Chapter 1
New Numerical Results for the Optimization of Interior Neumann Eigenvalues

D. Abele and A. Kleefeld

Abstract We present new numerical results for shape optimization problems of interior Neumann eigenvalues. This field is not well understood from a theoretical standpoint. The existence of shape maximizers is not proven beyond the first two eigenvalues, so we study the problem numerically. We describe a method to compute the eigenvalues for a given shape that combines the boundary element method with an algorithm for nonlinear eigenvalues. As numerical optimization requires many such evaluations, we put a focus on the efficiency of the method and the implemented routine. The method is well suited for parallelization. Using the resulting fast routines and a specialized parametrization of the shapes, we found improved maximums for several eigenvalues.

1.1 Introduction

We will discuss the optimization of interior Neumann eigenvalues with respect to the shape of the domain. To state the problem precisely, consider an open, possibly disconnected set $\Omega \in \mathbb{R}^2$ with smooth boundary $\partial\Omega$. The normal onto the boundary at point $x \in \partial\Omega$ directed into the exterior is $\nu := \nu(x)$. Interior Neumann eigenvalues are values $\lambda = \kappa^2 \in \mathbb{R}$ for which the boundary value problem (BVP)

$$\Delta u + \kappa^2 u = 0 \quad \text{in } \Omega \tag{1.1a}$$

$$\frac{\partial u}{\partial \nu} = 0 \quad \text{on } \partial\Omega \tag{1.1b}$$

has non-trivial solutions. Precisely, equation (1.1a) is the Helmholtz equation with wavenumber κ in the interior of Ω and equation (1.1b) is the homogeneous Neumann boundary condition. The problem arises in the study of acoustic scattering

D. Abele and A. Kleefeld at Forschungszentrum Jülich, Jülich Supercomputing Centre, 52425 Jülich, Germany, e-mail: d.abele@fz-juelich.de, e-mail: a.kleefeld@fz-juelich.de

[CoKr13]. It is well-known that the eigenvalues are discrete, real and nonnegative:

$$0 = \lambda_0 \leq \lambda_1 \leq \lambda_2 \leq \dots .$$

The eigenvalues depend on the domain. The optimization problem for the k-th eigenvalue is

$$\max_{\Omega} \{\lambda_k(\Omega)\}$$
$$\text{s.t. } |\Omega| = 1$$

with $|\Omega|$ denoting the area of the domain. The area must be constrained as the eigenvalues are inversely proportional to the area. So the goal is to find the shape of the domain that maximizes λ_k for $k > 0$ among all domains of constant area. The eigenvalue λ_0, for which (1.1) has only constant solutions, is ignored here as it is always zero.

There has been some theoretical and numerical work in this area. Szegö and a little later Weinberger have shown that the first eigenvalue is maximized by a disk [Sz54, We56]. The second eigenvalue is maximized by the union of two disjoint disks of the same size [GiNaPo09]. It is so far unknown if maximizers for higher eigenvalues exist. However, it has been shown that disjoint unions of disks do not maximize all eigenvalues [PoRo10], so there is room for exploration. Recent numerical results suggest that maximizers for the first ten eigenvalues exist and follow a certain system [AnOu17, AnFr12]. That system has been exploited to get more precise results for some eigenvalues [Kl19]. Those numerical results are summarized in Table 1.1.

k	[AnFr12]	[AnOu17]	[Kl19]
1	–	10.66 (2)	–
2	–	21.28 (4)	–
3	32.79 (3)	32.90 (3)	32.9018 (3)
4	43.43 (5)	43.86 (3)	43.8694 (3)
5	54.08 (7)	55.17 (3)	–
6	67.04 (4)	67.33 (4)	–
7	77.68 (6)	77.99 (6)	–
8	89.22 (4)	89.38 (4)	–
9	101.73 (4)	101.83 (4)	–
10	113.86 (5)	114.16 (5)	–

Table 1.1 Recent results for the numerical optimization of interior Neumann eigenvalue λ_k. The value in parantheses is the multiplicity.

Contribution

This work expands on the idea of [Kl19]. We show that the parametrization of shapes presented there is not very successful beyond the fourth eigenvalue. By introducing additional parameters we managed to get improved optimization results for some eigenvalues, while still using fewer parameters than a general Fourier series approach. As the performance of the eigenvalue solver directly affects the achievable precision, we discuss the employed methods and the implementation in greater detail and explain some adaptations that make optimization feasible on a larger scale than before: more eigenvalues, more degrees of freedom and greater precision. In particular, we developed a strongly scaling parallelization scheme.

Outline

In Section 1.2 we present the method of computing the eigenvalues and its implementation. The numerical methods - boundary element method to discretize the BVP and the contour integral method of Beyn to solve the nonlinear eigenvalue problem - are discussed in detail in Sections 1.2.1 and 1.2.2. That discussion motivates the parallelization scheme that is explained in Section 1.2.3. Section 1.3 is dedicated to the actual optimization. After a quick summary of the shape parametrization of [Kl19] we present the disappointing results of using that parametrization in the maximization of further eigenvalues. We then extend the parameter space and show the much improved results. Finally we will give our conclusion and a small outlook in Section 1.4.

1.2 Computation of Eigenvalues

The process to compute eigenvalues is the same as in [Kl19] with the exception of parallelization and some other important modifications that have major implications on the required computational effort. First, the BVP (1.1) is discretized using the boundary element method. The resulting homogeneous linear system is a nonlinear eigenvalue problem that is solved with the method of Beyn. To motivate the parallelization, we will give a quick summary of these methods while the modifications are highlighted and discussed in detail.

1.2.1 The Boundary Element Method

The theory of this method is covered in [CoKr83]. Using a single layer potential ansatz, the BVP (1.1) is first converted into the integral equation of the second kind

$$\frac{1}{2}\psi(x)+\int_{\partial\Omega}\frac{\partial}{\partial\nu(x)}\Phi_\kappa(x,y)\,\psi(y)\,\mathrm{d}s(y)=0\,,\quad x\in\partial\Omega \tag{1.2}$$

whose solution $\psi \in C(\partial\Omega)$ is the density of the solution of the BVP. The kernel $\Phi_\kappa(x,y) := \frac{\mathrm{i}}{4}H_0^{(1)}(\kappa\|x-y\|)$ with $H_0^{(1)}$ the Hankel function of the first kind of order zero is the fundamental solution of the PDE (1.1a). We choose n points $x_i, i = 1,\dots,n$ on the boundary that form $n/2$ boundary elements with two endpoints and one midpoint. The integral equation (1.2) is discretized using piecewise quadratic interpolation of ψ and the boundary. Collocation results in a homogeneous linear system

$$\underbrace{\left(\frac{1}{2}\mathbf{I}+\mathbf{M}(\kappa)\right)}_{\mathbf{T}(\kappa)}\vec{\psi} = 0 \tag{1.3}$$

