EIGENVALUE CHARACTERIZATION AND COMPUTATION FOR THE LAPLACIAN ON GENERAL 2-D DOMAINS

PATRICK GUIDOTTI* AND JAMES V. LAMBERS †

Abstract. In this paper we address the problem of determining and efficiently computing an approximation to the eigenvalues of the negative Laplacian $-\triangle$ on general domain $\Omega \subset \mathbb{R}^2$ subject to homogeneous Dirichlet or Neumann boundary conditions. The basic idea is to look for eigenfunctions as the superposition of generalized eigenfunctions of the corresponding free space operator in the spirit of the classical Method of Particular Solutions. The main advantages of the proposed approach are the possibility of targeting each eigenvalue independently without need for extensive scanning of the positive real axis and the use of small matrices.

Key words. Characterization and computation of eigenvalues, general domain, Laplacian.

1. Introduction. The numerical computation of eigenvalues of differential operators is usually performed following two main philosophies. On the one hand, one starts out with a discretization of the differential operator by a finite dimensional matrix approximation obtained by finite differences, finite elements or any other numerical method. The spectrum of the approximation is then taken as an approximation to the spectrum of the original differential operator. This procedure has two main disadvantages. First, it introduces a number of spurious modes which have nothing to do with the differential operator. Second, the size of the discretization matrix grows with the number of eigenvalues to be computed. For this type of approach we refer to [4] and the references therein. The other, known as Method of Particular Solutions (MPS), uses special function series and seeks the singular values of matrices obtained by imposing boundary conditions to their discretization. It was first introduced in [5] and has recently been revived by [3], where the authors resolve a major shortcoming of the method which had been already observed in the original [5]. In its current form as formulated by [3] we shall refer to it as the *modified MPS method*.

Other more specialized methods can also be found in the literature [11], but they are restricted to special cases where the domain has very high symmetry properties. As for the characterization of eigenvalues the undoubtedly most used one is that by the Rayleigh quotient which can equally be applied to continuous as well as discrete approximations.

Here an alternative characterization and a related computational scheme are proposed which are in the MPS spirit. Whereas it does not seem, at first sight, as widely applicable as other methods mentioned above based on the discretization of the differential operator, it has the advantage of better exploiting the analytical properties of the operator before proceeding to discretization. More specifically, an equation characterizing the eigenfunctions is discretized instead of the differential operator itself. In so doing the computation/approximation of any given eigenvalue is effectively decoupled from that of the others and, most importantly, computations need only to be performed on small matrices which leads to improved efficiency.

A common problem encountered when using MPS is the appearance of spurious solutions. Since the method proposed is closely related to MPS, it is only natural to expect such a phenomenon to occur. In all experiments performed so far, the

*Department of Mathematics, University of California at Irvine, Irvine, CA 92697-3875
†Department of Petroleum Engineering, Stanford University, Stanford, CA 94305-2220

proposed method never failed to deliver a real eigenvalue. As opposed to [3] we do not need to introduce interior points to achieve that. Rather, the very architecture of our algorithm seems responsible for it. We in fact always start with 1×1 matrices and increase their size only in the vicinity of the eigenvalue to be computed. In all experiments the algorithm would settle on an approximation of the desired eigenvalue before the size of the matrices (and their bad conditioning) could possibily lead to spurious solutions. We also observe that, as opposed to the modified MPS method, ours allows for a trivial computation of approximate eigenfunctions once a singular value is found just as in the original MPS.

The method we propose produces the best results for smooth domains or domains containing only "regular corners". It remains stable for more challenging domains such as the famous isospectral drums. In the latter case it can be improved by using ad-hoc expansions in eigenfuctions for wedge domains and otherwise using our algorithm as it is presented in this paper. In this case our approach can also be combined with the modified MPS to deliver increased efficiency for the same accuracy. It is in particular possible to avoid to extensively scan the positive real axis in search for the eigenvalues as is done in the modified MPS.

We point out, however, that the proposed method remains stable across a wide range of domains and does not use any domain specific information. Latter quality combined with its efficiency makes it particularly appealing for use in the resolution of inverse spectral problems. We present an experiment in this direction in Subsection 5.5.

In a recent paper [10], the authors also point out the computational cost related to the use of discretizations of the differential operator and suggest two perturbative methods in order to reduce the computational complexity. One is based on Taylor expansion of the domain about a circle combined with the Rayleigh quotient characterization and the other deals with the specific case of regular polygons. Our method, while offering the same benefits computationally, applies to more general domains.

We also mention our motivation for the study of this problem. We envision the use of the computed eigenfunctions as trial functions for Galerkin methods for PDEs. In particular, such trial functions are the best choices for Krylov Subspace Spectral Methods (see [9]) applied to time-dependent problems.

The rest of the paper is organized as follows. In the next Subsection the problem is formulated and the central Ansatz is introduced. In Section 3 equation (2.5) is derived and some of its analytical properties are highlighted which turn out to be useful in the related problem of numerically computing approximations to eigenvalue-eigenfunction pairs. The algorithm is described in Section 4 and a number of numerical experiments and comparisons are performed in Section 5.

2. Preliminaries. Consider the eigenvalue problem

$$\begin{cases} -\triangle\varphi = \lambda\varphi & \text{in } \Omega\,, \\ \quad \varphi = 0 & \text{on } \partial\Omega\,. \end{cases} \tag{2.1}$$

The approach proposed here stems from the natural generalization of the one dimensional Ansatz for the eigenfunctions and reads in this case

$$\varphi = \int_{\mathbb{S}_r^{n-1}} e^{i\,x\cdot\xi} f(\xi)\, d\sigma_{\mathbb{S}_r^{n-1}}(\xi) = \langle f, e^{i\,x\cdot\xi}\rangle_{\mathbb{S}_r^{n-1}} \tag{2.2}$$

where $\mathbb{S}_r^{n-1}$ denotes the sphere of radius $r > 0$ in $\mathbb{R}^n$. For any choice of a density function $f \in \mathcal{D}'(\mathbb{S}_r^{n-1})$ (2.2) is an eigenfunction of the negative Laplacian to the

eigenvalue r^2. The density needs to be allowed to be a distribution in order to cover all domains. We refer to [8, Theorems 2.1, 4.4] for an abstract treatment of this topic where, in the context of groups, precise characterization theorems can be proven. An eigenvalue of (2.1) is obtained whenever

$$\int_{\mathbb{S}_r^{n-1}} e^{i\,x\cdot\xi} f(\xi)\, d\sigma_{\mathbb{S}_r^{n-1}}(\xi) = 0\,,\ x \in \partial\Omega\,, \tag{2.3}$$

possesses a non-trivial solution f. Even though this procedure is dimension independent, we now specialize to the case $n = 2$ since the special functions needed change with dimension. Expanding the density f in a Fourier series

$$f = \sum_{m\in\mathbb{Z}} \alpha_m e^{i\,m\theta}\,,\ \theta \in [0, 2\pi)\,, \tag{2.4}$$

and substituting in (2.3) the problem can be recast as

$$\sum_{m\in\mathbb{Z}} \alpha_m\, J_m(r|x|)\, e^{i\,m\theta_x} = 0\,,\ x \in \partial\Omega\,, \tag{2.5}$$

where $\theta_x = \arg(x_1 + i\,x_2)$. We refer to lemma 3.1 for the details.

REMARK 2.1. *From this representation one clearly readily recovers the eigenvalues for the circle as the zeros* $(r_k^m)_{m,k\in\mathbb{N}}$ *of* $J_m\,(m \in \mathbb{N})$ *and the associated eigenfunctions* $\left(J_m(r_k^m|x|)\, e^{i\,m\theta_x}\right)_{m,k\in\mathbb{N}}$ *corresponding to the non-trivial solutions*

$$\alpha = e_m = (0,\dots,0,\underbrace{1}_{m\text{-th entry}},0,\dots)\,,\ m \in \mathbb{N}\,,$$

of (2.5). *Observe that* $J_{-m} = (-1)^m\, J_m$.

This remark is going to play a crucial role in the design of a stable, efficient algorithm for the computation of the eigenvalues and of the Fourier coefficients for the density function in the case of arbitrary domains. Other boundary conditions can also be considered. We refer the reader to Subsection 4.4.

