

# Computation of Scattering Matrices and their Derivatives for Waveguides

Greg Roddick

28 November 2020; 3<sup>rd</sup> Revision

## Abstract

This paper describes a new method to calculate the stationary scattering matrix and its derivatives for Euclidean waveguides. This is an adaptation and extension to a procedure developed by Levitin and Strohmaier which was used to compute the stationary scattering matrix on surfaces with hyperbolic cusps [1], but limited to those surfaces. At the time of writing, these procedures are the first and only means to explicitly compute such objects. In this context the challenge we faced was that on Euclidean waveguides, the scattering matrix naturally inhabits a Riemann surface with a countably infinite number of sheets making it more complicated to define and compute. We overcame this by breaking up the waveguide into compact and non-compact components, systematically describing the resolvent for the Neumann Laplace operator on both of them, giving a thorough treatment of the Riemann surface, and then using a “gluing” construction [2] to define the resolvent on the whole surface. From the resolvent, we were able to obtain the scattering matrix. The algorithm we have developed to do this not only computes the scattering matrix itself on such domains, but also arbitrarily high derivatives of it directly. We have applied this, together with the finite element method, to calculate resonances for a selection of domains and will present the results of some numerical calculations in the final section. Whilst this is certainly not the first, nor only method to compute resonances on these domains, ie. Levitin and Marletta have done so previously [3] and Aslanyan, Parnovski and Valiliev before them [4] and other techniques, such as perfectly matched layers may be adapted for this purpose [5]. The method described here has several advantages in terms of speed and accuracy and moreover, provides more information about the scattering phenomena.

*Keywords:* scattering matrix, scattering theory, waveguide, resonances

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Background overview and outline of the paper . . . . .	2
1.2	Waveguides . . . . .	4
<b>2</b>	<b>A description of the domain of our resolvent</b>	<b>6</b>
<b>3</b>	<b>Generalised Eigenfunctions and the Scattering Matrix</b>	<b>8</b>
<b>4</b>	<b>The Neumann to Dirichlet map</b>	<b>10</b>
4.1	Calculating the Neumann to Dirichlet map in the internal domain . . . . .	11
4.2	The Levitin-Marletta method for indirect calculation of the Neumann to Dirichlet map . . . . .	11

---

<b>5</b>	<b>Calculating the scattering matrix</b>	<b>12</b>
5.1	From the Neumann to Dirichlet map to the $S$ matrix . . . . .	12
5.2	Some words on computational cost . . . . .	14
<b>6</b>	<b>Derivatives of the <math>S</math> matrix</b>	<b>15</b>
6.1	A Neumann to Dirichlet map for the system on the external domain . . . . .	15
6.2	A Neumann to Dirichlet map for the system on the internal domain . . . . .	17
6.3	Extracting $S_J^{(n)}(\lambda)$ . . . . .	19
<b>7</b>	<b>Embedded eigenvalues and resonances</b>	<b>20</b>
7.1	Embedded eigenvalues . . . . .	20
7.2	Complex resonances . . . . .	20
<b>8</b>	<b>Conclusion and presentation of results</b>	<b>21</b>
8.1	Cylinders with a circular obstacle . . . . .	22
8.2	Some notes on mesh refinement . . . . .	23
8.3	Some notes on the number of eigenvalues . . . . .	24
8.4	Some notes on the “length” of the domains . . . . .	26
8.5	The importance of the number of modes and choice of auxiliary point . . . . .	26
8.6	Observing a resonance transition from one sheet to another . . . . .	26
8.7	Adding “Ends” to a Circle . . . . .	27
8.8	Animated motion of resonances as the domain is continuously altered . . . . .	31
8.9	Time delay and scattering length . . . . .	31
8.10	Limitations and future directions . . . . .	32
<b>9</b>	<b>References</b>	<b>32</b>

## 1 Introduction

### 1.1 Background overview and outline of the paper

Over the course of this paper, a method to calculate the stationary scattering matrix, and its derivatives for Euclidean waveguides with cylindrical ends will be presented, along with the results from some numerical calculations making use of it.

We will proceed as follows: For the remainder of this section, we will give the reader an overview of the background material and introduce the notion of waveguides and scattering matrices. The main body of the paper will introduce and describe each concept necessary for the end result and either provide the reader with proofs or references where appropriate. Section 2 will be devoted to describing the domain of the resolvent function for the Neumann Laplace operator on  $\mathbb{R}_+$ , the set of real numbers with positive values. This will be used in section 3 to define and describe the concept of generalised eigenfunctions and the scattering matrix itself, and show how the latter may be obtained from the former. Section 4 will focus on the Neumann to Dirichlet map, both its definition and calculation, and its use for computing the scattering matrix. We will then go on to explain how to compute the scattering or  $S$  matrix in section 5, culminating in an explicit formula for it in equation (23). The method from section 5 will be extended in section 6 to calculate derivatives. With a fast and efficient method of computing the  $S$  matrix and its derivatives, we apply it and produce numerical results. The background to the numerical computations will be presented in section 7 and tabulated data will be presented in section 8 along with links to some animations.

Waveguides are piece-wise, path connected subsets of  $\mathbb{R}^n$  that can be written as the union

of a compact domain and any number of non-compact, cylindrical ends. The compact and non-compact parts share a common boundary (see Figure 1). Whilst the ends are non-compact and infinite, they have a certain regularity that allows explicit calculation on them, the compact part may have a more complicated geometry which precludes this. The stationary scattering matrix, in this context, describes the outcome of a scattering event; the scattering of a wave packet in the compact domain, originating at infinity. In any such event, a proportion of the incoming wave packet will be transmitted and a proportion reflected; the coefficients of the scattering matrix contain this information.

The ends of the waveguide can be thought of as the Cartesian product of their boundary with the positive, real half-line  $\mathbb{R}_+$ . A notable feature of Euclidean waveguides is that the scattering matrix admits a meromorphic continuation to a certain Riemann surface with a countably infinite number of sheets [6]. We will present the reader with a proof of this. The existence of a meromorphic continuation is a property inherited from the resolvent of the Laplace operator on these domains whose integral kernel is made up a direct sum involving complex square roots. In order to construct this meromorphic continuation, one must first construct a meromorphic continuation of the resolvent. To do this, we will use a well-known “gluing” construction described in detail by Melrose (see [2]) which we have adapted for waveguides. The construction makes use of the meromorphic Fredholm theorem and the fact that the resolvent, for the Neumann Laplace operator on the ends of the waveguide, can be easily computed as an integral kernel. The resolvent can then be used to construct generalised eigenfunctions, a generalisation of the notion of eigenfunctions, and from them the scattering matrix. This is the first time such techniques have been explicitly adapted to prove this and, at the time of writing, they remain the only such method.

The Neumann to Dirichlet map, takes boundary data for solutions to partial differential equations with Neumann boundary conditions and maps it to the corresponding data for solutions to the same equations with Dirichlet boundary conditions. It is a vital component of this algorithm and we make heavy use of the quick and flexible method developed by Levitin and Marletta [3] to compute it. They were able to formulate the Neumann to Dirichlet map in terms of an infinite sum of Dirichlet data of Neumann eigenvalues on the compact, part of the waveguide. The advantage of this method is that the computation of the Dirichlet data of Neumann eigenvalues, the most computationally costly step, typically using the Finite Element Method, has to be performed only once. Though there are other methods to numerically solve partial differential equations, the point stands. Previously, if one wished to compute the Neumann to Dirichlet map, multiple numerical solutions of the underlying pde. would be required each time (See Definition 5 and Section 4.2 which follows it for the contrasting techniques). We will present both methods and we will also show that the scattering matrix, as well as its derivatives, can be obtained from this data directly. Levitin and Strohmaier have already used this technique to obtain the scattering matrix on finite volume, non-compact hyperbolic surfaces [1]. Due to the more complicated nature of the Riemann surface on which the resolvent sits, the problem of determining the scattering matrix for waveguides, which we provide a solution to here, is more complex.

Being in possession of a fast and efficient algorithm to compute the scattering matrix and its derivatives enables the calculation of resonances which we define to be poles of the scattering matrix. We are able to do this using a combination of numerical contour integration and Newton’s method. The time delay and scattering length [7][8] can also be computed from the scattering matrix.

Evans, Levitin and Vassiliev's 1994 paper [9] proved the existence of embedded eigenvalues (eigenvalues for an operator embedded within its continuous spectrum) for the Neumann Laplacian on two dimensional waveguides with an obstacle and/or deformation of the waveguide, so long as the domain has cross-sectional symmetry. This was further generalised to waveguides with cylindrical ends by Davies and Parnovski [10]. Parnovski and Levitin have, amongst others, produced two other papers on this topic [11] [12]. Embedded eigenvalues can be calculated numerically with their method and we have included some examples in the results section. The main focus for our numerical experiments, however, has been on complex resonances.

Levitin and Marletta [3] and Aslanyan, Parnovski and Vassiliev [4], were able to compute complex resonances for a collection of waveguides. In Levitin and Marletta's case, they computed embedded eigenvalues for a domain with cross-sectional symmetry and observed them decaying to complex resonances when a small perturbation of the domain destroyed that symmetry. We have been able to replicate their results for the same domains with our method, making a slight improvement on accuracy. We have also performed some additional numerical calculations of our own, the results of which are presented in section 8. This is certainly not the only method for computing resonances on Waveguides, perfectly matched layers are a feasible alternative [5], though to replicate the results we provide, existing work will need adaptation.

## 1.2 Waveguides

Let us denote the waveguide by  $M$ ; it is embedded in  $n$ -dimensional Euclidean space with  $K$  cylindrical ends.  $M$  can be written as  $M = E \cup X$ , where  $X$  is a compact, piece-wise connected manifold, with a piece-wise smooth Lipschitz boundary. We can write the ends (the non-compact part of  $M$ )  $E$ , whose  $k$  connected components will be denoted as  $E_k$ , as follows:

$$E = \Gamma \times \mathbb{R}_+ = \bigcup_{k=1}^K [\Gamma_k \times \mathbb{R}_+] = \bigcup_{k=1}^K E_k.$$

Where each  $\Gamma_k \subset \mathbb{R}^{n-1}$  is a compact and connected domain, with smooth boundary, and for any  $i \neq j$ ,

$$[\Gamma_i \times \mathbb{R}_+] \cap [\Gamma_j \times \mathbb{R}_+] = \emptyset.$$

Define

$$E_k = \Gamma_k \times \mathbb{R}_+ \quad \text{and} \quad \Gamma = E \cap X = \{0\} \times \Gamma = \bigcup_{k=1}^K [\{0\} \times \Gamma_k].$$

We will call the boundary of the whole waveguide  $M$ ,  $\Sigma$ .

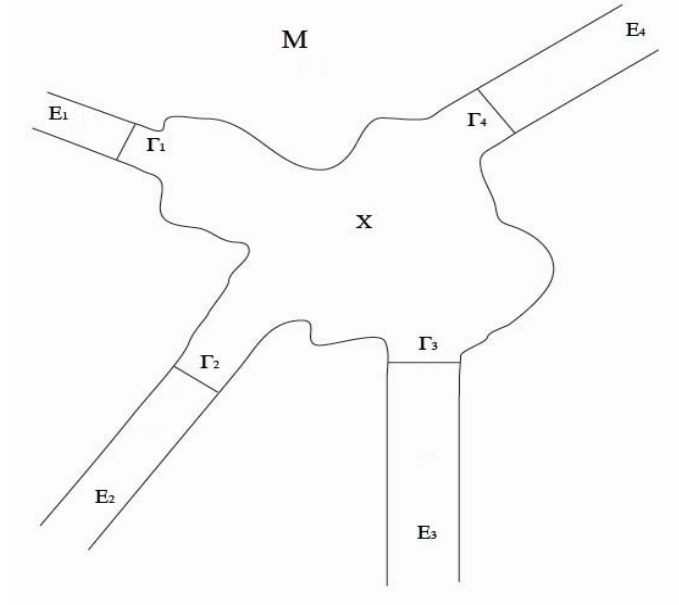


Figure 1: Waveguide

On the ends  $E$  we can use separation of variables to deal with the compact  $\Gamma$  and non-compact  $\mathbb{R}_+$  individually. Reed and Simon have given a rigorous definition of this in their text which we will reproduce. The original, with proof, can be found in here [13, p. 52,II.10]:

**Theorem 1.** *Let  $\langle M_1, \mu_1 \rangle$  and  $\langle M_2, \mu_2 \rangle$  be measure spaces so that  $L^2(M_1, d\mu_1)$  and  $L^2(M_2, d\mu_2)$  are separable.*

*Then there is a unique isomorphism  $\cong$ , from  $L^2(M_1, d\mu_1) \hat{\otimes} L^2(M_2, d\mu_2)$  to  $L^2(M_1 \times M_2, d\mu_1 \otimes d\mu_2)$ , where the  $\hat{\otimes}$ , denotes the completion of the tensor product, so that  $f \otimes g \mapsto fg$ .*

**Corollary 2.** *With the ends of the waveguide  $E$  defined as above, we can now write*

$$L^2(E) \cong L^2(\mathbb{R}_+) \hat{\otimes} L^2(\Gamma).$$

$\Delta_E$ , the Laplace operator on  $L^2(E)$  can now be written in the form

$$\Delta_E = -\frac{\partial^2}{\partial x^2} \otimes 1 - 1 \otimes \Delta_\Gamma, \quad (1)$$

where  $\Delta_\Gamma$  is the Laplace operator acting on  $L^2(\Gamma)$ ,  $1 \otimes \Delta_\Gamma$  is the tensor product of  $\Delta_\Gamma$  with the identity operator on  $L^2(\mathbb{R}_+)$  and  $-\frac{\partial^2}{\partial x^2} \otimes 1$  is the tensor product of the differential operator  $-\frac{\partial^2}{\partial x^2}$  acting on  $L^2(\mathbb{R}_+)$  with the identity operator on  $L^2(\Gamma)$ . The compactness of  $\Gamma$  means that for any  $\lambda$  in the resolvent set of  $\Delta_\Gamma$ ,  $L^2(\Gamma)$  has an orthonormal basis, consisting of eigenfunctions of  $(\Delta_\Gamma - \lambda)^{-1}$ . An application of the discrete Fourier transform yields:

$$L^2(\Gamma) \cong \bigoplus_{j=0}^{\infty} \mathbb{C} = l^2.$$

We conclude that

$$L^2(E) \cong L^2(\mathbb{R}_+) \otimes L^2(\Gamma) \cong L^2(\mathbb{R}_+) \otimes l^2 \cong \bigoplus_{j=0}^{\infty} L^2(\mathbb{R}_+). \quad (2)$$

$(\Delta_{E_j} - \lambda)$  acts on each direct summand in (2) by  $-\frac{\partial^2}{\partial x^2} - \lambda + \mu_j$ , and the  $\mu_j$  are the, not necessarily distinct, Neumann eigenvalues of  $\Delta_\Gamma$ , enumerated in ascending order and repeated with multiplicity taken into account. We shall henceforth refer to each of the summands in Equation 2 as **modes**.  $\Delta_{E_j} - \lambda$  acts on each mode as multiplication by  $\xi^2 - \lambda + \mu_j$  in the Fourier spectral representation (after an application of the Fourier transform).