with system matrix $\mathbf{T}(\kappa)$ and identity matrix $\mathbf{I} \in \mathbb{C}^{n\times n}$. The entries of matrix $\mathbf{M}(\kappa) \in \mathbb{C}^{n\times n}$ are integrals over the quadratic boundary elements of the form

$$\int_0^1 \frac{\partial}{\partial \nu_{i,k}} \Phi_\kappa(x_i, g_k(t))L_j(t)\left\|g_k'(t)\right\| \,\mathrm{d}t \tag{1.4}$$

with x_i the collocation point, $k = 1,\dots,n/2$ the index of the boundary element, $g_k : [0,1] \ni t \mapsto L_1(t)x_{2k-1} + L_2(t)x_{2k} + L_3(t)x_{2k+1}$ the quadratic interpolation polynomial of the k-th boundary element, $\nu_{i,k}$ some approximation of the normal onto the boundary at x_i that may depend on the boundary element (see below), and $L_j, j = 1,2,3$ the j-th quadratic Lagrange basis polynomial. The integral describes the influence of the j-th point of boundary element k on collocation point x_i, thus index i is the row index of the matrix entry and indices j and k depend on the column index. Elements in odd indexed rows correspond to the endpoints of boundary elements $(x_1,x_3,\dots)$ and as such are the sum of two such integrals with different j,k as they belong to two different boundary elements. The kernel

$$\frac{\partial}{\partial \nu_{i,k}}\Phi_\kappa(x,y) = -\frac{\mathrm{i}\kappa}{4}H_1^{(1)}(\kappa\|x-y\|)\frac{1}{\|x-y\|}\left\langle x-y, \nu_{i,k}\right\rangle \tag{1.5}$$

has a singularity at $x = y$ the type of which depends on the choice of $\nu_{i,k}$.

Handling of the Singular Kernel

The singularity of the kernel (1.5) must be handled correctly when evaluating the integrals (1.4) numerically. Note first that there is only a singularity in integral (1.4) if collocation point x_i is part of boundary element k, i.e. $g_k(t_0) = x_i$ for some $t_0 \in [0,1]$. Otherwise the integrand is continuous and does not present any specific challenge to numerical quadrature. Let us now assume that there is a singularity in the k-th boundary element. We will examine the limit of the kernel by isolating the singular part. First the Hankel function is replaced by $H_1^{(1)}(z) = J_1(z) + \mathrm{i}Y_1(z)$ where J_1 and Y_1 are the Bessel functions of the first and second kind of order one. As $z \to 0$, $J_1(z)$ tends to zero, so we have

$$\begin{aligned}&\lim_{y\to x}\left(-\frac{\mathrm{i}\kappa}{4}H_1^{(1)}(\kappa\|x-y\|)\left\langle\frac{x-y}{\|x-y\|},\nu_{i,k}\right\rangle\right)\\ =&\lim_{y\to x}\left(\frac{\kappa}{4}Y_1(\kappa\|x-y\|)\left\langle\frac{x-y}{\|x-y\|},\nu_{i,k}\right\rangle\right).\end{aligned} \tag{1.6}$$

We replace Y_1 with its power series expansion

$$\begin{aligned}Y_1(z)=&-\frac{2}{\pi z}+\frac{2}{\pi}\ln(\frac{1}{2}z)J_1(z)\\ &+\frac{1}{\pi}\sum_{k=0}^{\infty}\frac{(-1)^k}{k!(k+1)!}(\psi_0(k+1)+\psi_0(k+2))\left(\frac{1}{2}z\right)^{2k+1}\end{aligned}$$

with ψ_0 the digamma function. Of this expansion only the first summand is infinite. The remaining summands again tend to zero as $z\to 0$. This reduces (1.6) to

$$\lim_{y\to x}\left(-\frac{1}{2\pi}\left\langle\frac{x-y}{\|x-y\|^2},\nu_{i,k}\right\rangle\right). \tag{1.7}$$

The limit does not exist for any general vectors x, y, and $\nu_{i,k}$, so further constraints are applied. The points x and y lie on the graph of g_k with $x=g(t_0)$. By Taylor's theorem there exists a function $h:\mathbb{R}\to\mathbb{R}^2$ with $\lim_{t\to t_0}h(t)=0$ such that

$$g_k(t)=g_k(t_0)+g_k'(t_0)(t-t_0)+\frac{1}{2}g_k''(t_0)(t-t_0)^2+h(t)(t-t_0)^2.$$

Inserting this into (1.7), the remainder term vanishes as it tends to zero. Let $\nu_{i,k}$ be the normal vector onto g_k at t_0. Then, the linear term vanishes as $g_k'(t_0)$ is the tangent onto g_k at t_0 and $\langle g_k'(t_0),\nu_{i,k}\rangle=0$. The constant terms cancel and only the quadratic term remains. Hence, we are left with

$$\lim_{t\to t_0}\left(\frac{1}{4\pi}\left\langle\frac{g_k''(t_0)(t-t_0)^2}{\|g_k(t_0)-g_k(t)\|^2},\nu_{i,k}\right\rangle\right).$$

The difference quotient tends to the derivative and we find the result

$$\frac{1}{4\pi}\left\langle\frac{g_k''(t_0)}{\|g_k'(t_0)\|^2},\nu_{i,k}\right\rangle.$$

So under the assumptions from the beginning that x_i is part of element k, the singularity in the kernel (1.5) is removable and the integrals (1.4) are proper if $\langle g_k'(t_0),\nu_{i,k}\rangle=0$. Note that the existence of the integral in (1.2) can be shown with the same argument. In [Kl19], the normal onto the exact boundary, which is generally not normal to g_k, was used in the numeric computation. This results in infinite singularities that are difficult to handle. Here, we will use normals onto the interpolation polynomials. We denote the normal onto g_k at t_0 as $g_k^{\perp}(t_0)$. For even indices i (which correspond to the midpoint of element $i/2$), we always use

$g_{i/2}^{\perp}(1/2)$. For odd indices i, the point x_i is part of two boundary elements and we have $g_{(i+1)/2-1}(1) = g_{(i+1)/2}(0) = x_i$. We have a choice of two different normals at x_i. When we integrate over element $(i+1)/2-1$, we need to choose $g_{(i+1)/2-1}^{\perp}(1)$ and analogously for element $(i+1)/2$. Otherwise the choice is arbitrary, so we alternate to avoid introducing bias. In summary

$$\nu_{i,k} = \begin{cases} g_{i/2}^{\perp}(\frac{1}{2}) & \text{if } i \bmod 2 = 0 \\ g_{(i+1)/2-1}^{\perp}(1) & \text{if } i \bmod 2 = 1 \text{ and } k \bmod 2 = (\frac{i+1}{2}-1) \bmod 2 \\ g_{(i+1)/2}^{\perp}(0) & \text{if } i \bmod 2 = 1 \text{ and } k \bmod 2 = \frac{i+1}{2} \bmod 2 \end{cases} .$$

Now the integrals can be evaluated without any special handling of the singularity. This makes the quadrature less expensive. We have used a routine that uses the 15 point Gauss-Kronrod rule to solve the integrals to within a relative and absolute tolerance of 10^{-10}.

