgoing waves. Many different boundary conditions have been devised for this purpose, known as transparent-influx (TIBC), nonreflecting and absorbing boundary conditions (see e.g. Chapters 6 and 7 in [23] and [8, 2, 10, 3, 9]). All of them can be divided into two groups, namely non-local and local boundary conditions [10, 16]. Citing [16], "concerning non-local TIBC, the main ingredient is the Dirichlet-to-Neumann operator... which is typically exact, whereas local TIBC are usually approximate." The TIBC used in this paper are the non-local ones from [16]. Unlike the local conditions, the non-local TIBC usually do not require special tuning (see e.g. [4, 14]) and are known not to yield spurious solutions [16, 22]. In addition, the TIBC from [16] allow to obtain, by analytical calculations outside the computational domain, the solution on the whole plane. However, the application of the non-local TIBC leads to a nonlinear, nonpolynomial dependence of the discretized Helmholtz operator on the frequency, so that a nonlinear eigenvalue problem has to be solved.

In this paper, an approach is proposed for numerical solution of this eigenproblem. The approach can be sketched as follows. Since the boundary conditions are essentially a boundary (and, hence, lower dimensional) operator, the nonlinearity appears in the problem as a very sparse matrix depending nonlinearly on the frequency. At the first step, these very sparse matrices are sampled for different values of the frequency in a range of interest. The obtained data are then approximated with high accuracy through the truncated singular value decomposition (SVD) (see e.g. [11]) and a least-square polynomial fit. This leads to a polynomial approximation of the nonlinear contributions in the matrix of the eigenproblem. The approximate polynomial eigenproblem is solved with the Jacobi-Davidson method [21] which is readily applicable to large-scale polynomial eigenproblems [20, 19]. Note that the proposed truncated SVD approximation is of interest on its own, since it can also be used in combination with other eigensolvers (see Remark at the end of Section 3.2).

The remainder of this paper is organized as follows. We formulate the problem in Section 2, the truncated SVD approximation is then described in Section 3, Section 4 presents results of numerical experiments and, finally, some conclusions are drawn in Section 5.

# 2 Problem formulation

We are interested in numerical eigenmode solution of Helmholtz equation (1) posed in an infinite domain (in this paper $\mathbb{R}^2$) and supplied with TIBCs meant to restrict the domain to a finite computational window $\Omega$. The TIBCs we use, proposed in [16], result from a hybrid

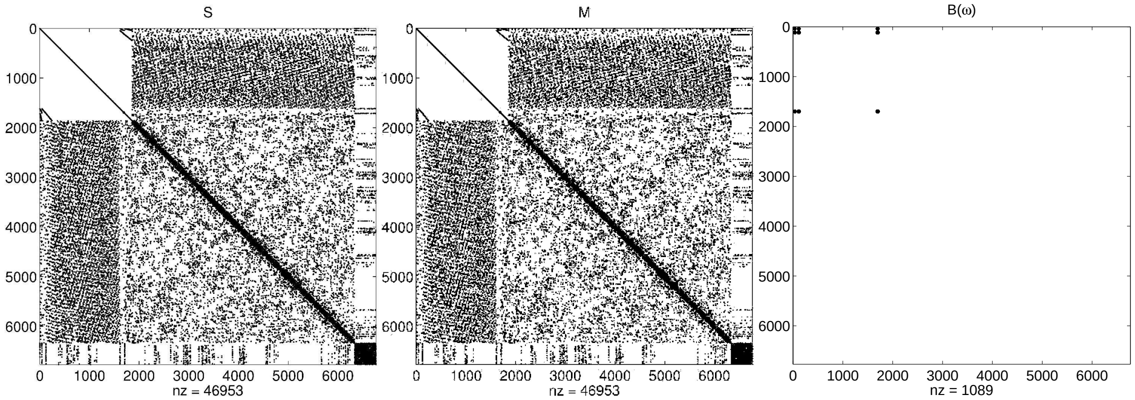

Figure 1: Sparsity patterns of the matrices $S$, $M$ and $B(\lambda)$ for a FEM mesh with $n = 6\,769$ degrees of freedom. Below the plots, the numbers of nonzero entries for each matrix are given.

analytic-numeric method designed for rectangular computational domains $\Omega$. The approach behind TIBCs, which is based on plane wave decomposition and a construction of a suitable Dirichlet-to-Neumann operator [16, 22], can be sketched as follows. The TIBCs are posed for a given influx $E^{\text{in}}|_{\partial\Omega}$ into $\Omega$ from the exterior and allow to calculate numerically the solution $E_\Omega$ inside $\Omega$. Next, based on the computed solution $E_\Omega$, the exterior outgoing field $E^{\text{out}}$ can be analytically determined from the Dirichlet data $E_\Omega|_{\partial\Omega} - E^{\text{in}}|_{\partial\Omega}$. Once $E^{\text{out}}$ is found, the solution $E^{\text{ext}}$ in the whole exterior domain is available as $E^{\text{ext}} = E^{\text{in}} + E^{\text{out}}$. The interior solution $E_\Omega$ and the exterior solution $E^{\text{ext}}$ can be shown to satisfy the correct continuity conditions on $\partial\Omega$ [16], thus producing a rigorous Helmholtz solution for the whole plane.

The TIBCs enter a weak formulation of Helmholtz equation (1) as follows [16]:

$$\int_\Omega \left(\nabla v \cdot \nabla E_\Omega - \omega^2 n^2 v E_\Omega\right) \mathrm{d}s - \int_{\partial\Omega} v \partial_n E^{\text{ext}} \mathrm{d}l = 0, \quad \forall v \in \mathrm{H}^1(\Omega), \tag{2a}$$

$$\partial_n E^{\text{ext}} = D^+(E_\Omega|_{\partial\Omega}) + D^-(E^{\text{in}}|_{\partial\Omega}) - D^+(E^{\text{in}}|_{\partial\Omega}), \tag{2b}$$

where $D^\pm$ are Dirichlet-to-Neumann (or Poincaré-Steklov) operators mapping a function defined on $\partial\Omega$ to the normal derivative on $\partial\Omega$ of the solution of the Dirichlet boundary value problem:

$$D^+(g) = \partial_n u|_{\partial\Omega}, u \text{ is outgoing solution of (1) with } u|_{\partial\Omega} = g,$$
$$D^-(g) = \partial_n u|_{\partial\Omega}, u \text{ is ingoing solution of (1) with } u|_{\partial\Omega} = g.$$

Details of definition and numerical implementation of these Dirichlet-to-Neumann operators can be found in [16, 22]. Here, we only give some more explanation concerning the test problem presented in Section 4. The relation (2b) gives a general form of the TIBCs. For the test problem (so-called "leaky mode computations"), the ingoing field $E^{\text{in}}$ is set to zero, so that the TIBCs (2b) reduce to

$$\partial_n E^{\text{ext}} = D^+(E_\Omega|_{\partial\Omega}). \tag{3}$$

Let $\Gamma \subset \partial\Omega$ be a straight wall of the computational window $\Omega$, with $z = \text{const}$ on $\Gamma$ and $E_\Omega|_\Gamma = g(x) = \int_{-\infty}^{+\infty} \hat{g}(k) e^{ikx} \mathrm{d}k$. Then one can show [22] that

$$D^+(E_\Omega|_\Gamma) = i \int_{-\infty}^{+\infty} \sqrt{\omega^2 n^2 - k^2}\, \hat{g}(k) e^{ikx} \mathrm{d}k, \tag{4}$$