The Schwartz Kernel Theorem proves existence and uniqueness of an integral kernel for the Neumann resolvent on a half-line and comes about after a simple calculation. See [14, Page 22] for an explicit calculation. If we take the domain be  $\mathbb{R}_+$  with Neumann boundary conditions at 0, the kernel of the resolvent will be of the form

$$\frac{-e^{i|x+y|\sqrt{\lambda}}}{4i\sqrt{\lambda}} + \frac{-e^{i|x-y|\sqrt{\lambda}}}{4i\sqrt{\lambda}}. \quad (3)$$

We should take note of the fact that when  $x \neq y$  the kernel is holomorphic when defined as a function of  $\lambda$  in the branch of the square root where  $\text{Im}(\sqrt{\lambda}) > 0$ . The anti-diagonal doesn't cause problems, because  $x$  and  $y$  are non-negative. Due to the ellipticity of  $(\Delta - \lambda)$ , elliptic boundary regularity can be invoked to show that the resolvent kernel is smooth away from the diagonal also.

The resolvent for  $\Delta_E$ , written as it is in (1), can now be written as

$$R_0(\lambda) = \bigoplus_j r^j(\lambda), \quad (4)$$

where  $r^j(\lambda) = \frac{1}{\xi^2 + \mu_j - \lambda}$  in the spectral representation. This means that equation (3) can be slightly modified to give the kernel on each mode as

$$r^j(\lambda) = \frac{-e^{i|x+y|\sqrt{\lambda-\mu_j}}}{4i\sqrt{\lambda-\mu_j}} + \frac{-e^{i|x-y|\sqrt{\lambda-\mu_j}}}{4i\sqrt{\lambda-\mu_j}}. \quad (5)$$

The existence of square roots, and their branches, adds extra complexity to this resolvent kernel. Rather than talking about  $R_0(\lambda)$  as being defined on  $\mathbb{C}$ , we must instead talk about it being defined on a Riemann surface  $Z$ , on which it is single valued function of  $\lambda$  (see [6]).

## 2 A description of the domain of our resolvent

We will denote the the Neumann eigenvalues of  $\Delta_\Gamma$ , enumerated in ascending order with multiplicity taken into account, by  $\{\mu_j\}$  and the set of such eugenvalues discarding repeated entries by  $\{\eta_j\}$ . This follows a convention of notation established by Christiensen [6, Page5]. The reason for this is that for each  $j \in \mathbb{N}$  corresponds to a branch point for  $\sqrt{\lambda - \eta_j}$ . A complete description of this domain has already appeared in the paper by Guillopé [15], we will, nevertheless, present a brief overview here.

In section 1.2, we saw that the resolvent for  $\Delta_E$  is made up of the direct sum of the  $r^j(\lambda)$ , acting on the direct sum of  $L^2$  spaces; equation (5). The Riemann surface for each individual summand that makes up this resolvent will have a branch at each of the  $\eta_j$ . We define the **physical sheet** of  $Z$  to be the sheet of the surface, which can be identified with  $\mathbb{C} \setminus \mathbb{R}_+$ , for which all the  $\sqrt{\lambda - \eta_j}$ , have positive imaginary part and identify it with  $\mathbb{C} \setminus \mathbb{R}_+$ . The whole surface  $Z$  is made up of a countable number of "sheets" of this nature, each of which represents a choice as to whether each  $\sqrt{\lambda - \eta_j}$  has a strictly positive imaginary

part or not.

The resolvent of  $\Delta_E$  is analytic on the physical sheet for values of  $\lambda$  where it is indeed the resolvent operator and not just a continuation of it. When we wish to extend the resolvent from the physical sheet to other sheets, we must do so along a path. It is necessary to have some kind of a coherent system to categorise such paths.

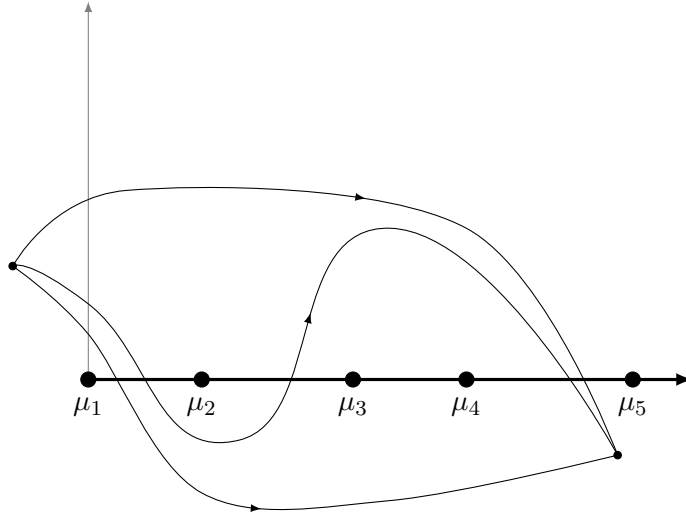


Figure 2: Inequivalent paths that lift to paths with endpoints in sheets  $(1, 2, 3, 4)$ ,  $(1, 3, 4)$ , and  $(1)$ , of  $Z$  respectively.

A path in  $Z$  will remain a path in  $\mathbb{C} \setminus \{\eta_j\}$  under the covering projection. Similarly, a path in  $\mathbb{C} \setminus \{\eta_j\}$ , lifts to a path in  $Z$  if the location (in terms of which sheet) of the pre-image of one of the endpoints is known (or given). We note that paths crossing of the intervals  $(\eta_j, \eta_{j+1})$  on the real line in  $\mathbb{C} \setminus \{\eta_j\}$  correspond with crossings of the boundaries between sheets in  $Z$ . This means that homotopy equivalent paths in  $Z$  originating in the physical sheet, paths in  $\mathbb{C} \setminus \{\eta_j\}$ , and sheets of  $Z$  are all in one-to-one correspondence with each other. Each class of paths in  $\mathbb{C} \setminus \{\eta_j\}$  can be indexed by a finite subset of  $\mathbb{N}$  that we call  $J$  constructed by counting the number of times the path crosses the  $n^{\text{th}}$  interval mod 2. Equivalently  $J$  may be defined as follows:

$$J = \{j \in \mathbb{N} : \text{Im}(\sqrt{\lambda - \mu_j}) \leq 0\}. \quad (6)$$

The monodromy theorem can be used to show that meromorphic continuations along paths are unique and hence any meromorphic continuation from the physical to a non-physical sheet is unique no matter which path is taken. Melrose and others have already demonstrated the existence of analytic continuations for the individual summands  $r_j(\lambda)$  that make up the resolvent kernel for  $\Delta_E$  [2, Page 11]. The “gluing” constructions and the meromorphic Fredholm theorem are used to prove the existence of a meromorphic continuation of the resolvent to  $\Delta$  on the whole waveguide  $M$  (this can be found in more detail in Melrose’s text and Guilope’s paper [15][2]). This has been adapted by us for use on Euclidean waveguides with minimal modifications, the full proof of the existence and uniqueness of a meromorphic continuation of the resolvent on  $M$  can be found in the author’s thesis [14, Page 30].

### 3 Generalised Eigenfunctions and the Scattering Matrix

Let  $\chi$  be a function on the waveguide  $M$  with support on  $E$  equal to 1 outside a compact set, and fix an orthonormal basis of Neumann eigenfunctions of  $\Delta_\Gamma$ , namely  $\{\nu_j(y)\}$ . As described in section 2, we may identify this sheet with  $\mathbb{C} \setminus \min_{j \in J} [\mu_j, \infty)$ . We will generally be working with either the physical sheet of  $Z$ , or the sheet defined by  $J$  which we shall refer to as the non-physical sheet from now on. When identified in this way, every  $\lambda$  in the non-physical sheet of  $Z$  has its counterpart in the physical sheet which is identical as a complex number, but not as on the Riemann surface.

We can now define:

$$\varphi_J(\lambda, x, y) = \sum_{j \in \mathbb{N}} \chi e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) - R(\lambda) \left[ (\Delta - \lambda) \left( \chi e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) \right) \right]. \quad (7)$$

for  $\lambda$  in the physical sheet, and for  $\lambda$  in the non-physical sheet:

$$\begin{aligned} \varphi_J(\lambda, x, y) = & \sum_{j \notin J} \chi e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) - R(\lambda) \left[ (\Delta - \lambda) \left( \chi e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) \right) \right] \\ & + \sum_{j \in J} \chi e^{+i\sqrt{\lambda - \mu_j}x} \nu_j(y) - R(\lambda) \left[ (\Delta - \lambda) \left( \chi e^{+i\sqrt{\lambda - \mu_j}x} \nu_j(y) \right) \right]. \end{aligned} \quad (8)$$

The formulae in Equations (7) and (8) describe a **Generalised Eigenfunction**. A generalised eigenfunction, we define to be a solution of  $(\Delta - \lambda)\varphi_J(\lambda, x, y) = 0$  which is not square integrable, unlike an eigenfunction which is. A reader may observe the  $R(\lambda)(\Delta - \lambda)$  in the right hand side of the equations above and mistakenly conclude that these two operators are mutually inverse and so equations (7) and (8) sum to zero. This is not the case, the reason is that  $R(\lambda)$ , unless  $\lambda$  is on the physical sheet, is a meromorphic continuation of the resolvent and not the resolvent itself, and when  $\lambda$  is in the physical sheet, the resolvent is only an inverse of  $(\Delta - \lambda)$  for  $L^2$ , square integrable functions. These functions have a number of properties:

**Proposition 3.** 1.  $\varphi_J(\lambda, x, y)$  is a meromorphic function of  $\lambda$  for any  $\lambda \in Z$  and holomorphic if  $\lambda$  is in the physical sheet.

2. For  $j \in J$  and  $\lambda$  in the physical sheet of  $Z$ ;

$$R(\lambda) \left[ (\Delta - \lambda) \left( \chi e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) \right) \right] \in L^2(\mathbb{R}_+). \quad (9)$$

3. There exists a unique, meromorphic  $S_{J,j,k}(\lambda)$  such that on  $E$ , and with  $\lambda$  in the physical sheet;

$$\begin{aligned} \varphi_J(\lambda, x, y) = & \sum_{j \in J} \left( e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) + \sum_{k \in J} S_{J,j,k}(\lambda) e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y) \right) \\ & + \sum_{j \notin J} T_j(\lambda) e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y) \end{aligned} \quad (10)$$

and

$$\begin{aligned} \varphi_J(\lambda, x, y) = & \sum_{j \in J} \left( e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y) + \sum_{k \in J} S_{J,j,k}(\lambda) e^{-i\sqrt{\lambda - \mu_k}x} \nu_k(y) \right) \\ & + \sum_{j \notin J} T_j(\lambda) e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y), \end{aligned} \quad (11)$$



for  $\lambda$  in the non-physical sheet. In the case where  $\lambda$  is in the physical sheet, the  $S_{J,j,k}(\lambda)$  are holomorphic.

*Proof.*

1. Follows from the meromorphicity of the various functions whose products make up  $\varphi_J(\lambda, x, y)$ .
2. This is due to the square integrability of the resolvent kernels in the physical sheet.
3. Observe that for any  $j \in J$ , each summand in equation (7) becomes zero when acted upon by  $\Delta - \lambda$ . Using a simple separation of variables on  $E$  we see that any solution to  $(\Delta - \lambda)F(\lambda, x, y) = 0$ , including the one we have, will be of the form,

$$\sum_{k=1}^{\infty} \left( A_{j,k}(\lambda) e^{-i\sqrt{\lambda - \mu_k}x} + B_{j,k}(\lambda) e^{+i\sqrt{\lambda - \mu_k}x} \right) \nu_k(y). \quad (12)$$

We can see that in part 2) of the proposition, the requirement that when we subtract  $\chi e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y)$  the result be square integrable, means that  $A_{j,k} = \delta_{j,k}$ .

In order to reconcile equation (12) with equation (7), we rename the  $B_{j,k}(\lambda)$ ,  $S_{J,j,k}(\lambda)$  for  $k \in J$ . For  $k \notin J$  and see that the remaining terms;

$$B_{j,k}(\lambda) e^{+i\sqrt{\lambda - \mu_k}x} \quad (13)$$

are all square integrable as  $k \notin J \Rightarrow \text{Im}(\sqrt{\lambda - \mu_k}) > 0$ .

When we combine the all such summands to get  $\varphi(\lambda, x, y)$ , as described in equation (7), we get

$$\sum_{j \in J} \left( e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) + \sum_{k \in J} S_{J,j,k}(\lambda) e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y) \right) + \sum_{k \notin J} \sum_{j \in J} B_{j,k}(\lambda) e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y),$$

where we can define, for  $j \notin J$ ,  $T_j(\lambda) = \sum_{k \in J} B_{k,j}(\lambda)$  to finish. The result for the non-physical sheet is due to meromorphic continuation. □

Those used to dealing with the dynamic scattering matrix defined, for example, in Reed-Simon [16], should be aware that the two definitions can be proved to be equivalent. A fundamental property of the generalised eigenfunctions and scattering matrix are their uniqueness in the  $L^2$  norm. This means that  $S_J(\lambda)$  is also uniquely determined by the geometry of  $M$ , the choice of basis for the  $J$  Neumann eigenfunctions and  $J \subset \mathbb{N}$  itself. For any given  $\lambda$  in the non-physical sheet of  $Z$ , given by  $J$  and identified with a suitable subset of the complex plane, we shall denote its counterpart in the physical sheet by  $\lambda^*$ .

**Theorem 4.**  $S_J(\lambda^*) = \overline{S_J^{-1}(\lambda)}$

*Proof.* Observe that

$$\langle \Delta \varphi_j(\lambda, x, y), \varphi_J(\bar{\lambda}, x, y) \rangle - \langle \varphi_J, \Delta \varphi_J \rangle = (\bar{\lambda} - \lambda) \langle \varphi_J, \varphi_J \rangle = 0.$$

The right hand side of the equation above, namely the  $(\bar{\lambda} - \lambda)$  is not an issue, as  $\lambda$  is simply a number in this context and the  $\bar{\lambda}$ , its complex conjugate, will be in the same

sheet of  $Z$  as  $\lambda$ .

