TECHNISCHE UNIVERSITÄT WIEN
Vienna | Austria

DIPLOMARBEIT

Computing resonances in metallic photonic crystals

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Technische Mathematik

eingereicht von

Michael Leumüller

Matrikelnummer 01225300

ausgeführt am Institute for Analysis and Scientific Computing
der Fakultät für Mathematik und Geoinformation
der Technischen Universität Wien

Betreuer: Associate Prof. Dipl.-Math. Dr.rer.nat. Lothar Nannen

Wien, 10.09.2018 _____ _____

(Unterschrift Verfasser) (Unterschrift Betreuer)

Abstract

Resonance problems arise in many fields of research. An example are photonic crystals, in which the propagation of waves is defined by resonances. Photonic crystals with band gaps are of special interest. Band gaps are regions of frequencies which cannot propagate through the crystal. To calculate the band structure of photonic crystals many linear resonance problems need to be solved. Fast and reliable linear eigenvalue solvers are needed. Lately, metallic photonic crystals have become more interesting. Differently from photonic crystals the electric permittivity of metallic photonic crystals depends on the frequency leading to rational resonance problems. These problems come with a high computational cost.

In this thesis, we introduce an efficient eigenvalue solver for large rational eigenvalue problems. At first, the resonance problems for two and three dimensional metallic photonic crystal are derived from Maxwell's equations. Then, they are discretised with Bloch periodic high order finite elements in **Netgen/NGSolve**. The arising large rational matrix eigenvalue problems are linearised with a rational linearisation schema and solved by the shift-and-invert Arnoldi method. By combining linearisation with the shift-and-invert Arnoldi, systems of linear equations with dimensions larger than the original matrix size have to be solve in each iteration. With the introduced rational linearisation these large systems of linear equations can be reduced to the original problem size.

The proposed combination of these two algorithms is applied to two and three dimensional metallic photonic crystals and compared to the shift-and-invert Arnoldi with a standard polynomial linearisation. The appearance of plasmon frequencies is witnessed and the influence on the solver is studied.

We show that the proposed method is a reliable and fast solver for large rational eigenvalue problems.

Acknowledgements

This thesis would not have been possible without the support of many people.

First and foremost, I would like to offer my special thanks to my supervisor, Prof. Lothar Nannen, for his support over the past five years. His council has been essential in my time as a student and I could not have wished for a better supervisor.

I would like to thank to Prof. Schöberl for his important help with **Netgen/NGSolve** and for letting me be a part of his workgroup.

A special thanks goes to Michael Neunteufel. Without him my time on the third floor would have been half the fun.

Moreover, I would like to acknowledge the support of the Austrian Science Fund (FWF) under grant P26252.

Finally, I wish to thank the moon of my life.

Contents

1 Introduction

Resonance problems arise in many fields of research. An example are photonic crystals (PHCs), in which the propagation of waves is defined by resonances. Photonic crystals with band gaps are of special interest. Band gaps are regions of frequencies which cannot propagate through the crystal. Usually, non-metallic materials are used as dispersive material, but lately also metallic materials have been taken into account. In the non-metallic material linear resonance problems appear. This changes in metallic materials. They give rise to rational resonance problems, which are numerically more challenging.

There are different ways to tackle these problems. For example, the finite-difference time-domain method in [FYS+13, IS01] uses excitation and Fourier analysis. Another method is the multiple multipole method proposed in [MEH02] and a Newton-type method is proposed in [HLM16]. The Newton-type method is specialised on calculating the smallest resonances of a three dimensional metallic photonic crystal (MPHC). In this thesis we introduce a method based on linearisation and the shift-and-invert Arnoldi method.

In a metallic photonic crystal the electric and magnetic fields are governed by Maxwell's equations, introduced in Section 2. From them the resonance problems for the electric and magnetic field are derived and put in the framework given by metallic photonic crystals. This leads to two and three dimensional rational resonance problems with Bloch periodic boundary conditions.

