

Computation of Topological Interface Modes

Lara Vrabac

MSc CSE, D-MATH, February 2025

Semester project supervised by
Prof. Dr. Ralf Hiptmair

ETH zürich

Contents

1	Introduction	2
2	Interface Modes in 1D	2
2.1	Mathematical Background and Numerical Methods	3
2.1.1	Implicit Midpoint Rule	3
2.1.2	Bloch Modes and the Monodromy	4
2.1.3	Secant Method	10
2.1.4	Extrapolation to Zero	14
2.2	Errors and Convergence	16
2.2.1	Aitken-Neville Results for Fixed SM and Tolerance (Implicit Midpoint Rule)	16
2.2.2	Omega Stepping Results for Fixed AN/SM	17
2.2.3	Secant Method and Time	17
2.2.4	Drawing (Implicit Midpoint Rule) Errors	17
2.3	Results	18
2.3.1	Inversion-Symmetric Interface Modes	18
2.3.2	Non-Inversion-Symmetric Interface Modes	22
3	Interface Modes in 1.5D	26
3.1	Review - Infinite Stripes in 2D	26
3.2	Results	28
4	Zak Phase and Topologically (Non) Protected Interface Modes	31
4.1	Zak Phase	31
4.1.1	Trapezoidal Rule and Finite Differences	31
4.2	Topologically Protected Interface Modes	34
4.2.1	Tiling	34
4.2.2	Persistent Modes	34
5	Conclusion and Further Work	35
6	Using the Code	35
6.1	List of Documented Class Members	37
7	References	38

1 Introduction

This project presents a way of computing general interface modes in 1- and 1.5D for two joint semi-infinite photonic crystals with periodic material parameters ε and μ based on the transfer matrix method.

The period does not need to be the same and the material parameters are piecewise continuous functions on both sides. An interface mode is a Bloch mode localized at the interface, and it shows exponential decay on both sides. It is the solution to the joint left- and right-side equations, which is continuous both in value and derivative across the interface. In addition, there is a computation of band diagrams with marked interface modes and the full Zak phase calculator.

Some interface modes are topologically protected - or guaranteed to exist - if the Zak phases of the bands below a band gap, which houses this mode, induce different topological indices. In the case of inversion symmetric μ and ε it is sufficient to check the parity of the first Bloch mode that belongs to a band below the band gap. The core numerical methods used are the implicit midpoint rule, extrapolation to zero using the Aitken-Neville scheme, and the secant method. The mathematical background, implementation, and application for each are discussed in the following sections.

$$\nabla \times \mathbf{e}(\mathbf{x}) = -i\omega\mu(\mathbf{x})\mathbf{h}(\mathbf{x}) \quad (1)$$

$$\nabla \times \mathbf{h}(\mathbf{x}) = i\omega\varepsilon(\mathbf{x})\mathbf{e}(\mathbf{x}) \quad (2)$$

The starting point are Maxwell's equations, shown in [equation 1](#) and [equation 2](#). Everything is normalized so that $c = 1$. The transfer matrix method involves finding two orthogonal solutions with unit initial conditions and constructing a matrix from them, that can *transfer* any initial conditions across the unit cell it was defined for. With transfer matrices for both spaces, we are equipped to construct any Bloch mode if the solution for just one unit cell per space is available. Going in reverse is also simple: the inverse of the transfer matrix is used.

This matrix also solves the original ordinary differential equation, meaning it is a fundamental matrix, and it is evaluated at the end of the unit cell (period), so it is also the *monodromy* matrix. The eigenvalues of the monodromy matrix are the characteristic multipliers of the Bloch waves, and their graph over a range of frequencies is the band diagram of the unit cell. After solving for the monodromy across one unit cell, the solution Bloch can be constructed from the monodromy and the waves that the two orthogonal solutions trace.

2 Interface Modes in 1D

Now we will pick two orthogonal initial conditions (and solutions) to find the monodromy matrix, and the Bloch waves in the sections that follow.

Explanation of some terms that will make the equations easier to understand:

***y*-polarized wave** A wave with the electric field in the *y*-direction.

***z*-polarized wave** A wave with the electric field in the *z*-direction.

TE (transversal electric) Describes the initial condition: electric field has a value of 1, magnetic field has a value of 0

TM (transversal magnetic) Describes the initial condition: magnetic field has a value of 1, electric field has a value of 0

For example, to get the system transfer matrix, both TE and TM systems will be solved. In both cases these are 2D vectors. These are directly derived from [equation 1](#) and [equation 2](#). There are two options that we get, and here is an example of the z -polarized TE and TM vectors: $\begin{bmatrix} e_{zTE}(x) \\ h_{yTE}(x) \end{bmatrix}$ and $\begin{bmatrix} e_{zTM}(x) \\ h_{yTM}(x) \end{bmatrix}$

These two vectors can be concatenated into a 2×2 z -polarized system matrix (see [lemma 1](#)) which will turn out to be the transfer matrix itself when evaluated at the end of the unit cell.

2.1 Mathematical Background and Numerical Methods

This section will provide a selection of definitions and proofs that explain the choice of numerical methods. First, we must satisfy the continuity and Bloch mode conditions. This, in the context of Maxwell's equations, implies a certain behavior of solutions that can be used to refine the search for frequencies at which interface modes exist.

Definition 1 *The Bloch mode is any function that satisfies*

$$\psi(x + a) = e^{ika}\psi(x) \quad (3)$$

for the period a and some complex constant k .

Definition 2 *The tangential components of \mathbf{e} and \mathbf{h} are continuous across the interface.*

$$\begin{aligned} \mathbf{h}(0^+) &= \mathbf{h}(0^-) \\ \mathbf{e}(0^+) &= \mathbf{e}(0^-) \end{aligned} \quad (4)$$

2.1.1 Implicit Midpoint Rule

The following definitions and proofs will show why energy-preserving Runge-Kutta methods work well for this problem.

The following lemma shows the equivalence of 1D systems.

Lemma 1 *The y - and z -polarized solutions are complex conjugates of each other.*

Proof. The z -polarized system is

$$\begin{bmatrix} e_{zTE}(x) & e_{zTM}(x) \\ h_{yTE}(x) & h_{yTM}(x) \end{bmatrix}' = \begin{bmatrix} 0 & i\omega\mu(x) \\ i\omega\varepsilon(x) & 0 \end{bmatrix} \begin{bmatrix} e_{zTE}(x) & e_{zTM}(x) \\ h_{yTE}(x) & h_{yTM}(x) \end{bmatrix}, \quad (5)$$

and the y -polarized system is

$$\begin{bmatrix} e_{yTE}(x) & e_{yTM}(x) \\ h_{zTE}(x) & h_{zTM}(x) \end{bmatrix}' = \begin{bmatrix} 0 & -i\omega\mu(x) \\ -i\omega\varepsilon(x) & 0 \end{bmatrix} \begin{bmatrix} e_{yTE}(x) & e_{yTM}(x) \\ h_{zTE}(x) & h_{zTM}(x) \end{bmatrix}. \quad (6)$$

If we conjugate the z -polarized system, the result is:

$$\begin{bmatrix} e_{zTE}^*(x) & e_{zTM}^*(x) \\ h_{yTE}^*(x) & h_{yTM}^*(x) \end{bmatrix}' = \begin{bmatrix} 0 & -i\omega\mu(x) \\ -i\omega\varepsilon(x) & 0 \end{bmatrix} \begin{bmatrix} e_{zTE}^*(x) & e_{zTM}^*(x) \\ h_{yTE}^*(x) & h_{yTM}^*(x) \end{bmatrix}, \quad (7)$$

from which it follows that $e_{zTE}^* = e_{yTE}$, $e_{zTM}^* = e_{yTM}$, $h_{yTE}^* = h_{zTE}$, and $h_{yTM}^* = h_{zTM}$ as desired. \blacksquare

Knowing the system, we can use the implicit midpoint rule like described in the following pseudocode to find the z -polarized solution matrix. Note that saving the full solution is not necessary (and it is expensive) to get the monodromy matrix. It is included in the listing below to illustrate how it is stored in a $4 \times N_{\text{pts}}$ solution matrix.

```

1 function systemZ(Npts, T, solution,  $\omega$ ,  $\mu$ ,  $\varepsilon$ )
2   h = T/Npts
3   Matrix A(2,2), B(2,2), C(2,2)
4   Matrix I = Identity2x2()
5   Matrix Y = I //the initial monodromy matrix
6
7   for i = 0 : N-1 do
8     x = (i + 0.5) * h
9     A = 0, i *  $\omega$  *  $\mu$ (x)
10      i *  $\omega$  *  $\varepsilon$ (x), 0
11
12     B = I - 0.5 * h * A
13     C = I + 0.5 * h * A
14
15     // implicit midpoint rule
16     Y = B.inverse() * C * Y
17     solution.col(i) = Y.reshaped(4,1) // 2x2 matrix
18                                         // to 4-vector
19
20 end
return Y

```

2.1.2 Bloch Modes and the Monodromy

Here we will see some additional properties of the monodromy matrix Y , and how these can be used to construct Bloch modes using the Monodromy and its eigenvalues.

The general structure of these proofs follows that from Lin and Zhang [1]; however, proofs that are similar are not the same because they had to take into account complex rather than real numbers for this project.

Lemma 2 *The electric field Bloch wave of the system above is given by*

$$e_{\text{Bloch}}(x) = e_{TX}e_{TE}(x) + h_{TX}e_{TM}(x), \quad (8)$$

and the magnetic field Bloch mode is given by

$$h_{\text{Bloch}}(x) = e_{TX}h_{TE}(x) + h_{TX}h_{TM}(x), \quad (9)$$

where $\begin{bmatrix} e_{TX} \\ h_{TX} \end{bmatrix}$ is an eigenvector of the monodromy matrix $\mathbf{M}(\omega)$. These Bloch modes are continuous across the cells.

Proof. Eigenvectors of the monodromy satisfy the following

$$\mathbf{M}(\omega) \begin{bmatrix} e_{TX}(x) \\ h_{TX}(x) \end{bmatrix} = e^{ika} \begin{bmatrix} e_{TX}(x) \\ h_{TX}(x) \end{bmatrix} = \begin{bmatrix} e_{TX}(x+a) \\ h_{TX}(x+a) \end{bmatrix} \quad (10)$$

(1) Let us construct a Bloch wave of the system solutions of the TE and TM electric fields:

$$e_{Bloch}(x) = Ae_{TE}(x) + Be_{TM}(x) \quad (11)$$

If this is a Bloch wave then

$$e_{Bloch}(x+a) = e^{ika} e_{Bloch} = e^{ika} (Ae_{TE}(x) + Be_{TM}(x)) \quad (12)$$

by [definition 1](#). Combining terms and evaluating at $x = a$ yields

$$\begin{aligned} e_{Bloch}(a) &= e^{ika} e_{Bloch}(0) \\ e^{ika} e_{Bloch}(0) &= Ae^{ika} \\ \Rightarrow A &= e_{Bloch}(0) \end{aligned} \quad (13)$$

Similarly, if the equation is differentiated and multiplied by $\frac{1}{i\omega\epsilon(x)}$,

$$h_{Bloch}(x) = Ah_{TE}(x) + Bh_{TM}(x) \quad (14)$$

$$h_{Bloch}(a) = Ah_{TE}(a) + Bh_{TM}(a) \quad (15)$$

$$h_{Bloch}(a) = e^{ika} h_{Bloch}(0) \quad (16)$$

$$e^{ika} h_{Bloch}(0) = Be^{ika} \quad (17)$$

$$\Rightarrow B = h_{Bloch}(0) \quad (18)$$

Because we had

$$\begin{bmatrix} e_{Bloch}(a) \\ h_{Bloch}(a) \end{bmatrix} = e^{ika} \begin{bmatrix} e_{Bloch}(0) \\ h_{Bloch}(0) \end{bmatrix} = e^{ika} \begin{bmatrix} A \\ B \end{bmatrix} \quad (19)$$

the vector $\begin{bmatrix} A \\ B \end{bmatrix}$ is an eigenvector of the monodromy by definition, see [equation 10](#).

(2) If we construct a wave using some eigenvector of the monodromy $\begin{bmatrix} e_{TX}(0) \\ h_{TX}(0) \end{bmatrix}$ so that the initial values of the constructed waves match with the initial values of the eigenvector, eg. at $x = 0$, then it holds for the constructed wave

$$e_{Bloch}(x) = e_{TX}(0)e_{TE}(x) + h_{TX}(0)e_{TM}(x), \quad (20)$$

that

$$e^{ika}e_{Bloch}(x) = e^{ika}e_{TX}(0)e_{TE}(x) + e^{ika}h_{TX}(0)e_{TM}(x) \quad (21)$$

$$e^{ika}e_{Bloch}(x) = e_{TX}(a)e_{TE}(x) + h_{TX}(a)e_{TM}(x) \quad (22)$$

but $e_{TX}(a)e_{TE}(x) + h_{TX}(a)e_{TM}(x)$ is the wave that matches the eigenvector values at the next cell so it is the constructed wave evaluated at $x + a$ or

$$e^{ika}e_{Bloch}(x) = e_{Bloch}(x + a), \quad (23)$$

which is the definition of a Bloch mode, so the constructed wave is a Bloch mode. If this equation is differentiated and multiplied by $\frac{1}{i\omega\varepsilon(x)}$, we get

$$e^{ika}h_{Bloch}(x) = h_{Bloch}(x + a), \quad (24)$$

so [equation 8](#) and [equation 9](#) give the Bloch modes, as desired. \blacksquare

And not only that; the Bloch modes are by this construction continuous across all cells, because the following cell must start at the point the previous cell finished.