Exploiting Symmetries of the Domain

The shapes considered in this work are all symmetric to some degree. This fact can be exploited to reduce the required work. In the integrand of the integrals (1.4) with kernel (1.5), the collocation point x_i, normal $\nu_{i,k}$, and integration point $y = g_k(t)$ exclusively exist in norms and scalar products. Those are invariant under rotation:

$$\|\mathbf{R}v\| = \|v\|$$
$$\langle \mathbf{R}v, \mathbf{R}w \rangle = \langle v, w \rangle$$

for all $v, w \in \mathbb{R}^2$ and all rotation transformations $\mathbf{R}$. Under reflection, scalar products switch signs but this is compensated by the switching of the integration bounds, so the resulting integral is again invariant. Thus, if x_a is the image of x_b and x_c is the image of x_d under reflection or rotation, then $\mathbf{M}(\kappa)_{ac} = \mathbf{M}(\kappa)_{bd}$. For this to work, n must be divisible by two times the degree of symmetry and the boundary elements must have the same symmetries as the shape itself (Fig. 1.1).

As an example for what effect this has on matrix $\mathbf{M}(\kappa)$ we will discuss the suspected shape maximizer of λ_3, which has degree of symmetry six (three rotations times two for reflection symmetry, Fig. 1.1). For simplicity, we assume that the first collocation point x_1 lies on a symmetry axis. Threefold rotational symmetry and the corresponding shifting of rows and columns by $n/3$ leads to the matrix having 3×3 block structure

$$\mathbf{M}(\kappa) = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{C} \\ \mathbf{C} & \mathbf{A} & \mathbf{B} \\ \mathbf{B} & \mathbf{C} & \mathbf{A} \end{pmatrix}$$

with some blocks $\mathbf{A}, \mathbf{B}, \mathbf{C} \in \mathbb{C}^{n/3 \times n/3}$ that each have to be evaluated only once, reducing the required work to a third. Reflection symmetry and the corresponding reflection of both rows and columns of the matrix leads to the matrix being centrally

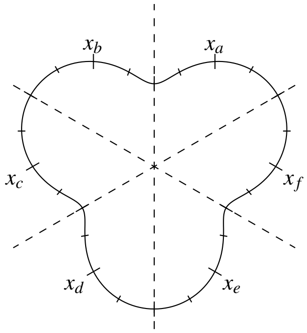


Fig. 1.1 Suspected shape maximizer of λ_3, discretized using $n = 24$ points that form 12 boundary elements. Long ticks mark the endpoints, short ticks the midpoints. The shape has degree of symmetry six, three rotations times two for reflection symmetry. The discretization parameter n is divisible by 12, two times the degree of symmetry. The boundary elements have the same symmetry as the shape itself. The points x_a, x_b, x_c, x_d, x_e, and x_f are images of each other.

symmetric with respect to the entry at row and column $n/2+1$, e.g. for $n = 4$ the matrix

$$\begin{pmatrix} a & e & b & e \\ f & g & h & i \\ c & j & d & j \\ f & i & h & g \end{pmatrix}$$

is point symmetric with respect to entry $(3,3)$ using periodic indices. Note that this example is just for illustration as $n = 4$ would not be valid for degree of symmetry six due to the restrictions mentioned above. Each value repeats twice except where row index and column index both correspond to fixed points of the reflection, i.e. to points that lie on the symmetry axis. As the boundary elements are symmetric and x_1 lies on the axis, so does $x_{n/2+1}$. In our example, values of entries $(1,1)$, $(1,n/2+1)$, $(n/2+1,1)$, and $(n/2+1,n/2+1)$ (or a, b, c, and d) do not repeat. Rotation and reflection symmetry combine so that all values repeat six times, except for the fixed points of reflection which only repeat three times.

Rotation symmetry of degree r reduces the number of matrix entries that must be evaluated to n^2/r. Reflection symmetry further reduces the amount to $n^2/2r+4$. Except for infinitely symmetric shapes like a circle we usually have $n \gg r$, so the complexity of the method with respect to the number of collocation points does not change, but the total amount of work is reduced by a factor approaching $2r$.

1.2.2 Beyn's Contour Integral Method

The eigenvalues of the nonlinear eigenvalue problem (1.3) are computed using the contour integral method W.-J. Beyn presented in [Be12]. For simplicity, we assume here that all eigenvalues are simple, but the method works identically for multiple eigenvalues. Given an operator $\mathbf{T}:\Gamma\to\mathbb{C}^{n\times n}$ that is holomorphic on a domain $\Gamma\subset\mathbb{C}$ and a closed contour $C\subset\Gamma$ with its interior $\operatorname{int}(C)\subset\Gamma$, the method computes all eigenvalues of $\mathbf{T}$ in the interior $\operatorname{int}(C)$.

Let $\kappa_i, i=1,\dots,k$ be the eigenvalues of $\mathbf{T}$ in the interior $\operatorname{int}(C)$ and v_i and w_i the corresponding left and right eigenvectors that are normalized so that $w_i^{\mathrm{H}}\mathbf{T}'(\kappa_i)v_i=1$. Then the equation

$$\frac{1}{2\pi\mathrm{i}}\int_C f(\kappa)\mathbf{T}(\kappa)^{-1}\,\mathrm{d}\kappa=\sum_{i=1}^{k} f(\kappa_i)v_iw_i^{\mathrm{H}} \tag{1.8}$$

holds for all holomorphic functions $f:\Gamma\to\mathbb{C}$ [Be12, Theorem 2.9]. We additionally assume that $k<n$, which is sufficient for our purposes. Beyn describes an extension to the method for $k\geq n$. Applying (1.8) to the functions $f_0(\kappa)=1$ and $f_1(\kappa)=\kappa$ and multiplying with a random matrix $\mathbf{Z}\in\mathbb{C}^{n\times m}$ from the right yields two equations

$$\mathbf{A}_0=\int_C \mathbf{T}(\kappa)^{-1}\mathbf{Z}\,\mathrm{d}\kappa=\mathbf{V}\mathbf{W}^{\mathrm{H}}\mathbf{Z} \tag{1.9a}$$

$$\mathbf{A}_1=\int_C \kappa\mathbf{T}(\kappa)^{-1}\mathbf{Z}\,\mathrm{d}\kappa=\mathbf{V}\mathbf{K}\mathbf{W}^{\mathrm{H}}\mathbf{Z} \tag{1.9b}$$