Most of the time these resonance problems cannot be solved analytically. Therefore, in Section 3 the finite element method is used to discretise the resonance problems into rational matrix eigenvalue problems of the form: Find pairs (ω, u) with $\omega \in \mathbb{C}, u \in \mathbb{C}^N \setminus \{0\}, N \in \mathbb{N}$ such that

$$\mathrm{T}(\omega)u = 0, \qquad \mathrm{T}(\omega) \in \mathbb{C}^{N\times N}.$$

The rational problem depends linearly on the eigenfunction, but the matrix valued operator $\mathrm{T}(\omega)$ depends rationally on the eigenvalue. In the generalised linear eigenvalue problem the operator would be $\mathrm{T}(\omega) = \mathrm{A} - \omega\mathrm{B}$ with $\mathrm{A}, \mathrm{B} \in \mathbb{C}^{N\times N}$.

The generated matrices are large and sparse making it necessary to apply iterative eigenvalue solvers. Such a solver is the shift-and-invert Arnoldi, introduced in Section 4. The solver is restricted to linear eigenvalue problems, hence it is adapted to fit the rational eigenvalue problem setting. This is done by linearising the rational problem first. As linearisation a rational and a polynomial approach are introduced. One of the topics of this thesis is to highlight the differences and the similarities of the two linearisations.

After the linearisation we end up with a k-times larger eigenvalue problem: Find pairs (ω, v) with $\omega \in \mathbb{C}, v \in \mathbb{C}^{kN} \setminus \{0\}$ such that

$$\hat{\mathrm{A}}v = \omega\hat{\mathrm{B}}v, \qquad \hat{\mathrm{A}}, \hat{\mathrm{B}} \in \mathbb{C}^{kN\times kN}.$$

The factor k is fixed for a rational eigenvalue problem. In the shift-and-invert Arnoldi algorithm the system of linear equations $(\hat{\mathrm{A}} - \sigma\hat{\mathrm{B}})\hat{x} = \hat{y}$ with $\hat{x}, \hat{y} \in \mathbb{C}^{kN}, \sigma \in \mathbb{C}$ has to be solved for $\hat{x}$ in each iteration. Usually, solving these systems of linear equations is more expensive than solving systems of linear equations of

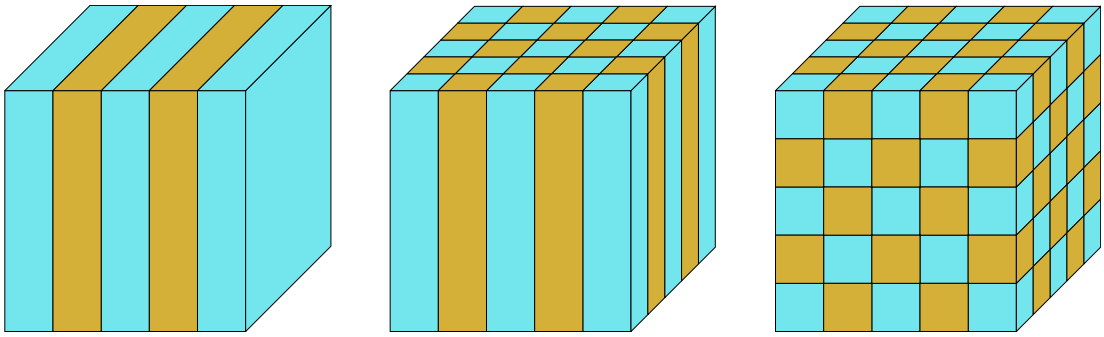


Figure 1: Example of 1D, 2D and 3D photonic crystal structure. The dispersive and non-dispersive materials are indicated in different colours.

the original problem size. The rational and the polynomial linearisation have been chosen so that only the system of linear equations $\mathrm{T}(\sigma)x = y$ with $x, y \in \mathbb{C}^N, \sigma \in \mathbb{C}$ has to be solved for x. The main idea is that a big part of the large matrix of the linear system of equations can be analytically inverted by applying a Schur complement.