Table 1: Number of nonzero entries in the matrices $S$ and $M$ and in the matrix $B(\lambda)$ for three FEM meshes

| | mesh 1 | mesh 2 | mesh 3 |
|---|---|---|---|
| number of degrees of freedom, $n$ | 6 769 | 26 861 | 107 017 |
| number of nonzero entries in $S$ and $M$ | 46 953 | 187 173 | 747 417 |
| number of nonzero entries in $B$, $N$ | 1 089 | 4 225 | 16 641 |

which illustrates that the frequency $\omega$ enters the boundary integral of (2a) in a nonlinear, nonrational way. Discretization of (2) by a finite element method (FEM) then leads to a nonlinear eigenvalue problem

$$(S - \lambda M + B(\lambda))\, x = 0, \quad \lambda \equiv \omega^2, \quad S, M \in \mathbb{R}^{n\times n}, \quad B(\lambda) \in \mathbb{C}^{n\times n}, \tag{5}$$

where $n$ is the number of degrees of freedom in the FEM, $S$ and $M$ are respectively stiffness and mass matrices approximating the first integral term in (2a) and $B(\lambda)$ is a discrete version of the boundary integral in the same equation. Note that the eigenvector $x$ in (5) should not be confused with the spatial coordinate variable in (1). The nonrational dependency of the boundary integral on the frequency $\lambda \equiv \omega^2$ is inherited by the matrix $B(\lambda)$ whose dependence on $\lambda$ can not, at least, easily, be expressed by an explicit formula. Being an approximation of a lower dimensional operator, $B(\lambda)$ is yet much sparser than the sparse stiffness and mass matrices (see Figure 1 and Table 1). The matrices $S$ and $M$ are real symmetric, whereas the matrices $B(\lambda)$ are not symmetric and, depending on the value of $\lambda$, can have complex entries.

# 3 Polynomial approximation through truncated SVD

One of the solvers directly applicable to large scale nonlinear eigenvalue problems is the Jacobi-Davidson (JD) method [21, 20, 19, 6, 26] (for a detailed overview of eigenvalue solvers, including ones for nonlinear problems, see [1]). The JD method is a subspace projection method and, as such, essentially involves three steps: projection onto a subspace of a moderate dimension, solution of a small scale projected eigenvalue problem and extension of the search subspace. It is desired (though, in general, not required) in the JD algorithm that the nonlinearity of the eigenvalue problem is explicitly given. This information is exploited in the solution of the projected problem. For example, if a large scale polynomial eigenvalue problem

$$(A_0 + \lambda A_1 + \lambda^2 A_2 + \cdots + \lambda^p A_p)x = 0, \quad \text{for given } A_0, A_1, \ldots, A_p \in \mathbb{C}^{n\times n}, \tag{6}$$

is solved then each iteration of the JD method involves solution of the following projected problem

$$(H_0 + \lambda H_1 + \lambda^2 H_2 + \cdots + \lambda^p H_p)y = 0, \quad H_0, H_1, \ldots, H_p \in \mathbb{C}^{k\times k},\ k \ll n. \tag{7}$$

In this paper, we propose an approach for solving nonlinear eigenvalue problem (5) which is based on reduction of the problem to a polynomial eigenvalue problem of the form (6). Once such a reduction is done, the JD method can readily be applied [20, 19]. Needless to say, other eigensolvers can be employed either for the approximate polynomial eigenproblem or directly for the nonlinear eigenproblem (see Remark at the end of Section 3.2 and numerical tests in Section 4.3).

## 3.1 Truncated SVD approximation

To reduce the problem (5) to a polynomial eigenvalue problem, we use a low-rank approximation of $B(\lambda)$ obtained by the singular value decomposition (SVD), see e.g. [11]. The approximation can be computed as follows. First, $n_s$ samples $B(\lambda_i)$, $i = 1, \ldots, n_s$, of the matrix $B(\lambda)$ are computed for frequencies $\lambda_1, \ldots, \lambda_{n_s}$ lying in a region of interest in the complex plain $\mathbb{C}$. Such a frequency region is usually known beforehand or may be obtained by some analytical considerations [22]. The number of samples $n_s$ is usually taken between say 10 and 30. We discuss further how to choose $n_s$ in Section 4. The nonzero entries of the sample matrices are put into columns of a matrix $\mathcal{B} \in \mathbb{C}^{N \times n_s}$ such that the $j$-th column of $\mathcal{B}$ contains all the nonzero entries of the sample $B(\lambda_j)$. The order of the nonzero sample entries in each column is arbitrary but must, of course, be the same for all the columns. Recall that, as evidenced by Figure 1 and Table 1, the number $N$ of nonzero entries in $B(\lambda)$ is always much smaller than the number of nonzero entries in the sparse mass and stiffness matrices.

Next, let $b_j$ be the $j$-th column of $\mathcal{B}$ and let $\tilde{\mathcal{B}} \in \mathbb{C}^{N \times n_s}$ be a matrix obtained by column averaging of the matrix $\mathcal{B}$, i.e., $\tilde{\mathcal{B}}$ contains $n_s$ identical columns $b_0 = \frac{1}{n_s}(b_1 + \cdots + b_{n_s})$. We compute the thin SVD of the difference

$$\mathcal{B} - \tilde{\mathcal{B}} = U\Sigma V^*, \qquad U \in \mathbb{C}^{N \times n_s}, \quad \Sigma, V \in \mathbb{C}^{n_s \times n_s}, \tag{8}$$

where the matrices $U$ and $V$ have orthonormal columns and $\Sigma$ is a diagonal matrix with diagonal entries

$$\sigma_1 \geqslant \sigma_2 \geqslant \ldots \geqslant \sigma_{n_s - 1} \geqslant \sigma_{n_s} = 0.$$

Note that $\sigma_{n_s} = 0$ since, by construction of $\tilde{\mathcal{B}}$, at least one column of $\mathcal{B} - \tilde{\mathcal{B}}$ is a linear combination of the other columns, so that the matrix $\mathcal{B} - \tilde{\mathcal{B}}$ has rank $n_s - 1$ at most. The SVD relation (8) can be rewritten for the columns $b_j$ of $\mathcal{B}$ as

$$b_j = b_0 + (\sigma_1 v_{j,1})u_1 + (\sigma_2 v_{j,2})u_2 + \cdots + (\sigma_{n_s} v_{j,n_s})u_{n_s}, \quad j = 1, \ldots, n_s, \tag{9}$$

making explicit that every column of $\mathcal{B} - \tilde{\mathcal{B}}$ is a linear combination of the columns of $U$. Note that, since the column $b_j$ is a reshaped matrix $B(\lambda_j)$, (9) can be written in the matrix form

$$B(\lambda_j) = B_0 + (\sigma_1 v_{j,1})U_1 + (\sigma_2 v_{j,2})U_2 + \cdots + (\sigma_{n_s} v_{j,n_s})U_{n_s}, \quad j = 1, \ldots, n_s, \tag{9$'$}$$

where the matrices $B_0$ and $U_j$ are respectively the column-vectors $b_0$ and $u_j$ cast into the matrix form. By truncating the expansion in (9) approximations to the columns $b_j$ can be obtained (such approximations are widely used e.g. in statistics, model reduction, data analysis and signal processing [5, 7, 13, 12]):

$$b_j \approx b_j^{(m)} \equiv b_0 + (\sigma_1 v_{j,1})u_1 + (\sigma_2 v_{j,2})u_2 + \cdots + (\sigma_m v_{j,m})u_m, \quad j = 1, \ldots, n_s, \quad m < n_s - 1. \tag{10}$$

The vectors $u_1, \ldots, u_m$ can then be seen as "principal components" which are characteristic for the data set represented by the columns $b_1, \ldots, b_{n_s}$.