with $\mathbf{V}=(v_1,\dots,v_m)$, $\mathbf{W}=(w_1,\dots,w_m)$, and $\mathbf{K}=\operatorname{diag}(\kappa_1,\dots,\kappa_m)$. The dimension m is an initial guess for k with $k\leq m\leq n$. Therefore, the matrix $\mathbf{Z}$ reduces the dimensions of $\mathbf{A}_0$ and $\mathbf{A}_1$ without reducing the rank k. Singular value decomposition (SVD) of $\mathbf{A}_0$ in reduced form yields

$$\mathbf{A}_0=\mathbf{V}_0\mathbf{S}_0\mathbf{W}_0^{\mathrm{H}}$$

with matrices $\mathbf{V}_0\in\mathbb{C}^{n\times k}$ and $\mathbf{W}_0\in\mathbb{C}^{m\times k}$ and the diagonal matrix of singular values $\mathbf{S}_0=\operatorname{diag}(\sigma_1,\dots,\sigma_k)$. With this, the correctness of the initial guess m can be confirmed by comparing it with the actual computed rank. In case the check is failed, the method is started again with a higher guess m. Finally a matrix

$$\mathbf{B}=\mathbf{V}_0^{\mathrm{H}}\mathbf{A}_1\mathbf{W}_0\mathbf{S}_0^{-1}=\mathbf{Q}\mathbf{K}\mathbf{Q}^{-1}$$

is computed. The matrix is diagonalizable by construction with $\mathbf{Q}=\mathbf{V}_0^{\mathrm{H}}\mathbf{V}$ and has eigenvalues $\kappa_i, i=1,\dots,k$. Thus, the nonlinear eigenvalue problem is converted into a linear eigenvalue problem. If the eigenvalues are not simple, the method works identically but $\mathbf{K}$ will have Jordan normal form. The stucture of multiplicity is preserved. We have enough knowledge about the location of eigenvalues to make the choice of C trivial. All that is left is the discretization of the contour integrals (1.9). Let the contour be described by a smooth mapping $h:[0,2\pi]\to C$ with $h(0)=h(2\pi)$, e.g. the simplest contour, a circle with center μ and radius r, is described by $h(t)=\mu+r\mathrm{e}^{\mathrm{i}t}$. The interval $[0,2\pi]$ is partitioned by the equally spaced nodes $t_j=2\pi j/N, j=1,\dots,N$ with N a chosen discretization parameter.

The approximations

$$\mathbf{A}_0 \approx \frac{1}{N\mathrm{i}} \sum_{j=1}^{N} \mathbf{T}(h(t_j))^{-1}\mathbf{Z}h'(t_j) \tag{1.10a}$$

$$\mathbf{A}_1 \approx \frac{1}{N\mathrm{i}} \sum_{j=1}^{N} h(t_j)\mathbf{T}(h(t_j))^{-1}\mathbf{Z}h'(t_j)\,. \tag{1.10b}$$

are obtained by transforming the integrals onto the partitioned interval and applying the trapezoidal quadrature rule. Beyn shows that the error in the eigenvalues decays exponentially with N [Be12, Corollary 4.8].

The operator $\mathbf{T}$ must be evaluated N times. This is by far the most expensive part of the algorithm. As k is usually much smaller than n, the introduction of the random matrix $\mathbf{Z}$ makes the matrices small enough so that the effort required for linear algebra operations is small. While the solving of linear systems $\mathbf{T}(h(t_j))^{-1}\mathbf{Z}$ is still noticeable, the other operations (SVD, solving the linear eigenvalue problem) are completely negligible.

1.2.3 Parallelization

For optimization with many iterations, the eigenvalue solver is required to be fast. The evaluation of both the matrix $\mathbf{M}(\kappa)$ and the contour integrals in Beyn's method are well suited for parallelization. This covers almost the entire computation. In our implementation for $n = 1152$ and $N = 48$, almost 100% of the time is spent on the evaluation of the contour integrals (1.10), of which the evaluation of $\mathbf{M}(\kappa)$ requires 98.8% and solving the linear systems $\mathbf{T}(\kappa)^{-1}\mathbf{Z}$ requires 1.2%. Everything else is completely negligible. While the exact gains that can be expected depend on the implementation and the system, the principles outlined here are universal. Most relevant for performance are the routines that evaluate the Hankel function and perform quadrature. Our program is implemented in C, using GNU Scientific Library (GSL) [Ga09] as a general framework and for quadrature specifically. For the Hankel function we use the FORTRAN routine provided by Amos [Am86].

Each entry of the $n \times n$ matrix can be evaluated independently without any synchronization or communication. Row cyclical distribution is a simple and effective way to balance the workload (Fig. 1.2). Columns that correspond to an endpoint of a boundary element require the evaluation of two integrals, whereas columns that correspond to a midpoint require only one. Additionally, the time necessary to evaluate a single integral depends strongly on the distance from the singularity of the kernel, i.e. the diagonal. Row distribution removes both these imbalances. For shapes less regular than a disk and collocation points that are not perfectly evenly spaced, the number of collocation points that are close to the singularity varies smoothly between rows. So the rows need to be distributed cyclically. Without any communication or synchronization and with 98.8 % of the computation parallelized, strong scaling is expected for this strategy. The implementation, e.g. using OpenMP, is trivial.

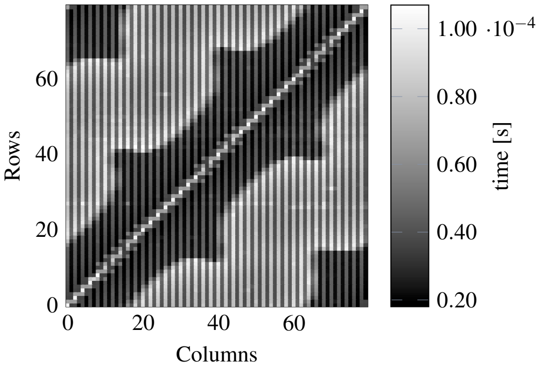


Fig. 1.2 Time required to evaluate each entry of $\mathbf{M}(\kappa)$ for a domain with threefold rotational symmetry. Vertical stripes correspond to alternating end- and midpoints of boundary elements. The singularity of the kernel of the integral equation causes a band around the diagonal whose boundary follows the shape of the domain.

Each of the N summands of the trapezoidal rule that approximates the contour integrals in Beyn's method can also be evaluated independently without any communication except for one sum reduce operation at the end. In regards to workload balancing, the time to evaluate $\mathbf{T}(\kappa)$ in our implementation depends mainly on the sign of the imaginary part of wave number κ (Fig. 1.3). The integrals involving the Hankel function are more expensive for $\mathrm{Im}(\kappa) < 0$. The eigenvalues are real, so a contour that is centered on the real axis is used. Cyclical distribution of summands is generally solid, although not very flexible regarding the number of tasks. Some tasks may end up with fewer summands from one half space than the other. The parallelized part is slightly larger than for the first strategy as basically 100 % of the computation is covered. This improves scaling compared to the first strategy. However, the parameter N is typically not very high (< 50) so the degree of parallelism is severely constrained. It is advisable to employ both strategies, e.g. in a hybrid (mixed shared memory and distributed memory) application. Our implementation achieves a speedup of ~520 with 576 physical/1152 virtual cores on the JURECA cluster [Ju18] which likely can be further improved. The time to evaluate the eigenvalues is reduced to below one second, which allows large scale optimization.