Lemma 3 *The off-diagonal solution entries of the y - and z - polarized transfer matrix systems are imaginary, and the diagonal solution entries are real.*

Proof. Take any system, z - or y - polarized. Here, the z -polarized one is used. Set the initial condition to $\begin{bmatrix} e_{zTE}(x) & e_{zTM}(x) \\ h_{yTE}(x) & h_{yTM}(x) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Because the identity matrix is the initial condition and the system matrix $\mathbf{A} = \begin{bmatrix} 0 & i\omega\mu(x) \\ i\omega\varepsilon(x) & 0 \end{bmatrix}$ always affects the other entry, and its non-zero terms are imaginary, the off-diagonal solution entries will be imaginary (since we started with real numbers), and the diagonal solution entries will be real:

$$\Im(e_{zTE}) = \Im(h_{yTM}) = 0 \quad \text{and} \quad \Re(e_{zTM}) = \Re(h_{yTE}) = 0. \quad (25)$$

Another way to see this is to rewrite the system as a Helmholtz equation for one of the entries, for example, the electric field in the z -direction:

$$-\frac{\partial}{\partial x} \left(\frac{1}{\mu(x)} \frac{\partial}{\partial x} e_z \right) - \omega^2 \varepsilon(x) e_z = 0 \quad (26)$$

and notice the cancellation of imaginary terms. This means that if e_z is imaginary, then e_z will stay imaginary, and likewise, if it is real, it will stay real. Even from this equation it is clear this holds for h_y as well, because for this Helmholtz equation $h_y = \frac{1}{i\omega\mu(x)} \frac{\partial}{\partial x} e_z$. \blacksquare

Definition 3 *The impedance of the electric and magnetic Bloch modes constructed from the monodromy system solution is given by*

$$Z(\omega) = \frac{e_{Bloch}(0)}{h_{Bloch}(0)} = \frac{e_{TX}(0)}{h_{TX}(0)}, \quad (27)$$

where $\begin{bmatrix} e_{TX} \\ h_{TX} \end{bmatrix}$ is an eigenvector of the monodromy matrix $\mathbf{M}(\omega)$ from [lemma 2](#).

The impedance function in [definition 3](#) is the central part of this project as it provides the sufficient and necessary condition for the existence of interface modes.

Definition 4 *The sufficient and necessary condition for the existence of an interface mode in the band gap is*

$$Z_R(\omega) = Z_L(\omega), \quad (28)$$

where Z_L is the impedance of the left half-space, and Z_R is the impedance of the right half-space.

The existence of an interface mode implies that the condition in [definition 2](#) is satisfied, after the left or right side Bloch wave is multiplied by an appropriate constant, which is shown below.

Lemma 4 *The impedance function defined by*

$$Z(\omega) = \frac{e_{zTM}(1)}{e^{ika} - e_{zTE}(1)} \text{ for the } z\text{-polarized wave, and} \quad (29)$$

$$Z(\omega) = \frac{e_{yTM}(1)}{e^{ika} - e_{yTE}(1)} \text{ for the } y\text{-polarized wave} \quad (30)$$

enforces the continuity condition between the half-spaces by having $Z_L = Z_R$, and it is purely imaginary.

Proof. The continuity follows from [definition 3](#). The impedance is given by [equation 27](#): $Z(\omega) = \frac{e_{Bloch}(0)}{h_{Bloch}(0)} = \frac{e_{TX}(0)}{h_{TX}(0)}$. Having $Z_L = Z_R$ means that

$$\frac{e_{Bloch,L}(0)}{h_{Bloch,L}(0)} = \frac{e_{Bloch,R}(0)}{h_{Bloch,R}(0)}. \quad (31)$$

From [equation 31](#) it follows that $e_{Bloch,L}(0) = Ce_{Bloch,R}(0)$ and $h_{Bloch,L}(0) = Ch_{Bloch,R}(0)$, where $C \in \mathbb{C}$, and this constant can be set arbitrarily as any solution to the systems from [lemma 1](#) (system [equation 1](#) and [equation 2](#)) multiplied by a constant is still a solution.

From [lemma 3](#) we know that $\Im(e_{zTE}) = \Im(h_{yTM}) = 0$ and $\Re(e_{zTM}) = \Re(h_{yTE}) = 0$. Likewise, from [definition 3](#) we know that the impedance is the ratio of the components of the monodromy eigenvector. This requires finding an eigenvector of a 2×2 matrix:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} x \\ y \end{bmatrix} \quad (32)$$

$$\implies (\lambda - a)x = by, \quad (33)$$

$$(\lambda - d)y = cx. \quad (34)$$

Now [equation 33](#) and [equation 34](#) imply two eigenvectors:

$$v_1 = C_1 \begin{bmatrix} b \\ \lambda - a \end{bmatrix}, \text{ and } v_2 = C_2 \begin{bmatrix} \lambda - d \\ c \end{bmatrix} \quad (35)$$

Picking the first one and substituting $a = e_{xTM}(1)$ and $b = e_{xTE}(1)$ yields the eigenvector $\begin{bmatrix} e_{TX} \\ h_{TX} \end{bmatrix} = \begin{bmatrix} e_{zTM} \\ e^{ika} - e_{zTE} \end{bmatrix}$ or $\begin{bmatrix} e_{yTM} \\ e^{ika} - e_{yTE} \end{bmatrix}$ as desired. ■

Lemma 5 Given the monodromy $\mathbf{M}(\omega) \in \mathbb{C}^{2,2}$ of the system described by [equation 1](#) and [equation 2](#):

$$\det \mathbf{M}(\omega) = 1. \quad (36)$$

Proof. By direct differentiation and combining with any system from [lemma 1](#):

$$\begin{aligned} & \frac{\partial}{\partial x} \left(\det \begin{bmatrix} e_{TE} & e_{TM} \\ h_{TE} & h_{TM} \end{bmatrix} \right) \\ &= \frac{\partial}{\partial x} (e_{TE} h_{TM} - e_{TM} h_{TE}) \\ &= h_{TM} \frac{\partial}{\partial x} e_{TE} + e_{TE} \frac{\partial}{\partial x} h_{TM} - h_{TE} \frac{\partial}{\partial x} e_{TM} - e_{TM} \frac{\partial}{\partial x} h_{TE} \\ &= h_{TM} (\pm i \omega \mu h_{TE}) + e_{TE} (\pm i \omega \varepsilon e_{TM}) - h_{TE} (\pm i \omega \mu h_{TM}) - e_{TM} (\pm i \omega \varepsilon e_{TE}) \\ &= 0 \end{aligned}$$

And because $\det \mathbf{M}(\omega) = 1$ at $x = 0$, and its derivative is 0 for all x , it must be constant, $\det \mathbf{M}(\omega) = 1$. See also [lemma 9](#). ■

Lemma 6 Given the monodromy $\mathbf{M}(\omega) \in \mathbb{C}^{2,2}$ of the system described by [equation 1](#) and [equation 2](#):

$$\text{Tr } \mathbf{M}(\omega) \in \mathbb{R}. \quad (37)$$

Proof. From [lemma 3](#) we know that the diagonal entries of the system solution are real. Since the monodromy is the solution evaluated at the end of the cell, its diagonal is also real, and hence its trace is real. ■

Theorem 1 As a consequence of [lemma 5](#) and [lemma 6](#), the eigenvalues of the monodromy are given by

$$e^{\pm ika} = \frac{D(\omega) \pm \sqrt{D^2(\omega) - 4}}{2}, \quad (38)$$

where $D(\omega) = \text{Tr } \mathbf{M}(\omega)$.

Proof. The eigenvalue equation for a 2×2 matrix $\mathbf{M} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is $(\lambda - a)(\lambda - d) - bc = 0$. Rewriting, we get $\lambda^2 - (a + d)\lambda + ad - bc = \lambda^2 - \text{Tr } \mathbf{M} + \det \mathbf{M} = 0$. Setting $D(\omega) = \text{Tr } \mathbf{M}$ and knowing that, for the monodromy by [lemma 5](#), $\det \mathbf{M} = 1$, directly results in [theorem 1](#). ■

The result of [theorem 1](#) allows for sweeping a range of frequencies, finding the monodromy matrix as a solution to one of the systems in [lemma 1](#), and finding its eigenvalues without explicitly saving the Bloch modes.

Definition 5 If and only if the eigenvalues of the monodromy $e^{\pm ika} \in \mathbb{R}$ and $|e^{ika}| \neq 1$, then they belong to a band gap.

Lemma 7 *The off-diagonal entries of the monodromy $\mathbf{M}(\omega) \in \mathbb{C}^{2,2}$ of the system described by [equation 1](#) and [equation 2](#) are pure imaginary:*

$$\Re(\mathbf{M}(\omega)_{0,1}) = \Re(\mathbf{M}(\omega)_{1,0}) = 0. \quad (39)$$

Proof. From [lemma 3](#) we know that the off-diagonal entries of the system solution are imaginary. Since the monodromy is the solution evaluated at the end of the cell, its off-diagonal terms must also be imaginary. ■

The eigenvalues can easily be computed using [theorem 1](#), and used to find the band diagram by sweeping the frequency ω . This is shown in the listing below:

```

1 for i = 0 : N_ω do
2   ω = (ω_end - ω_start) / N_ω*i + ω_start
3
4   Matrix ML=systemZ(Npts,T_left,solutionL,ω,μ_left,ε_left)
5   Matrix MR=systemZ(Npts,T_right,solutionR,ω,μ_right,ε_right)
6
7   DL = (ML0,0 + ML1,1).real()
8   DR = (MR0,0 + MR1,1).real()
9
10  // right halfspace
11  if (abs(DR) < 2)
12     k+ = arccos(0.5*DR) // 0.5 * D is the real part of
13                        // a complex number with modulus 1
14  else
15     k+ = NaN
16  end
17
18  // left halfspace
19  if (abs(DL) < 2)
20     k- = -arccos(0.5*DL) // property of cos: cos(-x)=cos(x)
21  else
22     k- = NaN
23  end
24
25  // mutual bandgap
26  if (abs(DR) > 2 && abs(DL) > 2)
27     track_impedance(ω, ML, MR, DR, DL)
28  end
29
30  save(ω, k+, k-)
31 end

```

In the following subsection we will see why impedance is being tracked in the band gap, and how this is useful for finding a more accurate frequency at which the interface modes appear.

2.1.3 Secant Method

The reasoning behind using the secant search method to get a more precise value of the interface mode frequencies ω is shown here.

Lemma 8 *The Bloch impedance as defined in [definition 3](#) is pure imaginary in the band gap:*

$$\Re(Z(\omega)) = 0. \quad (40)$$

Proof. The term e_{TM} in the impedance equation in [lemma 4](#) is an off-diagonal term of the monodromy which is imaginary by [lemma 7](#), and the term e_{TE} is a diagonal term so it is real by the proof of [lemma 6](#). We are in the band gap, so by [definition 5](#), the eigenvalue $e^{ika} \in \mathbb{R}$. This means that the first entry of the eigenvector $\begin{bmatrix} e_{TM} \\ e^{ika} - e_{TE} \end{bmatrix}$ is imaginary and the second real. Their ratio, the impedance, is therefore imaginary. ■

Another result from the proof of [lemma 8](#) is the fact that electric field Bloch waves are imaginary, and magnetic field Bloch waves are real in the band gap.

Theorem 2 *Away from its poles, the real function $f : \mathbb{R} \rightarrow \mathbb{R}$*

$$f(\omega) = \Im(Z_R(\omega)) - \Im(Z_L(\omega)) \quad (41)$$

is monotonously decreasing or increasing with ω in the band gaps.

Proof. This proof is the electromagnetic analog of Lemma 4.2 by Coutant and Lombard [4]. Since the impedance depends only on electric fields, the electric field by [lemma 4](#), the Helmholtz [equation 26](#) from the system of [equation 1](#) and [equation 2](#) can be used:

$$-\frac{\partial}{\partial x} \left(\frac{1}{\mu(x)} \frac{\partial}{\partial x} e_z \right) - \omega^2 \varepsilon(x) e_z = 0 \quad (26)$$

Differentiating this with respect to ω and dividing by $i\omega$ yields

$$\frac{\partial}{\partial x} \left(\frac{1}{i\omega\mu(x)} \frac{\partial}{\partial x} \frac{\partial e_z}{\partial \omega} \right) = 2i\varepsilon(x)e_z + i\omega\varepsilon(x) \frac{\partial e_z}{\partial \omega} \quad (42)$$

The dependencies on x will now be removed and the derivatives replaced by $\frac{\partial}{\partial x} e_z = e'_z$ for readability.

From [definition 3](#) and [equation 1](#) we have the impedance derivative

$$\frac{\partial \Im\{Z_R\}}{\partial \omega} = \frac{\partial}{\partial \omega} \Im \left\{ \frac{e(0^+)}{\frac{1}{i\omega\mu} e'(0^+)} \right\} = \frac{\partial}{\partial \omega} \left(\frac{e(0^+)}{\frac{1}{\omega\mu} e'(0^+)} \right) \quad (43)$$

Now:

$$\frac{\partial \Im\{Z_R\}}{\partial \omega} = \left(\frac{e}{\frac{1}{\omega\mu} e'} \right)' = \frac{\frac{1}{i\omega\mu} \left(\frac{ee'}{\omega} + e' \frac{\partial e}{\partial \omega} - e \frac{\partial e'}{\partial \omega} \right)}{\frac{1}{i(\omega\mu)^2} (e')^2} \quad (44)$$

$$= \frac{\mathcal{W}}{\frac{1}{i(\omega\mu)^2} (e')^2}, \quad (45)$$

where \mathcal{W} is the analog of the Wronskian in [4].