It is rather straightforward to obtain the following estimates for the error in approximation (10):

$$\begin{aligned}
\|b_j - b_j^{(m)}\|_2^2 &\leqslant C_j(\sigma_{m+1}^2 + \cdots + \sigma_{n_s-1}^2) \\
&\leqslant \max_{1\leqslant i\leqslant n_s} C_i(\sigma_{m+1}^2 + \cdots + \sigma_{n_s-1}^2) \\
&\leqslant (\sigma_{m+1}^2 + \cdots + \sigma_{n_s-1}^2) \\
&\leqslant (n_s - 1 - m)\sigma_{m+1}^2, \quad j = 1, \ldots, n_s, \\
\text{with} \quad C_i &= \max_{m+1\leqslant k\leqslant n_s-1} |v_{i,k}|^2 \leqslant 1.
\end{aligned} \tag{11}$$

Indeed, since $\sigma_{n_s} = 0$ and the columns $u_j$ of $U$ are orthonormal, we have

$$\begin{aligned}
\|b_j - b_j^{(m)}\|_2^2 &= \|(\sigma_{m+1}v_{j,m+1})u_{m+1} + (\sigma_{m+2}v_{j,m+2})u_{m+2} + \cdots + (\sigma_{n_s-1}v_{j,n_s-1})u_{n_s-1}\|_2^2 \\
&= |\sigma_{m+1}v_{j,m+1}|^2 + |\sigma_{m+2}v_{j,m+2}|^2 + \cdots + |\sigma_{n_s-1}v_{j,n_s-1}|^2,
\end{aligned}$$

where $|v_{j,k}| \leqslant 1$ since the matrix $V$ is unitary. From this, estimates in (11) immediately follow. Note that the estimates in (11) are easily computable and allow to choose $m$ such that $\|b_j - b_j^{(m)}\|_2$ does not exceed a certain tolerance. One practical criterion for choosing $m$ is to take $m$ such that

$$\sigma_{m+1}^2 < \delta^2(n_s - 1 - m),$$

which guarantees that $\|b_j - b_j^{(m)}\|_2 < \delta$.

We remark that if the vectors $b_1, \ldots, b_{n_s}$ are either very big or very small in norm, it might be sensible to compute the truncated SVD approximation (8),(10) for the normalized data set $\mathcal{B} = [\hat{b}_1, \ldots, \hat{b}_{n_s}]$, with $\hat{b}_j = b_j/\|b_j\|_2$. In this case the approximation in (10) changes as

$$\begin{aligned}
b_j &= \|b_j\|_2 \cdot \hat{b}_j \approx \|b_j\|_2 \cdot \hat{b}_j^{(m)} \\
&= \|b_j\|_2 \big[b_0 + (\sigma_1 v_{j,1})u_1 + (\sigma_2 v_{j,2})u_2 + \cdots + (\sigma_m v_{j,m})u_m\big], \\
&\qquad\qquad j = 1, \ldots, n_s, \quad m < n_s - 1,
\end{aligned} \tag{12}$$

so that the estimates in (11) are directly applicable to the relative error norm

$$\frac{\|b_j - b_j^{(m)}\|_2}{\|b_j\|_2} = \frac{\|b_j\|_2 \cdot \|\hat{b}_j - \hat{b}_j^{(m)}\|_2}{\|b_j\|_2} = \|\hat{b}_j - \hat{b}_j^{(m)}\|_2.$$

Finally, we note that the computation of the average $b_0$ in the algorithm above can be skipped (more precisely, we can set $b_0$ and $\tilde{\mathcal{B}}$ to zero) without losing in approximation quality. We prefer to include $b_0$ since in this case the computed terms preserve their statistical meaning (see e.g. Chapter 7 in [13]).

## 3.2 From truncated SVD to a polynomial approximation

The next step is crucial in the whole approximation procedure. We consider the samples $b_j \in \mathbb{C}^N$ as the values of some vector function $b(\lambda)$, with $b(\lambda_j) := b_j$, and rewrite (10) as

$$\begin{aligned}
\text{entries}(B(\lambda_j)) = b(\lambda_j) \approx b^{(m)}(\lambda_j) &= b_0 + (\sigma_1 v_{j,1})u_1 + \cdots + (\sigma_m v_{j,m})u_m, \\
&= b_0 + f_1(\lambda_j)u_1 + \cdots + f_m(\lambda_j)u_m, \\
&\qquad\qquad j = 1, \ldots, n_s, \quad m < n_s - 1,
\end{aligned}$$

Table 2: Algorithm of the truncated SVD polynomial approximation to reduce nonlinear eigenproblem (5) to polynomial eigenproblem (6)

```
(1)  compute $n_s$ samples $B(\lambda_i)$, $i = 1, \ldots, n_s$ for $\lambda_i$ in the region of interest,
     choose $m < n_s$ and $p$ (usually $p \leqslant 4$)

(2)  put nonzero entries of each sample $B(\lambda_i)$ into column $b_i$ of $\mathcal{B} = [b_1, \ldots, b_{n_s}]$
     compute $\tilde{\mathcal{B}} = [b_0, \ldots, b_0] \in \mathbb{C}^{N \times n_s}$ with $n_s$ identical columns $b_0 := \frac{1}{n_s} \sum_{k=1}^{n_s} b_k$

(3)  compute the thin SVD of $\mathcal{B} - \tilde{\mathcal{B}}$: $\mathcal{B} - \tilde{\mathcal{B}} = U \Sigma V^*$, $U \in \mathbb{C}^{N \times n_s}$, $\Sigma, V \in \mathbb{C}^{n_s \times n_s}$

(4)  compute $p$-order least square polynomials $P_p^{(1)}(\lambda), \ldots, P_p^{(m)}(\lambda)$:
     $P_p^{(1)}(\lambda_j) \approx (\sigma_1 v_{j,1}), \ldots, P_p^{(m)}(\lambda_j) \approx (\sigma_m v_{j,m})$, $j = 1, \ldots, n_s$

(5)  approximation: $\text{entries}(B(\lambda)) \approx b_0 + P_p^{(1)}(\lambda) u_1 + \cdots + P_p^{(m)}(\lambda) u_m$
     or, collecting powers of $\lambda$, $B(\lambda) \approx B_{\text{svd}}(\lambda) = B_0 + \lambda B_1 + \lambda^2 B_2 + \cdots + \lambda^p B_p$

(6)  set $A_0 := B_0 + S$, $A_1 := B_1 - M$, $A_2 := B_2, \ldots, A_p := B_p$
```

where the expansion coefficients $\sigma_1 v_{j,1}, \ldots, \sigma_m v_{j,m}$ are again seen as the values of some functions $f_j(\lambda)$. These functions are approximated by the polynomial least squares fit at $\lambda_j$, so that polynomials $P_p^{(j)}(\lambda)$ of degree $p$ are obtained such that

$$\text{entries}(B(\lambda)) \approx b^{(m)}(\lambda) \approx b_0 + P_p^{(1)}(\lambda) u_1 + \cdots + P_p^{(m)}(\lambda) u_m. \tag{13}$$