1.3 Optimization of Eigenvalues

Recall that the constrained optimization problem we are trying to solve is

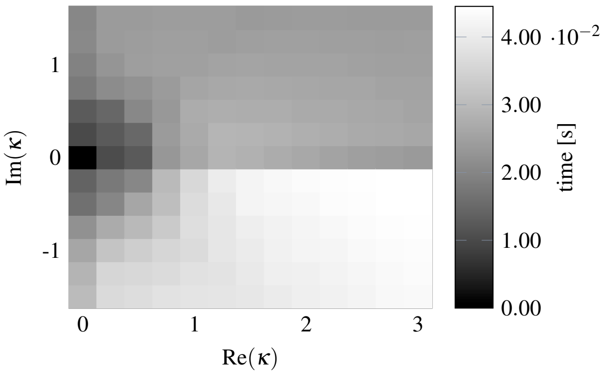


Fig. 1.3 Time required to evaluate $\mathbf{T}(\kappa)$ for different wave numbers κ. The evaluation of integrals involving the Hankel function $H_1^{(1)}$ is significantly more expensive for arguments with negative imaginary part. The singularity at the origin is of no concern as we are not interested in zero eigenvalues.

$$\begin{aligned} &\max_{\Omega}\{\lambda_k(\Omega)\} \\ &\text{s.t. } |\Omega| = 1 \end{aligned} \tag{1.11}$$

for some fixed $k \in \mathbb{N}$ with $|\Omega|$ denoting the area of the domain Ω. By applying the known relations

$$\begin{aligned} \lambda_k(a\Omega) &= a^{-2}\lambda_k(\Omega) \\ |a\Omega| &= a^2\,|\Omega| \end{aligned}$$

with $a\Omega$ denoting the homothety of Ω by the factor a, we can convert (1.11) into the equivalent unconstrained problem

$$\max_{\Omega}\{\lambda_k\,|\Omega|\}\,.$$

In the numeric treatment, it is sufficient to consider connected domains. The spectrum of a disconnected domain $\Omega_1 \cup \Omega_2$ with $\Omega_1 \cap \Omega_2 = \emptyset$ is the ordered union of the spectrums of the component domains Ω_1 and Ω_2. If the maximums $\lambda_i^* = \max_\Omega\{\lambda_i\}$ and corresponding maximizers Ω_i^* (connected or disconnected) for $i = 1,\dots,k-1$ are known, then the maximum of λ_k over disconnected domains is

$$\lambda_k^* = \max_{1\le i\le \frac{n}{2}}\{\lambda_i^* + \lambda_{k-i}^*\} \tag{1.12}$$

and the corresponding disconnected maximizer is

$$\Omega_k^* = \left(\sqrt{\frac{\lambda_j^*}{\lambda_j^* + \lambda_{k-j}^*}}\,\Omega_j^*\right) \cup \left(\sqrt{\frac{\lambda_{k-j}^*}{\lambda_j^* + \lambda_{k-j}^*}}\,\Omega_{k-j}^*\right)$$

where j is the integer that maximizes (1.12). If the maximum over all domains is greater than (1.12), then the maximizer must be connected [PoRo10].

In [Kl19], Kleefeld introduced equipotentials to parameterize the domain. They are described by the implicit function

$$\sum_{i=1}^{m} \frac{1}{\|x-p_i\|^{2\alpha}} = c \tag{1.13}$$

with m fixed base points p_i and free parameters c and α. To match the shapes reported in [AnOu17, Fig. 2], the base points are chosen so they form equilateral triangles of side length $\sqrt{3}/2$, three points on one triangle to maximize λ_3 and four points on two triangles to maximize λ_4 (see first row of Fig. 1.4). Points on the boundary required by the boundary element method are generated by transforming the equation into polar coordinates and using a root finding algorithm to compute the radiuses r_i for evenly spaced angles ϕ_i. The points (r_i, ϕ_i) are then transformed back into Cartesian coordinates. The area of the domain is computed to high accuracy by approximating the domain as a polygon with $100 \cdot n$ sides. With this method, Kleefeld improved on the values found by Antunes and Oudet with the maximum of λ_3 to 32.9018 over 32.90 and the maximum of λ_4 to 43.8694 over 43.86.

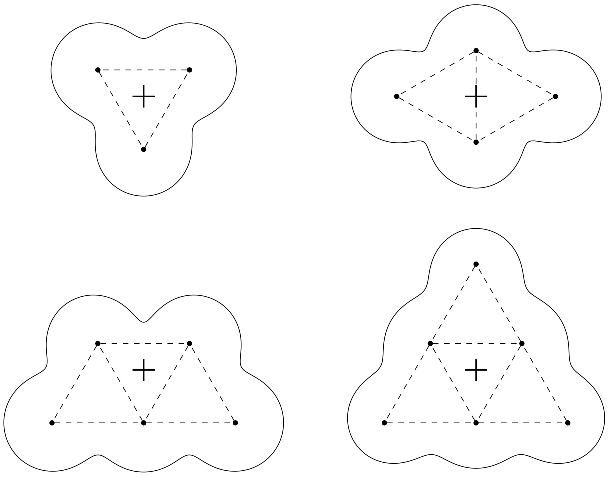


Fig. 1.4 Arrangement of base points for equipotentials to maximize λ_3, λ_4, λ_5, and λ_6. The points form a regular triangular grid. A cross marks the origin and rotation center.

With the improvements for the method of computation presented above, we can now try the scheme on the higher eigenvalues. Unless otherwise noted, we have used discretization parameters $n = 1152$ and $N = 48$. Convergence experiments suggest this is generally enough for six significant digits in the eigenvalues. To be safe, we check the results of optimization with finer discretization. As the gradient of the objective function is not trivially computable, we use a routine provided by GSL that implements the Nelder-Mead simplex method. To avoid local maximums, we try different starting values. For two parameters, it is possible to exhaustively probe the parameter space to find good starting values. The extended, higher dimensional parameter space that is presented later is randomly probed instead. Simplex algorithms are known to terminate prematurely even without the presence of local minimums. So after the optimization routine terminates, it is restarted with the step size reset to its initial value. If the previous result is indeed the maximum (or close enough), the restarted optimization routine quickly terminates again. Eigenvalues are truncated to six significant digits. Shape parameters are given with more significant digits so that the results can be reliably reproduced. The precise error propagation is unknown.