This is now differentiated with respect to x :

$$\mathcal{W}' = \frac{\partial}{\partial x} \left[\frac{1}{i\omega\mu} \left(\frac{ee'}{\omega} + e' \frac{\partial e}{\partial \omega} - e \frac{\partial e'}{\partial \omega} \right) \right] \quad (46)$$

$$= \frac{\partial}{\partial x} \left(\frac{1}{i\omega\mu} \frac{ee'}{\omega} \right) + \frac{\partial}{\partial x} \left(e' \frac{\partial e}{\partial \omega} - e \frac{\partial e'}{\partial \omega} \right) \quad (47)$$

The first term can be rewritten with the help of [equation 26](#) as

$$\frac{\partial}{\partial x} \left(\frac{1}{i\omega\mu} \frac{ee'}{\omega} \right) = \left(\frac{1}{i\omega\mu} e' \right)' \frac{e}{\omega} + \frac{e'}{\omega} \left(\frac{1}{i\omega\mu} e' \right) \quad (48)$$

$$= i\varepsilon e^2 - \frac{i}{\omega^2\mu} (e')^2 \quad (49)$$

The second term, with substitutions from [equation 26](#) and [equation 42](#) is

$$\frac{\partial}{\partial x} \left(e' \frac{\partial e}{\partial \omega} - e \frac{\partial e'}{\partial \omega} \right) = \frac{\partial e}{\partial \omega} \left(\frac{1}{i\omega\mu} e' \right)' - e \left(\frac{1}{i\omega\mu} \frac{\partial e'}{\partial \omega} \right)' \quad (50)$$

$$= i\omega\varepsilon e \frac{\partial e}{\partial \omega} - \left(2i\omega\varepsilon e + i\omega\varepsilon \frac{\partial e}{\partial \omega} \right) e \quad (51)$$

$$= -2i\varepsilon e^2. \quad (52)$$

Combining [equation 49](#) and [equation 52](#), we get that

$$\mathcal{W}' = i\varepsilon e^2 - \frac{i}{\omega^2\mu} (e')^2 - 2i\varepsilon e^2 \quad (53)$$

$$= -\frac{i}{\omega^2\mu} (e')^2 - i\varepsilon e^2 \quad (54)$$

Following the analogy, we can integrate the imaginary (and only) part of this Wronskian derivative with respect to x from 0 to $+\infty$ and see that $\mathcal{W}(+\infty) = 0$ because of the evanescent field:

$$\Im\{\mathcal{W}\} = \Im\{\mathcal{W}(+\infty) - \mathcal{W}(0)\} = -\mathcal{W}(0^+) = \int_0^\infty \Im\{\mathcal{W}'(x, \omega)\} dx = - \int_0^\infty \frac{1}{\omega^2\mu} (e')^2 + \varepsilon e^2 dx < 0, \quad (55)$$

since μ and ε are both positive, so $\Im\{\mathcal{W}(0^+)\} > 0$. This means that the imaginary constants cancel in the fraction with the Wronskian and the denominator of the derivative

$$\frac{\partial \Im\{Z_R\}}{\partial \omega}(0^+) = \frac{\mathcal{W}(0^+, \omega)}{i(\omega\mu)^2 (e')^2} = -\frac{\Im\{\mathcal{W}(0^+, \omega)\}}{(\omega\mu)^2 (e')^2} < 0, \quad \forall \omega. \quad (56)$$

Since the derivative of the imaginary part of the right side impedance is always negative, it means it is monotonically decreasing. A similar conclusion can be reached for the left side impedance, except this time, the integration is from $-\infty$ to 0 which flips the sign so it is monotonically increasing.

Subtracting a monotonically increasing function from a monotonically decreasing function will result in a monotonically decreasing function. ■

As a consequence of [theorem 2](#), we are allowed to use a zero-finding algorithm in the band gaps in the vicinity of points that cross the ω -axis. For this project, the secant method was used. Pseudocode for the secant method is given in the listing below:

```

1 function secant(f, x0, x1, maxiter, tol)
2   for i = 0 : maxiter do
3     f_at1 = f(x1)
4     if (abs(f_at1) < tol)
5       print("Converged with " + i + " iterations.")
6       return (x1, f_at1)
7     else
8       x2 = x1 - f_at1 * (x1 - x0) / (f_at1 - f(x0))
9       x0 = x1
10      x1 = x2
11    end
12  end
13  print("Maxiter reached!")
14 return (x1, f_at1)

```

Now we are equipped to properly track impedance. The "track_impedance()" function from the listing about computing the band diagram is calculating the impedance using the expressions from [lemma 4](#) which require the monodromy matrix and its eigenvalues only. This function works together with the function which is tracking bands and band gaps. The idea is to calculate the impedance and save its sign. If the sign changes at some frequency, that frequency is saved to an array. The following pseudocode illustrates the idea behind the impedance sign and band tracking. The idea is, if the current frequency is in a band gap, and the previous one was not, then a band gap was just entered. Likewise, if the current frequency is on a band, and the previous one was not, then it just entered a band, etc. This is also helpful to know when to start and end the Zak phase calculations, for example.

If the impedance in the band gap is negative, it is assigned a -1, and if it is positive a +1. On bands, the impedance is assigned a 0 to avoid spurious interface modes and ensure a proper sign initialization when entering the band gap. If the previous "oldsign" and current "newsign" give a negative number when multiplied, that registers as a sign change.

```

1 for i = 0 : N_ω do
2   ω = (ω_end - ω_start) / N_ω*i + ω_start
3
4   array = []

```

```

5      oldsign = 0 // avoid band edge and spurious modes
6
7      bandgapL = false
8      bandgap_newL = false
9      bandgapR = false
10     bandgap_newR = false
11
12     ..calculating ML, MR etc..
13
14     // right halfspace -----
15
16     // we are on a right band
17     if (abs(DR) < 2)
18         bandgap_newR = false
19         ..eigenvalue and other calculations..
20
21         if (bandgap_newR != bandgapR)
22             // we are on a new band
23             ..calculations..
24         end
25     else
26         bandgap_newR = true
27         if (bandgap_newR != bandgapR)
28             // we just entered a band gap
29             ..calculations..
30             oldsign = 0
31         end
32     end
33
34     bandgapR = bandgap_newR
35
36     // left halfspace -----
37
38     // we are on a left band
39     if (abs(DL) < 2)
40         bandgap_newL = false
41         ..eigenvalue and other calculations..
42
43         if (bandgap_newL != bandgapL)
44             // we are on a new band
45             ..calculations..
46         end
47     else
48         bandgap_newL = true
49         if (bandgap_newL != bandgapL)
50             // we just entered a band gap

```

```

51         .. calculations ..
52         oldsign = 0
53     end
54 end
55
56 bandgapL = bandgap_newL
57
58 // mutual bandgap -----
59
60 if (abs(DR) > 2 && abs(DL) > 2)
61     .. find impedance and track the sign ..
62     // the sign can change if Z_R - Z_L
63     // encounters a singularity
64     newsign = sign(Z_R - Z_L)
65     if (oldsign * newsign < 0)
66         print("Change of sign at " + ω)
67         array.push(ω)
68     end
69 end
70
71 .. other code ..
72 end

```

After this part of the program is run, we have an array filled with potential interface mode frequencies. We already know that the step in the loop was $\Delta\omega = \frac{\omega_{\text{end}} - \omega_{\text{start}}}{N_\omega}$ so the interval that will be provided to the secant is simply $[\omega - \Delta\omega, \omega]$.

The function that is provided to the secant method is a functor that numerically finds the monodromy matrices at the provided frequency and calculates the impedance difference:

$$f(\omega) : \omega \rightarrow Z_R - Z_L \quad (57)$$

and this function is the function of which the zero needs to be found. Before relaying further calculations to the secant method, it is recommended to set the desired meshwidth for the plot and adjust the Aitken-Neville algorithm (explained below in Extrapolation to Zero) tolerances accordingly, since these parameters will be applied to $f(\omega)$.

2.1.4 Extrapolation to Zero

The monodromy matrix found in the above sections will satisfy [lemma 5](#) because of energy preservation. However, in Deuffhard and Bornemann's book [\[7\]](#), the energy conservation can be traded for higher precision since for reversible timestepping methods, one of which is the implicit midpoint rule:

Theorem 4.42 (Deuffhard, Bornemann, p. 167, eBook: 6 December 2012)

Let Ψ be a reversible discrete evolution. Then there exists a sequence e_0, e_1, e_2, \dots of smooth functions with initial values $e_k(t_0) =$ such that for each $k \in \mathbb{N}_0$, the asymptotic

expansion in even powers of τ ,

$$x_\tau(t) = x(t) + e_0(t)\tau^{2q} + \dots + e_{k-1}\tau^{2(q+k-1)} + \mathcal{O}(\tau^{2(q+k)}) \quad (58)$$

Here, the steps are in space, but the conclusion holds: the error of the calculation with the implicit midpoint rule is an even polynomial in h , specifically $\mathcal{O}(h^{2p+1})$ where p is the number of monodromy matrices calculated for a series of steps. The steps h were created by halving $h_i = h_0/2^i$.

The Aitken-Neville scheme was used to extrapolate the monodromy polynomial in h to $h \rightarrow 0$, with predefined tolerance. The above conclusions lead to the fact that instead of using the original steps in the Aitken-Neville scheme, their squares can be used. This means that for using h_n we will get the same accuracy as if we used h_n^2 with a regular Aitken-Neville scheme.

A pseudocode similar to Hiptmair [6] code 5.2.3.19. is provided below. Note that the **square** of the meshwidth is used instead of the meshwidth in the standard Aitken-Neville scheme.

```

1 function monodromyAtZero(N0, T, maxiter, tol,  $\omega$ ,  $\mu$ ,  $\varepsilon$ )
2   Vector h(maxiter)
3   h(0) = 1/N0
4   Matrix Y(4,maxiter)
5   N = N0
6
7   // monodromyZ() is the same as systemZ() except it
8   // does not save the full solution as we do not need it
9   Y.col(0) = monodromyZ(N, T,  $\omega$ ,  $\mu$ ,  $\varepsilon$ ).reshaped(4,1)
10
11  for i = 1 : maxiter - 1 do
12    h(i) = 0.5 * h(i-1)
13    N *= 2
14    Y.col(i) = monodromyZ(N, T,  $\omega$ ,  $\mu$ ,  $\varepsilon$ ).reshaped(4,1)
15
16    for k = i - 1 : -1 : 0 do // stepping backwards
17      Y.col(k) = Y.col(k + 1) - (Y.col(k + 1) - Y.col(k))
18      *h(i)*h(i) / (h(i)*h(i) - h(k)*h(k))
19    end
20
21    if ((Y.col(1) - Y.col(0)).norm() < tol)
22      print("Extrapolation to 0 converged with -"
23          + i + "-iterations.")
24      break
25    end
26  end
27 return Y.col(0).reshaped(2,2)

```


2.2 Errors and Convergence

The best error and convergence test case is the interface mode at $\omega = 15.9449$ ($f = 2.5377$) for both varying ε and μ . See figure 6 and equation 64. This mode is topologically protected according to table 3 and it is quite close to the edge of the band. It will be the first interface mode to be missed by the program if errors get too high. The theoretical expectation is to obtain identical results with respect to h set by Aitken-Neville and $\Delta\omega$ since the results are limited by predefined Aitken-Neville (AN) and secant method (SM) maximum iterations and tolerances. The interface mode in question can be seen in figure 1.

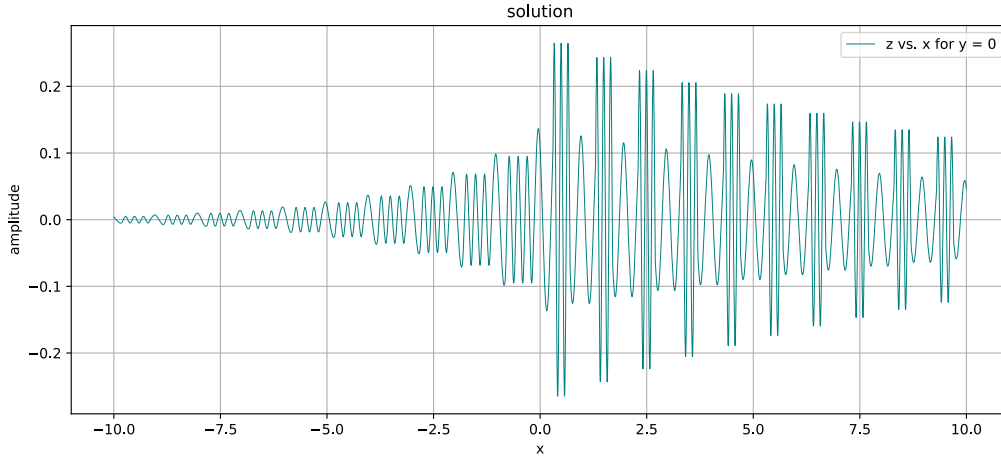


Figure 1: Electric field of the interface mode at $\omega = 15.9449$

2.2.1 Aitken-Neville Results for Fixed SM and Tolerance (Implicit Midpoint Rule)

The secant method was used with tolerance 10^{-12} and maximum number of iterations 100, with its Aitken-Neville parameters of 14 for maximum iterations, 10^{-12} for absolute tolerance, and $N_0 = 40$. There were 10 000 ω -steps in the range $\omega : 0 - 20$. The sweep Aitken-Neville tolerance was set to 10^{-6} . This is without explicit computation of the interface mode. The result is the following:

N0	ω , maxiter=5	ω , maxiter=8
8	miss	1.594490e+01
10	1.594490e+01 slows secant	1.594490e+01
20	1.594490e+01	1.594490e+01
40	1.594490e+01	1.594490e+01
80	1.594490e+01	1.594490e+01

Table 1: Interface mode frequency vs. initial number of steps, for maxiter = 10, 11 and sweep AN tolerance 10^{-6}

In [table 1](#) only maxiter 5 and 8 were shown since the table looks almost identical to 5 for maxiters 5-7, and the first one for which all are correct is maxiter 8 and above. Stepping that produces intervals to be searched without any interface modes is marked as "slows secant" since it will cause it to break the 100 iterations, thus slowing the program. We obtain nearly identical results as expected since they are limited by the secant method and its Aitken-Neville parameters.