As stated above, there are different methods to solve the rational eigenvalue problems. Two of them are shortly introduced in Section 5. The first one is the Newton-type method proposed in [HLM16] and the second one is Beyn's method proposed in [Bey12]. Beyn's method is an eigenvalue solver that uses complex analysis and can be applied to a large variety of non-linear eigenvalue problems. Making it the most versatile method mentioned in this thesis.

In Section 6 the rational and polynomial Arnoldi algorithms are tested on two and three dimensional metallic photonic crystals. Especially interesting is the calculation of plasmon frequencies in a two dimensional metallic photonic crystal.

We conclude our findings in Section 7 and give a short overview on future prospects in Section 8.

2 Maxwell's equations in metallic photonic crystals

At first, we introduce the necessary objects to state Maxwell's equations in metallic photonic crystals, starting with PHCs.

A photonic crystal is a combination of non-dispersive and dispersive materials with a periodic structure. This structure can be periodic in one, two or three directions resulting in one, two or three dimensional photonic crystals, see Figure 1. In dept information about photonic crystals can be found in [JJWM08].

The periodic structure of a PHC is formed by unit cells. For example in Figure 2 a two dimensional PHC with its unit cell can be seen and Figure 3 shows a three dimensional PHC with diamond structure. These crystals are the objects of interest in this thesis and used for the numerical tests. In a 2D photonic crystal the extension in the z-direction can be omitted, because it is unnecessary for computations, as will be seen later. Each crystal has a lattice constant a describing its diameter. The up to three lattice translation vectors $\mathbf{a}_i$ span the

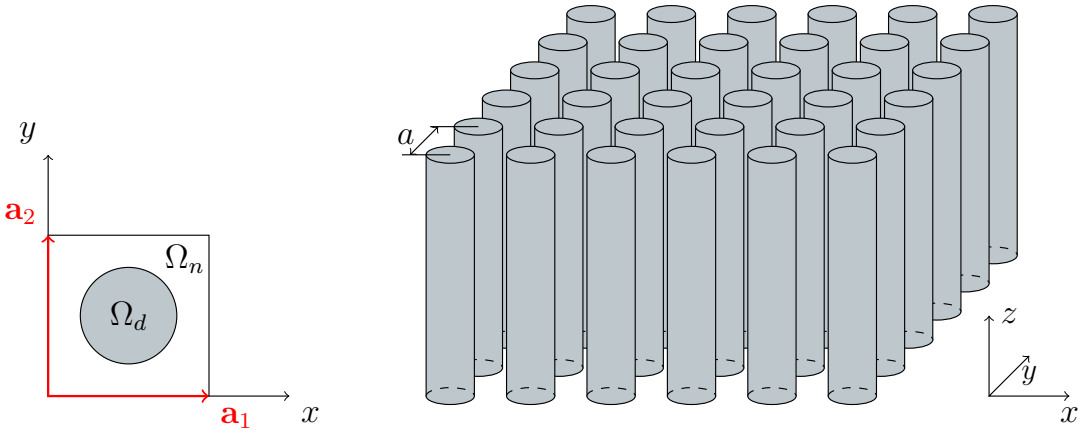


Figure 2: On the right, a 2D photonic crystal with rods as dispersive material and on the left, the according unit cell with the lattice translation vectors $\mathbf{a}_i$.