We use the results of Antunes and Oudet [AnOu17] as references for comparison. But there is some uncertainty regarding the precision of those values. The first eigenvalue λ_1 has been proven to be maximized by a disk. The spectrum for the disk can be stated analytically. It is composed of values $\pi j'^2_{pq}$ where j'_{pq} is the q-th positive zero of J'_p, the first derivative of the Bessel function of the first kind of order p. These values can be computed very accurately with root finding algorithms. The first eigenvalue is approximately 10.649866. The second eigenvalue λ_2 is maximized by the union of two disks of the same size. Following the rules for spectrum of disconnected shapes outlined above, the maximum is precisely $2\lambda_1$ or approximately 21.299733. However, the values given in [AnOu17] are 10.66 for λ_1 and 21.28 for λ_2. This discrepancy is not discussed in their paper. While a value lower than the analytical value could be simply be caused by incomplete optimization, a higher value calls into question the precision of the eigenvalue solver. This has direct implications for λ_7 as well, for which the maximizer found by Antunes and Oudet is a disjoint union of the maximizers of λ_1 and λ_6. The maximum value for λ_7 therefore should be the sum of maximums for λ_1 and λ_6. It appears like the authors used their own inaccurate value for λ_1, so λ_7 is inaccurate as well. Of course all their results could be affected by an inaccurate solver, so any comparison can only be tentative.

The base points for the maximizers of $\lambda_k, k = 5, 6, 8, 9, 10$ are k points on a regular equilaterial triangular grid as they were for $k = 3, 4$ (see Fig. 1.4). For λ_7, Antunes and Oudet did not find a connected shape maximizer. The maximizer can then be constructed from the maximizers for $k < 7$, so it is improved automatically with those. The positive results from [Kl19] have been confirmed with 32.9018, 32.9018, 32.9018 for λ_3 and 43.8693, 43.8693, 43.8693 for λ_4. The parameters differ slightly from [Kl19] with $c = 1.687730810, \alpha = 2.019822714$ and $c = 2.084610015, \alpha = 2.541256146$, respectively. This might be explained by the finer discretiztion used. Unfortunately, equipotentials work less and less well for higher k and less symmetric shapes. For λ_5, which in [AnOu17] has multiplicity three, we have 54.5401, 54.5401, 56.0889 for $c = 2.380671137$ and $\alpha = 3.914738607$. This is significantly lower than the reference value 55.17 and the multiplicity is not reproduced. For λ_6 (multiplicity four), where the suspected shape is more regular, we get 67.0440, 67.0440, 67.0440, 67.0440 with $c = 2.849410261, \alpha = 0.660868556$, which is a bit closer to the reference value of 67.33 but still some distance away. For λ_{10} (multiplicity five), we even get 109.988 109.988 109.988 118.955 118.955 instead of the previous value 114.16 with $c = 1.567009307, \alpha = 5.196376634$. The eigenvalues for $k = 7,8,9$ that have been skipped have not been tried as there was no reason to believe they would fare better.

The results strongly suggests that equipotentials as they are in (1.13) are not general representations of the shape maximizers. They have shown potential but need refinement. So far, the base points of the equipotentials have been arranged on a completely regular triangular grid and all base points are weighted equally. As there is no particular reason for this regularity other than visual intuition, breaking it might prove beneficial. So the base points will be allowed to deviate slightly from their regular position. The weight for the base points in the sum of potentials will be allowed to deviate from the regular weight of one. The imagined balls around base points expand as their weight increases. In general, the boundary of the shape moves away from such points. This final equation reads

$$\sum_{i=1}^{m} \frac{1+\hat{\delta}_i}{\|x-(p_i+\hat{\varepsilon}_i)\|^{2\alpha}} = c$$

where $\hat{\varepsilon}_i \in \mathbb{R}^2$ is the irregularity of position and $\hat{\delta}_i \in \mathbb{R}$ is the irregularity of weight of base point i.

Some of the new parameters are fixed so that the shapes are unique and there are no dependencies between parameters. The eigenvalues do not depend on the absolute position of the domain in space, only on the relative position of its base points, so at least one base point should remain fixed during optimization. One weight should remain fixed to avoid a dependency between the weights and parameter c. It is always possible to normalize one weight to a value of one without changing the shape by dividing all weights and c by that weight. The degrees of freedom are further reduced by a requirement that no rotation or reflection symmetries of the regular base points are broken. It must be said that at this point the conjecture that the symmetries are meaningful is unproven. However, based on the results of Antunes and Oudet, the conjecture seems reasonable and it keeps the number of parameters low. So most of the parameters $\hat{\delta}_i$ and $\hat{\varepsilon}_i$ will be fixed to zero. E.g. the shape for λ_3 will have no additional free parameters. One point must be fixed for uniqueness, the others to preserve rotation symmetry. The shape for λ_4 consists of two pairs of points that are images of each other. One pair must be fixed. The other is free but can only move along the symmetry axis. So there is one free coordinate and one free weight. The free, non-zero parameters will be denoted as $\varepsilon_i, i = 1,\ldots,f_\varepsilon$ and $\delta_i, i = 1,\ldots,f_\delta$ with f_δ and f_ε the number of degrees of freedom (Table 1.2). The free parameters can be assigned to base points almost arbitrarily as long as symmetry is conserved. We included our chosen assignment in the tables of results (Table 1.3 and 1.4). Due to rotation symmetry, some irregularities of position ε_i are not axis aligned but point towards the rotation center. For convenience of implementation we avoided irregularities of position that point away from the domain so that a positive first optimization step does not tear the shape apart.

The introduction of ε and δ drastically improves the results over just two parameters. Fig. 1.5 shows the shapes and optimized eigenvalues. Tables 1.3 and 1.4

k	f_ε	f_δ	f
3	0	0	2
4	1	1	4
5	3	2	7
6	1	1	4
8	6	4	12
9	7	4	13
10	1	2	5

Table 1.2 Degree of freedom of positions f_ε (each coordinate is counted separately) and weights f_δ and total degree of freedom f (including c, α) of the equipotential that is used to maximize λ_k after symmetry and uniqueness of the shape and independence of parameters is handled. Degree of freedom is generally greater for shapes with fewer symmetries or more base points.

show the full numerical results including parameters. The maximum for λ_3 remains unchanged as it did not gain any additional free parameters. The value of λ_4 got another small boost to 43.8700. For $k = 4, 6, 8, 10$, we achieved higher maximums than the reference value, sometimes by a small amount like $\lambda_6 = 67.3364$ over 67.33 that could be interpreted as just an increase in precision, sometimes significantly so with $\lambda_8 = 89.8620$ over 89.38 and $\lambda_{10} = 114.187$ over 114.16. Using the improved value for λ_6 and the precise value for λ_1 (see above), λ_7 can also be considered improved even though the reference value is higher. The result for λ_5 is now much closer than it was using just two parameters but is still short of the reference value by about $2 \cdot 10^{-2}$. The difference is small enough that it may still be caused by an inaccurate reference value. For λ_9, which is also too small, the distance to the reference value is almost certainly too big to be explained in that way. Maybe coincidentally, similar trapzeoid shapes are used in both cases (λ_9 and λ_5) where the reference value has not been matched or exceeded.