The proposed algorithm is very efficient since it relies on the manipulation of small matrices and allows for spotting individual eigenvalues independently of one another. It also produces approximate eigenfunctions for which

$$-\triangle\varphi = \lambda\varphi$$

holds exactly but which only satisfy the boundary condition approximately.

3. Characterization. Generalized eigenfunctions of the "free space" Laplacian are well-known to be given by

$$\phi_\xi(x) = e^{i\,x\cdot\xi}\,,\ x \in \mathbb{R}^n\,. \tag{3.1}$$

It is therefore a natural extension of one dimensional thinking to look for eigenfunctions φ_λ to a potential eigenvalue $\lambda^2 > 0$ for $-\triangle_D$ on a domain Ω in the form

$$\varphi_\lambda(x) = r \int_{\mathbb{S}^1} e^{ir\,x\cdot\xi} f(\xi)\, d\sigma_{\mathbb{S}^1}(\xi) \tag{3.2}$$

with $r > 0$ and $r^2 = \lambda^2$. (3.2) is only a minimal modification of (2.2) performed in order to have the integration performed on the unit circle. Obviously $-\triangle\varphi = \lambda^2\varphi$

is satisfied for any choice of (possibly generalized) density function $f \in \mathcal{D}'(\mathbb{S}^1)$. The requirement that (3.2) satisfy given boundary conditions on $\partial\Omega$ then provides a means of selecting the actual values of $r > 0$ which do indeed lead to a nontrivial solution of

$$\int_{\mathbb{S}^1} e^{ir\,x\cdot\xi} f(\xi)\, d\sigma_{\mathbb{S}^1}(\xi) = 0\,,\ x \in \partial\Omega \tag{3.3}$$

and therefore an eigenfunction. This amounts to the kernel of the operator

$$R_{\partial\Omega}\mathcal{F}^{-1}R'_{\mathbb{S}^1} \tag{3.4}$$

being non-trivial for a specific value of $r > 0$. Here, for any subset A of $\mathbb{R}^2$, R_A denotes the restriction operator, $\mathcal{F}$ the Fourier transform and R'_A the dual to the restriction operator (or, in other words, the trivial extension operator). By introducing the notation (3.4) we do not intend to further study the properties of such operators but rather simply point out the analytic structure of the problem at hand. The expected smoothness of the eigenfunctions (for smooth domains) is of course reflected in the fact that they can be synthesized from compactly supported density functions in frequency space. This is a consequence of the classical Payley-Wiener theorem. Since f is a (possibly generalized) function of the circle, it makes sense to consider its Fourier expansion

$$f = \sum_{m\in\mathbb{Z}} \alpha_m e^{i\,m\theta}\,,\ \theta \in [0, 2\pi)\,, \tag{3.5}$$

This leads to the computation of

$$\int_{\mathbb{S}^1} e^{ir\,x\cdot\xi} e^{i\,m\theta_\xi}\, d\sigma_{\mathbb{S}^1}(\xi)$$

which is done in the next lemma.

LEMMA 3.1. *For any $r > 0$ one has that*

$$\frac{1}{2\pi i^m}\int_0^{2\pi} e^{ir\,x\cdot\xi_\theta} e^{i\,m\theta}\, d\theta = J_m(r|x|)e^{i\,m\theta_x}\,,\ x \in \mathbb{R}^2\,. \tag{3.6}$$

Hereby $\xi_\theta = \big(\cos\theta, \sin\theta\big)$. We also make use the standard notation J_m for the Bessel functions of integer order.

Proof. First write

$$x = |x|\big(\cos\theta_x, \sin\theta_x\big)\,,\ \theta_x = \arg(x_1 + i\,x_2)$$

and observe that

$$e^{ir\,x\cdot\xi_\theta} = e^{ir|x|(\cos\theta_x\cos\theta + \sin\theta_x\sin\theta)} = e^{ir|x|\,\cos(\theta-\theta_x)}$$

Then, by virtue of the periodicity of the integrand, it follows that

$$\int_0^{2\pi} e^{ir\,x\cdot\xi_\theta} e^{i\,m\theta}\, d\theta = \int_0^{2\pi} e^{ir|x|\,\cos(\theta-\theta_x)} e^{i\,m\theta}\, d\theta = e^{im\,\theta_x}\int_0^{2\pi} e^{ir|x|\,\cos\theta} e^{i\,m\theta}\, d\theta\,.$$

Finally

$$\frac{1}{2\pi i^m}\int_0^{2\pi} e^{ir|x|\,\cos\theta} e^{i\,m\theta}\, d\theta = J_m(r|x|)\,.$$

gives the claim. □
We can now conclude that

$$\int_{\mathbb{S}^1} e^{ir\,x\cdot\xi} f(\xi)\, d\sigma_{\mathbb{S}^1}(\xi) = \sum_{m\in\mathbb{Z}} \alpha_m J_m(r|x|) e^{i\,m\theta_x} \tag{3.7}$$

Before turning to the analysis of some properties of this series we consider a few illustrative and motivating examples.

EXAMPLE 3.2. *The simplest example in the described setup is obviously the circle as we already mentioned in the introduction. The well-known eigenfunctions can be readily obtained from the series.*

EXAMPLE 3.3. *To show that the density function f in* (3.2) *can indeed be a distribution, we consider the unite square centered in the origin. It is elementary to verify that*

$$f = \frac{1}{4}\sum_{j=0}^{3} \delta_{\xi_j}\,,\ \xi_j = \big(\cos(\pi/4 + j\pi/2), \sin(\pi/4 + j\pi/2)\big)$$

is the density for the first eigenfunction and that any other eigenfunction also has generalized kernel.
Finally we consider a class of domains which we shall revisit in the numerical experiments to be presented later.

EXAMPLE 3.4. *Let $\partial\Omega$ be a star-shaped domain described by*

$$\rho(\theta)\big(\cos(\theta), \sin(\theta)\big)\,,\ \theta \in [0, 2\pi)\,, \tag{3.8}$$

for a given (continuous) periodic function $\rho : [0, 2\pi) \to [0, \infty)$. In this case the determining equation reads

$$\sum_{m\in\mathbb{Z}} \alpha_m J_m(r\rho(\theta)) e^{i\,m\theta} = 0\,, \theta \in [0, 2\pi)\,. \tag{3.9}$$

It seems natural to set up a system of equations for a truncated version of the series appearing in (3.9) *and taking an inner product with finitely many basis functions like, e.g., $e^{i\,k\theta}$, $\theta \in [0, 2\pi)$.*
Notice that a translation in the domain only results in the multiplication of the density function by a non-vanishing function so we can assume without loss of generality that the domain contain the origin, or, in this case, even be star-shaped with respect to the origin. In the one dimensional case there is a natural ordering of the eigenvalues according to their size and, more interestingly, by the number of nodal points of the associated eigenfunction. A similar characterization in higher dimension is most likely out of reach except in very special geometries. There is, however, a way of predicting the rough frequency content of an eigenfunction to an eigenvalue of a given size. This is related to the properties of Bessel functions and is presented in the next proposition.

Proposition 3.1. *Consider a domain of type* (3.8). *Then, if one is interested in eigenvalues of maximal approximate size $R > 0$, the corresponding eigenfunction will contain leading order frequencies determined by J_m of order at most $m \leq R\|\rho\|_\infty$.*

Proof. The claim follows from classical estimates for Bessel functions which can be found in [1, pp. 366 and 448]. They imply that $J_m(mt)$ is exponentially decaying in m for $0 < t < 1$. □
This property will play a crucial role in the design of an effective and stable algorithm based on the appropriate discretization of (3.9).

4. Computation. In this section we assume Ω is a star-shaped domain with boundary $\partial\Omega$ described by (3.8). We will describe an iterative method for computing an approximate solution of the determining equation (3.9) for the coefficients $\{\alpha_m\}$ in the Fourier expansion (3.5) of the density $f(\theta)$.

The main idea is to begin with a domain for which we know exact eigenpairs, a circle. The eigenvalues will serve as our initial guesses for the eigenvalues of $-\Delta$ on Ω, and the eigenfunctions serve as trivial expansions of the form (3.7), containing only one term of the form $J_{m_0}(\sqrt{\lambda})e^{im_0\theta}$ for some integer m_0. Then, our iteration will deform the circle into the shape defined by $\partial\Omega$ and add terms to the eigenfunction expansions until eigenpairs of Ω are found.