Thus, a polynomial approximation to the matrix function $B(\lambda)$ is obtained, valid for frequencies $\lambda$ in the range of interest. Collecting in (13) the terms corresponding to different powers of $\lambda$ we arrive at

$$B(\lambda) \approx B_{\text{svd}}(\lambda) = B_0 + \lambda B_1 + \lambda^2 B_2 + \cdots + \lambda^p B_p, \tag{14}$$

where the sparse $n \times n$ matrix $B_0$ contains not only the entries of $b_0 \in \mathbb{C}^N$ but also zero degree terms of all the vectors $P_p^{(1)}(\lambda) u_1, \ldots, P_p^{(m)}(\lambda) u_m$. It turns out in the experiments (see Section 4) that it suffices to take $p$ small, say upto 4. Moreover, if necessary, a higher accuracy can always be attained be tightening the frequency range in the course of Jacobi-Davidson iterations.

We have summarized the algorithm of the truncated SVD approximation in Table 2.

**Remark** We note that, once the truncated SVD approximation (14) is built, as an alternative to the Jacobi-Davidson method, one can also apply linearization to the obtained polynomial eigenproblem and arrive to a standard generalized eigenvalue problem of order $np$ (see e.g. [1], Section 9.2.2 or [24]). Taking into account that the matrices appearing in (14) contain a lot of zero rows and columns (see Figure 1), the size of this eigenproblem could be reduced to $n + (p-1)n_B$, with $n_B$ being the number of nonzero rows and columns in $B(\lambda)$. We have chosen to use the Jacobi-Davidson method because it is able to handle polynomial eigenproblems of arbitrary order directly, without any additional transformation (such as linearization). The question whether linearization in combination with other eigensolvers can be computationally attractive, is left beyond the scope of this paper.

### 3.3 Costs of truncated SVD approximation

The main computational costs are spent for the thin SVD in (8) and amount to $\mathcal{O}(Nn_s^2 + n_s^3)$ flops (see e.g. Section 5.4.5 in [11]). These costs, thus, grow linearly with the number of nonzero entries in $B(\lambda)$. As discussed earlier, this number is always much smaller than the number of nonzero entries in the stiffness and mass matrices. The costs for the least squares fit of $m$ scalar functions in (13) are negligible with respect to the costs for the thin SVD.

### 3.4 SVD-Jacobi-Davidson method

The SVD-Jacobi-Davidson algorithm essentially involves two steps. First, the nonlinear operator $B(\lambda)$ is approximated by a matrix polynomial via the truncated SVD approximation (cf. (14) and Table 2) and nonlinear eigenproblem (5) is thus reduced to polynomial eigenproblem (6). Second, the Jacobi-Davidson eigensolver for polynomial eigenproblems is applied to (6). The first step (SVD approximation) is almost negligible in costs as compared to the second step. Indeed, the costs for the SVD approximation grow only linearly with the number $N$ of the nonzero entries in $B(\lambda)$ (see Section 3.3 and Table 1). The Jacobi-Davidson eigensolver for polynomial eigenproblems is outlined in Table 3 (cf. [19] and Algorithm 9.1 in [1]). For implementation details we refer to Section 4.3.

## 4 Numerical experiments

### 4.1 Test problem

The test problem, taken from [22], Chapter 6, and coming from the field of integrated optics, arises in modeling of the photonic crystal microcavities. Here one is interested in computation of the so-called leaky modes of a photonic crystal (see e.g. the same chapter in [22]). The leaky modes are solutions of the Helmholtz problem with the influx $E^{\text{in}}$ set to zero in (2), which simplifies the TIBCs to (3). The corresponding weak formulation can then be obtained as explained in Section 2 (for a detailed description see [22], Section 6.5).

The photonic crystal in this test problem consists of $5 \times 5$ rods with the refractive index $n_r = 3.4$ with one thicker "defect" rode in the centrum forming the cavity (see Figure 2). Due to the symmetry of the chosen geometry, the computations were performed in a quarter of the square computational window (see the left plot in Figure 2). The eigenvalues of interest for this problem have small imaginary parts which correspond to a weak damping of the modes (which gives the name "leaky modes"). For the numerical tests presented in Section 4.3 the eigenvalue is approximately $\sqrt{\lambda} \equiv \omega \approx 2\pi(0.33510 + 0.00034\mathrm{i})$. For further details of the test problem we refer to [22], Chapter 6.

All numerical tests presented in this section were done in Matlab on a computer with a 2 GHz core-duo processor and 3.2 Gb memory.

### 4.2 Performance of the truncated SVD approximation

In this section we test the performance of the truncated SVD approximation (14). We use three FEM meshes with parameters presented in Table 1.

In this test problem, the values $\lambda_i$ used for the samples were taken real, since we are interested in real eigenvalues with very small imaginary parts. As the results of numerical experiments in this section suggest, the approximation quality is uniform for the whole

Table 3: Algorithm of the Jacobi-Davidson method for polynomial eigenproblem (6)

```
(1)  choose initial subspace $\mathsf{colspan}(V)$, $V \in \mathbb{C}^{n\times k}$
(2)  for $i = 0, \ldots, p$
         compute $W_i := A_i V$ and $H_i := V^* W_i$
     endfor
iterate until convergence
(3)  compute the desired eigenpair $(\theta, y)$ of the small projected problem
     $(H_0 + \theta H_1 + \theta^2 H_2 + \cdots + \theta^p H_p)y = 0$, $\|y\|_2 = 1$,
     set $u := Vy$, $w := (A_0 + \theta A_1 + \cdots + \theta^p A_p)'_\theta u$
(4)  compute residual $r := (A_0 + \theta A_1 + \cdots + \theta^p A_p)u$
(5)  if ($\|r\|_2 \leqslant$ tolerance) then
         $x := u$, $\lambda := \theta$, convergence, stop
     end if
(6)  approximately solve a $t$ from
     $\left(I - \dfrac{wu^*}{u^*w}\right)(A_0 + \theta A_1 + \cdots + \theta^p A_p)(I - uu^*)t = -r$
(7)  orthogonalize $t$ against $V$ with modified Gram-Schmidt, $v := t/\|t\|_2$
(8)  for $i = 0, \ldots, p$
         compute $w_i := A_i v$, expand $H_i := \begin{bmatrix} H_i & V^* w_i \\ v^* W_i & v^* w_i \end{bmatrix}$, $W_i := [W_i, w_i]$
     endfor
(9)  expand $V := [V, v]$
NB   at steps (3), (4) $w$ and $r$ are computed as
     $w := W_1 y + 2\theta W_2 y + \cdots + p\theta^{p-1} W_p y$, $r := W_0 y + \theta W_1 y + \cdots + \theta^p W_p y$
     matrices $A_i$ are only needed at steps (6) and (8)
```