Green's second identity can now be invoked to give that

$$\int_{\Gamma} \left( \frac{\partial \varphi_J(\lambda^*, x, y)}{\partial n} \overline{\varphi_J(\bar{\lambda}, x, y)} - \varphi_J(\lambda^*, x, y) \frac{\partial \overline{\varphi_J(\lambda, x, y)}}{\partial n} \right) = 0.$$

In particular, this means that

$$\begin{aligned} & \sum_{j,k \in J} i\sqrt{\lambda^* - \mu_k} (\delta_{k,j} - S_{J,j,k}(\lambda^*)) \sum_{l,m \in J} (\delta_{l,m} + \overline{S_{J,m,l}(\bar{\lambda})}) \int_{\Gamma} \nu_k(y) \nu_l(y) \\ & - \sum_{j,k \in J} (\delta_{k,j} + S_{J,j,k}(\lambda^*)) \sum_{l,m \in J} -i\sqrt{\lambda - \mu_l} (\overline{S_{J,m,l}(\bar{\lambda})} - \delta_{l,m}) \int_{\Gamma} \nu_k(y) \nu_l(y) \\ = & \sum_{j,k \in J} i\sqrt{\lambda^* - \mu_k} (\delta_{k,j} - S_{J,j,k}(\lambda^*)) \sum_{l,m \in J} (\delta_{l,m} + \overline{S_{J,m,l}(\bar{\lambda})}) \int_{\Gamma} \nu_k(y) \nu_l(y) \\ & - \sum_{j,k \in J} (\delta_{k,j} + S_{J,j,k}(\lambda^*)) \sum_{l,m \in J} i\sqrt{\lambda^* - \mu_l} (\overline{S_{J,m,l}(\bar{\lambda})} - \delta_{l,m}) \int_{\Gamma} \nu_k(y) \nu_l(y) \\ = & 0 \end{aligned}$$

From this, we obtain

$$\sum_{k,j,m \in J} \left( (S_{J,j,k}(\lambda^*) - \delta_{k,j})(\delta_{k,m} + \overline{S_{J,m,k}(\bar{\lambda})}) + (\delta_{k,j} + S_{J,j,k}(\lambda^*))(\overline{S_{J,m,k}(\bar{\lambda})} - \delta_{k,m}) \right) = 0$$

This is due to the orthonormality of the  $\{\nu_j\}$  and the way in which we have identified our sheet of  $Z$  with a  $\mathbb{C}$ . Multiplying these out and simplifying shows that, for fixed  $j, m \in J$ , gives

$$\sum_{k \in J} S_{J,j,k}(\lambda^*) \overline{S_{J,m,k}(\bar{\lambda})} = \delta_{j,m}.$$

Since  $S_J(\lambda)$  and  $S_J(\bar{\lambda})$  are, by construction, the same when  $\lambda$  and  $\bar{\lambda}$  are in the same sheet of  $Z$  the result follows.  $\square$

#### 4 The Neumann to Dirichlet map

Given a solution to a pde with Neumann boundary conditions, the Neumann to Dirichlet map takes the Dirichlet boundary data and maps it to the corresponding Dirichlet boundary conditions for this solution. The Neumann to Dirichlet map will be a vital intermediate step between the resolvents, which we have extensively covered in previous sections, and the scattering matrix. On the internal domain, or compact part of the waveguide  $X$ , the Neumann to Dirichlet map can be numerically computed either directly using the Finite Element Method (4.1) or indirectly using a combination of the Finite Element Method and Levitin Marletta's technique (see 4.2). It is an elementary calculation on the "ends"  $E$ . To give a rigorous definition, let  $X$  be a Lipschitz domain in  $\mathbb{R}^n$ ,  $n \geq 2$ . Define the map

$$N : H^{-1/2}(\partial X) \rightarrow H^{1/2}(\partial X) \quad (14)$$

acting on  $g \in H^{-1/2}(\partial X)$  by

$$\mathcal{D}g = \varphi|_{\partial X},$$

where  $\varphi$  is the solution to the Neumann problem, with  $g$  as the boundary derivative. This is the inverse of the Dirichlet to Neumann map

$$\mathcal{D} : H^{1/2}(X) \rightarrow H^{-1/2}(X),$$

whose action on  $f \in H^{1/2}$  is  $\mathcal{D}f = \frac{\partial}{\partial n}(Hf)$ .  $H$  is an extension of  $f$  to a solution of  $(\Delta - \lambda)(Hf) = 0$  on  $X$ . [17]

#### 4.1 Calculating the Neumann to Dirichlet map in the internal domain

It is known that the Neumann eigenfunctions of  $(\Delta - \lambda)$  on  $\Gamma$ , the common boundary between  $X$  and  $E$ , form an orthonormal basis of  $L^2(\Gamma)$ , with Neumann eigenvalues  $\mu_j$ ,  $j \in \mathbb{N}$  [18]. Given a basis of  $L^2(\Gamma)$ , we may compute the Neumann to Dirichlet map in matrix form. This is the first step towards viewing the Neumann to Dirichlet map as a concrete, computable object. With the correspondence between the Neumann to Dirichlet map, the scattering matrix and the resolvent, computing the Neumann to Dirichlet map in this way allows us to realise these other objects in a similar manner.

**Definition 5.** *Neumann to Dirichlet map, associated to  $(\Delta - \lambda)$  on a basis*

*Let us consider an ordered orthonormal basis of  $L^2(\Gamma)$ ,  $\{\nu_j\}_{j=0}^\infty$ , and  $\varphi_k$ , such that*

$$(\Delta - \lambda)\varphi_k = 0, \quad \frac{\partial \varphi}{\partial n}|_\Sigma = 0, \quad \frac{\partial \varphi_k}{\partial n}|_\Gamma = \nu_k. \quad (15)$$

$\Sigma$  here denotes the boundary of  $M$ ,  $\Gamma$  the common boundary by which  $X$  is jointed to  $E$ . Then the  $k, l^{\text{th}}$  element of the Neumann to Dirichlet map, in matrix form, with respect to basis  $\{\nu_k\}$ , will be given by

$$\langle \varphi|_\Gamma, \nu_l \rangle_{L^2(\Gamma)}.$$

Obviously, when doing this calculation practically, we must truncate after a finite number of entries.

#### 4.2 The Levitin-Marletta method for indirect calculation of the Neumann to Dirichlet map

This technique was devised by Levitin and Marletta in [3]. The following formula for the  $k, l^{\text{th}}$  entry of  $N(\lambda)$  acting on a basis of the Neumann subspace of  $L^2(\Gamma)$ ,  $\{\phi_i\}_{i=1}^\infty$  is taken from their paper where it is derived.

$$N_{k,l}(\lambda) = \sum_{m=1}^{\infty} \frac{1}{\lambda - \mu_m} \langle \phi_k, U_m|_\Gamma \rangle_{L^2(\Gamma)} \cdot \langle U_m|_\Gamma, \phi_l \rangle_{L^2(\Gamma)}. \quad (16)$$

$U_m$  and  $\mu_m$  are the eigenfunctions and eigenvalues of the homogeneous Neumann problem on  $X$ , namely the solutions to;

$$(\Delta - \mu_m)U_m = 0, \quad \frac{\partial U_m}{\partial n}|_{\partial X} = 0.$$

Equation (16) gives us a method to compute  $N_{k,l}$ . Truncations of both (17) and (16) can be computed using the finite element method, but (16) is a significant improvement over direct calculation, as once we have obtained the eigenvalues and Fourier coefficients of their associated eigenfunctions, computing  $N(\lambda)$ , for any  $\lambda$  we wish, now only involves matrix multiplication and not numerical solutions of partial differential equations. The latter is considerably slower, more computationally costly. Levitin and Marletta presented

a simple trick or method to further improve the rate of convergence, or accuracy given a fixed number of eigenvalues and eigenfunctions; the derivation of this can also be found in their paper. Where  $\bar{\lambda}$  is fixed and  $N_{k,l}(\bar{\lambda})$  a directly computed Neumann to Dirichlet map we get:

$$N_{k,l}(\lambda) = N_{k,l}(\bar{\lambda}) + \sum_{m=1}^{\infty} \frac{\bar{\lambda} - \lambda}{\mu_m^2 - \lambda\mu_m - \bar{\lambda}\mu_m + \lambda\bar{\lambda}} \langle \nu_k, U_m|_{\Gamma} \rangle_{\Gamma} \cdot \langle U_m|_{\Gamma}, \nu_l \rangle_{\Gamma}. \quad (17)$$

This now gives, quadratic, as opposed to linear convergence. This process can be repeated as many times as one desires to further increase the speed of convergence. In practice, to do this once is sufficient and any repetitions of this process would greatly complicate our extension of this algorithm used for calculating derivatives of the scattering matrix.

## 5 Calculating the scattering matrix

### 5.1 From the Neumann to Dirichlet map to the $S$ matrix

In 4.2 we described a procedure to calculate the Neumann to Dirichlet map from the the Dirichlet data of Neumann eigenfunctions on  $X$ . Here we will describe a procedure to extract the the scattering matrix from such data, culminating in a formula for the  $S$  matrix in (see (23)). As usual, we fix a sheet of  $Z$  and define  $J$ , the indexing set, to be the  $j \in \mathbb{N}$  such that  $\text{Im}(\sqrt{\lambda - \mu_j}) < 0$ . We can proceed as follows:

Fix a basis  $\{\nu_j(y)\}$  for the space of Neumann eigenfunctions on  $\Gamma$ , corresponding to Neumann eigenvalues  $\mu_j$ . Proposition 3, tells us that that the generalised eigenfunction on the cylindrical ends  $E$  of  $M$ , with homogeneous Neumann boundary conditions on the boundary will be of the form:

$$\varphi_J(\lambda, x, y) = \sum_{j \in J} \left( e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) + \sum_{k \in J} S_{J,j,k}(\lambda) e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y) \right) + \sum_{j \in \mathbb{N} \setminus J} T_j(\lambda) e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y),$$

At  $x = 0$ ,  $\varphi_J(\lambda, x, y)$  simplifies to

$$\varphi_J(\lambda, 0, y) = \sum_{j \in J} \left( \nu_j(y) + \sum_{k \in J} S_{J,j,k}(\lambda) \nu_k(y) \right) + \sum_{j \in \mathbb{N} \setminus J} T_j(\lambda) \nu_j(y).$$

At  $x = 0$ , the normal derivative with respect to  $x$  will be

$$\sum_{j \in J} \left( i\sqrt{\lambda - \mu_j} \nu_j(y) - \sum_{k \in J} i\sqrt{\lambda - \mu_k} S_{J,j,k}(\lambda) \nu_k(y) \right) - \sum_{j \in \mathbb{N} \setminus J} i\sqrt{\lambda - \mu_j} T_j(\lambda) \nu_j(y).$$

We ought to now be able to see that, on the external domain or “ends”  $E$ , the Neumann to Dirichlet map will be the inverse matrix of the following:

**Definition 6.**

$$\tilde{N}(\lambda) = \begin{pmatrix} i\sqrt{\lambda - \mu_1} & \cdots & \cdots & \cdots \\ \vdots & i\sqrt{\lambda - \mu_2} & \cdots & \cdots \\ \vdots & \vdots & i\sqrt{\lambda - \mu_3} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

Unfortunately  $\tilde{N}(\lambda)^{-1}$  is not quite yet the Neumann to Dirichlet map for  $(\Delta - \lambda)$  on  $E$ , as such a map only acts on the boundary data of square integrable functions (see equation (14)), thus we must project out the non square integrable modes of any such generalised eigenfunction beforehand.

**Definition 7.** *Define*

$$P : L^2(\Gamma) \longrightarrow L^2(\Gamma),$$

to be the projection whose kernel spanned by the Neumann eigenfunctions of  $\Gamma$  associated to  $\mu_j$  for  $j \in J$ .

Now observe that the internal domain  $X$  and the external domain (or ends)  $E$  share a common boundary  $\Gamma$ . This means that, for any generalised eigenfunction defined on  $M$ , the action of the Neumann to Dirichlet map calculated on  $X$  composed with  $P$  and the Neumann to Dirichlet map calculated on  $E$  should coincide.

Set

$$L = \left( PN - P\tilde{N}^{-1} \right). \quad (18)$$

The matrix  $L$ , will have a null-space of dimension  $|J|$ . Each element of  $\text{Ker}(L)$  can be equated with a one of the  $|J|$  summands of  $\varphi(\lambda, 0, y)$ , as defined in (7) in terms of some, possibly unknown, orthonormal basis of Neumann eigenfunctions of  $\Delta_\Gamma$ .

Applying a singular value decomposition algorithm, or some other procedure to find the kernel of a matrix, i.e.  $QR$ , gives  $|J|$  kernel vectors.

Let  $W$  the null-space of  $L$

$$W = \{w^1, \dots, w^J\}.$$

For each  $\omega_j \in W$  we have a representation of a generalised eigenfunction, evaluated at 0, of the form:

$$\omega^j = \bigoplus_{k \in J} (\delta_{j,k} + S_{j,k}(\lambda)) \bigoplus_{j \in \mathbb{N} \setminus J} T_j(\lambda). \quad (19)$$

We could now, in theory, extract the scattering matrix from this, but before we are able to do such things, and for our scattering matrix to be of any use to us, we need to control the basis of Neumann eigenfunctions. A numerical algorithm for singular value decomposition will not necessarily give us  $S_J(\lambda)$  in terms of the basis we want; the basis of Neumann eigenfunction of  $\Gamma$  that was carefully chosen when we began the calculation of  $N(\lambda)$  in 4.1.

Having found a basis for the null space of (18), we restrict our attention to the elements of these vectors that represent the  $J$  Fourier modes and discard the rest by means of application of the operator  $(1 - P)$ . The image of  $W$  under both  $(1 - P)$  and  $(1 - P)N(\lambda)$  forms a basis in  $\mathbb{R}^J$ .

The linear map  $\tau : \mathbb{R}^J \longrightarrow \mathbb{R}^J$ , defined on the  $(1 - P)w^j$ , by

$$(1 - P)\omega^j \mapsto (1 - P)N\omega^j,$$

can now be thought of as the identity map from the basis  $\{(1 - P)\omega^1, \dots, (1 - P)\omega^J\}$  of  $\mathbb{R}^J$ , to basis  $\{(1 - P)N\omega^1, \dots, (1 - P)N\omega^J\}$  of  $\mathbb{R}^J$ .

$\tau$  must be rewritten in terms of the standard basis, whose elements represent the chosen basis of the Neumann eigenfunctions of  $\Gamma$ . Thus, when acting our chosen basis of  $L^2(\Gamma)$ ,  $\tau$  can be written as,

$$\tau(\lambda) = \{(1 - P)N\omega^1, \dots, (1 - P)N\omega^J\}^{-1} \{(1 - P)\omega^1, \dots, (1 - P)\omega^J\}. \quad (20)$$

Now we note that applying  $N$  to each  $\omega_j$  gives:

$$N(\lambda)\omega^j = \bigoplus_{j \in J} \frac{S_{J,j,k}(\lambda) - \delta_{j,k}}{i\sqrt{\lambda - \mu_j}} \bigoplus_{j \in \mathbb{N} \setminus J} \frac{T_j(\lambda)}{i\sqrt{\lambda - \mu_j}}. \quad (21)$$

Let us define  $D(\lambda, k)$ ,  $k \in \mathbb{N}$  to be the  $|J| \times |J|$  matrix acting on the set of  $J$  modes for which  $\text{Im}(\sqrt{\lambda - \mu_j}) < 0$ . For the calculation of  $S_J(\lambda)$ , we will only be using this with  $k = 0$ .