2.2.2 Omega Stepping Results for Fixed AN/SM

Just like above, the secant method was used with tolerance 10^{-12} and maximum number of iterations 100, with its Aitken-Neville parameters of 14 for maximum iterations, 10^{-12} for absolute tolerance, and $N_0 = 40$. The Aitken-Neville algorithm parameters for the sweep are default: tolerance of 10^{-6} , 10 for maximum iterations, and the starting number of points $N_0 = 20$. The range was kept to be $\omega : 0 - 20$. Again, this is without explicit computation of the interface mode.

The first miss occurs around $N_\omega = 300$. At and above 400, all interface modes are found, and the most difficult one attains the same value as those in [table 1](#).

2.2.3 Secant Method and Time

In both results above, if an interface mode exists, it converges in less than 10 iterations. The only times when it went above 10 were when it missed an interface mode or it searched for a non-existent interface mode. The run with highest accuracy took 1473 milliseconds, and the successful run with "least" accuracy 29 milliseconds on Intel(R) Core(TM) i7-10875H CPU @ 2.30GHz. The entire [table 1](#) took less than 10 seconds per run. The graphs are drawn with 40 000 points per cell (with Python's automatic choice from these points to plot). Running the secant with lower tolerances (10^{-13} and Secant AN 10^{-15} , maximum AN iterations 15, everything else the same) produces 15.94489701693366, while the table above has values around 15.94489701693407. They differ in the last 3 of the 14 decimals as expected since the secant parameters for [table 1](#) set the errors to show the 12th decimal. Stricter secant tolerances will not converge with these parameters.

2.2.4 Drawing (Implicit Midpoint Rule) Errors

These errors are the error of the monodromy, last entry, and full solution matrix without the Aitken-Neville scheme in [figure 2](#). It shows second order convergence for the implicit midpoint rule and also justifies the use of the Aitken-Neville scheme. The total solution matrix is a $4 \times N$ matrix, which is not computed using the Aitken-Neville scheme. The reference monodromy matrix is the one with parameters: max iterations 14, tolerance 10^{-12} , and starting $N = 40$. The errors were plotted for 100, 200, 400, ..., 409600, 819200 points in the systemZ function, which uses the implicit midpoint rule only, for the left side of the $\omega = 15.9449$ interface mode. The reference solution had a million points. The entire error computation took 9741 milliseconds.

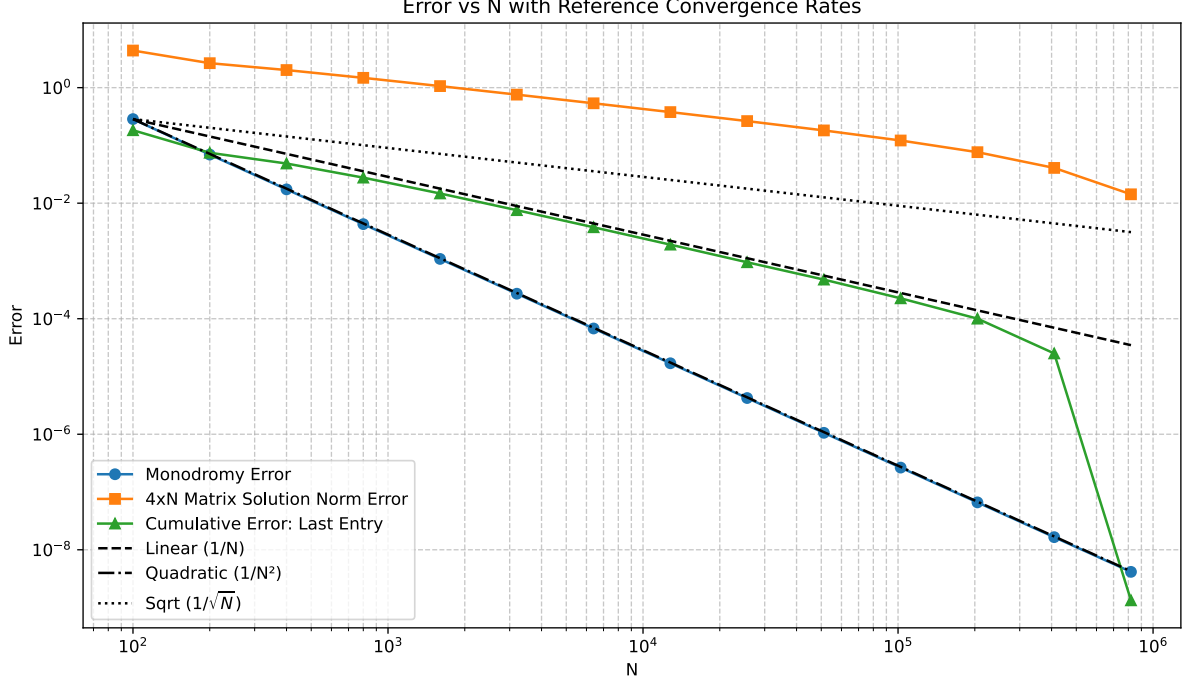


Figure 2: Error of the systemZ function without the Aitken-Neville scheme, used to compute the full solution

2.3 Results

2.3.1 Inversion-Symmetric Interface Modes

This subsection is mainly dedicated to comparisons with results from the literature Xiao, Zhang, Chan [2] and Kalozoumis [5].

Here the spatial distribution is inversion symmetric:

$$\varepsilon(x) = \varepsilon(a - x) \quad (59)$$

$$\mu(x) = \mu(a - x) \quad (60)$$

The structure is depicted on the left side in [figure 3](#).

The permeability is a constant 1 in the first example, and the permittivity function in the normalized unit cell for achieving a Dirac point (band crossing point) is

$$\varepsilon(x) = \begin{cases} 4 & \text{for } x < 0.2 \text{ or } x > 0.8, \\ 1 & \text{else.} \end{cases} \quad (61)$$

Opening the Dirac point with a perturbation like [5] will cause an interface mode to appear (green line):



Figure 3: Inversion-symmetric and non-symmetric unit cell

$$\varepsilon_{left}(x) = \begin{cases} 1.95^2 & \text{for } x < \frac{0.2}{0.95} \text{ or } x > 1 - \frac{0.2}{0.95}, \\ 1 & \text{else.} \end{cases} \quad (62)$$

$$\varepsilon_{right}(x) = \begin{cases} 2.05^2 & \text{for } x < \frac{0.2}{1.05} \text{ or } x > 1 - \frac{0.2}{1.05}, \\ 1 & \text{else.} \end{cases} \quad (63)$$

Material parameters from [equation 62](#) and [equation 63](#) create spaces with the band diagrams shown in [figure 4](#). These band diagram can be seen in [2], [5] and [4].

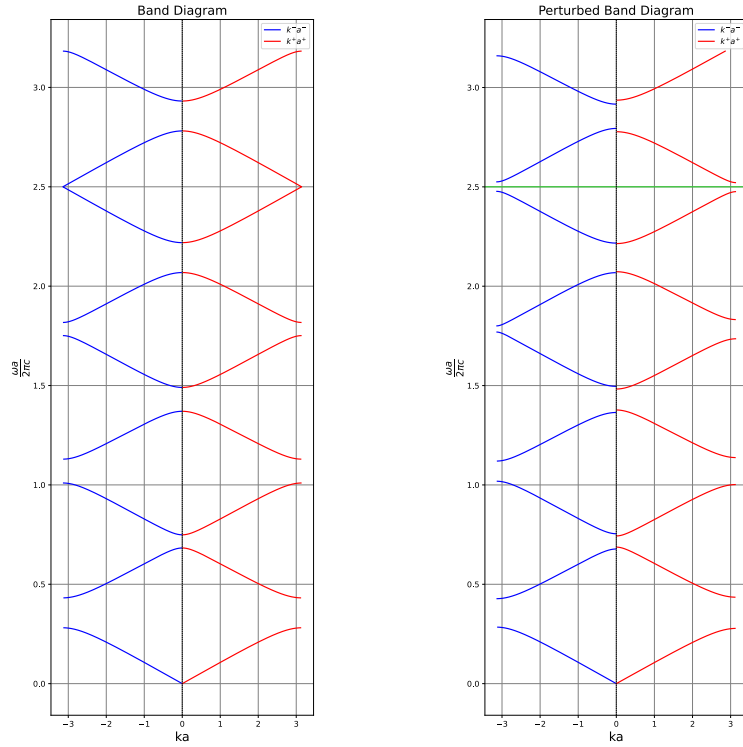


Figure 4: Dirac point space and perturbed space with interface mode (green)

The electric field of the expected $\omega \approx 5\pi$ or $f \approx 2.5$ interface mode is shown in [figure 5](#). The same interface mode was shown in [1], and mentioned in [2], [5]. The interface is at $x = 0$.

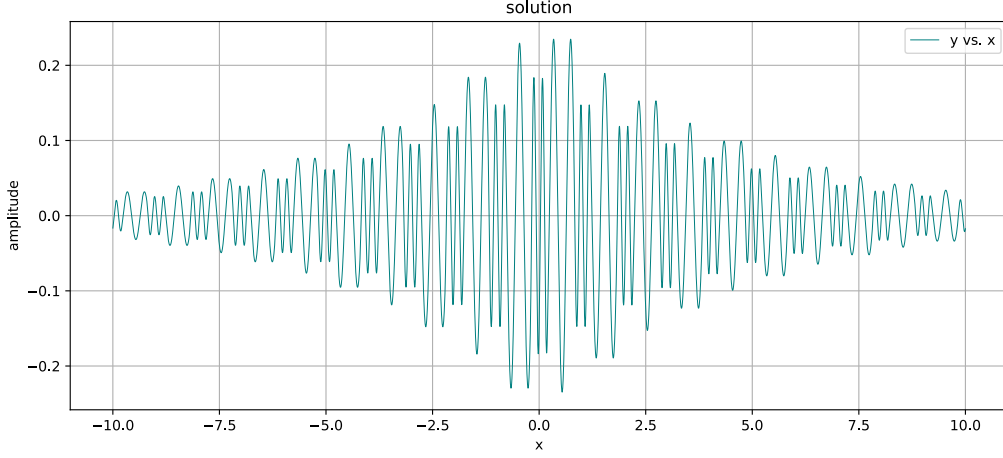


Figure 5: Electric field of the interface mode at $\omega = 1.0001 \cdot 5\pi$

Another configuration from [2] was tested:

$$\begin{aligned} \varepsilon_{\text{left}}(x) &= \begin{cases} 1 & \text{for } x < 0.175 \text{ or } x > 0.825, \\ 3.5 & \text{else.} \end{cases} \\ \mu_{\text{right}}(x) &= \begin{cases} 1 & \text{for } x < 0.3 \text{ or } x > 0.7, \\ 6 & \text{else.} \end{cases} \end{aligned} \quad (64)$$

with $\mu_{\text{left}} = \varepsilon_{\text{right}} = 1$. The configuration from [equation 64](#) has interface modes in the first, second, and fifth band gaps according to [2], and the interface modes in [figure 6](#) comply with this, and an extra interface mode that was not included in their frequency range (see [figure 1](#)). The transmission coefficient was calculated for 10 unit cells of each half-space.

The continuous case mentioned shifting the boundary to $z = 0$. It appears that the boundary begins at the center of the left half-space unit cell (shift by π). Due to this, the permittivity values adapted to the coordinate system in this report are

$$\begin{aligned} \varepsilon_{\text{left}}(x) &= 12 + 6 \sin[2\pi(x/a + 0.25) + \pi] \\ \varepsilon_{\text{right}}(x) &= 12 + 5 \sin[2\pi(x/a - 0.25)] + 5 \sin[4\pi(x/a + 0.125)] \end{aligned} \quad (65)$$

In [figure 7](#) created by [equation 65](#) interface modes appear in the second and third gap as denoted in [2], and the transmission coefficient approximation seems to roughly agree with the finite calculation in [2]. The transmission coefficient was calculated with 20 unit cells from the left and 10 unit cells from the right side.