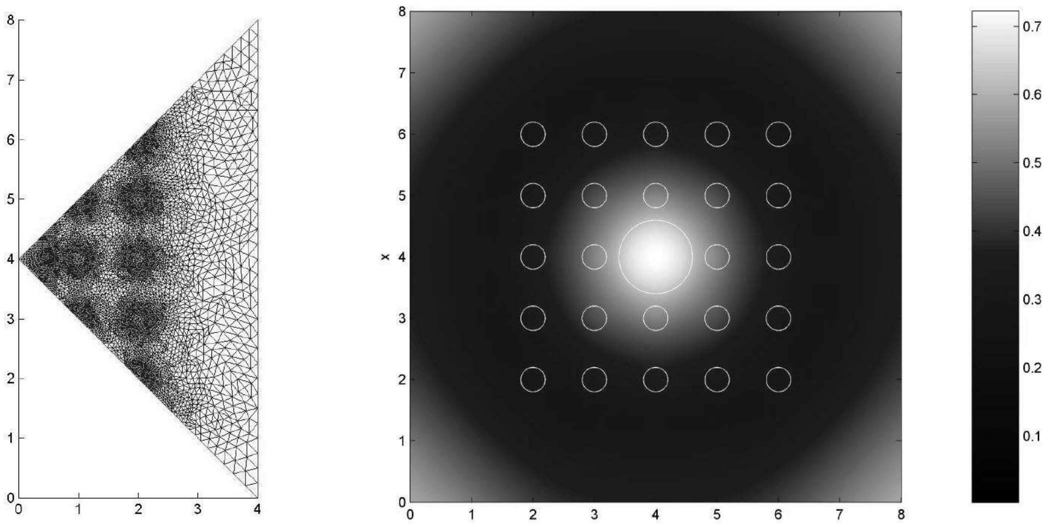

Figure 2: Computational domain with the FEM mesh 1 (left) and a computed eigenmode (right). The absolute value of the complex eigenmode is shown.

frequency range for which the approximation is computed. This means that the error of approximation in (14), computed in this section as

$$\text{error} = \frac{\|B(\lambda) - B_{\text{svd}}(\lambda)\|_1}{\|B(\lambda)\|_1}, \tag{15}$$

have the same order of magnitude for all values of $\lambda$ in the frequency range. Furthermore, since the matrix $B(\lambda)$ can easily be computed for any $\lambda$, it is easy to check the quality of approximation *a posteriori*, by computing error (15) for several values of $\lambda$. Although this has only a moderate influence, the sample frequencies $\lambda_i$ were in all tests taken as Chebyshev polynomial roots in the frequency interval (see Figure 8). Taking the uniform distribution of $\lambda_i$ results in errors which have the same order of magnitude but are slightly larger near the ends of the frequency range.

Figures 4 and 3 illustrate a weak influence of the number of samples $n_s$ on the approximation quality. In our experience it suffices to take $n_s$ between 10 and 30. To check in practice whether the number of samples is sufficient, one can compare the errors obtained with $n_s$ and $2n_s$ samples for several frequency values. If the errors are almost identical then $n_s$ is sufficiently large. Comparing Figures 4 and 3, we see that the approximation quality improves significantly as the frequency range gets tighter.

Next, we inspect the influence of the number of the truncated SVD terms $m$, see Figure 5. Usually, with $m$ is increasing, the approximation error drops for some value of $m$ to a certain value and is not further influenced by $m$. We have already discussed in Section 3 how to choose $m$ in practice. Comparing Figures 4 and 5, we see that $m$ and $n_s$, once taken sufficiently large, hardly influence the approximation error.

Figure 6 shows the effect of the least squares polynomial order $p$. The error is significantly reduced for larger values of $p$. However, as already said in Section 3, it suffices to have $p$ not too large, say $p \leqslant 4$. Higher accuracy, if necessary, can then be obtained by restricting the frequency range. At last, we examine in Figure 7 the robustness of the approximation in (14)

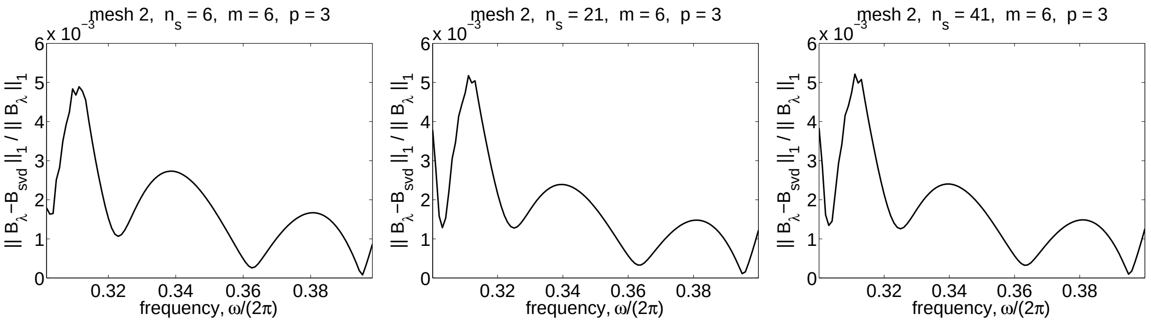

Figure 3: Approximation error (15) for different sample numbers $n_s$: $n_s = 6$ (left), $n_s = 21$ (center), $n_s = 41$ (right). Mesh 2 is used, $m = 6$, frequency range $\sqrt{\lambda} \equiv \omega \in 2\pi[0.30, 0.40]$, least squares polynomial order $p = 3$.

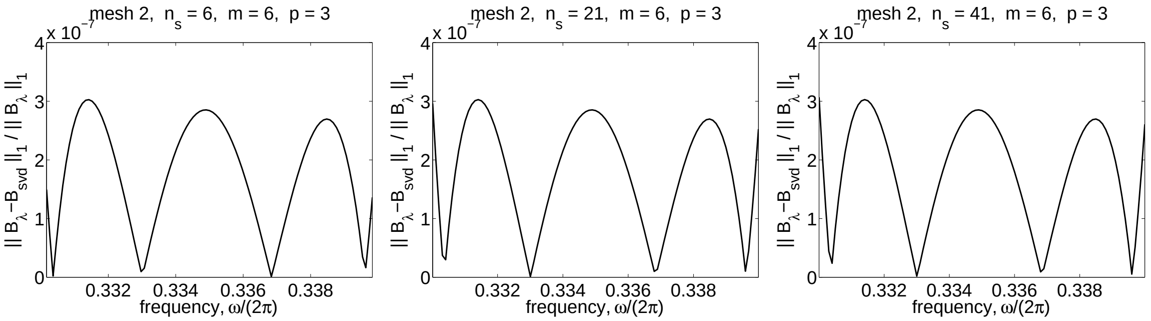

Figure 4: Approximation error (15) for different sample numbers $n_s$: $n_s = 6$ (left), $n_s = 21$ (center), $n_s = 41$ (right). Mesh 2 is used, $m = 6$, frequency range $\sqrt{\lambda} \equiv \omega \in 2\pi[0.33, 0.34]$, least squares polynomial order $p = 3$.