**Definition 8.**

$$D(\lambda, k) = \frac{\partial^k}{\partial \lambda^k} \begin{pmatrix} i\sqrt{\lambda - \mu_1} & \cdots & \cdots & \cdots & \cdots \\ \vdots & i\sqrt{\lambda - \mu_2} & \cdots & \cdots & \cdots \\ \vdots & \vdots & i\sqrt{\lambda - \mu_3} & \cdots & \cdots \\ \vdots & \vdots & \vdots & \ddots & \cdots \\ \vdots & \vdots & \vdots & \vdots & i\sqrt{\lambda - \mu_{|J|}} \end{pmatrix}$$

One can see now that, as  $(1 - P)$  will project out the second direct sum in (21), the map  $(1 - P)N$  will be a  $J \times J$  matrix defined as follows:

$$\tau(\lambda) = (D(\lambda, 0) + S_J(\lambda).D(\lambda, 0)).(\text{Id} + S_J(\lambda))^{-1}. \quad (22)$$

This means that finally

$$S_J(\lambda) = (\tau(\lambda) - D(\lambda, 0))^{-1}(-D(\lambda, 0) - \tau(\lambda)). \quad (23)$$

## 5.2 Some words on computational cost

When calculating the  $S$  matrix in practice, there are several variables governing computational cost.

- The number of Neumann eigenvalues of the internal domain taken. This is akin to truncating in series in equation (17) at  $m$ .
- The choice of sheet in  $Z$ , namely  $|J|$ : In the simplest case, this is simply the number of ‘ends’ of the domain, though it can be greatly higher.
- The number of modes chosen  $n$ . This is akin to the dimension of the matrix  $N(\lambda)$  where, equation (17) defines it for the  $k, l^{\text{th}}$  element.

The computational cost of the of computing the  $S$  matrix will depend on our choice of  $m$  and  $n$  as they are truncations of an infinite series. In 8.3, where the optimum number of modes and eigenvalues to use for the most accurate result is discussed; it was decided that 20 modes and 2000 eigenvalues was sufficient, though, in all cases  $m \gg n$ . This implies that the computation of  $N(\lambda)$  (see 17), which involves a the multiplication of an  $n \times m$  matrix with an  $m \times m$  diagonal matrix and then multiplication again with an  $m \times n$  matrix, will dominate any calculation off computational cost and thus the asymptotic complexity of the whole operation is that of rectangular matrix multiplication. This was classically treated as  $O(m^2n)$ , but reducing this continues to be an area of active research (for example, see [19]).

---

## 6 Derivatives of the S matrix

This section will show that an extension to the above method can be used to calculate  $S_J^{(n)}(\lambda) = \frac{\partial^n}{\partial \lambda^n} S_J(\lambda)$ . We will guide the reader through this process which culminates with an explicit formula in (37). This is an interesting endeavor in its own right but, our numerical calculations in section 8 make use of the argument principle and Newton's method (see Proposition 9), both of which require the first derivative of  $S$ . It could, in theory, also be used to write the scattering matrix as a power series. We will show that the accuracy of the computation is as accurate as the computation for the original  $S$  matrix and the asymptotic computational cost is no greater. This compares favorably with the, much cruder, alternative of picking a small  $h > 0$  and dividing the difference of  $S(\lambda + h)$  and  $S(\lambda)$  by it. We don't have to worry about the choice of  $h$ , the potential for magnification of errors or the possibility of strange behaviors when  $\lambda$  is a non-trivial fraction of  $h$  away from a resonance.

### 6.1 A Neumann to Dirichlet map for the system on the external domain

Fix a sheet of  $Z$ . If  $\varphi_J(\lambda, x, y)$  is a generalised eigenfunction, then  $(\Delta - \lambda)\varphi_J(\lambda, x, y) = 0$ . Thus, when we differentiate with respect to  $\lambda$ , we get.

$$\frac{\partial}{\partial \lambda}(\Delta - \lambda)\varphi_J(\lambda, x, y) = (\Delta - \lambda)\varphi_J'(\lambda, x, y) - \varphi_J(\lambda, x, y) = 0,$$

For any  $n$  we get

$$\frac{\partial^n}{\partial \lambda^n}(\Delta - \lambda)\varphi_J(\lambda, x, y) = (\Delta - \lambda)\varphi_J^{(n)}(\lambda, x, y) - \varphi_J^{(n-1)}(\lambda, x, y) = 0,$$

where  $\varphi_J^{(n)}(\lambda, x, y)$  denotes  $\frac{\partial^n}{\partial \lambda^n} \varphi_J(\lambda, x, y)$  for brevity. One can simply look for a solution to the resulting system of equations in a similar manner to (15):

$$\begin{aligned} (\Delta - \lambda)\varphi_J(\lambda, x, y) &= 0, & \frac{\partial \varphi_J}{\partial n} \Big|_{\Sigma} &= 0, \\ (\Delta - \lambda)\varphi_J'(\lambda, x, y) - \varphi_J(\lambda, x, y) &= 0, & \frac{\partial \varphi_J'}{\partial n} \Big|_{\Sigma} &= 0, \\ & \vdots & & \vdots \\ (\Delta - \lambda)\varphi_J^{(n)}(\lambda, x, y) - \varphi_J^{(n-1)}(\lambda, x, y) &= 0, & \frac{\partial \varphi_J^{(n)}}{\partial n} \Big|_{\Sigma} &= 0, \end{aligned} \tag{24}$$

On the other hand, since  $\varphi(\lambda, x, y)$  is known on  $E$ , we can recall equation (10) and see

$$\varphi_J(\lambda, x, y) = \sum_{j \in J} \left( e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) + \sum_{k \in J} S_{J,j,k}(\lambda) e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y) \right) + \sum_{j \notin J} T_j(\lambda) e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y),$$

and deduce that

$$\begin{aligned} \varphi'_J(\lambda, x, y) = & \sum_{j \in J} \left( \frac{-ix}{2\sqrt{\lambda - \mu_j}} e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) + \sum_{k \in J} \left[ S'_{J,j,k}(\lambda) - \frac{xS_{J,j,k}(\lambda)}{2i\sqrt{\lambda - \mu_k}} \right] e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y) \right) \\ & + \sum_{j \notin J} \left[ T'_j(\lambda) + \frac{ixT_j(\lambda)}{2\sqrt{\lambda - \mu_j}} \right] e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y), \end{aligned} \quad (25)$$

$$\begin{aligned} \frac{\partial}{\partial x} \varphi'_J(\lambda, x, y) = & \sum_{j \in J} \left[ \frac{-i}{2\sqrt{\lambda - \mu_j}} - x \right] e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) \\ & + \sum_{j,k \in J} \left[ i\sqrt{\lambda - \mu_k} S'_{J,j,k}(\lambda) + \frac{iS_{J,j,k}(\lambda)}{2\sqrt{\lambda - \mu_k}} - xS_{J,j,k}(\lambda) \right] e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y) \\ & + \sum_{j \notin J} \left[ i\sqrt{\lambda - \mu_j} T'_j(\lambda) + \frac{iT_j(\lambda)}{2\sqrt{\lambda - \mu_j}} - xT_j(\lambda) \right] e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y). \end{aligned} \quad (26)$$

At 0 these two generalised functions become:

$$\varphi'_J(\lambda, 0, y) = \sum_{j \in J} \left( \sum_{k \in J} S'_{J,j,k}(\lambda) \nu_k(y) \right) + \sum_{j \in \mathbb{N} \setminus J} T'_j(\lambda) \nu_j(y), \quad (27)$$

$$\begin{aligned} \frac{\partial}{\partial x} \varphi'_J(\lambda, 0, y) = & \sum_{j \in J} \left( \frac{1}{2i\sqrt{\lambda - \mu_j}} \nu_j(y) + \sum_{k \in J} \left[ i\sqrt{\lambda - \mu_k} S'_{J,j,k}(\lambda) - \frac{S_{J,j,k}(\lambda)}{2i\sqrt{\lambda - \mu_k}} \right] \nu_k(y) \right) \\ & + \sum_{j \in \mathbb{N} \setminus J} \left( i\sqrt{\lambda - \mu_j} T'_j(\lambda) - \frac{T_j(\lambda)}{2i\sqrt{\lambda - \mu_j}} \right) \nu_j(y). \end{aligned} \quad (28)$$

We can now see that the external Neumann to Dirichlet map, denoted  $\tilde{N}_1(\lambda)$ , for the system will take the form

$$\tilde{N}_1(\lambda) : l^2 \oplus l^2 \rightarrow l^2 \oplus l^2.$$

Using the  $\tilde{N}(\lambda)$  introduced in Definition 6, the fact that

$$\frac{\partial}{\partial x} \varphi'_J(\lambda, 0, y) = \tilde{N}(\lambda) \varphi'_J(\lambda, 0, y) + \frac{\partial}{\partial \lambda} \tilde{N}(\lambda) \varphi_J(\lambda, 0, y)$$

in addition to the fact that

$$\frac{\partial}{\partial x} \varphi_J(\lambda, 0, y) = \tilde{N}(\lambda) \varphi_J(\lambda, 0, y),$$

we can see that by defining

$$\tilde{N}_{(1)}(\lambda) = \begin{pmatrix} \tilde{N}(\lambda) & 0 \\ \frac{\partial}{\partial \lambda} \tilde{N}(\lambda) & \tilde{N}(\lambda) \end{pmatrix},$$

then, analogous to (18), the Neumann to Dirichlet map on  $E$  for this system will be  $P\tilde{N}_{(1)}(\lambda)^{-1}$ . There is no reason for us to limit ourselves to first derivatives. We should go further now and do the same thing for  $\varphi_J^{(n)}(\lambda, x, y)$ . It is at this point that we take note of the fact that each successive differentiation of  $e^{\pm i\sqrt{\lambda - \mu}x}$  respect to  $\lambda$ , produces a factor of  $x$ . Since we will be focusing on  $\varphi_J$  and  $\frac{\partial}{\partial x} \varphi_J$  at the boundary, where  $x = 0$ , it is unnecessary to differentiate  $e^{\pm i\sqrt{\lambda - \mu}x}$  more than once, and all terms that result from such actions, terms in these summands with a factor of  $x^2$  will simply be denoted them as h.o.t.



We will now introduce some new notation:  $P\tilde{N}_{(n)}(\lambda)^{-1}$  and  $PN_{(n)}(\lambda)$  we define to be the Neumann to Dirichlet maps for the system of  $\varphi_J^{(n)}(\lambda, x, y)$  on the external and internal domains respectively. Differentiating  $\varphi_J^{(n)}(\lambda, x, y)$   $n$  times gives us:

$$\begin{aligned} \varphi_J^{(n)}(\lambda, x, y) &= \sum_{j \in J} \left[ -x \frac{\partial^n \tilde{N}_j}{\partial \lambda^n}(\lambda) + \text{h.o.t} \right] e^{-i\sqrt{\lambda - \mu_j}x} \nu_j(y) \\ &+ \sum_{j, k \in J} \left[ x \sum_{q=1}^n \binom{n}{q} \frac{\partial^q \tilde{N}_k}{\partial \lambda^q}(\lambda) \cdot S_{J,j,k}^{(n-q)}(\lambda) + S_{J,j,k}^{(n)}(\lambda) + \text{h.o.t} \right] e^{i\sqrt{\lambda - \mu_k}x} \nu_k(y) \\ &+ \sum_{j \notin J} \left[ x \sum_{q=1}^n \binom{n}{q} \frac{\partial^q \tilde{N}_j}{\partial \lambda^q}(\lambda) \cdot T_j^{(n-q)}(\lambda) + T_j^{(n)}(\lambda) + \text{h.o.t} \right] e^{i\sqrt{\lambda - \mu_j}x} \nu_j(y). \end{aligned} \quad (29)$$

These expressions evaluated on the boundary become:

$$\varphi_J^{(n)}(\lambda, 0, y) = \sum_{j, k \in J} S_{J,j,k}^{(n)}(\lambda) \nu_k(y) + \sum_{j \notin J} T_j^{(n)}(\lambda) \nu_j(y), \quad (30)$$

$$\begin{aligned} \frac{\partial}{\partial x} \varphi_J^{(n)}(\lambda, 0, y) &= \sum_{j \in J} \left[ -\frac{\partial^n \tilde{N}_j}{\partial \lambda^n}(\lambda) \right] \nu_j(y) \\ &\sum_{j, k \in J} \left[ \sum_{q=0}^n \binom{n}{q} \frac{\partial^q \tilde{N}_k}{\partial \lambda^q}(\lambda) \cdot S_{J,j,k}^{(n-q)}(\lambda) \right] \nu_k(y) \\ &+ \sum_{j \notin J} \left[ \sum_{q=0}^n \binom{n}{q} \frac{\partial^q \tilde{N}_j}{\partial \lambda^q}(\lambda) \cdot T_j^{(n-q)}(\lambda) \right] \nu_j(y). \end{aligned} \quad (31)$$

$\tilde{N}_{(n)}(\lambda)$  will be a block-upper triangular matrix, ( $\tilde{N}_j$ , for  $j \in \mathbb{N}$  has been used above to be the  $j^{\text{th}}$  element of the diagonal matrix  $\tilde{N}$  so the reader should avoid confusing the two).

$$\tilde{N}_{(n)}(\lambda) : \bigoplus_{\{0, \dots, n\}} l^2 \longrightarrow \bigoplus_{\{0, \dots, n\}} l^2,$$

given by

$$\tilde{N}_{(n)}(\lambda) = \begin{pmatrix} \tilde{N}(\lambda) & 0 & \dots & \dots & \dots & 0 \\ \frac{\partial}{\partial \lambda} \tilde{N}(\lambda) & \tilde{N}(\lambda) & 0 & \dots & \dots & 0 \\ \binom{2}{0} \frac{\partial^2}{\partial \lambda^2} \tilde{N}(\lambda) & \binom{2}{1} \frac{\partial}{\partial \lambda} \tilde{N}(\lambda) & \binom{2}{2} \tilde{N}(\lambda) & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & 0 \\ \binom{n}{0} \frac{\partial^n}{\partial \lambda^n} \tilde{N}(\lambda) & \binom{n}{1} \frac{\partial^{n-1}}{\partial \lambda^{n-1}} \tilde{N}(\lambda) & \dots & \dots & \dots & \binom{n}{n} \tilde{N}(\lambda) \end{pmatrix}$$

## 6.2 A Neumann to Dirichlet map for the system on the internal domain

Now we turn our attention towards the internal domain  $X$ .  $N_{(n)}(\lambda)$  can be computed with an extension of Levitin-Matletta's method. Equation (16) describes the method when used to compute  $N(\lambda)$ . Their "trick" to increase the rate of convergence (17) is unaffected, so long as it is performed only once.