4.1. Definitions. Let M and m_0 be nonnegative integers representing the size and center, respectively, of the Fourier-Bessel expansion of an approximate eigenfunction. We define the set of indices $I(M, m_0)$ by

$$I(M, m_0) = \{-m_0 - M, \ldots, -m_0 + M\} \cup \{m_0 - M, \ldots, m_0 + M\}.$$

We denote by $|I(M, m_0)|$ the number of indices in this set, and we use the notation $I_j(M, m_0)$ to refer to the jth index in the set, when arranged in increasing order. These indices correspond to the orders of the Bessel functions that are included in an eigenfunction expansion of the form (3.7).

For convenience, we define

$$\varphi_m(r, \theta) = J_m(r\rho(\theta))e^{im\theta}, \quad m \in I(M, m_0).$$

We then define an n-point grid on the interval $[0, 2\pi)$ with gridpoints $0 \leq \theta_1 < \theta_2 < \cdots < \theta_n < 2\pi$. Next, we define $\Phi_n(r, M, m_0)$ to be an $n \times |I(M, m_0)|$ matrix with entries

$$[\Phi_n(r, M, m_0)]_{ij} = \frac{\varphi_{I_j(M,m_0)}(r, \theta_i)}{\left(\sum_{k=1}^{n} |\varphi_{I_j(M,m_0)}(r, \theta_k)|^2\right)^{1/2}}.$$

It can be seen from the definition that each column of $\Phi_n(r, M, m_0)$ has unit 2-norm.

Finally, we let $Q_n(r, M, m_0)$ be the matrix obtained by orthogonalizing the columns of $\Phi_n(r, M, m_0)$, and we define the matrix $A(r, M, m_0)$ by

$$A(r, M, m_0) = [Q_n(\epsilon, M, m_0)]^H \Phi_n(r, M, m_0).$$

The columns of $\Phi_n(r, M, m_0)$ correspond to the Fourier-Bessel functions in the series (3.7). We will seek a value of r such that a linear combination of these functions, when projected into the column space of $Q_n(\epsilon, M, m_0)$, nearly vanishes on $\partial\Omega$. The constant ϵ is independent of r; in practice ϵ can be chosen as large as 1 (but should not be larger) or as small as 10^{-4}.

This choice of a subspace in which to project the columns of $\Phi_n(r, M, m_0)$, as opposed to other choices that we could have made, is based on the observation that an approximate eigenfunction of the form (4.1), when evaluated at the points belonging to the discretization of $\partial\Omega$, will yield a vector that (approximately) belongs to $Q_n(\epsilon, M, m_0)^\perp$. Since this residual will also be rich in the directions of higher-order Fourier-Bessel functions, we can ensure a smaller residual by excluding lower-frequency components of higher-order Bessel functions from $Q_n(\epsilon, M, m_0)^\perp$ and including them in $Q_n(\epsilon, M, m_0)$. Since higher-order Bessel functions exhibit exponential

decay close to the origin, we can use lower-order Bessel functions as aliases for higher-order ones, by using a small value for ϵ. We do not claim that this is the best possible choice and intend to explore alternatives, but it does produce the best results of any that we have used thus far.

4.2. Motivation. For each eigenvalue λ_j, we seek an approximate eigenfunction $u_j(\rho, \theta)$ of the form

$$u_j(\rho, \theta) = \sum_{m \in I(M, m_0)} \alpha_m J_m(\sqrt{\lambda_j}\rho) e^{im\theta}. \tag{4.1}$$

The coefficients $\{\alpha_m\}_{m \in I(M,m_0)}$ and eigenvalue λ_j must be chosen so that u_j vanishes, at least approximately, on $\partial\Omega$. To compute these values, for given M and m_0, we can solve the nested minimization problem

$$\min_{r>0} \min_{\|\mathbf{a}\|_2=1} \|A(r, M, m_0)\mathbf{a}\|_2. \tag{4.2}$$

The vector $\mathbf{a}$ represents the coefficients α_m, for $m \in I_m(M, m_0)$, in the expansion (4.1), and r corresponds to the square root of the eigenvalue λ_j.

First, we examine the inner minimization problem for fixed values of r, M, and m_0. We can solve this problem by computing the singular value decomposition, or SVD (see [6]), of $A(r, M, m_0)$,

$$A(r, M, m_0) = U\Sigma V^H, \tag{4.3}$$

where U and V are unitary and Σ is diagonal with diagonal entries

$$0 \leq \sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_{|I(M,m_0)|}.$$

Then, $\|A(r, M, m_0)\mathbf{a}\|_2$ is minimized by setting $\mathbf{a} = \mathbf{v}_1$, where $\mathbf{v}_i$ is the ith column of V for $i = 1, 2, \ldots, |I(M, m_0)|$. It follows that $\|A(r, M, m_0)\mathbf{a}\|_2 = \sigma_1$.

Observe that it is not necessary to compute the entire SVD to obtain an approximation to $\mathbf{v}_1$; one can instead apply the symmetric Lanczos algorithm or Rayleigh quotient iteration (see [6]) to the matrix $B = [A(r, M, m_0)]^H A(r, M, m_0)$ to find the eigenvector corresponding to the smallest eigenvalue of B, and this eigenvector is equal to $\mathbf{v}_1$.

Let $\sigma^*(r, M, m_0)$ denote the smallest singular value of $A(r, M, m_0)$, which is also the solution of the inner minimization problem. If we can solve the outer minimization problem, seeking an r that minimizes $\sigma^*(r, M, m_0)$, and if the solution r^* is such that $\sigma^*(r^*, M, m_0)$ is nearly zero, then the coefficients $\{\alpha_m\}_{m \in I(M,m_0)}$ obtained from the corresponding singular vector $\mathbf{v}_1(A(r^*, M, m_0))$ and the value $\lambda_j = (r^*)^2$, when substituted into (4.1), yield an approximate eigenfunction of $-\Delta$ on Ω that satisfies Dirichlet boundary conditions.

4.3. Algorithm Description. We first describe how we choose our initial iterates. We use the eigenvalues for a circle of radius ρ_0, where

$$\rho_0 = \mathrm{Avg}_{\theta \in [0, 2\pi]}\, \rho(\theta).$$

These eigenvalues are simply the values of λ for which $J_m(\sqrt{\lambda}\rho_0) = 0$, for some nonnegative integer m. For $m = 0$, the eigenvalues are simple; otherwise they have

multiplicity 2, in which case they are included twice in the set of initial iterates and perturbed slightly, so that closely clustered eigenvalues can be found independently.

For each initial iterate r, we let m_0 be the integer such that $J_{m_0}(r\rho_0) = 0$, prior to any perturbation. Then, we begin an iteration to find an eigenvalue. For $M = 0, 1, 2, \ldots$, we minimize

$$f_M(r) = \sigma_1(A(r, M, m_0)) \tag{4.4}$$

using the minimizer r^* of $f_{M-1}(r)$ as the initial iterate when $M > 0$, and the value described in the previous paragraph as the initial iterate when $M = 0$. When $M = 0$, the expansion (4.1) only includes Bessel functions of order m_0 and $-m_0$. When $M = 1$, we add Bessel functions of order $m_0 \pm 1$ and $-m_0 \pm 1$; when $M = 2$, we add orders $m_0 \pm 2$ and $-m_0 \pm 2$, and so on. The set $I(M, m_0)$ describes all orders that are included in the expansion for any choice of M and m_0.

To minimize $f_M(r)$, we exploit the fact that near the minimizer, $f_M(r)$ is a continuous, approximately piecewise linear function of r, with the minimizer occurring at the boundary between two pieces. Therefore, we can use any derivative-free optimization method to quickly obtain an approximation to the minimizer. However, in using any such method, one should exploit this behavior of $f_M(r)$.

To that end, we perform an inner iteration in which we compute each iterate r_k by constructing a secant line using the previous iterate r_{k-1} and a small perturbation $r_{k-1} + \delta_k$. This perturbation is chosen by assuming that $f_M(r)$ is exactly piecewise linear and then choosing the sign of δ_k to ensure that $r_{k-1} + \delta_k$ lies on the same piece as r_{k-1}. We then obtain r_k by computing the r-intercept of this secant line.