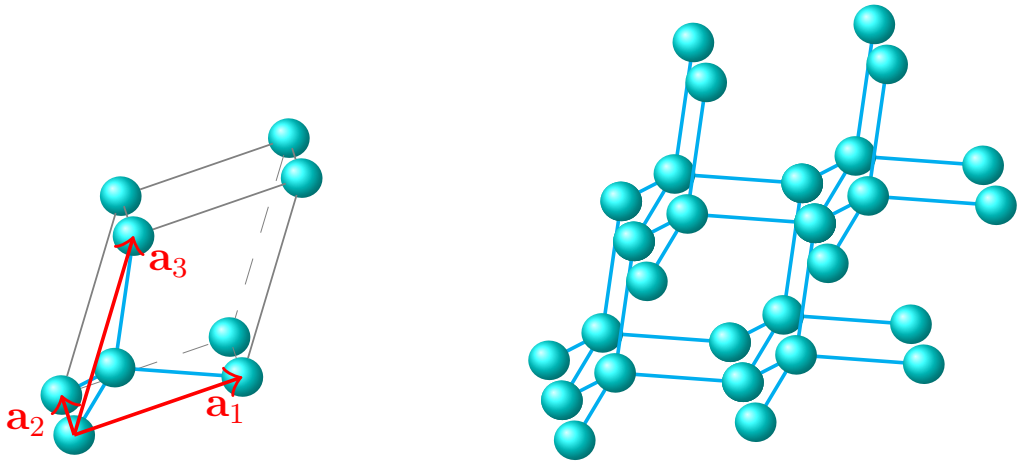


Figure 3: On the right, a 3D photonic crystal with a diamond structure as dispersive material and on the left, the according unit cell with the lattice translation vectors $\mathbf{a}_i$.

unit cell.

Let us have a look at the defining system of equations of a PHC. In such crystals the magnetic field $\mathbf{H}$ and the electric field $\mathbf{E}$ are governed by Maxwell's equations [JJWM08, p. 8]

$$\begin{aligned} \operatorname{curl} \mathbf{E}(\mathbf{r},t) &= -\partial_t \mathbf{H}(\mathbf{r},t), & \operatorname{div} \mathbf{H}(\mathbf{r},t) &= 0, \\ \operatorname{curl} \mathbf{H}(\mathbf{r},t) &= \epsilon(\mathbf{r})\partial_t \mathbf{E}(\mathbf{r},t), & \operatorname{div} \epsilon(\mathbf{r})\mathbf{E}(\mathbf{r},t) &= 0, \end{aligned}$$

if no external current density and charge density exist and the magnetic permeability is set to one, with $\mathbf{r} \in \Omega$ being the spatial dimension and $t \in (0,T)$ the temporal. At this point Ω is the unbounded $\mathbb{R}^3$. Later we will argue that Ω can be restrict to the unit cell. The scalar function $\epsilon(\mathbf{r})$ is the electric permittivity.

Applying the time harmonic ansatz [JJWM08, p. 9]

$$\mathbf{E}(\mathbf{r},t) := \mathbf{E}(\mathbf{r})\,\mathrm{e}^{-i\omega t}, \qquad \mathbf{H}(\mathbf{r},t) := \mathbf{H}(\mathbf{r})\,\mathrm{e}^{-i\omega t},$$

with $i := \sqrt{-1}$ being the imaginary unit and $\omega \in \mathbb{C}$, leads to the time independent system of equations

$$\begin{aligned} \operatorname{curl} \mathbf{E}(\mathbf{r}) &= i\omega \mathbf{H}(\mathbf{r}), & \operatorname{div} \mathbf{H}(\mathbf{r}) &= 0, \\ \operatorname{curl} \mathbf{H}(\mathbf{r}) &= -i\omega \epsilon(\mathbf{r}) \mathbf{E}(\mathbf{r}), & \operatorname{div} \epsilon(\mathbf{r}) \mathbf{E}(\mathbf{r}) &= 0. \end{aligned}$$

This is a coupled system of $\mathbf{E}$ and $\mathbf{H}$.