The  $\{U_m\}$  remain unchanged from (16) and are used to denote the orthonormal Neumann eigenfunctions and  $\mu_m$ , their corresponding eigenvalues.  $\{\nu_j\}$  will denote orthonormal an basis of  $L^2(\Gamma)$  and finally, we will use  $\{\Phi_k\}$  and  $\{\Phi_k^{(n)}\}$ , to denote solutions to the following system:

$$\begin{aligned} (\Delta - \lambda)\Phi_k(\lambda, x, y) &= 0, & \frac{\partial\Phi_k}{\partial n}|_{\Sigma} &= 0, & \frac{\partial\Phi_k}{\partial n}|_{\Gamma} &= \nu_{k_0} \\ (\Delta - \lambda)\Phi'_k(\lambda, x, y) - \Phi_k(\lambda, x, y) &= 0, & \frac{\partial\Phi'_k}{\partial n}|_{\Sigma} &= 0, & \frac{\partial\Phi'_k}{\partial n}|_{\Gamma} &= \nu_{k_1} \\ \vdots & & \vdots & & \vdots & \\ (\Delta - \lambda)\Phi_k^{(n)}(\lambda, x, y) - \Phi_k^{(n-1)}(\lambda, x, y) &= 0, & \frac{\partial\Phi_k^{(n)}}{\partial n}|_{\Sigma} &= 0, & \frac{\partial\Phi_k^{(n)}}{\partial n}|_{\Gamma} &= \nu_{k_n}, \end{aligned} \quad (32)$$

where  $k$  is the  $n$ -tuple  $\{k_0, k_1, \dots, k_n\}$ .

We begin by fixing  $\lambda$  and will now calculate elements of  $N_{(n)}(\lambda)$  which map to  $\Phi^{(n)}$ , where  $k_n \in \mathbb{N}$  and  $l \in \mathbb{N}^n$ :

$$\begin{aligned} N_{(n),k_n,l} &= \langle N\nu_{k_n}, \nu_l \rangle = \langle \Phi_k^{(n)}|_{\Gamma}, \nu_l \rangle = \langle \Phi_k^{(n)}|_{\Gamma}, \frac{\partial\Phi_l^{(n)}}{\partial n}|_{\Gamma} \rangle \\ &= \langle \nabla\Phi_k^{(n)}, \nabla\Phi_l^{(n)} \rangle + \langle \Delta\Phi_k^{(n)}, \Phi_l^{(n)} \rangle = \langle \nabla\Phi_k^{(n)}, \nabla\Phi_l^{(n)} \rangle + \lambda\langle \Phi_k^{(n)}, \Phi_l^{(n)} \rangle + \langle \Phi_k^{(n-1)}, \Phi_l^{(n)} \rangle. \end{aligned}$$

Since each  $\Phi_k$  can be written as  $\sum_m \Phi_k \langle \Phi_k, U_m \rangle$  (the same is true for  $\Phi_k^{(n)}$ ), and by definition,  $\Phi_k = (\Delta - \lambda)^{-j} \Phi^{(n-j)}$ , it follows that:

$$\begin{aligned} N_{(n),k_n,l} &= \sum_m (\langle \nabla U_m, \nabla U_m \rangle + \lambda) \langle \Phi_k^{(n)}, U_m \rangle \langle U_m, \Phi_l^{(n)} \rangle + \langle \Phi_k^{(n-1)}, \Phi_l^{(n)} \rangle \\ &= \sum_m (\langle \nabla U_m, \nabla U_m \rangle + \lambda) \langle \Phi_k^{(n)}, U_m \rangle \langle U_m, \Phi_l^{(n)} \rangle + \sum_m \langle \Phi_k^{(n-1)}, U_m \rangle \langle U_m, \Phi_l^{(n)} \rangle \\ &= \sum_m (\lambda - \mu_m) \langle \Phi_k^{(n)}, U_m \rangle \langle U_m, \Phi_l^{(n)} \rangle + \sum_m \langle \Phi_k^{(n-1)}, U_m \rangle \langle U_m, \Phi_l^{(n)} \rangle. \end{aligned} \quad (33)$$

Green's second identity, for any  $j$ , means that

$$\begin{aligned} \langle \Phi_j, U_m \rangle &= \sum_m \frac{1}{\lambda - \mu_m} (\langle \Delta\Phi_j, U_m \rangle - \langle \Phi_j, \Delta U_m \rangle) = \sum_m \frac{1}{\lambda - \mu_m} \langle \nu_j, U_m|_{\Gamma} \rangle \\ \langle \Phi_j^{(n)}, U_m \rangle &= \sum_m \frac{1}{\lambda - \mu_m} (\langle \Delta\Phi_j^{(n)}, U_m \rangle - \langle \Phi_j^{(n)}, \Delta U_m \rangle - \langle \Phi_j^{(n-1)}, U_m \rangle) \\ &= \sum_m \frac{1}{\lambda - \mu_m} (\langle \nu_{j_n}, U_m|_{\Gamma} \rangle - \langle \Phi_j^{(n-1)}, U_m \rangle). \end{aligned} \quad (34)$$

So then  $N_{(n),k_n,l}$  becomes

$$\sum_m (\langle \nu_k, U_m|_{\Gamma} \rangle - \langle \Phi_j^{(n-1)}, U_m \rangle) \langle U_m, \Phi_l^{(n)} \rangle,$$

and inductively, we see that

$$N_{(n),k_n,l} = \sum_m \langle \nu_{k_n}, U_m|_{\Gamma} \rangle \langle U_m, \Phi_l^{(n)} \rangle = \sum_m \sum_{p=1}^n \frac{(-1)^{p-1}}{(\lambda - \mu_m)^p} \langle \nu_{k_n}, U_m|_{\Gamma} \rangle \langle U_m|_{\Gamma}, \nu_{l_p} \rangle.$$

Thus the  $n^{\text{th}}$  block-row for  $N_{(n)}(\lambda)$  is made up of the direct sum of maps defined component-wise by

$$\eta_{(p)}(\lambda) = \sum_m \frac{(-1)^{p-1}}{(\lambda - \mu_m)^p} \langle \nu_k, U_m | \Gamma \rangle \langle U_m | \Gamma, \nu_l \rangle, \quad (35)$$

where  $p$  runs through  $1, \dots, n$ .

Now, for the system,  $N_n(\lambda)$  is a block upper-triangular matrix, acting on  $\Phi \oplus \dots \oplus \Phi^{(n)}$ , of the same form as  $\tilde{N}_{(n)}(\lambda)$ , given by

$$N_{(n)}(\lambda) = \begin{pmatrix} \eta_{(1)}(\lambda) & 0 & \dots & \dots & \dots & 0 \\ \eta_{(2)}(\lambda) & \eta_{(1)}(\lambda) & 0 & \dots & \dots & 0 \\ \eta_{(3)}(\lambda) & \eta_{(2)}(\lambda) & \eta_{(1)}(\lambda) & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & 0 \\ \eta_{(n)}(\lambda) & \eta_{(n-1)}(\lambda) & \dots & \dots & \dots & \eta_{(1)}(\lambda) \end{pmatrix} \quad (36)$$

We can assert here that the computational cost of such an approach is no greater than that of crudely approximating the derivatives from first principles. Each of the  $\eta_p$  in (35), that goes into making up the  $N_{(n)}$  matrix (36), comes about in a very similar way to (17) and has the same computational cost, namely that of  $n \times m$  by  $m \times n$  matrix multiplication. Even though the number of  $\eta_p$  required to make up the block-diagonal  $N_{(n)}$  rises arithmetically with the order of the derivatives, the number of unique  $\eta_p$  required for the  $n^{\text{th}}$  derivative is only  $n + 1$ . The reader will see in the following subsection, that the rest of the calculation consists of a series of additions, multiplications, inversions and one Singular Value Decomposition of (in practice) much smaller matrices. There are no issues relating to the speed of convergence either as the order of the denominators in (35) will always be greater or equal to 2.

### 6.3 Extracting $S_J^{(n)}(\lambda)$

The coefficients of  $\varphi_J^{(n)}(\lambda, 0, y)$  can now be computed by finding a basis for the null space of

$$L_n = (PN_{(n)} - P\tilde{N}_{(n)}^{-1}).$$

We will denote such a basis as  $\omega_1^{(n)}, \dots, \omega_J^{(n)}$ . Finally, using the same argument made when calculating  $S(\lambda)$ ,

$$\tau_n(\lambda) = \{(1 - P)\omega_1^{(n)}, \dots, (1 - P)\omega_J^{(n)}\} \{(1 - P)N\omega_1^{(n)}, \dots, (1 - P)N\omega_J^{(n)}\}^{-1},$$

thus

$$\begin{aligned} \tau_n(\lambda) &= \left( D(\lambda, n) - \sum_{q=0}^n \binom{n}{q} D(\lambda, q) \cdot S^{(n-q)}(\lambda) \right) S^{(n)}(\lambda)^{-1} \\ &= \left( D(\lambda, n) - \sum_{q=1}^n \binom{n}{q} D(\lambda, q) \cdot S^{(n-q)}(\lambda) - D(\lambda, 0) S^{(n)}(\lambda) \right) S^{(n)}(\lambda)^{-1}. \end{aligned}$$

All of the  $S_J^{(n-q)}(\lambda)$  are known, having been previously calculated. Finally we can say that

$$S_J^{(n)}(\lambda) = \left( \text{Id} - D(\lambda, 0)\tau_n(\lambda) \right)^{-1} \cdot \left( D(\lambda, n) \cdot \tau_n(\lambda) - \sum_{q=1}^n \binom{n}{q} D(\lambda, q) \cdot \tau_n(\lambda) \cdot S^{(n-q)}(\lambda) \right). \quad (37)$$

---

## 7 Embedded eigenvalues and resonances

This section will give an overview of embedded eigenvalues, complex resonances and describe how the scattering matrix can be used to compute the latter. It should be noted that computational methods are becoming more important generally in science, with new techniques under continuous development [20], [21], [22].

### 7.1 Embedded eigenvalues

An embedded eigenvalue of an operator is an isolated point within an operator's continuous spectrum which is actually an eigenvalue of finite multiplicity. The paper by Evans, Levitin and Vassiliev [9] proved the existence of embedded eigenvalues for the Neumann Laplacian on two dimensional waveguides with an obstacle, and/or deformation of the waveguide so long as the domain has cross-sectional symmetry.

This was further generalised to waveguides with cylindrical ends by Davies and Parnowski [10]. Parnowski and Levitin have, amongst others, produced two other papers on this topic [11] [12]. More recently, in Levitin and Strohmaier's paper [1, 6.3] they describe a technique to calculate embedded eigenvalues numerically by looking for real values of  $\lambda$  that make the sub-matrix of  $(N(\lambda) - \tilde{N}^{-1}(\lambda))$ , obtained by omitting the rows and columns representing non square integrable modes, singular (see (17) and definition 6 for definitions of  $N$  and  $\tilde{N}$ ). In this case, non square integrable modes will be the  $\mu < \lambda$ , as in such a situation  $e^{i\sqrt{\lambda-\mu}}$  will certainly not be in  $L^2$  for real  $\lambda$ . An eigenvector corresponding to a zero-eigenvalue will necessarily represent the boundary data of an eigenfunction and not a generalised eigenfunction as the Neumann to Dirichlet maps on the internal and external domains co-inside when acting on that vector and the modes have been chosen deliberately to be in  $L^2$ . The problem with such an approach is that it is hard to distinguish an embedded eigenvalue from a resonance very close to the real line [1, 6.3]. Like Levitin and Strohmaier, we have calculated a small number of embedded eigenvalues, but the main focus of the numerical calculations in section 8 has been to use the scattering matrix to locate resonances. The reason for this is that the properties of the scattering matrix lend themselves to use in algorithms that are able to "sweep" an area in search of them and this process can be automated to "zoom in" on any resonances found (see Proposition 9 and the preceding and following discussion).

### 7.2 Complex resonances

The scattering matrix, and its derivatives, can be used to calculate complex resonances on any sheet of  $Z$ . Previous work on this topic has focused on either providing asymptotic bounds for the number of resonances ([6], [23], [24], [25], [26]), or their calculation ([3], [27], [4], [28], [29], [30], [31], [1]). For the remainder of this paper, we will focus on the latter. As was alluded to previously, reason we have gone down this route is because the numerical technique we have just constructed is absolutely conducive to this. Levitin and Strohmaier have already used the scattering matrix to locate resonances on surfaces with hyperbolic cusps [1]; as we now are in possession of the scattering matrix for Euclidean waveguides, we will do the same for them.

We define a resonance here to be a pole of the scattering matrix. The relationship between the resolvent, the Neumann to Dirichlet map and the scattering matrix means that poles of the resolvent coincide with zeros of the determinant of the the inverse of the scattering matrix, and their multiplicities will be the same.

Theorem 4 can be used to show that every pole of  $S_J(\lambda)$ , on the sheet  $J$  of  $Z$ , coincides

---

with a zero of  $S_J(\lambda^*)$  in the physical sheet and vice versa. We have used  $\lambda^*$  to denote the canonical projection of  $\lambda$  to the physical sheet; when both  $\lambda$  and  $\lambda^*$  are identified with a subset of the complex plane, they will be in the same location. Since the resolvent and scattering matrix are holomorphic in the physical sheet, it can't have poles there, and we will have no zeros in a non-physical sheet of  $Z$ . This means we can now make use of the argument principle to locate resonances, and locating resonances in a non-physical sheet of  $Z$  has been reduced to locating zeros in the physical sheet. The argument principle together with the Jacobi formula gives:

**Proposition 9.** *Let  $C$  be a contour in a non-physical sheet of  $Z$ ,  $J$  with winding number one and let  $\#$  be the counting function  $\# : C \mapsto \mathbb{N}$  which counts the number of poles enclosed by  $C$ . Then*

$$\#(C) = \frac{1}{2\pi i} \oint_C \text{Tr}(S_J(\lambda)^{-1} \cdot S'_J(\lambda)).$$

This approach was featured in the paper by Davies and Aslanyan, but not applied to the scattering matrix [27]. If a contour can be found that contains one or more zeros, we can subdivide then integrate over the subdivisions and repeat the process until a small enough contour has been found containing a single resonance. Simpson's method for numerical integration will suffice for this. Once we are sufficiently close to any resonances we may find, Newton's method can then be used to obtain their location to a desired level of accuracy. We can then multiply the scattering matrix  $S_J(\lambda)$  by  $(\lambda_0 - \lambda)^{-1}$ , where  $\lambda_0$  is the location of the zero, then apply Newton's method again, repeating if necessary to find its order. This process can be automated to such an extent that one need only feed parameters for a search area into a script, wait, and then be rewarded with a list of locations for any resonances found.