The muliplicities given in [AnOu17] have not been precisely reproduced in all cases. For example there is a small gap inbetween the values for λ_8. For almost all shapes, the eigenvalues have multiplicities one or two. With changing shapes, some of those groups of identical eigenvalues increase, others decrease. In most cases, both Antunes and Oudet and us have found the optimum where two groups merge, producing multiplicities of three or four. Note that for unions of disjoint shapes, higher multiplicities are expected, as the multiplicities of the component shapes accumulate. Connected shapes where more than two groups merge may not exist and the values for λ_8 may simply be a near miss, where three groups almost merge. On the other hand it is possible that we are simply not able to represent such shapes with the chosen parametrization or that the optimization routine missed them.

1.4 Conclusion

We have presented a way to efficiently and precisely compute interior Neumann eigenvalues for two dimensional domains. Along the way, we highlighted a few

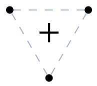
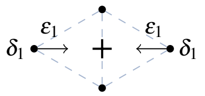
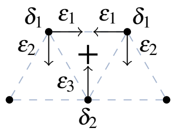
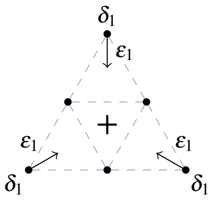
k	Reference	Maximum	Parameters	Base points and irregularities
3	32.90 (3)	32.9018 32.9018 32.9018	$c = 1.687730810$ $\alpha = 2.019822714$	
4	43.86 (3)	43.8700 43.8700 43.8700	$c = 1.942568636$ $\alpha = 2.751523202$ $\varepsilon_1 = -1.314531646 \cdot 10^{-2}$ $\delta_1 = -4.623467053 \cdot 10^{-2}$	 ε_1 ε_1 δ_1 δ_1
5	55.17 (3)	55.1498 55.1498 55.1498	$c = 1.548694899$ $\alpha = 2.231247849$ $\varepsilon_1 = -8.845230330 \cdot 10^{-2}$ $\varepsilon_2 = -4.509337199 \cdot 10^{-2}$ $\varepsilon_3 = -4.354727490 \cdot 10^{-2}$ $\delta_1 = -1.979312992 \cdot 10^{-1}$ $\delta_2 = -1.671890335 \cdot 10^{-1}$	 δ_1 ε_1 ε_1 δ_1 ε_2 ε_2 ε_3 δ_2
6	67.33 (4)	67.3364 67.3364 67.3364 67.3364	$c = 2.027170345$ $\alpha = 1.706097040$ $\varepsilon_1 = 1.577407017 \cdot 10^{-1}$ $\delta_1 = 6.001214705 \cdot 10^{-3}$	 δ_1 ε_1 ε_1 ε_1 δ_1 δ_1
7	77.99 (6)	77.9862 77.9862 77.9862 77.9862 77.9862 77.9862	/	/

Table 1.3 Optimization results for interior Neumann eigenvalues $\lambda_k, k = 3, \dots, 7$ using extended equipotentials. The second column gives the reference value of [AnOu17] with multiplicity in parentheses. The third column contains the maximal eigenvalue that was found by us and as many of the following eigenvalues as the multiplicity requires. The third column contains the equipotential parameters of the shape maximizer. The figures in the fourth column show the base points of the equipotential and the assignment of free irregularity parameters ε and δ. The shape maximizer for λ_7 in [AnOu17] is a disconnected shape that is a union of the shape for λ_6 with a disk and we did not run numerical optimization on it. The values are the sum of the analytical maximum for λ_1 and our new maximum for λ_6.

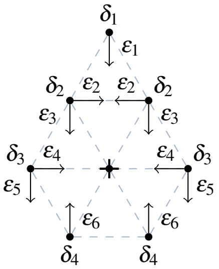
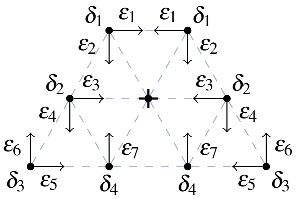
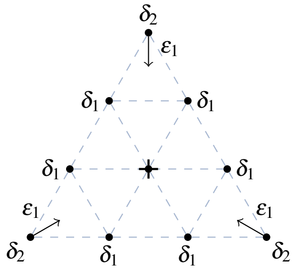
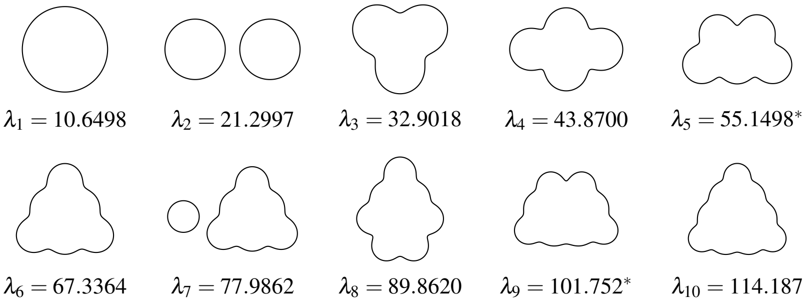
k	Reference	Maximum	Parameters	Base points and irregularities
8	89.38 (4)	89.8620	$c = 1.942964474$	 δ_1 ε_1 δ_2 ε_2 ε_2 δ_2 ε_3 ε_3 δ_3 ε_4 ε_4 δ_3 ε_5 ε_5 ε_6 ε_6 δ_4 δ_4
		89.8620	$\alpha = 1.810828390$	
		89.8620	$\varepsilon_1 = 1.219776174 \cdot 10^{-1}$	
		89.8621	$\varepsilon_2 = -9.776658965 \cdot 10^{-2}$	
			$\varepsilon_3 = -4.652290511 \cdot 10^{-2}$	
			$\varepsilon_4 = -6.000769737 \cdot 10^{-2}$	
			$\varepsilon_5 = -7.584457864 \cdot 10^{-2}$	
			$\varepsilon_6 = -2.247915505 \cdot 10^{-1}$	
			$\delta_1 = 5.396514489 \cdot 10^{-1}$	
			$\delta_2 = 2.082674393 \cdot 10^{-1}$	
			$\delta_3 = 1.353703658 \cdot 10^{-1}$	
			$\delta_4 = 9.643159176 \cdot 10^{-2}$	
9	101.83 (4)	101.752	$c = 1.506287804$	 δ_1 ε_1 ε_1 δ_1 ε_2 ε_2 δ_2 ε_3 ε_3 δ_2 ε_4 ε_4 ε_6 ε_7 ε_7 ε_6 δ_3 ε_5 δ_4 δ_4 ε_5 δ_3
		101.752	$\alpha = 1.928595020$	
		101.752	$\varepsilon_1 = -2.021261311 \cdot 10^{-1}$	
		101.752	$\varepsilon_2 = -1.184995442 \cdot 10^{-1}$	
			$\varepsilon_3 = -1.272843752 \cdot 10^{-1}$	
			$\varepsilon_4 = -1.075608953 \cdot 10^{-1}$	
			$\varepsilon_5 = -3.596435931 \cdot 10^{-2}$	
			$\varepsilon_6 = 7.343083116 \cdot 10^{-3}$	
			$\varepsilon_7 = 8.586462162 \cdot 10^{-2}$	
			$\delta_1 = -3.306712889 \cdot 10^{-2}$	
			$\delta_2 = 5.598216794 \cdot 10^{-1}$	
			$\delta_3 = 7.664620451 \cdot 10^{-3}$	
			$\delta_4 = 1.043695363 \cdot 10^{0}$	
10	114.16 (5)	114.187	$c = 0.899837214$	 δ_2 ε_1 δ_1 δ_1 δ_1 δ_1 ε_1 ε_1 δ_2 δ_1 δ_1 δ_2
		114.187	$\alpha = 2.708323325$	
		114.187	$\varepsilon_1 = -4.458971106 \cdot 10^{-2}$	
		114.187	$\delta_1 = 1.150148658 \cdot 10^{0}$	
		114.187	$\delta_2 = -2.824155602 \cdot 10^{-1}$	