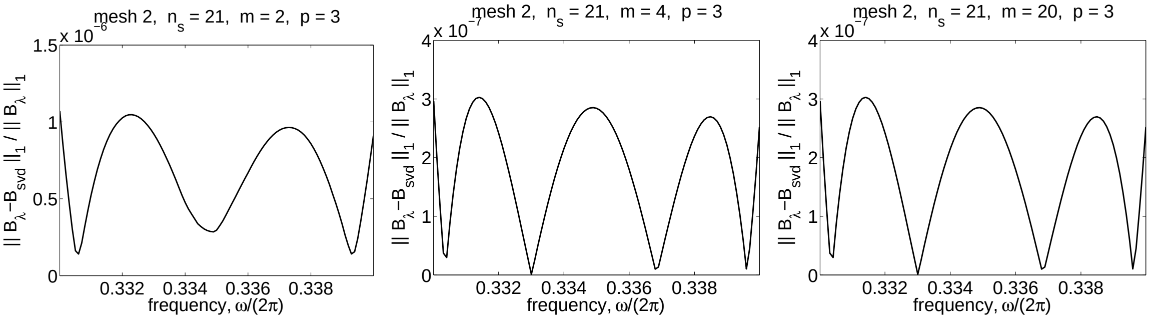

Figure 5: Approximation error (15) for different values of the truncated SVD terms $m$: $m = 2$ (left), $m = 4$ (center), $m = 20$ (right). Mesh 2 is used, $n_s = 21$, frequency range $\sqrt{\lambda} \equiv \omega \in 2\pi[0.33, 0.34]$, least squares polynomial order $p = 3$.

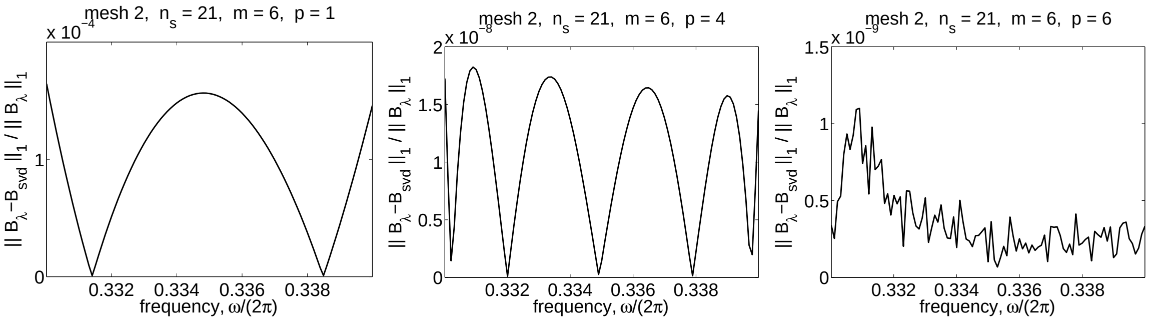

Figure 6: Approximation error (15) for different orders of least squares polynomials $p$: $p = 1$ (left), $p = 4$ (center), $p = 6$ (right). Mesh 2 is used, $n_s = 21$, $m = 6$, frequency range $\sqrt{\lambda} \equiv \omega \in 2\pi[0.33, 0.34]$.

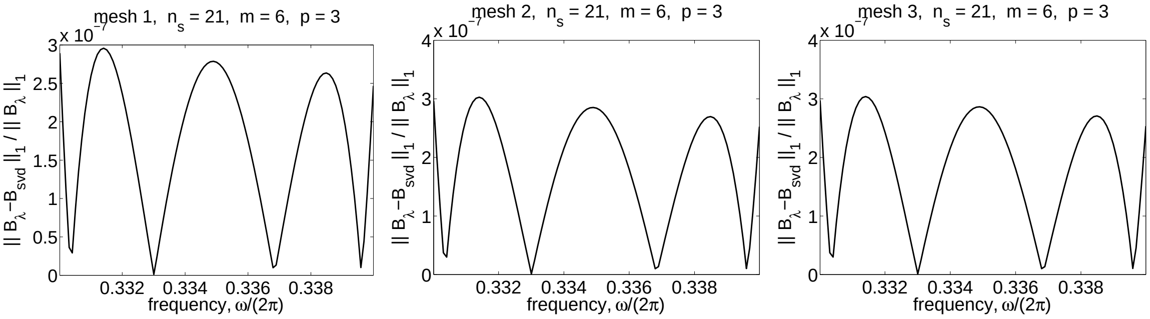

Figure 7: Approximation error (15) for different meshes: mesh 1 with $n = 6\,769$ (left), mesh 2 with $n = 26\,861$ (center), mesh 3 with $n = 107\,017$ (right). $n_s = 21$, $m = 6$, frequency range $\sqrt{\lambda} \equiv \omega \in 2\pi[0.33, 0.34]$, least squares polynomial order $p = 3$.

with respect to the number of degrees of freedom $n$. We see that there is no visible influence of the size of the problem on the approximation quality. This is to be expected as soon as the meshes used are sufficiently fine to adequately approximate the TIBC operator by the matrix $B(\lambda)$.

### 4.3 Testing performance of the SVD-Jacobi-Davidson eigensolver

In this section we test the Jacobi-Davidson method for polynomial eigenproblems combined with the proposed SVD approximation to the nonlinear operator $B(\lambda)$. We test this SVD-Jacobi-Davidson solver (denoted in this section as SVD-JD) against two other methods. The first one is the following fixed-point algorithm proposed in [22], Chapter 6: at $k$-th iteration, the matrix $B(\lambda)$ in (5) is replaced with $B(\lambda_{k-1})$ and the following standard generalized eigenproblem is solved:

(a) find an eigenpair $(\tilde{\lambda}, \tilde{u})$ of

$$\left(\tilde{S} - \lambda M\right) u = 0, \quad \tilde{S} = S + B(\lambda_{k-1}) \tag{16}$$

(b) set $\lambda_k := \tilde{\lambda}$.

The iterations are stopped as soon as a stagnation in $\lambda_k$ is observed. In all the experiments reported in this paper the fixed-point iterations were employed with the stopping criterion

$$\left|\frac{\omega_k - \omega_{k-1}}{\omega_{k-1}}\right| < 10^{-15}, \qquad \omega_k^2 \equiv \lambda_k.$$

Although it might not always be reliable, the stopping criterion appears to work in practice. In our limited experience, the fixed-point iteration usually converged, at least for a reasonably chosen initial guess $\lambda_0$, within up to 10 iterations. One drawback of the fixed-point iteration method is that it is in general difficult to guarantee its convergence in practice.

To solve the standard generalized eigenproblem (16a) at every fixed-point iteration, the implicitly restarted (IR) Arnoldi method (available in Matlab as function `eigs`) is used. Computational work in `eigs` was saved by computing a sparse Cholesky factorization of the mass matrix $M$ once before the fixed-point iteration loop. At each fixed-point iteration $k$, the IR Arnoldi method was run with the target parameter set to $\lambda_{k-1}$. The new iterant $\lambda_k$ was chosen among the eigenvalues delivered by IR Arnoldi to be the closest in real part to $\lambda_{k-1}$.

We note that, instead of IR Arnoldi, one could have combined the fixed-point iteration method with another eigensolver, for instance, Jacobi-Davidson. We have chosen not to do so and to use the combination fixed-point iteration – IR Arnoldi as a reference method since this method is known in the field of integrated optics [22]. In addition, the use of Jacobi-Davidson within the fixed-point iteration would ignore the potential of Jacobi-Davidson as a nonlinear solver.