## 8 Conclusion and presentation of results

We will conclude by presenting the reader with the results of some numerical calculations. In 8.1 we will replicate and improve somewhat the accuracy of the calculations produced by Aslanyan, Parnowski and Vassiliev [4], and Levitin and Marletta [3]. In 8.2 - 8.5 we will discuss the finite element method mesh refinement, number of eigenvalues and modes needed to obtain a desired accuracy and discuss where to "cut" the internal domain from the ends. All resonances up to this point will have been calculated for the sheet  $J = \{1\}$  (see 6), but in 8.6, we will track the location of a resonance and observe it cross from the sheet  $J = \{1\}$  to  $J = \{2\}$  as the domain is changed. In 8.7 we will present some tabulated results and graphs describing the motion of resonances in an area as the domain is continuously changed. 8.9 gives a brief overview of time delay and scattering length and we will finalise the paper by linking to some animations of plots of various resonances as the domain is continuously changed before suggesting additional areas of study.

All of the finite element method data used to obtain these results has been produced in FreeFem++ [32]. FreeFem++ was chosen at the time because it is free and open source, with no restrictions on installation and use. It has proved itself to be reliable and with good documentation. FreeFem++ contains methods to produce triangulations (or tetrahedralisation if working in 3-D) for domains given to it by the user in the form of parameterised curves, implements both the finite element method for solving PDEs and the implicitly restarted Arnoldi method for finding eigenvalues and eigenvectors by making use of the ARPACK subroutine [33] [34]. Though the results have not been presented here, it was able to perform similar calculations in three dimensions for ends with rectangular cross sections and circular cross-sections and could theoretically work for domains with

arbitrarily drawn cross sections and domains with potentials. If the reader wishes to compare finite element method software, the site <http://feacompare.com> has an exhaustive list that, at the time of writing, numbered 78. Boundary element method techniques could also be used in place of the finite element method here. In certain circumstances they may have several advantages [21]. At this stage, we have not explored this option any further because, for each domain, calculation of numerical solutions to pdes is only required once, and for the two dimensional domains we have presented results for here, the compute time has been “tolerable”, around five minutes for each domain. This will certainly change if higher dimensional domains are used with more complicated geometries. The method of particular solutions [35] is another computational technique that has been generating significant interest of late, and where software to implement it is starting to become easily available (see <https://github.com/ahbarnett/mpspack>).

The code used to extract the scattering matrix and its derivatives, from the output of the finite element method (then later search for resonances) was written in Wolfram Mathematica and has been subsequently rewritten in Sage. It is available at <https://bitbucket.org/greg5783/smcalc2/>. It was important to use a software package that was able to perform linear algebraic operations on vectors and matrices with complex values. MATLAB could also be used for this. Sage was found to have a slight speed advantage over Mathematica provided care was taken to write efficient code. Sage is more complicated to use, for example when parallelising operations (in mathematica, one only has to decorate the entire script with `parallel()` and parralelisation of any and all ensuing operations is implemented automatically in an optimised way.

### 8.1 Cylinders with a circular obstacle

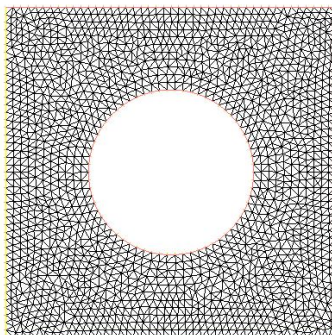


Figure 3: One of our triangulated interior domains produced by FreeFEM++ [32]. The waveguide is composed of this interior domain, with the two ends the same width as the interior domain joined on the left and right sides.

The first thing we wanted to do once we had a working algorithm was replicate the results of our predecessors and try to improve their accuracy. Amongst other things, Levitin and Marletta looked at an infinite rectangle of width 2, with a single circular obstruction (See Figure 3). The radius of this obstruction,  $R$ , is varied along with  $\delta$ , the position of its centre (vertical displacement) relative to the centre line of the rectangle. When the vertical displacement is 0, there exist embedded eigenvalues. The embedded eigenvalues decay to a resonances when the vertical displacement  $\delta$  becomes non-zero [3]. With the parameterisation of  $\lambda \mapsto \lambda^2$ , they presented a number of values for these resonances. We have performed our calculation to the highest accuracy our method and available hardware afforded us, namely 2000 eigenvalues, 20 modes (see 8.3) and a mesh refinement of 80 (see 8.2), and have displayed ours and our predecessors’ results in Table 4 below.



Any additional eigenvalues, modes and mesh refinements after these values did not affect the results. We have been able to offer a slight improvement on the number of decimal places.

$R$	$\delta$	Our calculation	Levitin-Marletta	Aslanyan et al
0.3	$\delta = 0$	1.50497	1.50486	1.5048
0.3	$\delta = 0.1$	1.50783 + 0.0001205 <i>i</i>	1.5078 + $10^{-4}i$	1.5102 + $\times 10^{-4}i$
0.3	$\delta = 0.2$	1.51651 + 0.0004740 <i>i</i>	1.5165 + $5 \times 10^{-4}i$	1.5188 = $5 \times 10^{-4}i$
0.5	$\delta = 0$	1.39138	1.39134	1.3913
0.5	$\delta = 0.1$	1.39785 + 0.0009255 <i>i</i>	1.3979 + $9 \times 10^{-4}i$	1.3998 + $9 \times 10^{-4}i$
0.5	$\delta = 0.2$	1.41779 + 0.0039101 <i>i</i>	1.4178 + $3.90 \times 10^{-3}i$	1.4196 + $3.93 \times 10^{-3}i$

Table 1: This table displays our own results alongside those of Levit-Marletta and Aslanyan, Parnovski and Vassiliev. Greyed out figures portray a lower level of confidence in their accuracy on our part.

## 8.2 Some notes on mesh refinement

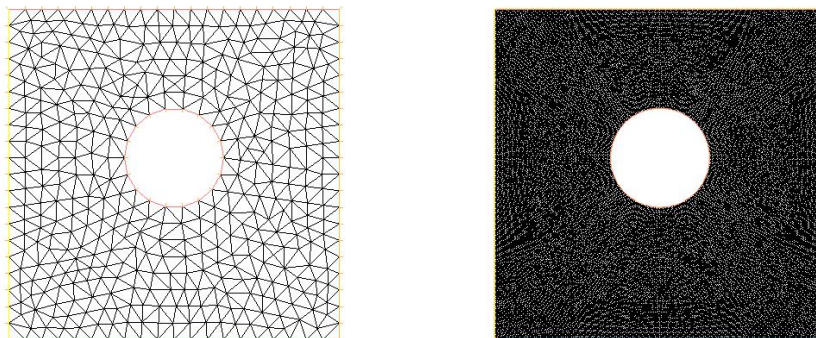


Figure 4: Some internal domains for  $R = 0.3$ ,  $\delta = 0.1$  with mesh refinement of 10 and 80 respectively.

We have looked at the effects of mesh refinements on the accuracy of the results obtained. Whilst it should be obvious that the more refined the mesh becomes, the more accurate the result, we have tabulated the results of some experiments to demonstrate just to what extent. The standard method for producing a triangulated domain in FreeFem++ is to draw the outline as a union of parameterised curves, then use the program's own triangulation algorithm after specifying the number of points on each such curve. Our scale of mesh refinement was taken to be the number of such points per unit length on the boundary. Using the same domain as [3](#), we have tabulated some results for the calculation of these resonances using 2000 eigenvalues and 20 modes for each connected component of  $\Gamma$  (see [Figure 4](#)). The reader can see from [Table 2](#) and [Table 3](#) how the result stabilises as the mesh refinement increases.

Mesh refinement	$R = 0.3 \delta = 0.1$	$R = 0.3 \delta = 0.2$
10	$1.50943 + 0.0001157i$	$1.51791 + 0.0004530i$
15	$1.50847 + 0.0001185i$	$1.51708 + 0.0004657i$
20	$1.50821 + 0.0001193i$	$1.51684 + 0.0004691i$
25	$1.50805 + 0.0001198i$	$1.51670 + 0.0004712i$
30	$1.50797 + 0.0001200i$	$1.51663 + 0.0004722i$
35	$1.50793 + 0.0001202i$	$1.51660 + 0.0004727i$
40	$1.50790 + 0.0001203i$	$1.51657 + 0.0004731i$
45	$1.50788 + 0.0001203i$	$1.51655 + 0.0004734i$
50	$1.50786 + 0.0001204i$	$1.51654 + 0.0004736i$
55	$1.50785 + 0.0001204i$	$1.51653 + 0.0004737i$
60	$1.50785 + 0.0001204i$	$1.51652 + 0.0004738i$
65	$1.50784 + 0.0001204i$	$1.51651 + 0.0004739i$
70	$1.50783 + 0.0001205i$	$1.51651 + 0.0004740i$
75	$1.50783 + 0.0001205i$	$1.51651 + 0.0004740i$
80	$1.50783 + 0.0001205i$	$1.51651 + 0.0004740i$

Table 2: This table shows the location of the resonances calculated as the number of mesh refinements is increased. The domain is unchanged from 3,  $R$  denotes the radius of the obstacle and  $\delta$ , the distance it is off-centre by.

Mesh refinement	$R = 0.5 \delta = 0.1$	$R = 0.5 \delta = 0.2$
10	$1.39874 + 0.0009122i$	$1.41857 + .00385170i$
15	$1.39822 + 0.0009199i$	$1.41811 + 0.0038854i$
20	$1.39805 + 0.0009225i$	$1.41796 + 0.0038971i$
25	$1.39797 + 0.0009237i$	$1.41789 + 0.0039021i$
30	$1.39793 + 0.0009243i$	$1.41786 + 0.0039049i$
35	$1.39791 + 0.0009247i$	$1.41784 + 0.0039063i$
40	$1.39789 + 0.0009249i$	$1.41782 + 0.0039075i$
45	$1.39788 + 0.0009251i$	$1.41781 + 0.0039083i$
50	$1.39787 + 0.0009252i$	$1.41781 + 0.0039089i$
55	$1.39786 + 0.0009253i$	$1.41780 + 0.0039092i$
60	$1.39786 + 0.0009254i$	$1.41780 + 0.0039095i$
65	$1.39786 + 0.0009254i$	$1.41779 + 0.0039097i$
70	$1.39785 + 0.0009255i$	$1.41779 + 0.0039098i$
75	$1.39785 + 0.0009255i$	$1.41779 + 0.0039101i$
80	$1.39785 + 0.0009255i$	$1.41779 + 0.0039101i$

Table 3: Like Table 2, this table shows the location of the resonances calculated as the number of mesh refinements is increased. The domain is still unchanged from 3,  $R$  still denotes the radius of the obstacle and  $\delta$ , the distance it is off-centre by.

### 8.3 Some notes on the number of eigenvalues

In subsection 8.1, we mentioned that we used 2000 eigenvalues but gave no justification for this. Here we will present the reader with some graphs and charts to demonstrate convincingly, why we chose this number. We have tabulated data describing what happens when number of eigenvalues is increased. We also present some convergence graphs that track the value of the leading coefficient of the scattering matrix as the number of eigenvalues increases for a selection of the domains at a randomly chosen point.



Number of eigenvalues	$R = 0.3 \delta = 0.1$	$R = 0.3 \delta = 0.2$
200	$1.50783 + 0.000120482i$	$1.51651 + 0.000474043i$
300	$1.50783 + 0.000120482i$	$1.51651 + 0.000474043i$
400	$1.50783 + 0.000120482i$	$1.51651 + 0.000474045i$
500	$1.50783 + 0.000120482i$	$1.51651 + 0.000474045i$
600	$1.50783 + 0.000120482i$	$1.51651 + 0.000474045i$
700	$1.50783 + 0.000120482i$	$1.51651 + 0.000474045i$
800	$1.50783 + 0.000120483i$	$1.51651 + 0.000474045i$
900	$1.50783 + 0.000120482i$	$1.51651 + 0.000474045i$
1000	$1.50783 + 0.000120483i$	$1.51651 + 0.000474046i$
1200	$1.50783 + 0.000120483i$	$1.51651 + 0.000474046i$
1400	$1.50783 + 0.000120483i$	$1.51651 + 0.000474046i$
1600	$1.50783 + 0.000120483i$	$1.51651 + 0.000474046i$
1800	$1.50783 + 0.000120483i$	$1.51651 + 0.000474046i$
2000	$1.50783 + 0.000120483i$	$1.51651 + 0.000474046i$

Table 4: This table shows the effect of increasing the number of eigenvalues on the accuracy of some selected resonances. We have used a mesh refinement of 80 here and 20 modes. The domain is from 3,  $R$  denotes the radius of the obstacle and  $\delta$ , the distance it is off-centre by.

Number of eigenvalues	$R = 0.5 \delta = 0.1$	$R = 0.5 \delta = 0.2$
200	$1.39785 + 0.000925529i$	$1.41779 + 0.00391010i$
300	$1.39785 + 0.000925538i$	$1.41779 + 0.00391011i$
400	$1.39785 + 0.000925536i$	$1.41779 + 0.00391013i$
500	$1.39785 + 0.000925536i$	$1.41779 + 0.00391013i$
600	$1.39785 + 0.000925536i$	$1.41779 + 0.00391014i$
700	$1.39785 + 0.000925536i$	$1.41779 + 0.00391014i$
800	$1.39785 + 0.000925537i$	$1.41779 + 0.00391014i$
900	$1.39785 + 0.000925536i$	$1.41779 + 0.00391014i$
1000	$1.39785 + 0.000925537i$	$1.41779 + 0.00391014i$
1200	$1.39785 + 0.000925537i$	$1.41779 + 0.00391014i$
1400	$1.39785 + 0.000925537i$	$1.41779 + 0.00391014i$
1600	$1.39785 + 0.000925537i$	$1.41779 + 0.00391014i$
1800	$1.39785 + 0.000925537i$	$1.41779 + 0.00391014i$
2000	$1.39785 + 0.000925537i$	$1.41779 + 0.00391014i$

Table 5: This table shows the effect of increasing the number of eigenvalues on the calculated position of some selected resonances. The domain remains the same,  $R$  denotes the radius of the obstacle and  $\delta$ , the distance it is off-centre by.

From this it might seem like it is unnecessary to use many eigenvalues, however the number of eigenvalues does have a significant impact on the coefficients of the scattering matrix as can be seen in Figure 5. We have picked, as an example, the domain where  $R = 0.3$  and  $\delta = 0.1$  with a mesh refinement of 80. We have plotted the real and imaginary components of the leading coefficient of the scattering matrix at the value  $1 + 0.1i$ . This is typical behavior for any arbitrarily chosen point.

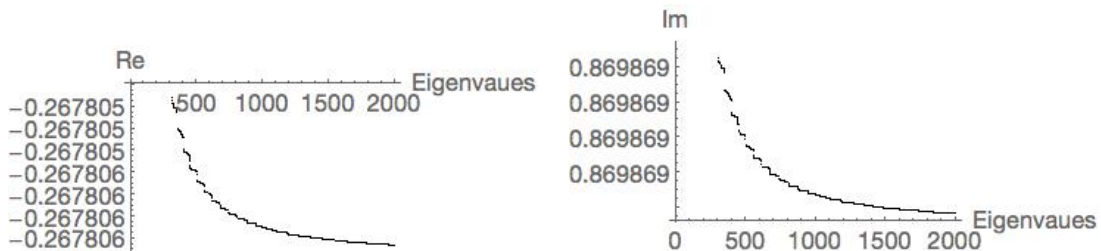


Figure 5: The real and imaginary components of the first coefficient of the scattering matrix plotted against the number of eigenvalues and eigenvectors used to compute it.