If $f_M(r_k)$ is sufficiently small, then the iteration terminates successfully. Otherwise, we repeat this process with r_k. If r_k is an outlying value relative to the previous iterates, then it is rejected and a new iterate is chosen based on the previous iterates and their slopes, but this is not a typical occurrence in practice.

The motivation for this approach is illustrated in Figure 4.1. Note that for smaller values of M, one must only have a reasonable initial guess in order to exploit the approximate piecewise linearity of $f_M(r)$ near the minimizer. As M increases, the "well" containing the minimizer begins to "close," thus requiring a much more accurate initial guess. If this requirement is not satisfied, the fact that $f_m(r)$ is very smooth away from the minimizer implies that this "well" would be missed entirely by a search algorithm that did not take very small steps.

The reason for the closing of the well is that as M increases, $A(r, M, m_0)$ becomes more ill-conditioned, and $\sigma_1(A(r, M, m_0))$ becomes smaller regardless of the value of r. This troublesome trend is mitigated significantly by normalizing the columns of $\Phi(r, M, m_0)$ and orthogonalizing the columns of $\Phi(\epsilon, M, m_0)$, but for sufficiently large M it can still cause difficulties if r is not already close to an eigenvalue.

If the boundary of Ω is not very smooth, then it is advisable to construct a homotopy from the circle of radius ρ_0 to Ω, resulting in a sequence of domains $\Omega_0, \Omega_1, \ldots, \Omega_{\tilde{M}}$ where each Ω_k is a star-shaped domain with boundary defined by $\rho_k(\theta) = t_k\rho(\theta) + (1 - t_k)\rho_0$ and $0 = t_0 < t_1 < \cdots < t_{\tilde{M}-1} < t_n = 1$. Then, when we are minimizing $f_M(r)$ for $M < \tilde{M}$, we are using Ω_M in place of Ω. This homotopy helps to ensure that as M increases, our initial iterate r_0 is sufficiently close to the minimizer.

Once we have found a minimizer r^* of $f_M(r)$, we can obtain the approximate solution $\mathbf{a}$ of the determining equation (3.9) by computing the right singular vector $\mathbf{v}_1$ of the matrix $A(r^*, M, m_0)$. This vector $\mathbf{v}_1$ represents the coefficients α_m, for

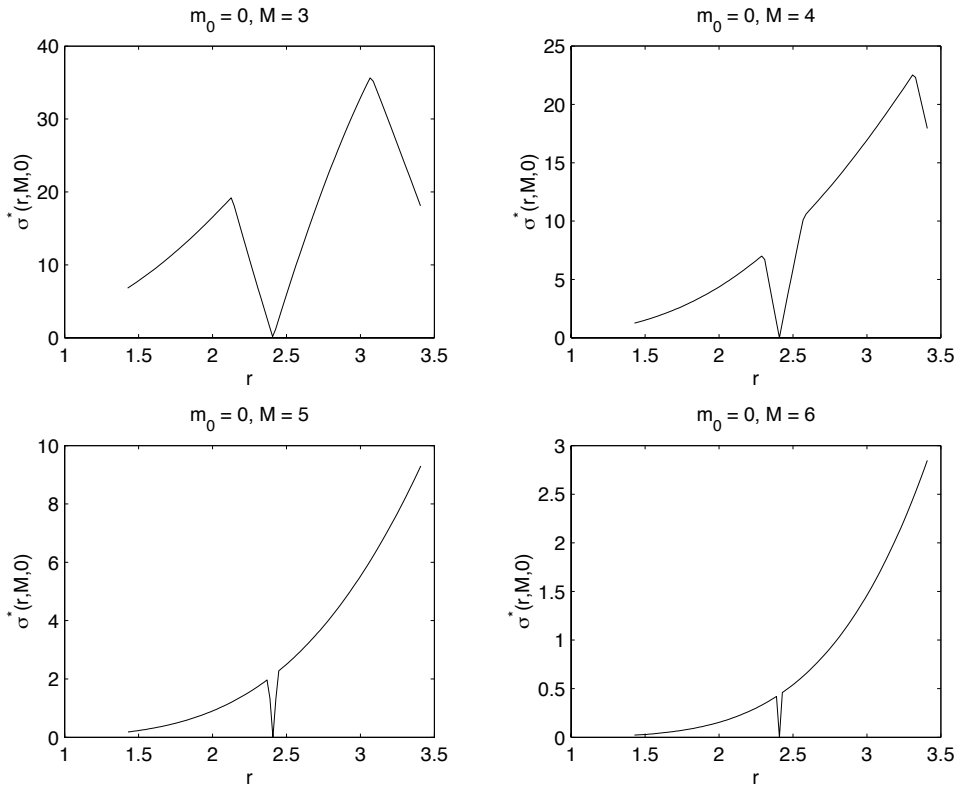


FIG. 4.1. *Plots of $f_M(r) = \sigma^*(r, M, m_0)$ for various values of M and $m_0 = 0$. The domain is defined by $\rho(\theta) = 1 + 0.05\sin(2\theta)$.*

$i \in I_m(r, M, m_0)$, in the expansion of the approximate eigenfunction. We then increase the value of M and repeat this process until our approximate eigenvalue r converges to within some tolerance. It should be noted that Proposition 2.1 suggests an upper limit on M.

4.4. Neumann Boundary Conditions. With minor modifications, we can find eigenfunctions that satisfy Neumann boundary conditions. For simplicity, we again consider a star-shaped domain Ω defined by the polar equation $r = \rho(\theta)$ for $\theta \in [0, 2\pi)$. For each θ, we define $\phi_N(\theta)$ to be the angle that the outward unit normal vector at the boundary point $(\rho(\theta)\cos\theta, \rho(\theta)\sin\theta)$ makes with the positive x-axis. Our goal is to solve an equation similar to (3.9),

$$\sum_{m\in\mathbb{Z}} \alpha_m \frac{\partial}{\partial n}[J_m(r\rho(\theta))e^{i\,m\theta}] = 0\,, \quad \theta \in [0, 2\pi)\,, \tag{4.5}$$

where

$$\frac{\partial}{\partial n}[J_m(r\rho(\theta))e^{i\,m\theta}] = r\left[J_{m-1}(r\rho(\theta))e^{i[(m-1)\theta+\phi_N(\theta)]} - J_{m+1}(r\rho(\theta))e^{i[(m+1)\theta-\phi_N(\theta)]}\right]. \tag{4.6}$$

This leads to an algorithm analogous to the Dirichlet case, in which we seek r that minimizes $\sigma_1(A_N(r, M, m_0))$, where M and m_0 are chosen as before and the matrix

$A_N(r, M, m_0)$ is defined as follows. As in the Dirichlet case, we begin with the set of functions $\{\varphi_m(r,\theta)\}_{m\in I(M,m_0)}$ defined earlier in this section, and a set of gridpoints $\theta_i{}_{i=1}^{N}$ chosen from $[0, 2\pi)$.

Using (4.6), we evaluate the normal derivatives of $\varphi_m(r, \theta_i)$ for each $m \in I(M, m_0)$ and $i = 1, \ldots, n$. These values are stored in a matrix $\Phi_N(r, M, m_0)$, in a manner analogous to the definition of $\Phi(r, M, m_0)$ in the Dirichlet case. We then normalize the columns of $\Phi_N(r, M, m_0)$ so that each has unit 2-norm, as before. We construct the matrix $\Phi_N(\epsilon, M, m_0)$ in the same manner and orthogonalize its columns to obtain $Q_N(\epsilon, M, m_0)$. Finally, we obtain

$$A_N(r, M, m_0) = Q_N(\epsilon, M, m_0)^H \Phi_N(r, M, m_0).$$

Using this matrix in place of $A(r, M, m_0)$, the algorithm proceeds exactly as in the Dirichlet case, except that if a homotopy from a circle to $\partial\Omega$ is used, the normal vectors for each intermediate boundary must be approximated numerically.