If $\mathbf{E}$ is smooth enough, then the curloperator can be applied to the first system of equations. Substituting $\operatorname{curl} \mathbf{H}(\mathbf{r})$ with the second system of equations afterwards leads to the, in w^2 linear, resonance problem: Find pairs $(\omega, \mathbf{E})$ with $\omega \in \mathbb{C}, \mathbf{E} \neq 0$ such that

$$\operatorname{curl} \operatorname{curl} \mathbf{E}(\mathbf{r}) = \omega^2 \epsilon(\mathbf{r}) \mathbf{E}(\mathbf{r}) \tag{1}$$

is satisfied.

Remark 1 *A solution of this resonance problem also satisfies the coupled problem, by introducing* $\mathbf{H} := \frac{1}{i\omega} \operatorname{curl} \mathbf{E}$. *With this definition* $\mathbf{H}$ *is automatically divergence free.*

Exchanging the roles of $\mathbf{E}$, $\mathbf{H}$ and assuming enough regularity of $\mathbf{H}$, leads to the resonance problem

$$\operatorname{curl} \epsilon(\mathbf{r})^{-1} \operatorname{curl} \mathbf{H}(\mathbf{r}) = \omega^2 \mathbf{H}(\mathbf{r}), \tag{2}$$

where $\omega \in \mathbb{C}$ and $\mathbf{H} \neq 0$.

We stated Maxwell's equations and derived resonance problems without applying it to the specific case of photonic crystals. For a photonic crystal, there are further simplifications. The structure and the electric permittivity are periodic:

$$\epsilon(\mathbf{r} + l\mathbf{a}_1 + m\mathbf{a}_2 + n\mathbf{a}_3) = \epsilon(\mathbf{r}), \qquad \forall l, m, n \in \mathbb{Z}.$$

A resonance function $\mathbf{H}$ or $\mathbf{E}$ can be translated by any arbitrary vector $\mathbf{t} = l\mathbf{a}_1 + m\mathbf{a}_2 + n\mathbf{a}_3$ with $l, m, n \in \mathbb{Z}$ and is still a resonance function. Therefore, the solutions have this property. An example of such functions are the plane waves

$$\mathbf{f}_{\mathbf{k}}(\mathbf{r}) := \mathbf{F}\, \mathrm{e}^{i\mathbf{k}\cdot\mathbf{r}}, \qquad \mathbf{F} \in \mathbb{C}^3, \mathbf{k} \in \mathbb{R}^3.$$

The vector $\mathbf{k}$ is called the wave vector. These are just examples, but it can be shown, that for a specific wave vector $\mathbf{k}$, a general solution has the form

$$\begin{aligned} \mathbf{f}_{\mathbf{k}}(\mathbf{r}) &:= \sum_{l,m,n \in \mathbb{Z}} \mathbf{F}_{\mathbf{k},l,m,n}\, \mathrm{e}^{i\mathbf{k}\cdot\mathbf{r} + i2\pi(m\mathbf{b}_1 + l\mathbf{b}_2 + n\mathbf{b}_3)\cdot\mathbf{r}} \\ &= \mathrm{e}^{i\mathbf{k}\cdot\mathbf{r}} \sum_{l,m,n \in \mathbb{Z}} \mathbf{F}_{\mathbf{k},l,m,n}\, \mathrm{e}^{i2\pi(m\mathbf{b}_1 + l\mathbf{b}_2 + n\mathbf{b}_3)\cdot\mathbf{r}}, \end{aligned}$$

where the vectors $\mathbf{b}_i$ build the dual basis to the lattice translation vectors $\mathbf{a}_i$ and span the reciprocal space containing the wave vectors. Therefore, the wave vector can also be written as $\mathbf{k} = k_1\mathbf{b}_1 + k_2\mathbf{b}_2 + k_3\mathbf{b}_3$. The function above is a combination of two parts. The first is a plane wave defined by the wave vector and the second is a Fourier series expansion of a periodic function. This results in the general form

$$\mathbf{f}_{\mathbf{k}}(\mathbf{r}) := \mathrm{e}^{i\mathbf{k}\cdot\mathbf{r}}\, \mathbf{u}_{\mathbf{k}}(\mathbf{r})$$

with an on the unit cell Ω periodic function $u_{\mathbf{k}}(\mathbf{r})$.