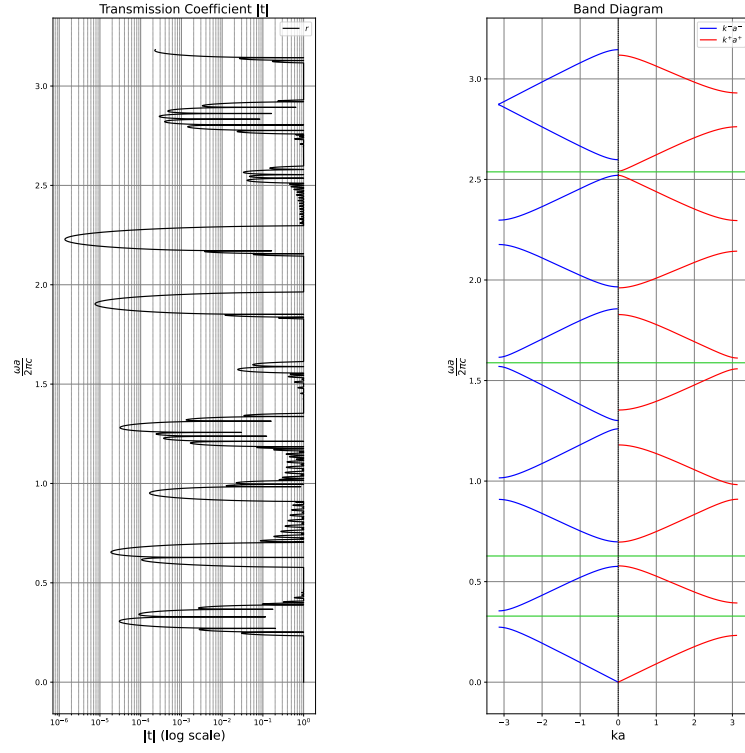


Figure 6: Approximate transmission coefficient and band diagram for varying ϵ and μ

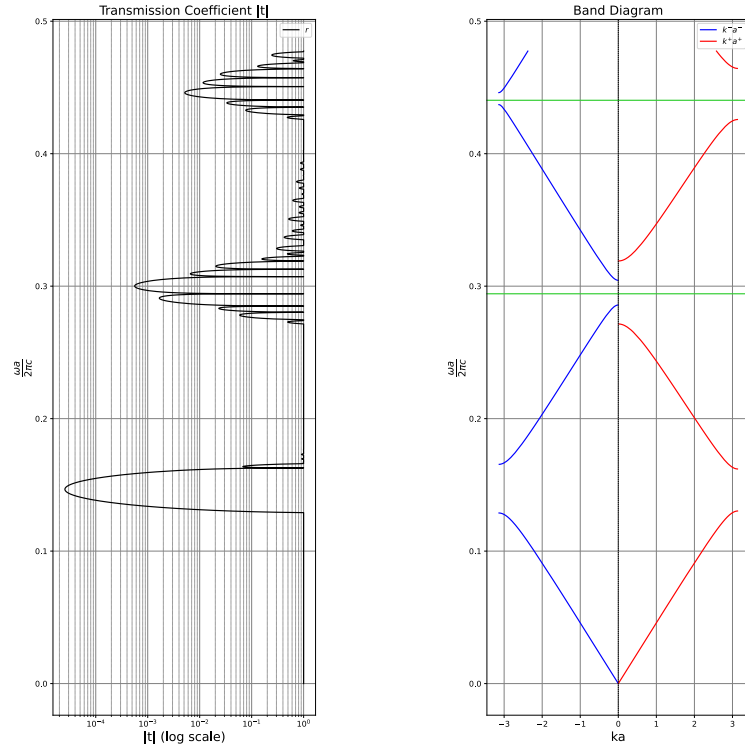


Figure 7: Approximate Transmission Coefficient and Band Diagram for Continuously Varying ϵ

Two joint spaces with different period values were also included. The parameters are the same as for [figure 6](#), but the left half-space has period 2 instead of 1 (note that the period in a functor must be set to 2, to match the space period). The results agree with those in [\[2\]](#).

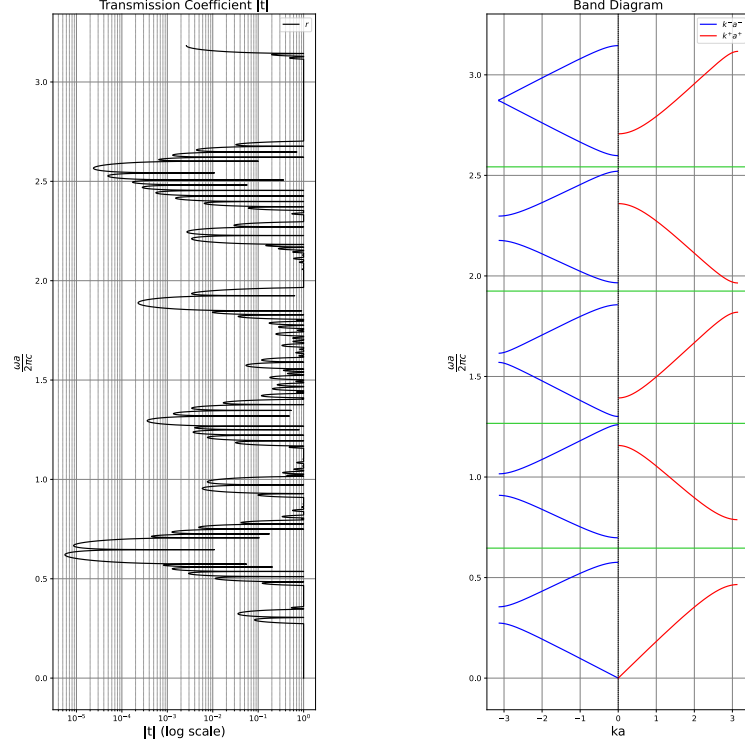


Figure 8: Approximate transmission coefficient and band diagram for varying ε and μ for period 2 and 1 at the left and right side respectively (the left space is the reference, hence no unitless frequency change)

2.3.2 Non-Inversion-Symmetric Interface Modes

A selection of interface modes for a non-symmetric space given by $\mu = 1$, [equation 66](#) and [equation 67](#) is provided in this section. Both electric and magnetic field solutions are provided.

$$\varepsilon_{left} = \begin{cases} 1 & \text{for } x > \frac{0.4}{0.95}, \\ 3.8025 & \text{otherwise.} \end{cases} \quad (66)$$

$$\varepsilon_{left} = \begin{cases} 1 & \text{for } x < 1 - \frac{0.4}{1.05}, \\ 4.2025 & \text{otherwise.} \end{cases} \quad (67)$$

Note that the 4 unit cells defined in [equation 62](#), [equation 63](#), [equation 66](#), and [equation 67](#) tile the same infinite spaces, up to mirroring. The band diagrams in this space look exactly the same as the ones in the perturbed space of [figure 4](#). They are shown next to each other in [figure 9](#).

Parameters: 10 000 drawing points, 40 000 frequency points, default sweep Aitken-Neville, secant Aitken-Neville 14 maximum iterations, tolerance 10^{-12} , and 40 starting points. Secant method tolerance 10^{-12} and 20 maximum iterations. Interestingly, the secant method succeeds in finding some modes multiple times from points further away.

Unfortunately, no non-symmetric interface mode representations were found in the literature so they currently have no reference to compare with.