Table 1.4 Optimization results for interior Neumann eigenvalues $\lambda_k, k = 8, 9, 10$ using extended equipotentials. The second column gives the reference value of [AnOu17] with multiplicity in parentheses. The third column contains the maximal eigenvalue that was found by us and as many of the following eigenvalues as the multiplicity requires. The third column contains the equipotential parameters of the shape maximizer. The figures in the fourth column show the base points of the equipotential and the assignment of free irregularity parameters ε and δ.



$\lambda_1 = 10.6498$ $\lambda_2 = 21.2997$ $\lambda_3 = 32.9018$ $\lambda_4 = 43.8700$ $\lambda_5 = 55.1498^*$

$\lambda_6 = 67.3364$ $\lambda_7 = 77.9862$ $\lambda_8 = 89.8620$ $\lambda_9 = 101.752^*$ $\lambda_{10} = 114.187$

Fig. 1.5 Shape maximizers for interior Neumann eigenvalues $\lambda_k, k = 1,\ldots,10$. An asterisk marks values where the reference value by [AnOu17] has not been matched or exceeded. The first two eigenvalues, which are proven theoretically, are included for completeness. For $k = 3,4,5,6,8,9,10$ the maximizers were found by optimizing the (extended) parameters of equipotentials. The maximizer for the seventh eigenvalue is a union of the scaled maximizers for the first and sixth eigenvalues and was not optimized on its own. All shapes have been scaled so they have the same area.

techniques to reduce the time to solution. The strongly scaling parallelization in particular allowed us to use the implemented solvers in the optimization of the eigenvalues with respect to the shape of the domain. We refined the parametrization of the shapes developed by previous research and found improved maximums for most of the first ten eigenvalues.

The very specialized parametrization we presented requires fewer parameters than more general approaches like Fourier series. This makes numerical optimization much cheaper. But the new parametrization is unfortunately far from compact, especially for higher eigenvalues. It is therefore unlikely to be helpful in any theoretical proof of shape maximizers. Numerical optimization was also not equally successful in all cases. The general idea seems promising, but further adaptations will be necessary. Ultimately, an entirely new idea might be called for.

It should prove insightful to study even higher eigenvalues than in this work. Both the method of solution and the parametrization can also be extended without great modifications to three dimensions. Similar results as for the interior Neumann problem also exist for Dirichlet boundary conditions. So research similar to the one in this work is possible in that area. We have concentrated on acoustic scattering. One could also study electromagnetic or elastic problems.

The source code for the program is available at the URL below. We invite researchers to check the program, extend it or use it in their own research.

`https://gitlab.version.fz-juelich.de/abele2/shapeopt`

Acknowledgements The authors gratefully acknowledge the computing time provided on the supercomputer JURECA at Jülich Supercomputing Centre (JSC).

References

[Am86] Amos, D.E.: Algorithm 644: A Portable Package for Bessel Functions of a Complex Argument and Nonnegative Order. *ACM Trans. Math. Softw.*, **12**, 3, 265–273 (1986).

[AnFr12] Antunes, P.R.S. and Freitas, P.: Numerical optimization of low eigenvalues of the Dirichlet and Neumann Laplacians. *J. Optim. Theory Appl.*, **154**, 235–257 (2012).

[AnOu17] Antunes, P.R.S. and Oudet, E.: Numerical results for extremal problem for eigenvalues of the Laplacian. In *Shape optimization and spectral theory*, A. Henrot (ed.), De Gruyter, Warzow/Berlin, 398–412 (2017).

[Be12] Beyn, W.-J.: An integral method for solving nonlinear eigenvalue problems. *Linear Algebra Appl.*, **436**, 3839–3863 (2012).

[CoKr83] Colton, D. and Kress, R.: *Integral Equation Methods in Scattering Theory*. John Wiley & Sons, New York (1983).

[CoKr13] Colton, D. and Kress, R.: *Inverse acoustic and electromagnetic sattering theory*. Springer, New York (2013).

[GiNaPo09] Girouard, A., Nadirashvili, N., and Polterovich, I.: Maximization of the second positive Neumann eigenvalue for planar domains. *J. Differ. Geom.*, **83**, 637–662 (2009).

[Ga09] Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Alken, P., Booth, M., and Rossi, F.: *GNU Scientific Library Reference Manual*, 3rd Edition. Network Theory Ltd., Bristol (2009).

[Ju18] Jülich Supercomputing Centre: JURECA: Modular supercomputer at Jülich Supercomputing Centre. *Journal of large-scale research facilities*, **4**, A132 (2018).

[Kl19] Kleefeld, A.: Shape optimization for interior Neumann and transmission eigenvalues. In *Integral Methods in Science and Engineering*, Christian Constanda and Paul Harris (eds.), Birkhäuser, Cham, 185–196 (2019).

[PoRo10] Poliquin, G. and Roy-Fortin, G.: Wolf-Keller theorem for Neumann eigenvalues. *Ann. Sci. Math. Québec*, **36**, 169–178 (2012).

[Sz54] Szegö, G.: Inequalities for certain eigenvalues of a membrane of given area. *Arch. Ration. Mech. Anal.*, **3**, 343–356 (1954).

[We56] Weinberger, H.F.: An isoperimetric inequality for the N-dimensional free membrane problem. *Arch. Ration. Mech. Anal.*, **5**, 633–636 (1956).