5. Numerical Results. We now demonstrate the accuracy of our approach on a number of examples.

5.1. Domains with Smooth Boundaries. First, we consider a star-shaped domain with a smooth boundary, defined by (3.8) with $\rho(\theta) = 1 + 0.05\sin(2\theta)$. The domain is shown in Figure 5.1(a). Table 5.1 lists approximations to the ten smallest

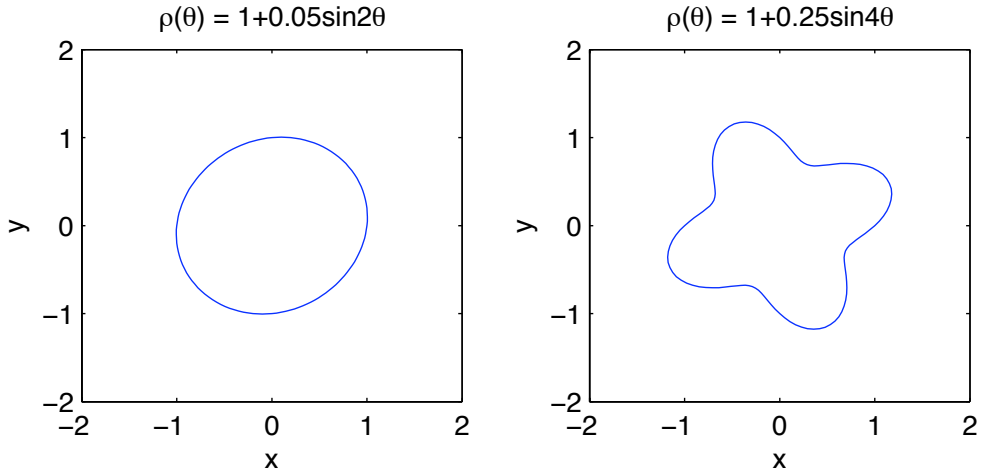


FIG. 5.1. *(a) Left plot: domain with boundary described by polar equation* $\rho(\theta) = 1+0.05\sin 2\theta$. *(b) Right plot: domain with boundary described by polar equation* $\rho(\theta) = 1 + 0.25\sin 4\theta$.

eigenvalues. Note that the accuracy is not much larger than machine precision, and that the number of iterations is comparable for all eigenvalues. We now examine the coefficients α_m of the corresponding eigenfunctions. It can be seen in Figure 5.2 that for fixed values of m, the coefficients $\{\tilde{\alpha}_m\}$, $\{\hat{\alpha}_m\}$ appear to converge rapidly as M increases.

We can also obtain high accuracy for domains whose boundaries exhibit greater oscillation. The domain defined by (3.8) with $\rho(\theta) = 1 + 0.25\sin(4\theta)$ is shown in Figure 5.1(b). Table 5.2 lists approximations to the smallest ten eigenvalues. As can be seen in the table, more iterations and more expansion terms are required than in the previous example.

k	m_0	M_{max}	$r_k = \sqrt{\mu_k}$	$\sigma^*(r_k, M_{max}, m_0)$	Iterations
1	0	12	2.40900956310833	5.1915e-015	13
2	1	14	3.74095865321159	1.4623e-015	17
3	1	14	3.93284180810808	8.2596e-016	17
4	2	14	5.07072160959815	1.3518e-015	30
5	2	14	5.13847966669082	2.843e-015	14
6	0	14	5.60117924855279	8.3542e-015	17
7	3	14	6.35615204542185	3.1301e-015	15
8	1	14	6.89470061520270	5.5334e-015	18
9	1	14	7.23051290051418	2.4392e-015	18
10	4	14	7.55755578029419	2.0736e-015	16

TABLE 5.1

Approximations $\{\mu_k\}_{k=1}^{10}$ to the ten smallest eigenvalues $\{\lambda_k\}_{k=1}^{10}$ of $-\Delta$ on the star-shaped domain with boundary given by $\rho(\theta) = 1 + 0.05 * sin(2\theta)$, $0 \le \theta < 2\pi$. The value M_{max} indicates the largest value of M used.

k	m_0	M_{max}	$r_k = \sqrt{\mu_k}$	$\sigma^*(r_k, M_{max}, m_0)$	Iterations
1	0	25	2.68969299056912	2.0328e-015	30
2	1	25	4.12955410262284	8.1337e-016	41
3	2	25	4.85603553243775	1.6213e-015	29
4	2	22	5.48806069658009	2.3608e-015	20
5	0	25	6.16258921311118	1.4208e-014	46
6	3	24	6.23430327124296	2.5041e-015	36
7	4	23	7.34380858405790	8.3404e-015	28
8	4	25	7.43120441792176	9.8004e-015	50
9	2	25	7.63915899496748	9.9413e-013	48
10	5	24	8.05954250413219	2.3375e-016	38

TABLE 5.2

Approximations $\{\mu_k\}_{k=1}^{10}$ to the ten smallest eigenvalues $\{\lambda_k\}_{k=1}^{10}$ of $-\Delta$ on the star-shaped domain with boundary given by $\rho(\theta) = 1 + 0.25 * sin(4\theta)$, $0 \le \theta < 2\pi$. The value M_{max} indicates the largest value of M used.

5.2. Polygons. Next, we move on to piecewise smooth domains, considering polygons. Table 5.3 lists approximations of the smallest ten eigenvalues on the unit square for the Dirichlet case. These eigenvalues can be computed analytically, and thus we can see that very high accuracy can be obtained in this case, just like for domains with smooth boundaries. Comparable accuracy can be obtained for the case of Neumann boundary conditions on the unit square, as shown in Table 5.4.

Next, we consider a regular polygon with 128 sides. The eigenvalues for this domain are not known, but in [10], the ten smallest simple eigenvalues were approximated using a three-term Taylor series centered at the eigenvalues of the circle. The approximations listed in Table 5.5 were obtained by considering only the case where $M = 0$ and $m_0 = 0$. The discretization of the polygon used to construct $A(r, 0, 0)$ consisted of 32 points per side, and experimentation with other discretizations showed quadratic convergence in the number of points. Because the boundary of the polygon is well-approximated by a circle, zeros of Bessel functions provide very accurate initial iterates, thus making iteration beyond $M = 0$ unnecessary. Non-simple eigenvalues can be obtained by considering other values of m_0.

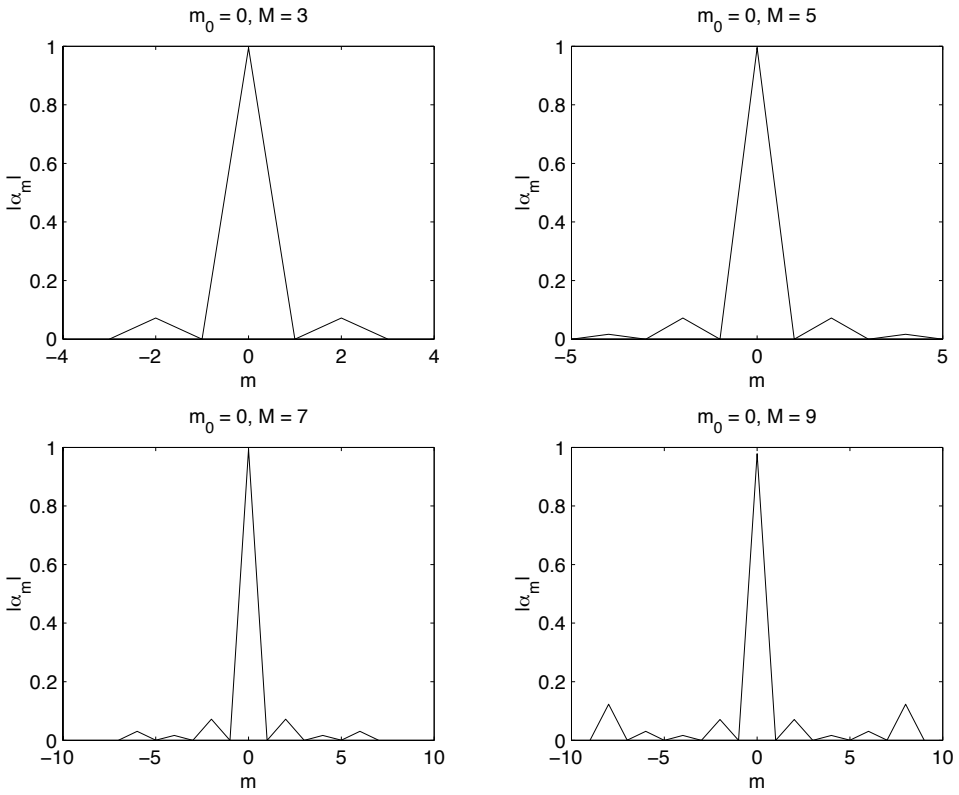


FIG. 5.2. *For increasing values of M, coefficients $\{\tilde{\alpha}_m\}_{m=m_0-M}^{m_0+M}$ that are approximate solutions of the determining equation (3.9) for eigenfunctions satisfying Dirichlet boundary conditions on the star-shaped domain defined by $\rho(\theta) = 1 + 0.05\sin(2\theta)$*

5.3. Detecting Multiple and Clustered Eigenvalues. Multiple eigenvalues and closely clustered eigenvalues can be detected using the smallest singular values of $A(r, M, m_0)$. Specifically, if r^2 is an eigenvalue of multiplicity k, then we can expect that the k smallest singular values of $A(r, M, m_0)$ should be nearly zero if M is sufficiently large. Furthermore, if the $(k+1)$st singular value is small, then this suggests that another eigenvalue is nearby.