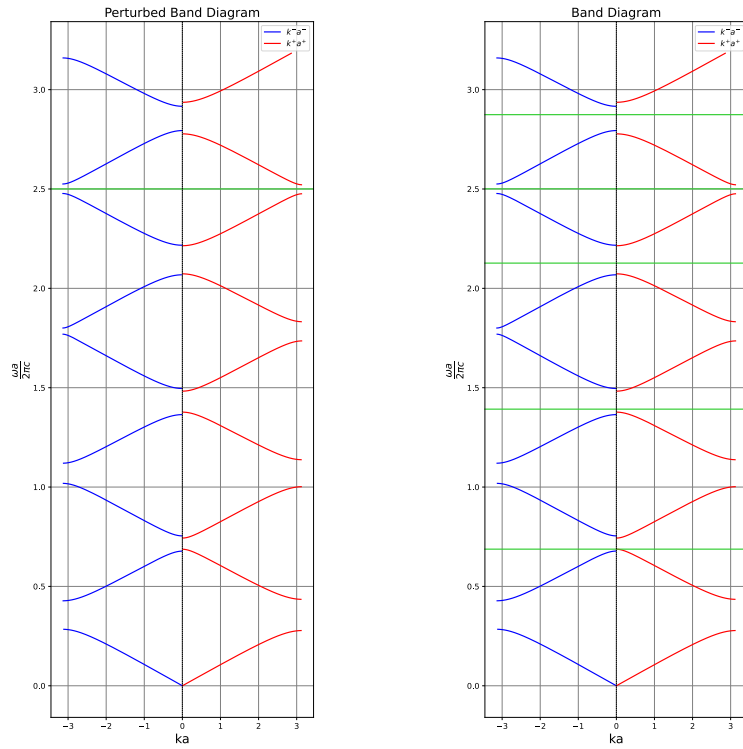


Figure 9: Perturbed space from [figure 4](#) and the non-symmetric space diagram

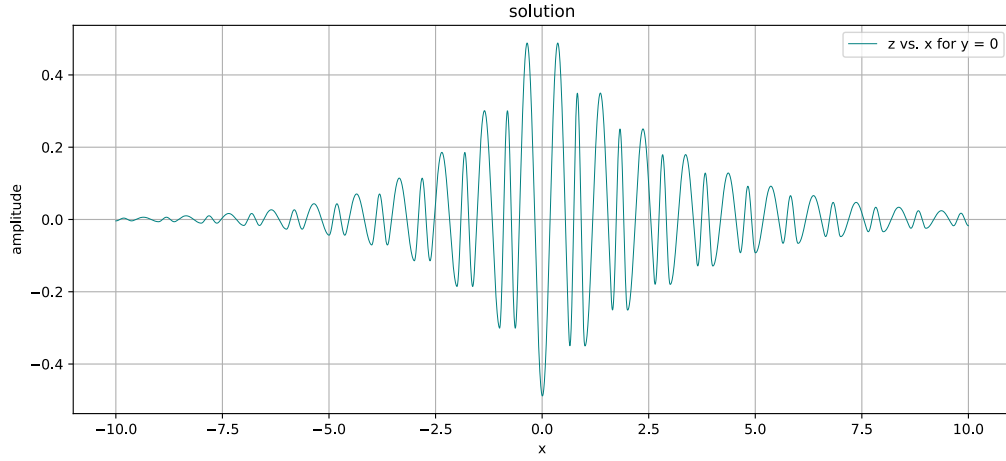


Figure 10: Electric field of the interface mode at $\omega = 8.7467$

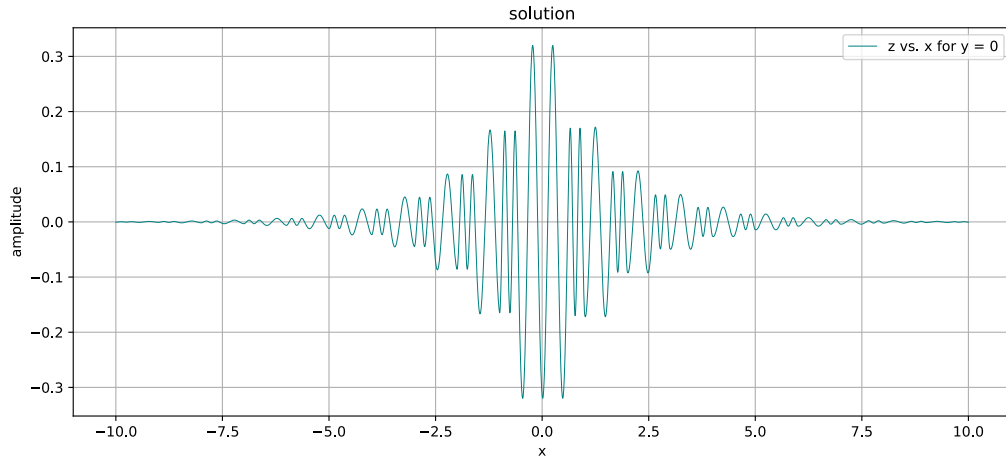


Figure 11: Electric field of the interface mode at $\omega = 13.3644$

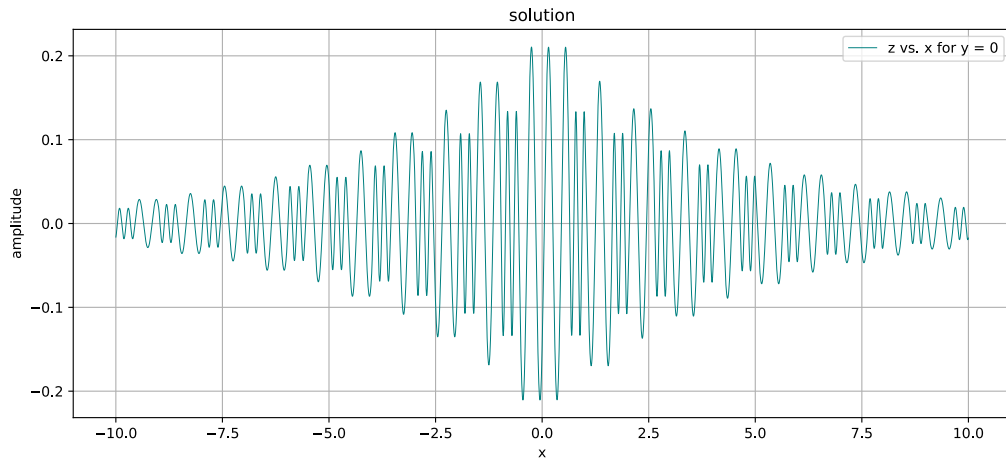


Figure 12: Electric field of the interface mode at $\omega = 15.7120$

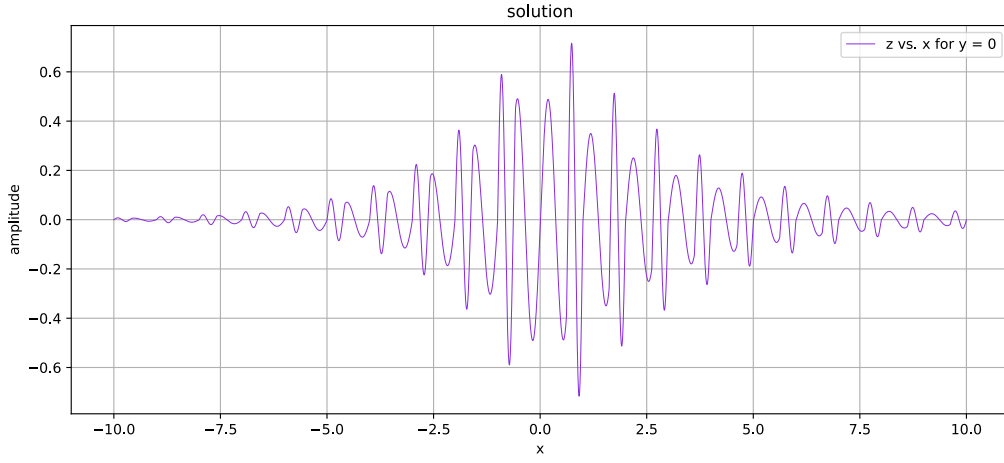


Figure 13: Magnetic field of the interface mode at $\omega = 8.7467$

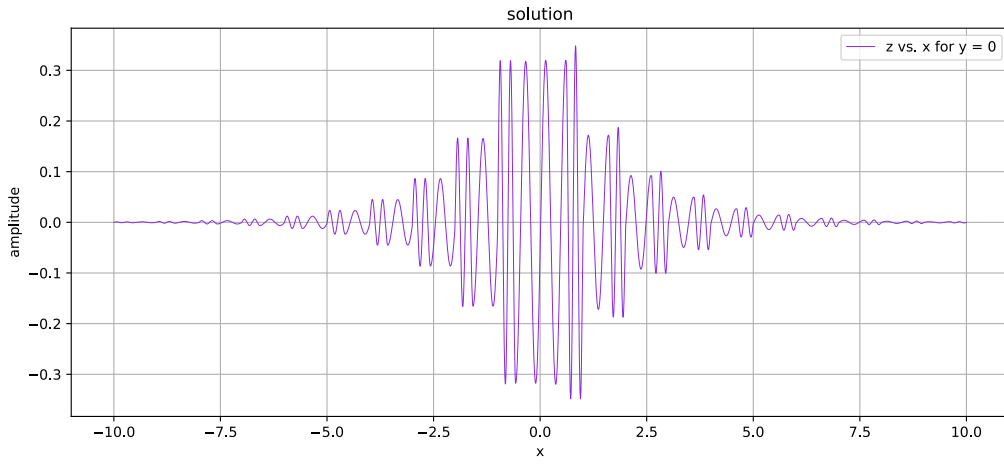


Figure 14: Magnetic field of the interface mode at $\omega = 13.3644$

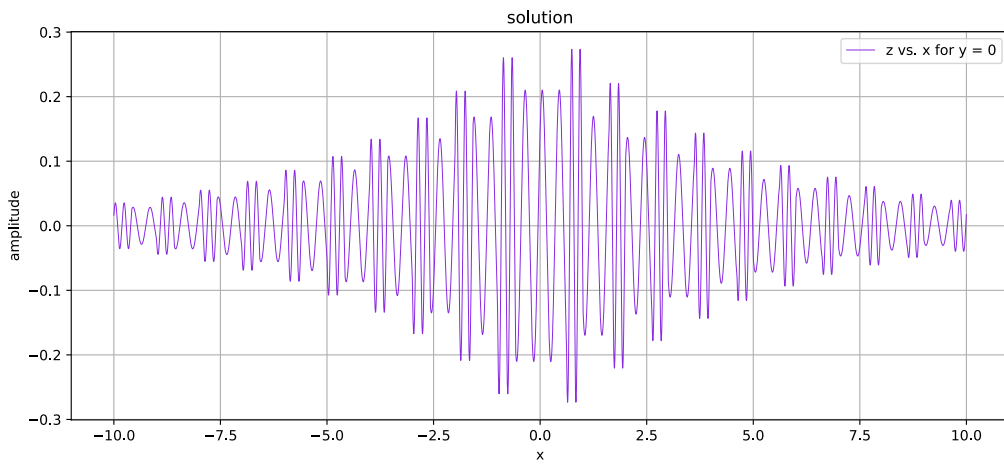


Figure 15: Magnetic field of the interface mode at $\omega = 15.7120$

3 Interface Modes in 1.5D

3.1 Review - Infinite Stripes in 2D

The problem of this subsection is almost identical to the one in 1D, except that now the y -axis is included so the space looks like a chain of periodic stripes infinite in the y -direction.

First, we pick a transversal electric (TE) or transversal magnetic (TM) solution. For TE, the electric field vector is confined to the z -axis and for TM, the magnetic field is confined to the z -axis.

For TE, the electric field will only have a z -component, and magnetic fields will be confined to the xy plane:

$$\mathbf{e}(x, y) = \begin{bmatrix} 0 \\ 0 \\ e_z(x, y) \end{bmatrix}, \quad \mathbf{h}(x, y) = \begin{bmatrix} h_x(x, y) \\ h_y(x, y) \\ 0 \end{bmatrix} \quad (68)$$

Because we have infinite stripes for which the medium parameters are uniform in the y direction, we can assume that the electric field can be separated into the following product

$$e_z(x, y) = e_z^x(x) e_z^y(y), \quad (69)$$

where

$$e_z^y(y) = \exp(ik_y y). \quad (70)$$

Together with [equation 1](#) and [equation 2](#), this yields a second order equation for $e_z^x(x)$ dependent only on x :

$$-\frac{\partial}{\partial x} \left(\frac{1}{\mu(x)} \frac{\partial e_z^x(x)}{\partial x} \right) - \left(\omega^2 \varepsilon(x) - \frac{k_y^2}{\mu(x)} \right) e_z^x(x) = 0. \quad (71)$$

This equation is almost the same as [equation 26](#), except for the extra $\frac{k_y^2}{\mu(x)}$ term. This means that almost all of the methods discussed for the 1D problem apply to this modified system, except [lemma 5](#) and [theorem 2](#), so new ones are included below.

The only significant differences would be the calculations with real numbers rather than complex, the impedances, and the monodromy matrix system, which now depends on $\psi = e_z^x(x)$ and $\psi' = \frac{1}{\mu(x)} \frac{\partial e_z^x(x)}{\partial x}$ instead of \mathbf{h} :

$$\frac{\partial}{\partial x} \begin{bmatrix} \psi'_1 & \psi'_2 \\ \psi_1 & \psi_2 \end{bmatrix} = \begin{bmatrix} 0 & \frac{k_y^2}{\mu(x)} - \omega^2 \varepsilon(x) \\ \mu(x) & 0 \end{bmatrix} \begin{bmatrix} \psi'_1 & \psi'_2 \\ \psi_1 & \psi_2 \end{bmatrix} \quad (72)$$

As a result, the band diagrams and interface mode frequencies shift depending on the choice of k_y . When $k_y = 0$, the problem reduces to 1D. The space chosen is the same as the one given by [equation 64](#).

Lemma 9 *Given the 1.5D monodromy $\mathbf{M}(\omega) \in \mathbb{C}^{2,2}$ of the system described by [equation 1](#) and [equation 2](#):*

$$\det \mathbf{M}(\omega) = 1. \quad (73)$$

Proof. For the TE system: by direct differentiation and combining with [equation 71](#):

$$\begin{aligned}
& \frac{\partial}{\partial x} \left(\det \begin{bmatrix} \psi'_1 & \psi'_2 \\ \psi_1 & \psi_2 \end{bmatrix} \right) \\
&= \frac{\partial}{\partial x} (\psi'_1 \psi_2 - \psi_1 \psi'_2) \\
&= \psi''_1 \psi_2 + \cancel{\psi'_1 \psi'_2} - \cancel{\psi'_1 \psi'_2} - \psi_1 \psi''_2 \\
&= \left(-\omega^2 \varepsilon(x) + \frac{k_y^2}{\mu(x)} \right) \psi_1 \psi_2 - \psi_1 \left(-\omega^2 \varepsilon(x) + \frac{k_y^2}{\mu(x)} \right) \psi_2 \\
&= 0
\end{aligned}$$

And because $\det \mathbf{M}(\omega) = 1$ at $x = 0$, and its derivative is 0 for all x , it must be constant, $\det \mathbf{M}(\omega) = 1$. ■

An identical procedure can be done for magnetic fields, yielding

$$-\frac{\partial}{\partial x} \left(\frac{1}{\varepsilon(x)} \frac{\partial h_z^x(x)}{\partial x} \right) - \left(\omega^2 \mu(x) - \frac{k_y^2}{\varepsilon(x)} \right) h_z^x(x) = 0, \quad (74)$$

and

$$\frac{\partial}{\partial x} \begin{bmatrix} \psi'_1 & \psi'_2 \\ \psi_1 & \psi_2 \end{bmatrix} = \begin{bmatrix} 0 & \frac{k_y^2}{\varepsilon(x)} - \omega^2 \mu(x) \\ \varepsilon(x) & 0 \end{bmatrix} \begin{bmatrix} \psi'_1 & \psi'_2 \\ \psi_1 & \psi_2 \end{bmatrix}. \quad (75)$$

Definition 6 *The TE impedance is defined as*

$$Z_{TE} = \frac{e_z^x}{\frac{1}{\mu} e_z^x}, \quad (76)$$

and the TM impedance is defined as

$$Z_{TM} = \frac{h_z^x}{\frac{1}{\varepsilon} h_z^x}, \quad (77)$$

Theorem 3 *Both the TE and TM impedances are monotonously decreasing functions on the right side, and increasing functions on the left side.*

Proof. This proof is very similar to that of [theorem 2](#). A part of the proof for right-side TE is provided below. TM is analogous. First, we differentiate [equation 71](#) with respect to ω :

$$-\frac{\partial}{\partial x} \left(\frac{1}{\mu(x)} \frac{\partial}{\partial x} \frac{\partial e_z^x}{\partial \omega} \right) - 2\omega \varepsilon(x) e_z^x - \left(\omega^2 \varepsilon(x) - \frac{k^2}{\mu(x)} \right) \frac{\partial e_z^x}{\partial \omega} \quad (78)$$

The dependencies on x and z will now be removed and the derivatives replaced by $\frac{\partial}{\partial x} e_z^x = e'$ for readability.

We have the TE impedance derivative

$$\frac{\partial Z_R}{\partial \omega} = \frac{\partial}{\partial \omega} \left(\frac{e(0^+)}{\frac{1}{\omega \mu} e'(0^+)} \right) \quad (79)$$

Now:

$$\frac{\partial Z_R}{\partial \omega} = \left(\frac{e}{\frac{1}{\mu} e'} \right)' = \frac{\frac{1}{\mu} (e' \frac{\partial e}{\partial \omega} - e \frac{\partial e'}{\partial \omega})}{\frac{1}{\mu^2} (e')^2} \quad (80)$$

$$= \frac{\mathcal{W}}{\frac{1}{\mu^2} (e')^2}, \quad (81)$$

This Wronskian is now differentiated with respect to x :

$$\mathcal{W}' = \frac{\partial}{\partial x} \left[\frac{1}{\mu} \left(e' \frac{\partial e}{\partial \omega} - e \frac{\partial e'}{\partial \omega} \right) \right] \quad (82)$$

The first term can be rewritten with the help of [equation 26](#) as

This, rewritten with substitutions from [equation 71](#) and [equation 78](#) is

$$\frac{\partial}{\partial x} \left[\frac{1}{\mu} \left(e' \frac{\partial e}{\partial \omega} - e \frac{\partial e'}{\partial \omega} \right) \right] = \frac{\partial e}{\partial \omega} \left(\frac{1}{\mu} e' \right)' - e \left(\frac{1}{\mu} \frac{\partial e'}{\partial \omega} \right)' \quad (83)$$

$$= - \left(\omega^2 \varepsilon(x) - \frac{k^2}{\mu(x)} \right) e \frac{\partial e}{\partial \omega} + e \left[2\omega \varepsilon e + \left(\omega^2 \varepsilon(x) - \frac{k^2}{\mu(x)} \right) \frac{\partial e}{\partial \omega} \right] e \quad (84)$$

$$= 2\omega \varepsilon e^2. \quad (85)$$

Since \mathcal{W}' is always positive, it will give a negative $\mathcal{W}(0^+)$ so the rest of the proof is analogous to that of [theorem 2](#). ■

The full solution can be reconstructed after solving for the x -components.

3.2 Results

The results for $k = 1$ are provided below, as well as the full 2D solution plots for TE and TM modes in [subsection 3.2](#). If $k = 0$, the problem reduces to 1D and the checks and comparisons for 1D interface modes hold. The only difference from the 1D presented above is that all fields are now real. One TM interface mode for different periods is included in [figure 20](#) for completeness. The space used is that of [equation 64](#) in all cases (except the one which has period 2 of ε on the left).