Remark 2 *With this identity, it is also possible to state the resonance problem for the periodic function. The operators in the systems of equations will be* $\mathbf{k}$ *dependent, but the space for the solution is the same for each wave vector.*

Resonance functions $\mathbf{E}$ and $\mathbf{H}$ have such a form. From this, the behaviour on the boundary Γ of the unit cell Ω can be derived. For all vectors $\mathbf{r}$ on one face of the unit cell with its opposing face being a translation along a vector $\mathbf{a}_i$ away, the periodicity

$$\mathbf{E}(\mathbf{r}+\mathbf{a}_i) = \mathrm{e}^{i\mathbf{k}\cdot\mathbf{a}_i}\,\mathbf{E}(\mathbf{r}), \qquad\qquad \mathbf{H}(\mathbf{r}+\mathbf{a}_i) = \mathrm{e}^{i\mathbf{k}\cdot\mathbf{a}_i}\,\mathbf{H}(\mathbf{r}).$$

has to hold. For a 3D photonic crystal, there are three pairs of faces and for a 2D photonic crystal two pairs of faces that are connected by this condition. This periodicity is called Bloch periodicity.

With the argumentation above we narrowed down the form of the resonances in PHCs. On the plus side, the problem only has to be solved on the unit cell, but the disadvantage is that the resonance problem, or more specific the boundary condition of the resonance problem, depends on the wave vector $\mathbf{k}$ in the reciprocal space. At this point the reciprocal space is the unbounded $\mathbb{R}^3$.

This is a problem which can be solved. For different wave vectors the according resonances are the same. For example all wave vectors, defined by

$$\mathbf{k}_{l,m,n} := \mathbf{k} + 2\pi m\mathbf{b}_1 + 2\pi n\mathbf{b}_2 + 2\pi l\mathbf{b}_3, \qquad\qquad \forall m,n,l \in \mathbb{Z}$$

result in the same resonance problem as with the original wave vector $\mathbf{k} = \mathbf{k}_{0,0,0}$. The solutions for the wave vectors in

$$\mathcal{R} := \{2\pi(a\mathbf{b}_1 + b\mathbf{b}_2 + c\mathbf{b}_3) : a,b,c \in [-0.5, 0.5]\}$$

are the same as the solutions for wave vectors in

$$\mathcal{R}_{l,m,n} := \mathcal{R} + 2\pi m\mathbf{b}_1 + 2\pi n\mathbf{b}_2 + 2\pi l\mathbf{b}_3, \qquad\qquad m,n,l \in \mathbb{Z}.$$

We can cut of every wave vector which is outside of the cube $\mathcal{R}$. There are still more possible ways to cut the necessary space resulting in the first Brillouin zone. The region which contains all necessary wave vectors. Examples of the first Brillouin zone for two dimensional and three dimensional PHCs can be seen in Figure 4.

The original resonance problems (1) and (2) evolve into:

Definition 1 (Resonance problem in a PHC for the electric field)

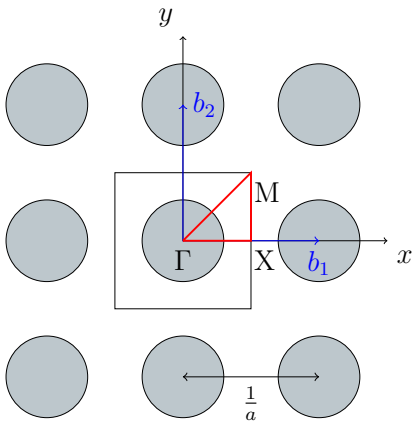
For $\mathbf{k}$ *in the first Brillouin zone, find* $(\omega_\mathbf{k}, \mathbf{E}_\mathbf{k}), \omega_\mathbf{k} \in \mathbb{C}, \mathbf{E}_\mathbf{k} \neq 0$ *such that* $\mathbf{E}_\mathbf{k}$ *is Bloch periodic and satisfies*

$$\operatorname{curl}\operatorname{curl}\mathbf{E}_\mathbf{k}(\mathbf{r}) = \omega_\mathbf{k}^2\epsilon(\mathbf{r})\mathbf{E}_\mathbf{k}(\mathbf{r}), \qquad\qquad \mathbf{r} \in \Omega.$$