To illustrate this, we consider the case of the rectangle $[0,1] \times [0, 1+\epsilon]$ which, for Dirichlet boundary conditions, has eigenvalues $\pi^2(m^2 + (n/(1+\epsilon))^2)$ for positive integers m and n. This leads to several double eigenvalues when $\epsilon = 0$, and clustered eigenvalues for small nonzero values of ϵ. In Table 5.6 we consider these cases: (1) $m = 1$ and $n = 2$, and (2) $m = 2$ and $n = 1$. For $\epsilon = 0$, these cases result in a double eigenvalue of $5\pi^2$, whereas for $\epsilon = 10^{-6}$, they yield two distinct eigenvalues that differ by 5.92×10^{-5}. Our algorithm will only find one of them, but for the computed eigenvalue r_1^2, the second smallest singular value of $A(r_1, M, m_0)$ is quite small, indicating that there exists a linear combination of functions of the form $J_m(r_1\rho(\theta))e^{im\theta}$ that, while orthogonal to the computed eigenfunction, nearly satisfies the boundary conditions. It is therefore reasonable to conclude that there is a second eigenvalue r_2^2 nearby, which can be found by scanning the graph of $\sigma_1(A(r, M, m_0))$ near r_1.

k	m_0	M_{max}	μ_k	$\lvert\mu_k - \lambda_k\rvert/\lvert\lambda_k\rvert$	Iterations
1	0	16	19.7392088021787	0	14
2	1	18	49.3480220054467	1.8718e-015	29
3	2	20	78.9568352087148	0	22
4	0	20	98.6960440108938	2.8797e-015	17
5	3	20	128.3048572141659	3.3228e-014	29
6	1	20	167.7832748172576	7.5186e-012	29
7	4	19	177.6528792162691	1.8797e-011	26
8	2	17	197.3920880213265	2.3337e-012	24
9	5	23	246.7401100490182	8.8288e-011	41
10	0	18	256.6097144238706	1.7352e-011	23

TABLE 5.3

Approximations $\{\mu_k\}_{k=1}^{10}$ to the ten smallest eigenvalues $\{\lambda_k\}_{k=1}^{10}$ of $-\Delta$ on the unit square for Dirichlet boundary conditions. The value M_{max} indicates the largest value of M used.

k	m_0	M_{max}	μ_k	$\lvert\mu_k - \lambda_k\rvert/\lvert\lambda_k\rvert$	Iterations
1	1	15	9.8696044010894	8.9991e-016	28
2	2	16	19.7392088021786	2.8797e-015	16
3	2	18	39.4784176043572	3.9596e-015	29
4	3	18	49.3480220054470	4.1756e-015	31
5	4	20	78.9568352087120	3.5097e-014	22
6	4	20	88.8264396098096	6.0954e-014	37
7	1	17	98.6960440063917	4.5614e-011	25
8	5	20	128.3048572141609	5.7594e-015	31
9	2	18	157.9136703981950	1.2181e-010	31
10	0	21	167.7832751306811	1.8605e-009	34

TABLE 5.4

Approximations $\{\mu_k\}_{k=1}^{10}$ to the ten smallest nonzero eigenvalues $\{\lambda_k\}_{k=1}^{10}$ of $-\Delta$ on the unit square for Neumann boundary conditions. The value M_{max} indicates the largest value of M used.

5.4. Domains with Both Singular and Regular Corners. For more challenging test cases, we consider domains that have both regular and singular corners, where a regular corner is a corner with interior angle π/α for an integer α and a singular corner is a corner that is not regular. First, we consider an L-shaped domain, defined by

$$\rho(\theta) = \frac{1}{\max(|\sin\theta|, |\cos\theta|)}, \quad 0 \le \theta < \frac{3\pi}{2},$$

for which eigenvalues were computed in [3] using a modification of the MPS introduced in [5]. This domain has a singular corner of angle $3\pi/2$. For this reason, instead of using the eigenfunctions for the free space operator, we use the eigenfunctions for the operator defined on an unbounded wedge of interior angle $3\pi/2$, which are the Fourier-Bessel functions

$$u_{r,m}(\rho,\theta) = J_{2m/3}(r\rho)\sin\frac{2m\theta}{3}, \quad m = 0, 1, \ldots,$$

which will vanish on the sides of the boundary that define the singular corner.

The algorithm proceeds as described in the previous section, except that the series (3.7) is a linear combination of these eigenfunctions, and the entries of the

k	m_0	μ_k	λ_k	$\lvert\mu_k - \lambda_k\rvert/\lvert\lambda_k\rvert$	Iterations
1	0	5.78551	5.78552	2.2801e-006	2
2	0	30.48349	30.48357	2.6001e-006	2
3	0	74.91706	74.91726	2.6745e-006	2
4	0	139.09608	139.09646	2.7135e-006	2
5	0	223.02177	223.02237	2.6975e-006	2
6	0	326.69441	326.69528	2.6759e-006	2
7	0	450.11409	450.11529	2.6638e-006	2
8	0	593.28086	593.28245	2.6746e-006	2
9	0	756.19474	756.19675	2.6567e-006	2
10	0	938.85573	938.85822	2.6481e-006	2

TABLE 5.5

Approximations $\{\mu_k\}_{k=1}^{10}$ to the ten smallest simple eigenvalues of $-\Delta$ on the 128-sided regular polygon. These approximations are compared to those computed in [10], denoted by λ_k, $k = 1, 2, \ldots, 10$.

ϵ	λ	mult(λ)	$\sigma_1(A(\sqrt{\lambda}, M, m_0))$	$\sigma_2(A(\sqrt{\lambda}, M, m_0))$
0	49.34802200544679	2	0	0
1e-6	49.34794304872997	1	0	3.8616e-006

TABLE 5.6

Smallest two singular values σ_1 and σ_2 of the matrix $A(\sqrt{\lambda}, M, m_0)$ for the domain $[0,1]\times[0,1+\epsilon]$, with $m_0 = 0$ and $M = 20$. For $\epsilon = 0$ the Laplacian has a double eigenvalue at 49.34802200544679 while for $\epsilon = 10^{-6}$, there are two simple eigenvalues at 49.34794304873002 and 49.34800226626760.

matrix $A(r, M, m_0)$ are obtained by taking inner products with functions of the form $\sin(2\pi\theta/3)$ rather than $J_m(\rho(\theta))e^{im\theta}$. Table 5.7 contains the resulting approximations $\{\lambda_k\}_{k=1}^3$ of the smallest three eigenvalues, compared to the approximations $\{\mu_k\}_{k=1}^3$ reported in [3]. It should be noted that both sets of results were obtained using MATLAB implementations of the respective algorithms, run on a Toshiba Satellite A75-211 Notebook with a Pentium 4 processor. Our algorithm computed these three eigenvalues in 1.11 seconds, while the modified MPS algorithm (see [2]) did so in 2.31 seconds. In both cases, 72 points were used in the discretization of the boundary, and the modified MPS algorithm also used 72 randomly chosen interior points.

k	μ_k	λ_k	$\lvert\lambda_k - \mu_k\rvert/\lvert\lambda_k\rvert$	Iterations
1	9.63971927923382	9.63972384464540	4.7360e-007	10
2	15.19727667220087	15.19725192576365	1.6283e-006	11
3	19.73920878892437	19.73920880208238	6.6659e-010	8

TABLE 5.7

Approximations $\{\mu_k\}_{k=1}^3$ to the three smallest eigenvalues of $-\Delta$ on an L-shaped domain using Fourier-Bessel functions of order $2m/3$ for each positive integer m, centered at the singular corner. These approximations are compared to those computed in [3], denoted by λ_k, $k = 1, 2, 3$.