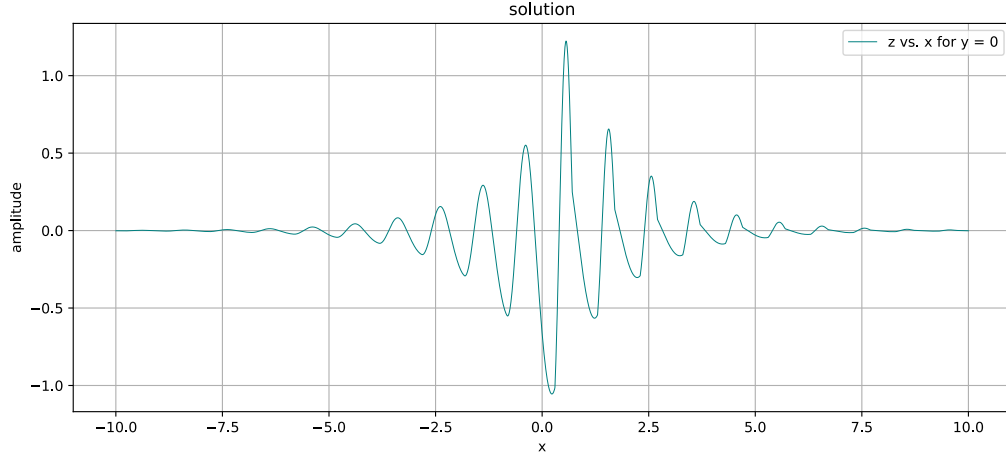


Figure 16: Interface Mode Slice at $y = 0$, $\omega = 4.0063$, Transversal Electric

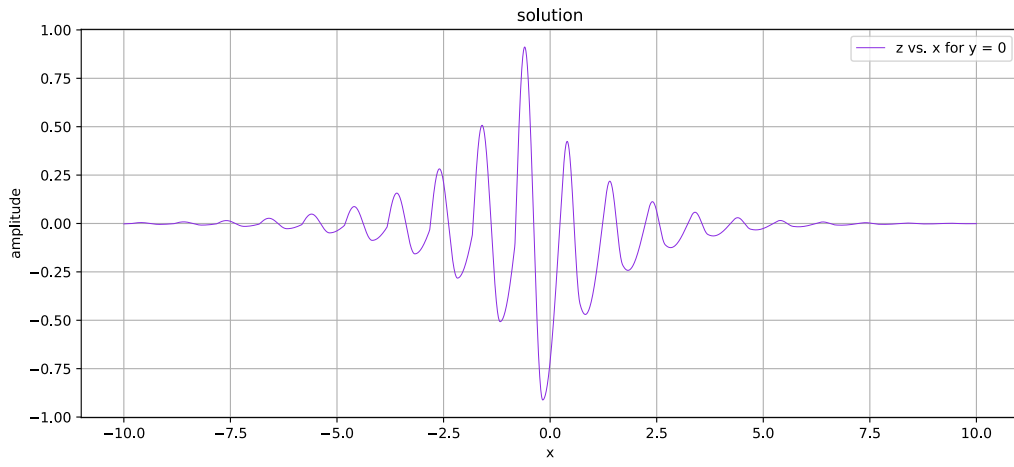


Figure 17: Interface Mode Slice at $y = 0$, $\omega = 4.0037$, Transversal Magnetic

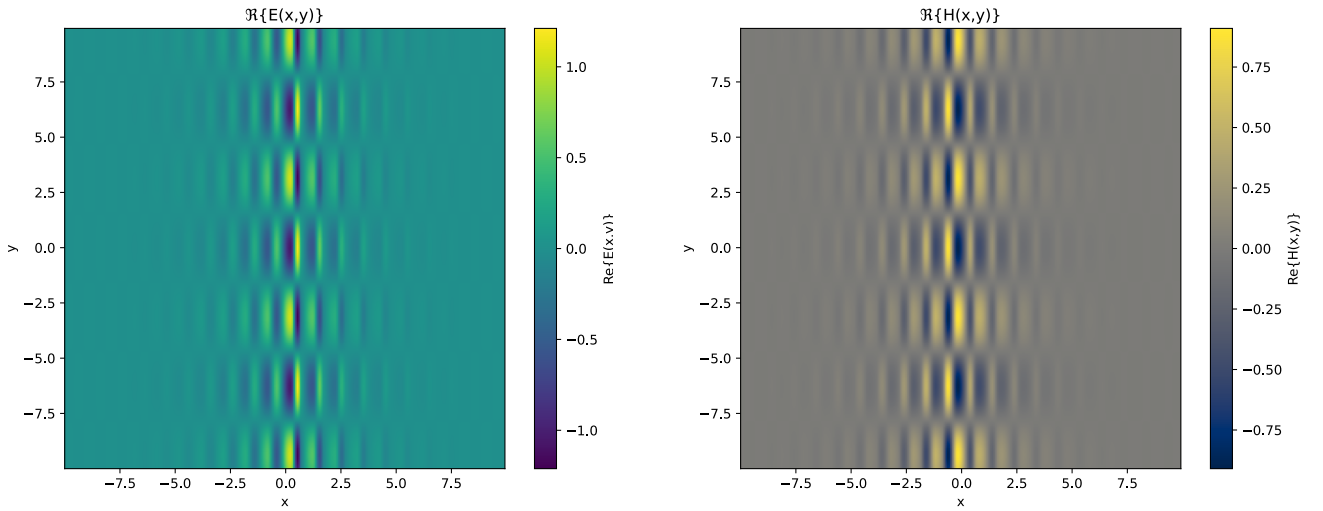


Figure 18: Real Parts of TE (e_z) and TM (h_z) waves on a $[-10, 10] \times [-10, 10]$ square with $k_y = 1$

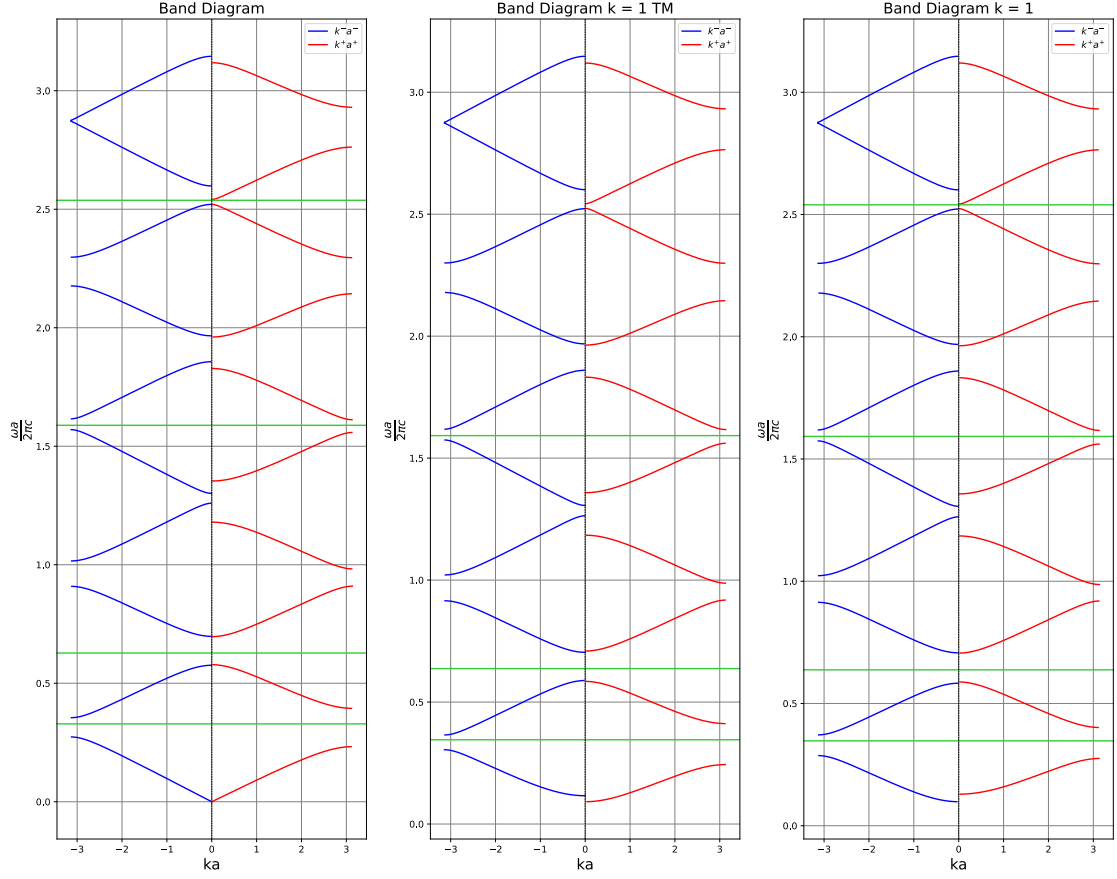


Figure 19: Band diagrams from figure 6 (left, same for $k = 0$ TE), for $k_y = 1$ TM (middle) and $k_y = 1$ TE (right)

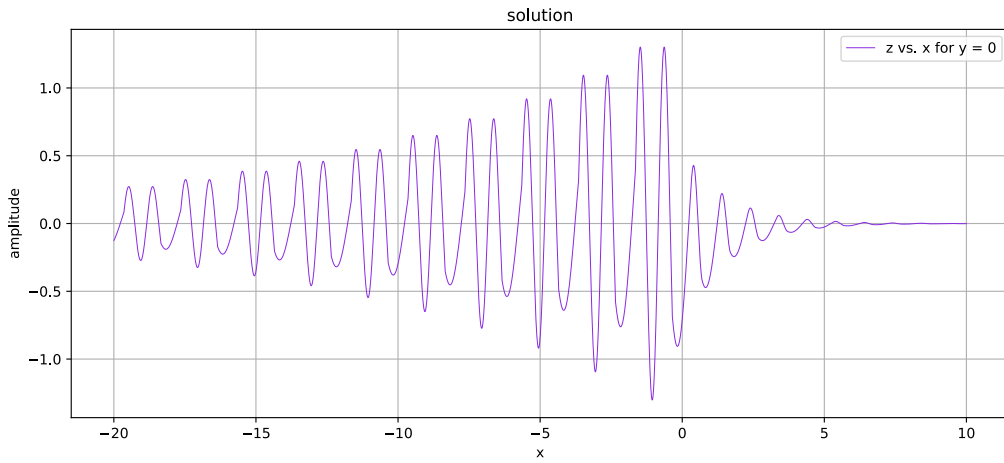


Figure 20: Interface mode slice at $y = 0$ with $k_y = 1$, $\omega = 7.9940$, Transversal Magnetic with period 2 on the left and 1 on the right

4 Zak Phase and Topologically (Non) Protected Interface Modes

4.1 Zak Phase

The Zak phase is the Berry phase accumulated by the Bloch mode photon as it moves through the Brillouin zone. It essentially measures the global phase shift from the periodic structure.

The Zak phase, from Fang, Li, Yang [3] and [1], of the j -th band can be computed as

$$\theta_j^{\text{Zak}} = i \int_{-\pi}^{\pi} \langle u_{j,k} | \frac{\partial u_{j,k}}{\partial k} \rangle dk \mod 2\pi, \quad (86)$$

where $u_{j,k}$ is the periodic part of a given Bloch mode $\psi(x) = e^{ikx} u_{j,k}(x)$ and the inner product is given by the integral over the unit cell of length T

$$\langle u|v \rangle = \int_0^T \varepsilon(x) u^*(x) v(x) dx. \quad (87)$$

An additional term is provided by [1] which takes jumps across 0^\pm and $\pm\pi$ into account, resulting in the following formula:

$$\theta_n^{\text{Zak}} = i \int_{-\pi}^0 \langle u_{j,k} | \frac{\partial u_{j,k}}{\partial k} \rangle dk + i \int_0^{\pi} \langle u_{j,k} | \frac{\partial u_{j,k}}{\partial k} \rangle dk + \hat{\theta}_j \mod 2\pi \quad (88)$$

$$\hat{\theta}_j = -\text{Im} \ln \langle u_{j,0-} | u_{j,0} \rangle - \text{Im} \ln \langle u_{j,\pi} | e^{-i2\pi x} u_{j,(-\pi)+} \rangle \quad (89)$$

4.1.1 Trapezoidal Rule and Finite Differences

In the inner product, the integrand is periodic so it makes sense to use the trapezoidal rule with uniform meshwidth since the cell has uniform meshwidth. This is given in the following listing with arrays u and v , and the period T :

```

1 function innerprod(u, v, T)
2     sum = 0
3     h = T / Npts
4
5     for i = 0 : Npts-2 do
6         sum = sum + ε(h*i) conj(u(i)) v(i)
7             + (ε(h*(i+1)) conj(u(i+1)) v(i+1))
8     end
9
10 return sum * h * 0.5 //even meshwidth permits extraction
11                      //of the product with h * 0.5

```

The finite differences will be applied to the full Zak phase integral, but with non-uniform meshwidth. The k wave vectors are unknown, and they are generated in the loop itself.

The idea is to sweep along the frequency ω as above, generate the two associated k^- and k^+ , find the periodic parts of the Bloch modes, and save them in arrays for use in the next step. We already keep track of the bands and band gaps as above, so this is simple to implement in that structure.