#### 8.4 Some notes on the “length” of the domains

The reader should recall that the waveguide is made up of a compact part, on which we have applied the finite element method, and the non-compact ends. The choice as to where the compact part finishes and the ends begin is somewhat arbitrary. It does, however, have an impact on the accuracy of any calculations. In practice, if the compact part is taken to be too large, the internal Neumann to Dirichlet map becomes ill conditioned and a source of major inaccuracy. If the compact part is cut too short then the effect of the obstacle is not completely captured. We settled on the lengths we did after careful experimentation. It should be noted here that all domains for which we have furnished results are comparatively simple, so performing these kind of experiments are “affordable” in terms of compute time; for more complicated and higher dimensional domains, it is a potential vulnerability of the method and extreme care must be taken to ensure that the matrix inversions are well conditioned.

#### 8.5 The importance of the number of modes and choice of auxiliary point

It should be noted that the number of modes and the auxiliary point (the  $\tilde{\lambda}$  from 17 used to speed up convergence) chosen had an undetectable effect on the accuracy of the calculations, so long as the choice was “sensible”. It should also be noted that increasing the number of modes is the most computationally costly action we can take, in theory as well as practice, and should be minimised. For the rest of the results presented, we have used 20 modes for each end, 1000 eigenvalues and a mesh refinement of 30.

#### 8.6 Observing a resonance transition from one sheet to another

This subsection will present the first explicit calculation of the scattering matrix beyond the first non-physical sheet  $J = \{1\}$  of  $Z$  ( $Z$  was defined by equation (6) in section 2). We will be looking at the same domain as we did in 3, with the same conventions; an infinite rectangle in  $\mathbb{R}^2$  of width 2 with a single, circular ‘obstruction’ of radius  $R$  whose centre is moved up by  $\delta$  from the middle of the waveguide. In 3 we used the parameterisation  $\lambda \mapsto \lambda^2$  to enable the comparison of our results with other peoples’, from now on we will no longer do this. Of particular interest here is the case where  $R = 0.2$ . As  $\delta$  increases from 0.6 to 0.7 the resonance moves from the sheet  $J = \{1\}$  to  $J = \{2\}$  as can be seen in Table 6. This was the first circumstance where such a phenomena has been observed.

$R = 0.2$	$J = \{1\}$	$J = \{2\}$
$\delta = 0$	2.4036	-
$\delta = 0.1$	$2.40712 + 0.00006i$	-
$\delta = 0.2$	$2.41709 + 0.00021i$	-
$\delta = 0.3$	$2.43170 + 0.00040i$	-
$\delta = 0.4$	$2.44777 + 0.00059i$	-
$\delta = 0.5$	$2.46101 + 0.00053i$	-
$\delta = 0.6$	$2.46725 + 0.00013i$	-
$\delta = 0.7$	-	$2.46475 + 0.00063i$

Table 6: Resonances for the domain  $R = 0.2$  showing the resonance moving to a different sheet of  $Z$  as  $\delta$  increases from 0.6 to 0.7

We will also include some contour plots of the absolute value of the determinant of the scattering matrix for this occurrence. We can observe that the “tail” of the resonance is visible on both sheets prior to the resonance crossing over.

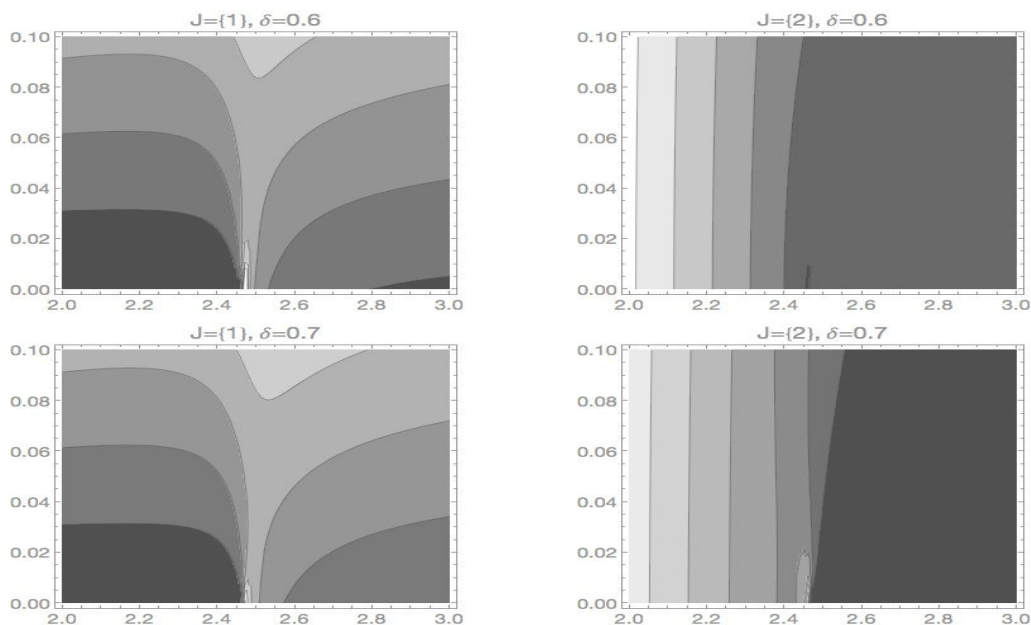


Figure 6: Contour plots of the absolute value of the the determinant of  $S_J(\lambda)$ . The resonance can be seen crossing from sheet  $J = \{1\}$  to  $J = \{2\}$  as  $\delta$  is increased from 0.6 to 0.7.  $R = 0.2$  throughout.

### 8.7 Adding “Ends” to a Circle

In this example, we have added a single, infinite rectangular ‘end’ to a circle of radius 2 and the width of the ‘end’ has been varied. We have searched for resonances, on sheets  $J = \{1\}$ ,  $J = \{2\}$  and  $J = \{3\}$ , within in the search area given by:

$$\{\lambda : 0 \leq \text{Re}(\lambda) \leq 15, -3 \leq \text{Im}(\lambda) \leq 3\}, \quad (38)$$

and recorded the results the results found in Table 8. The choice of sheets and and search area on which to look for resonances was somewhat arbitrary, though we believe they provide a wide enough ‘field of vision’ to observe interesting phenomena, but are limited enough to minimise compute time and cost. We have chosen this as an example because we feel it interesting to observe that when the width of the end is close to zero, the resonances

are very close to the Neumann eigenvalues of the circle of radius 2, of which we have given the first 9 non-zero examples in Table 7. When the width of the end is continuously increased, they migrate away from these values in distinct paths. We are confident of the accuracy of the resonances calculated to least three decimal places, though we have included the fourth place in a lighter shade for the reader’s information. In addition to this, we have plotted them on graphs in Figure 8 with colour coded markers indicating the respective sheet of  $Z$  they reside on; black for  $J = \{1\}$ , red for  $J = \{2\}$  and green for  $J = \{3\}$ . In the case of varying widths, the paths taken by the resonances as the widths increase are clearly visible, and we have included them.

0.8476	2.3323	3.6709	4.4130	7.0698	7.1068	10.2911	11.2442	12.3059
--------	--------	--------	--------	--------	--------	---------	---------	---------

Table 7: Neumann eigenvalues for a circle of radius 2.



Figure 7: An example of some internal domains produced by FreeFEM++ for these calculations.

$J = \{1\}$			
$w = 0.1$	$w = 0.2$	$w = 0.5$	$w = 1$
0.8496 + 0.0206 <i>i</i>	0.8541 + 0.0408 <i>i</i>	0.8753 + 0.0992 <i>i</i>	0.9255 + 0.1957 <i>i</i>
2.3388 + 0.0416 <i>i</i>	2.3525 + 0.0807 <i>i</i>	2.413 + 0.1848 <i>i</i>	2.5431 + 0.3334 <i>i</i>
3.6742 + 0.0145 <i>i</i>	3.6814 + 0.0265 <i>i</i>	3.7091 + 0.0512 <i>i</i>	3.7577 + 0.0802 <i>i</i>
4.4246 + 0.0664 <i>i</i>	4.4480 + 0.1285 <i>i</i>	4.5452 + 0.3009 <i>i</i>	4.7425 + 0.5764 <i>i</i>
7.0947 + 0.0022 <i>i</i>	7.0946 + 0.0011 <i>i</i>	7.0947 + 0.0004 <i>i</i>	7.8573 + 1.0573 <i>i</i>
7.1124 + 0.1328 <i>i</i>	7.1721 + 0.2541 <i>i</i>	7.4065 + 0.5657 <i>i</i>	10.7726 + 0.1699 <i>i</i>
10.3293 + 0.1186 <i>i</i>	10.4037 + 0.2071 <i>i</i>	10.6664 + 0.2940 <i>i</i>	12.2884 + 0.8945 <i>i</i>
11.2547 + 0.0572 <i>i</i>	11.2733 + 0.1151 <i>i</i>	11.368 + 0.35467 <i>i</i>	12.0991 + 0.3521 <i>i</i>
12.3122 + 0.0263 <i>i</i>	12.3237 + 0.0502 <i>i</i>	12.3567 + 0.1348 <i>i</i>	14.7205 + 0.6654 <i>i</i>
14.1139 + 0.1527 <i>i</i>	14.1949 + 0.2766 <i>i</i>	14.4649 + 0.5345 <i>i</i>	
$w = 1.5$	$w = 2$	$w = 2.5$	$w = 3$
0.9900 + 0.3010 <i>i</i>	1.0710 + 0.4264 <i>i</i>	1.1752 + 0.5898 <i>i</i>	1.3176 + 0.8290 <i>i</i>
2.6965 + 0.4627 <i>i</i>	2.8746 + 0.5614 <i>i</i>	3.0696 + 0.5767 <i>i</i>	3.1713 + 0.4476 <i>i</i>
3.8154 + 0.1224 <i>i</i>	3.9000 + 0.2054 <i>i</i>	4.0569 + 0.3750 <i>i</i>	4.5055 + 0.5486 <i>i</i>
4.9403 + 0.8597 <i>i</i>	5.0970 + 1.1417 <i>i</i>	5.1843 + 1.3491 <i>i</i>	5.3676 + 1.4029 <i>i</i>
7.1276 + 0.0014 <i>i</i>	7.1757 + 0.0151 <i>i</i>	9.2528 + 2.4257 <i>i</i>	
8.2933 + 1.6165 <i>i</i>	8.6102 + 2.1380 <i>i</i>		
10.7234 + 0.0883 <i>i</i>	11.9290 + 0.0758 <i>i</i>		
12.0129 + 0.1897 <i>i</i>	13.780 + 2.6635 <i>i</i>		
13.1459 + 2.0493 <i>i</i>			
14.5252 + 0.3368 <i>i</i>			

$J = \{2\}$			
$w = 0.1$	$w = 0.2$	$w = 0.5$	$w = 1$
			10.2536 + 0.11562 <i>i</i> 11.2435 + 0.01777 <i>i</i> 14.2167 + 0.42736 <i>i</i>
$w = 1.5$	$w = 2$	$w = 2.5$	$w = 3$
7.2439 + 0.4698 <i>i</i> 7.7433 + 0.9299 <i>i</i> 10.9022 + 0.7275 <i>i</i> 11.1970 + 0.1239 <i>i</i>	4.6065 + 0.4829 <i>i</i> 11.0769 + 0.08842 <i>i</i> 11.8264 + 1.38616 <i>i</i>	2.3695 + 0.3389 <i>i</i> 4.9869 + 0.8324 <i>i</i> 7.1156 + 0.0011 <i>i</i> 8.3670 + 1.44198 <i>i</i> 10.9968 + 0.0355 <i>i</i> 12.7631 + 2.1634 <i>i</i>	2.6042 + 0.6511 <i>i</i> 5.5217 + 1.2393 <i>i</i> 7.1356 + 0.0192 <i>i</i> 9.1096 + 1.9492 <i>i</i> 14.1616 + 2.7545 <i>i</i>
$J = \{3\}$			
$w = 0.1$	$w = 0.2$	$w = 0.5$	$w = 1$
$w = 1.5$	$w = 2$	$w = 2.5$	$w = 3$
	10.3673 + 0.2721 <i>i</i> 14.7867 + 0.8682 <i>i</i>	7.0639 + 0.2546 <i>i</i> 11.831 + 0.0873 <i>i</i> 11.161 + 1.0833 <i>i</i>	7.6306 + 0.7147 <i>i</i> 11.6268 + 0.0988 <i>i</i> 12.4059 + 2.0430 <i>i</i>

Table 8: Resonances calculated when a single rectangular ‘end’ is added to a circle of radius 2 in two dimensions when  $\lambda$  is in the sheets  $J = \{1\}$ ,  $J = \{2\}$  and  $J = \{3\}$  of  $Z$  as indicated.  $\omega$  denotes the width of the end which increases from 0.1 to 3.