The second solver to which our SVD-JD solver is compared, is the nonlinear Jacobi-Davidson method as discussed in [6, 26]. The only essential difference from SVD-JD is that the method is applied directly to nonlinear problem (5). Hence, the small projected nonlinear eigenproblem arising at each Jacobi-Davidson iteration (see step (3) in Table 3) has the form

$$V^* \left[S - \lambda M + B(\lambda)\right] V y = 0,$$

with $V$ being the search subspace matrix. To solve this projected problem, the residual inverse iteration [15] is used, which thus enters the outer Jacobi-Davidson iterations as an inner solver. We refer to this nonlinear Jacobi-Davidson method as JD-RI (Jacobi-Davidson – Residual Inverse iteration). Note that we can not use the safeguarded iteration for the inner solver as suggested in [6, 25] because the matrices $B(\lambda)$ are not Hermitian. Since the dependence $B(\lambda)$ is not known explicitly, the matrix of the projected eigenproblem has to be evaluated, at each inner iteration, as

$$\begin{gathered} H_0 + \lambda H_1 + V^* B(\lambda) V, \qquad \text{with} \\ H_0 = V^* W_0, \quad W_0 = SV, \quad H_1 = V^* W_1, \quad W_1 = -MV, \end{gathered}$$

where the matrices $V$, $W_0$, $W_1$, $H_0$ and $H_1$ are handled as shown in Table 3. The presence of the subspace matrix $V$ here makes the solution of the projected eigensystem in JD-RI rather expensive (see Table 4 and conclusions at the end of this section). The residual inverse iteration is implemented as described in [15], with the same tolerance as used in the outer Jacobi-Davidson iterations and maximum number of iterations set to 10.

The small projected polynomial eigenproblem arising in SVD-JD (step (3) in Table 3) was solved with the Matlab standard polynomial eigensolver `polyeig`. In all the experiments, to build the truncated SVD polynomial approximation in SVD-JD, $n_s = 21$ samples in the

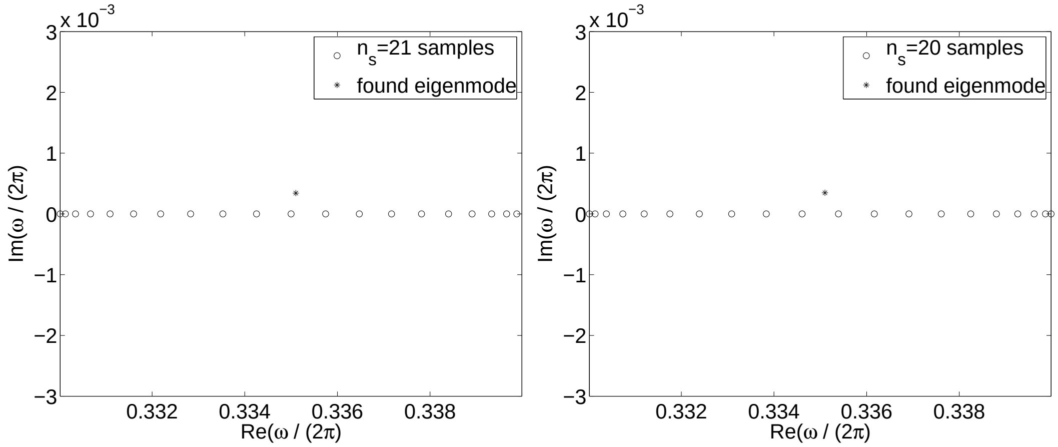

Figure 8: The position of the real sample values $2\pi\sqrt{\lambda_j}$ and the found eigenmodes on the complex plain, for $n_s = 21$ (left) and $n_s = 20$ (right) samples. In both cases mesh 2 is used. The found eigenvalues are $\omega \approx 2\pi(0.335101+0.00035\mathrm{i})$ (left) and $\omega \approx 2\pi(0.335102+0.00034\mathrm{i})$ (right).

frequency range $\sqrt{\lambda} \equiv \omega \in 2\pi[0.33, 0.34]$ were used, with $m = 6$ truncated SVD terms and the polynomial order $p = 4$. In both SVD-JD and JD-RI, the iterations were stopped as soon as the residual norm (computed in SVD-JD for the approximate polynomial eigenproblem) was less than $10^{-9}$. The stopping criteria in the fixed-point and in the two Jacobi-Davidson methods were chosen such that all three methods delivered eigenpairs of approximately the same accuracy. The initial guess $\lambda_0$ for the fixed-point method was taken to be the center of the frequency interval $\sqrt{\lambda_0} \equiv \omega_0 := 2\pi 0.335$. In SVD-JD and JD-RI, the initial subspace was set to a vector containing ones in all its components.

The Jacobi-Davidson method avoids the expensive shift-and-invert steps [19, 20], requiring instead approximate solution of the correction equation (step (6) in Table 3). In both SVD-JD and JD-RI, we solved the correction equation approximately with 10 (fixed number) iterations of full preconditioned GMRES [18]. The ILUT($\varepsilon$) preconditioner, built once before the iteration process for the matrix $S + B(\lambda_0) - \lambda_0 M$ with $\varepsilon = 10^{-4}$, was used [17].

In the first test, we check the robustness of the truncated SVD approximation in the following way. In the left plot of Figure 8 we show the position of the sample values $\sqrt{\lambda_j}$, $j = 1, ..., n_s = 21$, and the found eigenvalue (as found by the SVD-Jacobi-Davidson solver) on the complex plane. To make sure that the SVD approximation works well not only if one of the samples is chosen close to the eigenvalue, we repeated the computations with a different distribution of $n_s = 20$ samples. The new sample values and the found eigenvalue are shown in the right plot of Figure 8. As we see, the sample distribution appears to be of little influence on the computed eigenvalue.

The computational work required by the three methods is summarized in Table 4. We see that, despite fast convergence of the fixed-point iterations (only five iterations were needed), the fixed-point iteration method turns out to be rather expensive. On the other hand, both versions of the Jacobi-Davidson method works quite well, exhibiting its familiar robustness and convergence properties. Although the JD-RI method requires less iterations to converge, it is by far more expensive than SVD-JD. This is due to the expensive evaluations of $V^*B(\lambda)V$

Table 4: Computational costs, accuracy and CPU times of the three eigensolvers. Note that the CPU times are obtained for Matlab codes and, thus, give only an indication of the performance.

| | mesh 1 | mesh 2 | mesh 3 |
|---|---|---|---|
| # degrees of freedom, $n$ | 6 769 | 26 861 | 107 017 |
| costs fixed-point iteration IR Arnoldi | | | |
| # fixed-point iterations | 5 | 5 | out of memory |
| Cholesky factorizations $M$ | 1 | 1 | |
| LU factorizations $S - \sigma M$ | 5 | 5 | |
| matvecs with $M$ | 34 399 | 134 860 | |
| LU solves $S - \sigma M$ | 185 | 185 | |
| eigenpair residual norm | 7.9e−05 | 2.0e−05 | |
| total CPU time, s | 7 | 81 | |
| costs SVD-Jacobi-Davidson (SVD-JD) | | | |
| # iterations | 31 | 32 | 44 |
| matvecs $S - \theta M + \sum_{k=0}^{p} \theta^k B_k$ | 31 | 32 | 44 |
| vector updates with $W_k$ | 62 | 64 | 88 |
| matvecs correction equation | 300 | 310 | 430 |
| eigenpair residual norm | 7.9e−05 | 4.2e−05 | 9.3e−04 |
| total CPU time, s | 15 | 79 | 550 |
| costs Jacobi-Davidson–residual inverse iteration (JD-RI) | | | |
| # iterations | 17 | 25 | 24 |
| matvecs $S - \theta M + B(\theta)$ | 17 | 25 | 24 |
| vector updates with $W_k$ | 34 | 50 | 48 |
| matvecs correction equation | 160 | 240 | 230 |
| evaluations $V^* B(\theta) V$ | 556 | 612 | 1077 |
| eigenpair residual norm | 3.9e−03 | 6.7e−04 | 7.8e−04 |
| total CPU time, s | 90 | 440 | 3180 |

required by the residual inverse iterations to solve the projected problem. Although measuring the CPU time in Matlab might not be a good indication of the performance, we report that SVD-JD was approximately a factor six faster than JD-RI.