The formula will use a "pseudo-central" difference derivative for $\frac{\partial u_{k,n}}{\partial k}$:

$$\frac{\partial u_{k_{j+1/2},n}}{\partial k} \approx \frac{u_{k_{j+1}} - u_{k_j}}{k_{j+1} - k_j}, \quad (90)$$

which will be applied to a trapezoidal rule-like sum

$$\theta_j = \sum_{i=0}^{N-1} \frac{\langle u_{j,k_i} | \frac{\partial u_{k_{i+1/2},k}}{\partial k} \rangle + \langle u_{j,k_{i+1}} | \frac{\partial u_{j,k_{i+1/2}}}{\partial k} \rangle}{2} (k_{i+1} - k_i) \quad (91)$$

$$= \sum_{i=0}^{N-1} \frac{\langle u_{j,k_i} + u_{j,k_{i+1}} | \frac{\partial u_{j,k_{i+1/2}}}{\partial k} \rangle}{2} (k_{i+1} - k_i) \quad (92)$$

$$= \sum_{i=0}^{N-1} \frac{\langle u_{j,k_{i+1}} + u_{j,k_i} | u_{k_{i+1}} - u_{k_i} \rangle}{2}. \quad (93)$$

The name is pseudo-central because the k -step is not uniform and the backward and forward differences are averaged with a factor 1/2, which would, for a uniform mesh, produce central differences. The result will have error of first order $\mathcal{O}(\frac{1}{N_k})$ since forward and backward differences used here are of first order, and will conveniently avoid the division and multiplication by small meshwidth differences.

```

1 for i = 0 : N_omega do
2   omega = (omega_end - omega_start) / N_omega*i + omega_start
3   zak = 0
4
5   // finite differences over k
6   if (same_band)
7     u1 = extractPeriodic(bloch1, phase_k+)
8     u2 = extractPeriodic(bloch2, phase_k-)
9
10    zak = zak + 0.5*i*innerprod(u1 + u1_old, u1_old - u1_old)
11    zak = zak + 0.5*i*innerprod(u2 + u2_old, u2_old - u2_old)
12
13    u1_old = u1
14    u2_old = u2
15
16   // additional jump terms
17   else if (just_entered_new_band)
18     zak = 0
19     if (k=±pi) zak = zak - arg(innerprod(u1, e-2piixu2))
20     else if (k=0) zak = zak - arg(innerprod(u2, u1))
21   end

```

```

22
23     else if (just exited old band)
24         if (k= $\pm\pi$ ) zak = zak - arg(innerprod(u1_old,  $e^{-2\pi ix}$  u2_old))
25         else if (k=0) zak = zak - arg(innerprod(u2_old, u1_old))
26         end
27         save(zak)
28     end
29 end

```

The Zak phase calculator is quite sensitive to errors. It is best to run it with around 4000 system solution steps and about 2500 frequency steps per band. Otherwise it might encounter spurious π shifts. For 40 000 frequency steps in the range 0-20 and 4000 solution steps, the results for the band diagram from [1] in figure 4 are shown in table 2. Note that the bands are numbered from below: the first band is the lowest band, the second above that one etc.

band	1	2	3	4	5	6	7	8
expected L	0	0	π	0	π	0	0	π
result L	1.684e-5	-9.871e-5	-3.142	2.414e-5	3.142	-1.717e-4	- 1.173e-5	-3.142
expected R	0	0	π	0	π	0	π	0
result R	1.031e-5	1.387e-4	-3.142	2.142e-5	3.142	3.424e-4	3.142	-2.816e-5

Table 2: Band Zak phases for right and left space from [1]

Since "fmod" was used with real numbers, the results are not quantized π s and 0s but values between $\langle -2\pi, 2\pi \rangle$.

The following table 3 shows the Zak phases of the bands in figure 6 and and expected from [2].

band	1	2	3	4	5	6
expected L	?	π	π	π	0	π
result L	3.142	3.142	3.142	3.142	-8.849e-6	3.142
expected R	?	π	0	π	π	0
result R	-2.609e-5	3.142	6.283	3.142	3.142	-1.761e-5

Table 3: Band Zak phases for right and left space from varying μ, ϵ [2]

For table table 4 10 000 solution points and 50 000 frequency steps were used. These are Zak phases from bands shown in figure 8 and expected from [2].

Interestingly, the Zak phases disagree with the right side of figure 8 from [2]; however, since the unit cell was scaled uniformly, there should be no reason for a Zak phase change - and the results from this project agree with the non-scaled cell, while those from [2] do not. Perhaps the signs $\pi, 0, \pi$ were shifted one place lower since the first band was included in figure 8 of [2] and it is not included in most figures, or the Zak phase calculator from this project encounters issues with "stretched" spaces that evaded inspection.

band	1	2	3	4	5	6
expected L	?	π	π	π	0	π
result L	-3.122	-3.111	-3.109	-3.108	2.712e-2	-3.099
expected R	π	0	π	?	?	?
result R	-5.892e-6	3.142	6.283	3.142	3.142	-
stretched R	-2.609e-5	3.142	6.283	3.142	3.142	-1.761e-5

Table 4: Band Zak phases for right and left space from varying μ , ε , different period [2]

4.2 Topologically Protected Interface Modes

An interface mode is topologically protected for symmetric cells with quantized Zak phase, if the bulk topological index of the cells on the right and left side is different. In this case, the bulk topological index attains values of -1 or 1.

The bulk topological index [1], [3] for the j -th is defined as

$$\gamma_j := (-1)^{j+l-1} e^{i \sum_{m=1}^j \theta_m}, \quad (94)$$

where l is the number of Dirac points below that band, and θ_m is the Zak phase of the m -th band.

Additionally, [1] show that the bulk topological index is 1 if the Bloch mode at the edge of the j -th band is even, and it is -1 if the Bloch mode is odd.

4.2.1 Tiling

Unit cells that tile the same space show the same band diagrams. To find the characteristic multipliers of a Bloch mode, the starting position in the unit cell is not relevant since the periodicity must be preserved if it is indeed a Bloch mode; changing the unit cell in this regard is like changing the starting position which is equivalent to multiplying the solution by e^{ikx_0} so the quasi-periodicity and therefore the eigenvalues which constitute the band diagram remain unchanged.

4.2.2 Persistent Modes

From numerical simulations, it seems that changing the unit cell does not affect the existence of an interface mode in a given band gap, if it is protected for one choice of cell. From tiling, these different cells will yield the same band diagrams; however, the Zak phase, and consequently, the bulk topological index, will be changed which is confirmed by the full Zak phase calculator. If cells on both sides can be made symmetric, so that an interface mode is topologically protected in the n -th band gap, that band gap appears to have an interface mode for non-symmetric and mirrored cells, which do not have quantized Zak phase or bulk topological indices of 1 or -1.

The exact frequency of an interface mode depends on the unit cell because it defines the contact, and hence the associated impedance, which is changing at different positions of the unit cell.

5 Conclusion and Further Work

The code developed in this project can compute all interface modes that satisfy the sufficient and necessary condition in [definition 4](#) in 1D and 1.5D. Both can handle periodic piecewise continuous material parameters that do not need to be inversion symmetric, with different periods on different sides.

The 1D class also supports computations of Zak phase. The 1.5D class supports TE and TM modes. Both offer retrieval of the field or its derivative, as well as band gap start and end positions.

Further work would involve extending the ideas from 1D and 1.5D and applying them to 2D, with Chern numbers instead of Zak phase and impedance matrices instead of a scalar as described by Lawrence et al. [\[10\]](#).

6 Using the Code

The code is structured like a mini library and is intended to use in the way described by the C++ code listing. The reference space of this code is the left-halfspace. It dictates all normalizations.

First, a space is initialized with material parameters as `std::functions`, followed by the number of plotting (solution) points, the period values for the two media, number of discrete frequencies and the frequency range beginning and end.

After this, the space should be preprocessed by calling the `drawBands(...)` method. This will also create a band diagram `.csv` file. Assign the result of this method to a `std::vector` of possible interface mode frequencies.

Now we can loop through the vector of possible frequencies and call the secant method on the original space at that frequency. If there is an interface mode, the secant method will return true. After finding an interface mode we can simply run the `drawInterface(...)` method which will compute the interface mode at one cell each and save 20 cells, 10 on each side.

These functions all have additional parameters such as those for choosing the Y or Z polarizations, or the electric or magnetic fields.

The main differences between 1.5D and 1D are the wave vector that comes after the material parameters in the constructor, and the fact that it can compute the field itself or the x -derivative of the TE electric or TM magnetic fields since this is sufficient to reconstruct the vector field that is in the xy plane.

Zak phases and the inner product are available for 1D and can be computed using `zakPhases(...)` which otherwise has the same functionality as `drawBands(...)` but it is much slower and heavily depends on the number of plotting points and discrete frequencies.

All kernel methods are available for use as well, for example, the `MonodromyAtZeroX(...)` methods and `systemX(...)` methods. You can find more about them and methods mentioned earlier in the documentation.

After computing the band diagrams, transmission coefficients, and interface modes, they can be visualized using the Python scripts: `plot.py`, `plot_band.py`, and `plot_2d.py`.

```

1 #include "interface_modes_1_d.h"
2 #include "interface_modes_1_5_d.h"
3
4 int main() {
5     cout << "starting" << endl;
6
7     // define the material parameters, eg.
8     double period2 = 2.;
9     auto muR = [period2](double x) {
10         double xmod = std::fmod(x, period2);
11         if (xmod < 0) xmod += period2;
12         if (xmod <= 0.3 * period2 || xmod >= 0.7 * period2) {
13             return 1.;
14         }
15         return 6.;
16     };
17
18     InterfaceModes1D space(muL, muR, epsL, epsR, 10000, 1.,
19 period2, 40000, 0, 20);
20     space.setAitkenNeville(10, 1e-6, 20); //optional
21
22     std::vector<double> preproc = space.drawBands(10, 10);
23
24     //for higher precision
25     space.setAitkenNeville(14, 1e-12, 40);
26
27     for (int i = 0; i < preproc.size(); i++) {
28         if (space.findInterfaceModeSecant(preproc[i], 1e-12, 20,
29 false)) {
30             space.drawInterfaceMode();
31         } else {
32             cout << "There is nothing here." << endl;
33         }
34     }
35
36     return 0;
37 }

```

6.1 List of Documented Class Members

- `drawBands()` : `InterfaceModes1D`
- `drawBandsTE()` : `InterfaceModes1_5D`
- `drawBandsTM()` : `InterfaceModes1_5D`
- `drawInterfaceMode()` : `InterfaceModes1_5D` , `InterfaceModes1D`
- `findInterfaceModeSecant()` : `InterfaceModes1_5D` , `InterfaceModes1D`
- `innerprod()` : `InterfaceModes1D`
- `InterfaceModes1_5D()` : `InterfaceModes1_5D`
- `InterfaceModes1D()` : `InterfaceModes1D`
- `monodromyAtZeroTE()` : `InterfaceModes1_5D`
- `monodromyAtZeroTM()` : `InterfaceModes1_5D`
- `monodromyAtZeroY()` : `InterfaceModes1D`
- `monodromyAtZeroZ()` : `InterfaceModes1D`
- `monodromyTE()` : `InterfaceModes1_5D`
- `monodromyTM()` : `InterfaceModes1_5D`
- `monodromyY()` : `InterfaceModes1D`
- `monodromyZ()` : `InterfaceModes1D`
- `setAitkenNeville()` : `InterfaceModes1_5D` , `InterfaceModes1D`
- `setRKSSMiter()` : `InterfaceModes1_5D` , `InterfaceModes1D`
- `systemTE()` : `InterfaceModes1_5D`
- `systemTM()` : `InterfaceModes1_5D`
- `systemY()` : `InterfaceModes1D`
- `systemZ()` : `InterfaceModes1D`
- `zakPhases()` : `InterfaceModes1D`

7 References

- [1] Lin J. and Zhang H., Mathematical theory for topological photonic materials in one dimension 2022, J. Phys. A: Math. Theor. 55 495203 DOI: [10.1088/1751-8121/aca9a5](https://doi.org/10.1088/1751-8121/aca9a5)
- [2] Xiao, M. and Zhang, Z. Q. and Chan, C. T., Surface Impedance and Bulk Band Geometric Phases in One-Dimensional Systems Phys. Rev. X 4, 2014, 021017 DOI: <https://doi.org/10.1103/PhysRevX.4.021017>
- [3] Fang, Y. T. and Li, X. X. and Yang, L. X. Robust topological edge states from one-dimensional diatomic chain photonic crystals International Journal of Modern Physics B 2021 35:10 DOI: <https://doi.org/10.1142/S0217979221501460>
- [4] Coutant A, Lombard B. 2024, Surface impedance and topologically protected interface modes in one-dimensional phononic crystals. Proc. R. Soc. A 480: 20230533. DOI: <https://doi.org/10.1098/rspa.2023.0533>
- [5] Kalozoumis, P. A. and Theocharis, G. and Achilleos, V. and Félix, S. and Richoux, O. and Pagneux, V., Finite-size effects on topological interface states in one-dimensional scattering systems Phys. Rev. A 98, 2018, 023838 DOI: <https://doi.org/10.1103/PhysRevA.98.023838>
- [6] Hiptmair R., Numerical Methods for Computational Science and Engineering, Version: Nov 2022, ETH Zürich
- [7] Deuffhard P. and Bornemann F. (2002). Scientific Computing with Ordinary Differential Equations. Chapter 4: One-Step Methods for Nonstiff IVPs, Theorem 4.42. Springer. <https://link.springer.com/book/10.1007/978-0-387-21582-2>
- [8] Markos, P. and Soukoulis, C. (2008). Wave Propagation: From Electrons to Photonic Crystals and Left-Handed Materials. Chapter 1: Transfer Matrix, Princeton: Princeton University Press. DOI: <https://doi.org/10.1515/9781400835676>
- [9] Joannopoulos J. D. and Johnson S. G. and Winn J. N., and Meade R. D., Photonic Crystals: Molding the Flow of Light, second edition, 2008, Princeton University Press
- [10] Lawrence, F. J. and Botten, L. C. and Dossou, K. B. and McPhedran, R. C. and de Sterke, C. M., Photonic-crystal surface modes found from impedances Phys. Rev. A 82, 053840 (2010) Erratum Phys. Rev. A 83, 029907 DOI: <https://doi.org/10.1103/PhysRevA.82.053840>

Declaration of originality

The signed declaration of originality is a component of every written paper or thesis authored during the course of studies. In consultation with the supervisor, one of the following three options must be selected:

- ☒ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used no generative artificial intelligence technologies¹.
- ☐ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used and cited generative artificial intelligence technologies².
- ☐ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used generative artificial intelligence technologies³. In consultation with the supervisor, I did not cite them.

Title of paper or thesis:

Computation of Topological Interface Modes
--

Authored by:

If the work was compiled in a group, the names of all authors are required.

Last name(s):

Vrabac

First name(s):

Lara

With my signature I confirm the following:


- I have adhered to the rules set out in the Citation Guide.
- I have documented all methods, data and processes truthfully and fully.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for originality.

Place, date

Zurich, February 28, 2025

Signature(s)



If the work was compiled in a group, the names of all authors are required. Through their signatures they vouch jointly for the entire content of the written work.

¹ E.g. ChatGPT, DALL E 2, Google Bard

² E.g. ChatGPT, DALL E 2, Google Bard

³ E.g. ChatGPT, DALL E 2, Google Bard