Next, we use the original version of the algorithm, with free space eigenfunctions and an expansion centered at an interior point of the domain instead of at the singular corner. In this case, we obtain only a few correct digits for the first two eigenvalues, while still achieving high accuracy for the third, which coincides with an eigenvalue of the unit square.

Finally, we try our algorithm on one of the GWW drums described in [7], the

k	m_0	μ_k	λ_k	$\lvert\lambda_k - \mu_k\rvert/\lvert\lambda_k\rvert$	Iterations
1	0	9.72384639102523	9.63972384464540	0.00873	70
2	2	15.16260753106696	15.19725192576365	0.00228	72
3	2	19.73920880217876	19.73920880208238	1.2373e-011	30

TABLE 5.8

Approximations $\{\mu_k\}_{k=1}^3$ to the three smallest eigenvalues of $-\Delta$ on an L-shaped domain using free-space eigenfunctions centered in the interior, with $M = 40$. These approximations are compared to those computed in [3], denoted by λ_k, $k = 1, 2, 3$.

GWW1 drum shown in Figure 5.3. We again use free-space eigenfunctions centered in the interior, and a discretization of Chebyshev points on each side, so that points will be clustered near each corner. The results are shown in Table 5.9. As with the L-shaped domain, we obtain at most a few digits. However, it is clear that our algorithm can at least be used to obtain an initial approximation for a method such as the MPS, and future work will focus on improving our algorithm for such domains.

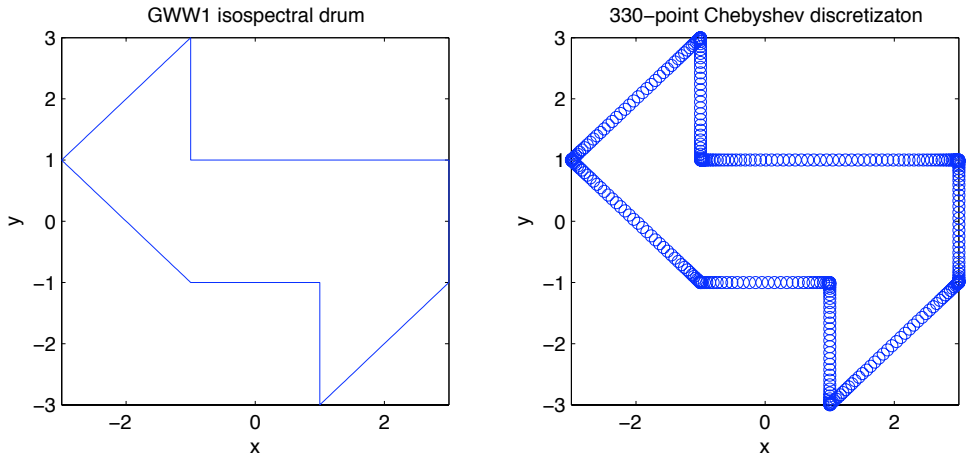


FIG. 5.3. *Left plot: the GWW1 drum. Right plot: a 330-point discretization of the drum, using Chebyshev points on each side.*

k	m_0	μ_k	λ_k	$\lvert\lambda_k - \mu_k\rvert/\lvert\lambda_k\rvert$	Iterations
1	0	2.55996542353383	2.537943999798	0.00868	78
2	1	3.67093788732147	3.655509713520	0.00422	78
3	2	5.16128092767208	5.175559356220	0.00458	78

TABLE 5.9

Approximations $\{\mu_k\}_{k=1}^3$ to the three smallest eigenvalues of $-\Delta$ on the GWW1 drum, with $M = 40$. These approximations are compared to those computed in [3], denoted by λ_k, $k = 1, 2, 3$.

5.5. An Inverse Problem. We now consider an inverse spectral problem, in which n eigenvalues $\lambda_1, \ldots, \lambda_n$ of $-\Delta$ are specified and our goal is to find a domain Ω such that $-\Delta$ has eigenfunctions that vanish on $\partial\Omega$ with corresponding eigenvalues equal to the prescribed values. The solution of this problem is not unique, but we hope to obtain a domain whose boundary is close to our initial guess. To that end,

we define the matrix $A(\rho, r, M, m_0)$ in the same way as we defined $A(r, M, m_0)$ in the previous section, except that in this case, the function $\rho(\theta)$ that defines the boundary is allowed to vary. Then, we solve the minimization problem

$$\rho^*(\theta) = \min_{\rho \in W_n} \sum_{i=1}^{n} \sigma_1(A(\rho, \sqrt{\lambda_i}, M^{(i)}, m_0^{(i)})), \tag{5.1}$$

where W_n is a subspace of $C_p([0, 2\pi])$, and $M^{(i)}$ and $m_0^{(i)}$ determine the size and center of the eigenfunction expansion corresponding to λ_i.

We illustrate the results of this approach with a simple example. We use eight prescribed eigenvalues of the domain

$$\rho(\theta) = 1 + 0.01(\cos\theta + \sin 2\theta + \cos 3\theta), \tag{5.2}$$

computed using the modified MPS, since it is ill-advised to test a method for solving an inverse problem on data obtained using the same method for the forward problem. We then define

$$W_8 = \text{span}\{1, \cos\theta, \cos 2\theta, \cos 3\theta, \sin\theta, \sin 2\theta, \sin 3\theta\},$$

and choose $M^{(i)} = 6$ and $m_0^{(i)} = 0$ for $i = 1, \ldots, 8$. Since our exact solution is a perturbation of the unit circle, we choose the unit circle as our initial guess.

We solve the problem (5.1) using MATLAB's `fminsearch` function with initial guess $\rho_0(\theta) = 1$ and a stopping criterion that the value of the objective function should be less than 10^{-10}. The output is a function $\rho_1(\theta)$ that, after rotation, satisfies $\|\rho_1 - \rho\| \approx 10^{-3}\|\rho\|$, whereas $\|\rho_0 - \rho\| \approx 10^{-2}\|\rho\|$. The domains defined by ρ, ρ_0 and ρ_1 are shown in Figure 5.4.

Figure 5.5 shows the value of the objective function from (5.1) after each iteration performed by `fminsearch`. Note that there are periods of relatively steep drops in the function value, between periods of relatively little change. This suggests that the algorithm is finding multi-dimensional "wells" as it searches W_n, corresponding to the one-dimensional wells observed in the solution of the forward problem. Future work will involve attempting to develop an optimization method that can find these wells as quickly as possible, even if minimizing over a higher-dimensional function space.

5.6. Comparison with the Method of Particular Solutions. The Method of Particular Solutions (MPS), introduced by Fox, Henrici and Moler in [5] and revived by Betcke and Trefethen in [3], can be used to compute eigenvalues of the Laplacian by solving an equation such as (3.9). Our approach is similar to the MPS in that both algorithms locate an approximate eigenvalue λ by minimizing a function $f(\lambda)$ whose value is the smallest singular value of some matrix $A(\lambda)$. In both algorithms, the columns of $A(\lambda)$ correspond to select eigenfunctions of the Laplacian on some unbounded domain, a linear combination of which is taken in (3.9). Aside from this, the algorithms differ in the following ways:

- The number of columns of the matrix $A(\lambda)$ used in the MPS needs to be sufficiently large, relative to the number of points used in the discretization, to obtain an approximate eigenvalue. If too few columns are used, then $A(\lambda)$ will not be close to a singular matrix for any choice of λ. In the original version in [5], the excess of columns leads to instability in some cases, while in the modified version in [3], unnecessary columns are removed by orthogonalization. With our approach, the number of columns is a *lower* bound on the