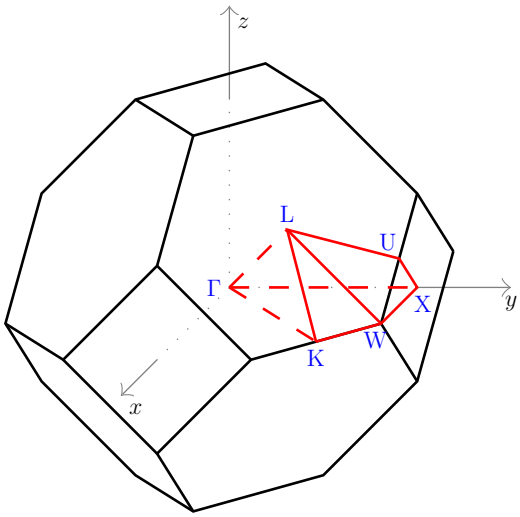
Definition 2 (Resonance problem in a PHC for the magnetic field)

For $\mathbf{k}$ *in the first Brillouin zone, find* $(\omega_\mathbf{k}, \mathbf{H}_\mathbf{k}), \omega_\mathbf{k} \in \mathbb{C}, \mathbf{H}_\mathbf{k} \neq 0$ *such that* $\mathbf{H}_\mathbf{k}$ *is Bloch periodic and satisfies*

$$\operatorname{curl}\epsilon(\mathbf{r})^{-1}\operatorname{curl}\mathbf{H}_\mathbf{k}(\mathbf{r}) = \omega_\mathbf{k}^2\mathbf{H}_\mathbf{k}(\mathbf{r}), \qquad\qquad \mathbf{r} \in \Omega.$$



(a) First Brillouin zone (square) of a 2D photonic crystal with cylinders as dispersive material. The points Γ, X, M and their connecting lines define the red bounded region for wave vectors **k**.



(b) First Brillouin zone of a face centred cubic lattice. The points Γ, X, U, L, W, K and their connecting lines define the region for wave vectors **k**.

Figure 4: Examples of the first Brillouin zone and its symmetric points.

With more symmetry arguments the region for the wave vectors can be further reduced to only a small fraction of the first Brillouin zone. Examples for these parts are marked red in Figure 4.

Remark 3 *Note that in the new resonance problems the frequency $\omega_{\mathbf{k}}$ depends on the wave vector. Sorting the frequencies with positive real part in ascending order, indexing them and interpreting them as a function in* **k** *results in functions $\omega_\tau(\mathbf{k})$. The index τ can be, but need not be, a natural number. These functions are the bands of the photonic crystal and define the band structure.*

Usually, the problems are solved on a couple of connecting lines between specific points, resulting in band structure diagrams [FYS+13, SC10]. More details on the Bloch periodicity and the first Brillouin zone can be found in the book [JJWM08], which was the source of this information.

Up to now, we analysed photonic crystals, but this work is about MPHCs. There is a small but significant difference between PHCs and MPHCs, the electric permittivity. The electric permittivity $\epsilon(\mathbf{r})$ has been introduced as a scalar function only depending on the spatial dimension, but in metallic photonic crystals it also depends on the frequency ω.

From now on the subscript $\cdot_n$ indicates that the object is confined to the non-dispersive material and the subscript $\cdot_d$ indicates connection to the dispersive material. For example the unit cell Ω is split into the non-dispersive part Ω_n and the dispersive part Ω_d. The electric permittivity in the non-dispersive media, always set to $\epsilon_n = 1$ in this thesis, is independent of the frequency, but ϵ_d depends on ω.