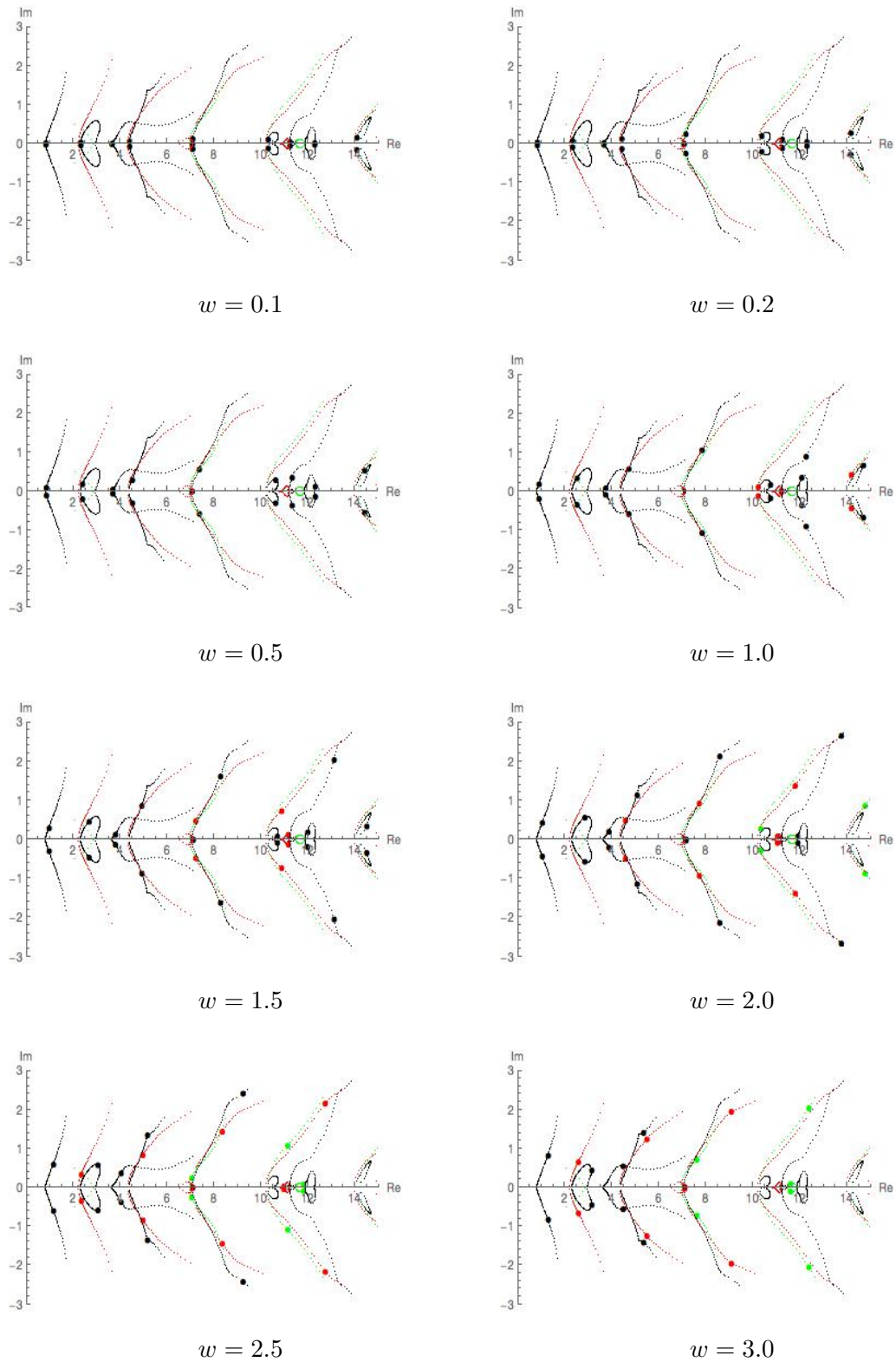


Figure 8: Some colour-coded plots of the location of resonances. In this instance the width of a single end is changed. The larger markers are the resonances for the domain indicated, the smaller markers are the entire family of domains with varying widths. This allows the reader to visualise the paths the resonances take as the width of the end increases. Similar, but not identical, paths can be observed with higher numbers of ends.

### 8.8 Animated motion of resonances as the domain is continuously altered

Due to the speed of the method it has been possible to animate the motion of the resonances as the domains are continuously changed. An animation of the results found in 8.7 can be found in the link below along with many more. Of particular interest to the reader are occurrences where resonances cross sheets, higher order resonances bifurcate, resonances loop etc. The same colour coding convention applies in the animations as in Figure 8. Many other similar animations have been produced

- Varying the width of one rectangular end  
<https://www.youtube.com/watch?v=6AryY-dHRAM>
- Varying the width of two equispaced rectangular ends  
<https://www.youtube.com/watch?v=GR13vY2IkOY>
- Varying the size of a centrally placed circular ‘obstruction’, one end of width 1  
<https://www.youtube.com/watch?v=T6Xap-VQHmQ>
- Varying the size of a centrally placed circular ‘obstruction’, two equispaced ends of width 1  
<https://www.youtube.com/watch?v=WhcdPBotn9A>
- Varying the size of a centrally placed circular ‘obstruction’, three equispaced ends of width 1  
<https://www.youtube.com/watch?v=UP4P9xW7ATI>
- Varying the angle between two rectangular ends of width 1  
<https://www.youtube.com/watch?v=8HDXRfMqxfU>
- Moving a circular ‘obstruction’ of radius 0.5 in the x axis, 3 equally spaced ends of width 1. <https://www.youtube.com/watch?v=B8WmXaDoObs>
- Moving a circular ‘obstruction’ of radius 0.5 in the y axis, 3 equally spaced ends of width 1. <https://www.youtube.com/watch?v=WncB1xoSSzs>

If the reader has any difficulty in accessing these videos, then they can be provided, on request, by other means.

### 8.9 Time delay and scattering length

The notions of time delay and scattering length hail from dynamic scattering theory; they can both be calculated using the scattering matrix. Reed and Simon mention time delay briefly in their text [16]. Müller and Strohmaier have also covered time delay and scattering length in their paper [36], where they give results that relate the time delay to the geometry of the internal domain. In this section, we will apply this to a pair of the domains featured above. It should be noted that the  $\lambda$  in this context will be a real number less than  $\mu_1$ , representing the energy of the system and not an element of the  $Z$ . The Appendix of Müller and Strohmaier’s paper provides an overview of the time delay in this setting [37]. We will take the (non standard) definition to be:

**Definition 10.** *Time delay*

$$T(\lambda) = -2\sqrt{\lambda}S^{-1}(\lambda).S'(\lambda)$$

when  $\lambda = 0$ , we define this to be the *scattering length*.



Wigner and Eisenbud were the first to present the time delay in this manner for potential scattering and  $T(\lambda)$  is often called the Eisenbud-Wigner time delay operator [7][8]. Müller and Strohmaier have, amongst other things, proved this formula for the case of manifolds with cylindrical ends and, in the case of a single end

$$T(0) = 2 \frac{\text{Vol}(X)}{\text{Vol}(\Gamma)}.$$

We will pick a selection of our single ended domains from above and plot their time delay as  $\lambda$  approaches 0.

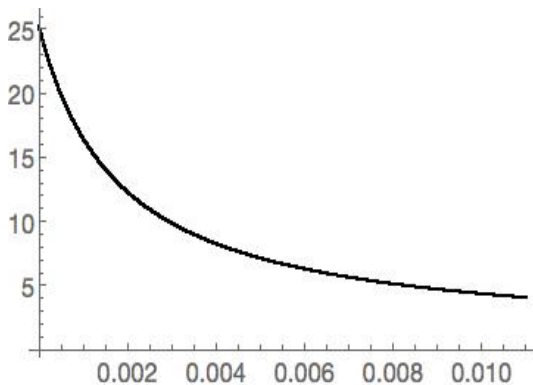


Figure 9: Circle of radius 2, with end width 1.5.  $T(0)$  should theoretically be 25.1327.

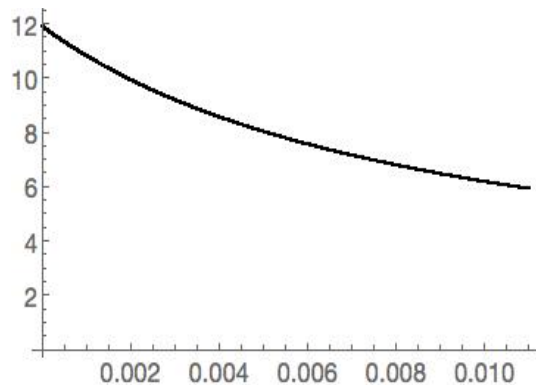


Figure 10: Circle of radius 2 with end width 1, obstacle radius 0.5.  $T(0)$  should theoretically be 11.781.

### 8.10 Limitations and future directions

We will finish by informing the reader of the limitations of what we have done as things stand and, give some suggestions for further work in this immediate area.

- The algorithm is currently limited to Neumann or Acoustic waveguides in Euclidian space; it can be extended, with minimal additional theoretical work, to those with Dirichlet or mixed boundary conditions, as well as those with non-Euclidian metrics on either the internal domain or the ends. There is also the possibility to look at potentials supported on the internal domain.
- Applying this method to higher dimensions and more complicated domains is possible, though the computational cost will rise very significantly and greater care needs to be taken with accuracy.
- Continuing to use the current method as it is to either search for interesting phenomena with respect to the distribution of resonances as David Borthwick has done for hyperbolic surfaces [28], or perform domain optimisation and create domains from given patterns of resonances.
- Adopting the boundary element method [21] in place of the finite element method or adopt finite element method techniques that minimise dispersion error and try to improve accuracy yet further [38].

## 9 References

- [1] M. Levitin and A. Strohmaier, “Computations of eigenvalues and resonances on perturbed hyperbolic surfaces with cusps,” 2019.



- 
- [2] R. Melrose, *Geometric Scattering Theory*. Stanford Lectures: Distinguished Visiting Lecturers in Mathematics, Cambridge University Press, 1995.
- [3] M. Levitin and M. Marletta, “A simple method of calculating eigenvalues and resonances in domains with infinite regular ends,” *Proceedings of the Royal Society of Edinburgh: Section A Mathematics*, vol. 138, pp. 1043–1065, 10 2008.
- [4] A. Aslanyan, L. Parnovski, and D. Vassiliev, “Complex resonances in acoustic waveguides,” *The Quarterly Journal of Mechanics and Applied Mathematics*, vol. 53, no. 3, pp. 429–447, 2000.
- [5] T. Jiang and Y. Xiang, “Perfectly-matched-layer method for optical modes in dielectric cavities,” *Physical Review A*, vol. 102, Nov 2020.
- [6] T. Christiansen, “Some upper bounds on the number of resonances for manifolds with infinite cylindrical ends,” *Annales Henri Poincaré*, vol. 3, no. 5, pp. 895–920, 2002.
- [7] E. P. Wigner, “Lower limit for the energy derivative of the scattering phase shift,” *Phys. Rev.*, vol. 98, pp. 145–147, Apr 1955.
- [8] L. Eisenbud, *Dissertation, Unpublished*. PhD thesis, Princeton University, 1948.
- [9] D. V. Evans, M. Levitin, and D. Vassiliev, “Existence theorems for trapped modes,” *Journal of Fluid Mechanics*, vol. 261, pp. 21–31, 2 1994.
- [10] E. B. Davies and L. Parnovski, “Trapped modes in acoustic waveguides,” *Quart. J. Mech. Appl. Math.*, vol. 51, no. 3, pp. 477–492, 1998.
- [11] H. Hawkins and L. Parnovski, “Trapped modes in a waveguide with a thick obstacle,” *Mathematika*, vol. 51, no. 1-2, pp. 171–186 (2005), 2004.
- [12] E. R. Johnson, M. Levitin, and L. Parnovski, “Existence of eigenvalues of a linear operator pencil in a curved waveguide—localized shelf waves on a curved coast,” *SIAM J. Math. Anal.*, vol. 37, no. 5, pp. 1465–1481 (electronic), 2006.
- [13] M. Reed and B. Simon, *I: Functional Analysis*. Methods of Modern Mathematical Physics, Elsevier Science, 1981.
- [14] G. Roddick”, “Computation of scattering matrices and resonances for waveguides,” ”6” ”2016”.
- [15] L. Guillopé, “Théorie spectrale de quelques variétés à bouts,” *Ann. Sci. École Norm. Sup. (4)*, vol. 22, no. 1, pp. 137–160, 1989.
- [16] M. Reed and B. Simon, *Methods of modern mathematical physics. III*. Academic Press [Harcourt Brace Jovanovich, Publishers], New York-London, 1979. Scattering theory.
- [17] A. Girouard and I. Polterovich, “Spectral Geometry of the Steklov Problem.” 2014.
- [18] M. Taylor, *Partial Differential Equations I: Basic Theory*. Applied Functional Analysis: Applications to Mathematical Physics, Springer, 1996.
- [19] F. L. Gall, “Faster algorithms for rectangular matrix multiplication,” *CoRR*, vol. abs/1204.1111, 2012.
- [20] “Casimir physics,” *Lecture Notes in Physics*, 2011.

- [21] W. undefinedmigaj, T. Betcke, S. Arridge, J. Phillips, and M. Schweiger, “Solving boundary integral problems with bem++,” *ACM Trans. Math. Softw.*, vol. 41, Feb. 2015.
- [22] A. Barnett, “Mpspack : 2d helmholtz scattering and eigenvalue problems via particular solutions and integral equations,” 2016.
- [23] R. B. Melrose, “Polynomial bound on the distribution of poles in scattering by an obstacle,” *Journaes Equations aux derivees partielles*, pp. 1–8, 1984.
- [24] M. Zworski, “Sharp polynomial bounds on the number of scattering poles of radial potentials,” *Journal of Functional Analysis*, vol. 82, no. 2, pp. 370 – 403, 1989.
- [25] Z. M. Christiansen, Tanya, “Spectral asymptotics for manifolds with cylindrical ends,” *Annales de l’institut Fourier*, vol. 45, no. 1, pp. 251–263, 1995.
- [26] L. Guillope and M. Zworski, “Scattering asymptotics for riemann surfaces,” *Annals of Mathematics*, vol. 145, no. 3, pp. pp. 597–660, 1997.
- [27] A. M. Aslanyan and E. B. Davies, “Separation of variables in perturbed cylinders,” Tech. Rep. math.SP/0012113, Dec 2000.
- [28] D. Borthwick, “Distribution of resonances for hyperbolic surfaces,” *Exp. Math.*, vol. 23, no. 1, pp. 25–45, 2014.
- [29] L. N. Trefethen and T. Betcke, “Computed eigenmodes of planar regions,” in *Recent advances in differential equations and mathematical physics*, vol. 412 of *Contemp. Math.*, pp. 297–314, Amer. Math. Soc., Providence, RI, 2006.
- [30] T. Betcke and L. N. Trefethen, “Reviving the method of particular solutions,” *SIAM Rev.*, vol. 47, no. 3, pp. 469–491 (electronic), 2005.
- [31] S. A. Nazarov, K. Ruotsalainen, and P. Uusitalo, “Bound states of waveguides with two right-angled bends,” *J. Math. Phys.*, vol. 56, no. 2, pp. 021505, 24, 2015.
- [32] F. Hecht, “New development in freefem++,” *J. Numer. Math.*, vol. 20, no. 3-4, pp. 251–265, 2012.
- [33] R. B. Lehoucq, D. C. Sorensen, and C. Yang, “Arpack users guide: Solution of large scale eigenvalue problems by implicitly restarted arnoldi methods.,” 1997.
- [34] D. C. Sorensen, “Implicitly restarted arnoldi/lanczos methods for large scale eigenvalue calculations,” 1996.
- [35] T. Betcke and L. Trefethen, “Reviving the method of particular solutions,” *SIAM Review*, vol. 47, pp. 469–491, 09 2005.
- [36] W. Müller, “On the analytic continuation of rank one eisenstein series,” *Geometric & Functional Analysis GAFA*, vol. 6, no. 3, pp. 572–586, 1996.
- [37] W. Müller and A. Strohmaier, “Scattering at low energies on manifolds with cylindrical ends and stable systoles,” *Geometric and Functional Analysis*, vol. 20, no. 3, pp. 741–778, 2010.
- [38] Z. C. He, A. G. Cheng, G. Y. Zhang, Z. H. Zhong, and G. R. Liu, “Dispersion error reduction for acoustic problems using the edge-based smoothed finite element method (es-fem),” *International Journal for Numerical Methods in Engineering*, vol. 86, no. 11, pp. 1322–1338, 2011.