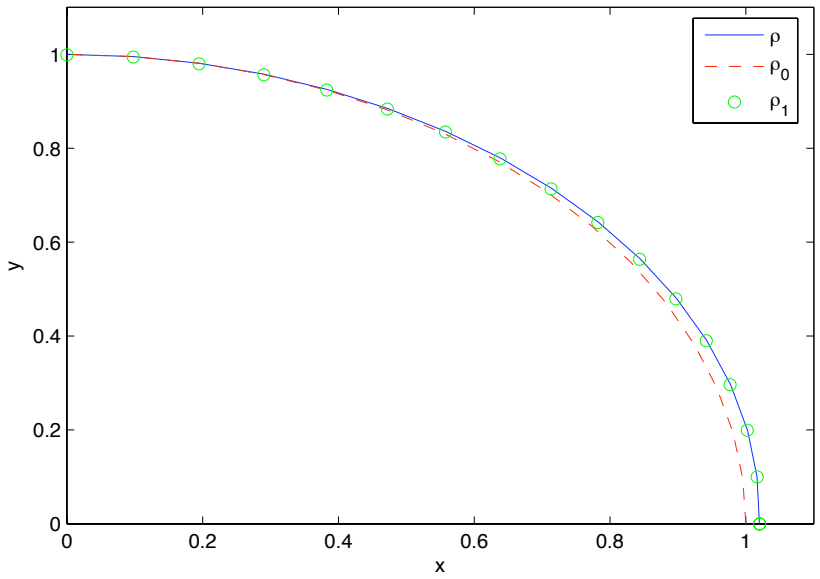
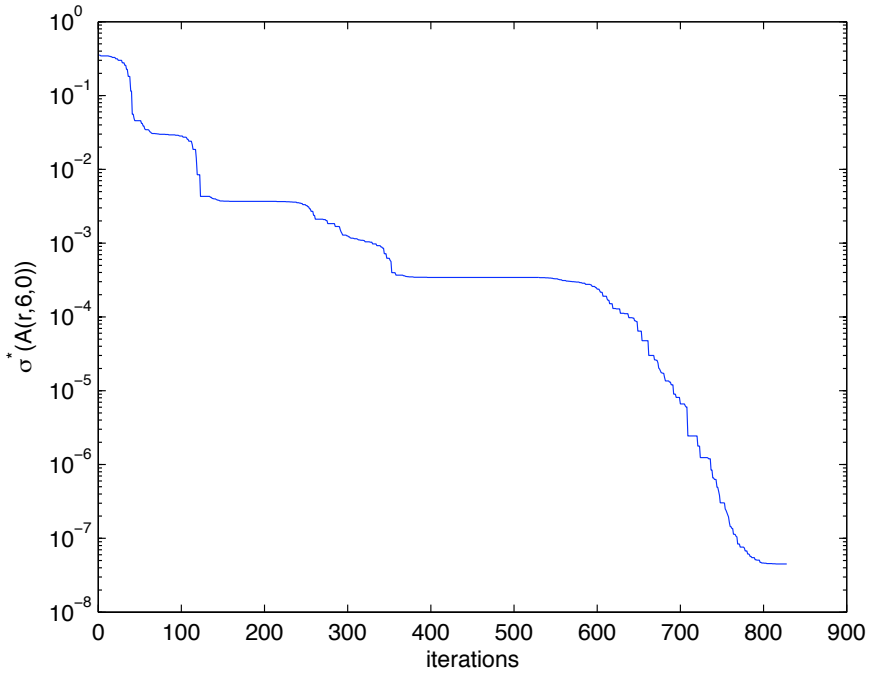


FIG. 5.4. *Results of solving the inverse eigenvalue problem described in Section 5.5. The solid curve is the exact solution $\rho(\theta)$ defined in (5.2), the dashed curve is the initial guess $\rho_0(\theta) = 1$, and the open circles describe the computed solution $\rho_1(\theta)$.*

number of terms needed in (3.9), beginning with only one column and increasing with M, independently of the size of the discretization. This difference in behavior is due to the fact that we require the approximate eigenfunction to vanish on the boundary only after projection into a subspace whose dimension increases with M. Furthermore, the size of the matrix whose SVD is computed depends only on the number of expansion terms, as opposed to the MPS in which the size is determined by the number of expansion terms and the number of points in the discretization.

- The MPS is not restricted to any one minimization algorithm. For example, MATLAB's `fminsearch` can be used, or one can proceed as in [2], evaluating $f(\lambda)$ at several equally spaced points, in order to find approximate minima, and then using inverse polynomial interpolation in order improve the approximation. This strategy can fail when attempting to compute larger eigenvalues, as they may become so clustered that they may elude detection unless an extremely small spacing is used. Because of the larger number of columns used, this approach can be prohibitively expensive. Our algorithm specifies a particular minimization method that attempts to take into account the behavior of $\sigma_1(A(r, M, m_0))$ and minimize the number of evaluations of the objective function. The MPS can be used with this minimization method as well, but it would be more expensive, in view of the previous discussion.
- In the MPS, the columns of $A(\lambda)$ are discretizations of functions defined on both the interior and the boundary of Ω. In our approach, the columns are discretizations of functions defined on only $\partial\Omega$. No interior points are necessary to stabilize the algorithm and prevent spurious solutions. This is because we use a minimal number of columns in our expansion, as previously



discussed. As with the original MPS, the problem of spurious solutions can still occur if too many columns are used, due to the exponential decay of higher-order Bessel functions near the origin, but the steps we have taken to improve the conditioning of $A(r, M, m_0)$ have prevented this from actually occurring in practice before an eigenvalue is found.

- In the MPS, the terms in the expansion are determined by the singular corners, if any. We use an expansion that does not take any such features of the domain into account. Even for difficult domains such as GWW1, we are still able to obtain a reasonable approximation to an eigenvalue that can be improved upon using MPS, using a far more localized search than with using the MPS alone.
- In the original version of the MPS, the coefficients of the eigenfunction expansion corresponding to (3.7) can readily be obtained from the singular vector of $A(\lambda)$ corresponding to the smallest singular value. In the modified MPS, this is no longer the case due to the orthogonalization that takes place; as discussed in [3], an ill-conditioned system of equations must be solved. In our algorithm, these coefficients are readily available once again because we do not orthogonalize $\Phi(r, M, m_0)$ in computing $A(r, M, m_0)$. We did perform this orthogonalization in some experiments, and it did improve the conditioning of $A(r, M, m_0)$, but in some cases the SVD would fail to converge.

We have seen in this section that these differences can lead to greater computational efficiency for our approach. For domains with both regular and singular corners, our algorithm can be used in conjunction with MPS to compute eigenvalues with high accuracy and greater efficiency than with either method used alone.

6. Conclusions. An iterative method for the numerical computation of eigenvalues has been proposed which relies on calculations performed on small matrices and which allows for the independent targeting of individual eigenvalues. Furthermore, the method does not require the evaluation of functions at points in the interior of the domain, and it does not require exhaustive scanning of the positive real axis. Numerical experiments and comparisons corroborate these claims. It is our hope that the approach described in this work will eventually lead to effective methods for solving a variety PDE on general 2-D and 3-D domains, as well as insight concerning the eigensystem of the Laplacian and related operators on such domains.

REFERENCES

[1] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*, Dover, New York, 1970.

[2] T. Betcke and L. N. Trefethen, *Ldrum.m - eigenvalues of laplacian on l-shaped region (matlab implementation)*, http://web.comlab.ox.ac.uk/oucl/work/nick.trefethen/Ldrum.m, (2003).

[3] ———, *Reviving the method of particular solutions*, SIAM Review, 47 (2005), pp. 469–491.

[4] T. A. Driscoll, *Eigenmodes of isospectral drums*, SIAM Review, 39 (1997), pp. 1–17.

[5] L. Fox, P. Henrici, and C. Moler, *Approximations and bounds for eigenvalues of elliptic operators*, SIAM J. Numer. Anal., 4 (1967), pp. 89–102.

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

[7] C. Gordon, G. Webb, and S. Wolpert, *Isospectral plane domains and surfaces via Riemannian orbifolds*, Invent. Math., 110 (1992), pp. 1–22.

[8] S. Helgason, *Groups and Geometric Analysis*, vol. 83 of Mathematical Surveys and Monographs, American Mathematical Society, Providence, 2000.

[9] J. V. Lambers, *Krylov subspace spectral methods for variable-coefficient initial-boundary value problems*, Elec. Trans. Numer. Anal., 20 (2005), pp. 212–234.

[10] G. Strang and P. Grinfeld, *The Laplacian Eigenvalues of a Polygon*, Computers and Mathematics with Applications, 48 (2004), pp. 1121–1133.

[11] H. Wu, D. Sprung, and J. Martorell, *Numerical investigations of isospectral cavities built from triangles*, Phys. Rev. E, 51 (1995), pp. 703–708.