## 5 Conclusions

We presented an approach for the solution of nonlinear Helmholtz eigenvalue problems arising when the nonlocal TIBCs are used. Since the nonlinearity results from the boundary conditions, the nonlinear contributions to the matrix of the eigenvalue problem can be seen as a smaller-dimensional discrete operator. This allows for a relatively cheap low-rank SVD parameterization of the nonlinear dependence, so that the boundary operator can be approximated by a low-degree matrix polynomial. Both analysis and numerical tests suggest that the truncated SVD approximation is computationally cheap, robust and reliable. Moreover, the quality of the truncated SVD approximation can easily be checked in practice *a posteriori*.

Once the nonlinear nonpolynomial eigenproblem is reduced to a nonlinear polynomial one, the Jacobi-Davidson method can readily be applied. Depending on the accuracy requirements of the eigenvalue problem, the truncated SVD polynomial approximation can be refined during the Jacobi-Davidson iterations.

Numerical tests were presented to compare the SVD-Jacobi-Davidson algorithm to a fixed-point iteration method [22] (which is chosen as a reference method known in the field of integrated optics) and to another nonlinear Jacobi-Davidson solver. The drawback of the fixed point iteration method is that it is, in general, difficult to guarantee its convergence. This nonlinear Jacobi-Davidson solver (cf. [25, 6, 26]) applies directly to the nonlinear eigenproblem and employs the residual inverse iteration [15] for the small projected eigenproblem. This method, however, appears too expensive since the nonlinearity is not known explicitly and evaluations of the form $u := V^*B(\lambda)Vy$ are required. The SVD-Jacobi-Davidson algorithm appears to be significantly cheaper than the other two methods in terms of the computational costs.

Finally, we note that the proposed truncated SVD polynomial approximation seems to be a promising approach as it can be combined with other nonlinear eigensolvers, inlcuding Arnoldi methods for the linearized eigenproblems and nonlinear Jacobi-Davidson solvers. An important issue for nonlinear eigensolvers is preventing the solver from converging to an already found eigenpair (see e.g. [25, 6, 26]). We believe that the truncated SVD polynomial approximation has a potential in this sense. This will be addressed in a future work.

**Acknowledgements** The first authors would like to thank Gene Golub for stimulating discussions at the initial stages of this work. We are very grateful to the referees for their useful comments and to the editors for making this special issue possible.

# References

[1] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. A. van der Vorst, editors. *Templates for the solution of algebraic eigenvalue problems: A practical guide*. Software, Environments, and Tools. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000. Available at `www.netlib.org/etemplates/`.

[2] A. Bayliss and E. Turkel. Radiation boundary conditions for wave-like equations. *Comm. Pure Appl. Math.*, 33(6):707–725, 1980.

[3] J. Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.*, 114:185–200, 1994.

[4] J.-P. Berenger. Improved PML for the FDTD solution of wave-structure interaction problems. *Antennas and Propagation, IEEE Transactions on*, 45(3):466–473, Mar 1997.

[5] G. Berkooz and E. S. Titi. Galerkin projections and the proper orthogonal decomposition for equivariant equations. *Phys. Lett. A*, 174(1-2):94–102, 1993.

[6] T. Betcke and H. Voss. A Jacobi-Davidson-type projection method for nonlinear eigenvalue problems. *Future Generation Computer Systems*, 20(3):363–372, 2004.

[7] E. A. Christensen, M. Brøns, and J. N. Sørensen. Evaluation of proper orthogonal decomposition-based decomposition techniques applied to parameter-dependent nonturbulent flows. *SIAM J. Sci. Comput.*, 21(4):1419–1434, 1999/00.

[8] B. Engquist and A. Majda. Radiation boundary conditions for acoustic and elastic wave calculations. *Comm. Pure Appl. Math.*, 32(3):314–358, 1979.

[9] S. D. Gedney. An anisotropic perfectly matched layer absorbing media for the truncation of FDTD latices. *Antennas and Propagation, IEEE Transactions on*, 44:1630–1639, 1996.

[10] D. Givoli. Nonreflecting boundary conditions. *J. Comput. Phys.*, 94(1):1–29, 1991.

[11] G. H. Golub and C. F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, Baltimore and London, third edition, 1996.

[12] I. T. Jolliffe. *Principal component analysis.* Springer Series in Statistics. Springer-Verlag, New York, second edition, 2002.

[13] D. C. Lay. *Linear Algebra and its Applications.* Addison Wesley, 2003.

[14] C. Michler, L. Demkowicz, J. Kurtz, and D. Pardo. Improving the performance of Perfectly Matched Layers by means of *hp*-adaptivity. *Numer. Methods Partial Differential Equations*, 23(4):832–858, 2007.

[15] A. Neumaier. Residual inverse iteration for the nonlinear eigenvalue problem. *SIAM J. Numer. Anal.*, 22(5):914–923, 1985.

[16] J. B. Nicolau and E. W. G. van Groesen. Hybrid analytic-numeric method for light through a bounded planar dielectric structure. *Journal of Nonlinear Optical Physics & Materials*, 14(2):161–176, 2005.

[17] Y. Saad. ILUT: a dual threshold incomplete *LU* factorization. *Numer. Linear Algebra Appl.*, 1(4):387–402, 1994.

[18] Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, 1986.

[19] G. Sleijpen, H. van der Vorst, and M. van Gijzen. Quadratic eigenproblems are no problem. *SIAM News*, 8:9–10, September 1996. Available at `http://igitur-archive.library.uu.nl/math/2001-0621-124617/UUindex.html`.

[20] G. L. G. Sleijpen, A. G. L. Booten, D. R. Fokkema, and H. A. van der Vorst. Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems. *BIT*, 36(3):595–633, 1996.

[21] G. L. G. Sleijpen and H. A. van der Vorst. A Jacobi-Davidson iteration method for linear eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 17:401–425, 1996. Reprinted in SIAM Review, 42(2), 2000, 267–293.

[22] A. Sopaheluwakan. *Characterization and Simulation of Localized States in Optical Structures.* PhD thesis, University of Twente, Enschede, December 2006. `http://eprints.eemcs.utwente.nl/8858/`.

[23] A. Taflove and S. C. Hagness. *Computational electrodynamics: the finite-difference time-domain method.* Artech House Inc., Boston, MA, third edition, 2005.

[24] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM Rev.*, 43(2):235–286, 2001.

[25] H. Voss. An Arnoldi method for nonlinear eigenvalue problems. *BIT Numerical Mathematics*, 44:387–401, 2004.

[26] H. Voss. A Jacobi-Davidson method for nonlinear and nonsymmetric eigenproblems. *Comput. & Structures*, 85(17-18):1284–1292, 2007.