Such a permittivity can be modelled by the Drude model

$$\epsilon_d(w) := 1 - \frac{\delta_p^2}{\omega(\omega + i\gamma_p)}$$

or the more involved Drude-Lorentz model

$$\epsilon_d(w) := \epsilon_\infty - \frac{\delta_p^2}{\omega(\omega + i\gamma_p)} + \sum_{j=1}^{2} \delta_j \rho_j \left(\frac{\mathrm{e}^{i\beta_j}}{\delta_j - \omega - i\gamma_j} + \frac{\mathrm{e}^{-i\beta_j}}{\delta_j + \omega + i\gamma_j} \right).$$

$\epsilon_\infty, \delta_p, \gamma_p, \delta_j, \rho_j, \beta_j, \gamma_j$ are material constants [ELRM06, ELRM07]. With these models, the resonance problems in Definition 1 and Definition 2 evolve into the non-linear resonance problems [HLM16]:

Definition 3 (Resonance problem in a MPHC for the electric field)
For $\mathbf{k}$ *in the first Brillouin zone, find* $(\omega_\mathbf{k}, \mathbf{E}_\mathbf{k}), \omega_\mathbf{k} \in \mathbb{C}, \mathbf{E}_\mathbf{k} \neq 0$ *such that* $\mathbf{E}_\mathbf{k}$ *is Bloch periodic and satisfies*

$$\operatorname{curl}\operatorname{curl} \mathbf{E}_\mathbf{k}(\mathbf{r}) = \omega_\mathbf{k}^2 \epsilon(\mathbf{r}, \omega_\mathbf{k}) \mathbf{E}_\mathbf{k}(\mathbf{r}), \qquad \mathbf{r} \in \Omega. \tag{3}$$

Definition 4 (Resonance problem in a MPHC for the magnetic field)
For $\mathbf{k}$ *in the first Brillouin zone, find* $(\omega_\mathbf{k}, \mathbf{H}_\mathbf{k}), \omega_\mathbf{k} \in \mathbb{C}, \mathbf{H}_\mathbf{k} \neq 0$ *such that* $\mathbf{H}_\mathbf{k}$ *is Bloch periodic and satisfies*

$$\operatorname{curl} \epsilon(\mathbf{r}, \omega_\mathbf{k})^{-1} \operatorname{curl} \mathbf{H}_\mathbf{k}(\mathbf{r}) = \omega_\mathbf{k}^2 \mathbf{H}_\mathbf{k}(\mathbf{r}), \qquad \mathbf{r} \in \Omega. \tag{4}$$

2.1 2D metallic photonic crystal

In case of a 2D metallic photonic crystal, further simplifications can be made. We will address the different coordinates of a specific vector or field $\mathbf{F}$ by $\mathbf{F} := (\mathbf{F}_x, \mathbf{F}_y, \mathbf{F}_z)^\top$. Because of the materials infinite extension in the z-direction, the solutions z-dependency is separable and has the form of a plane wave. In the other two spatial dimensions the same argument as in a three dimensional MPHC can be made. The resonance functions have the form

$$\mathbf{f}_\mathbf{k}(x, y, z) := \mathrm{e}^{i\mathbf{k}_z z}\, \mathrm{e}^{i(\mathbf{k}_x x + \mathbf{k}_y y)}\, \mathbf{u}_\mathbf{k}(x, y)$$

with $\mathbf{u}_\mathbf{k}(x, y)$ being a periodic function on the lattice projected onto the x, y plane. For simplifications in 2D photonic crystal the z-coordinate of the wave vector is set to zero ($\mathbf{k}_z := 0$). This reduces the three dimensional first Brillouin zone to a two dimensional projection, resulting in solutions of the form

$$\mathbf{f}_\mathbf{k}(x, y, z) = \mathbf{f}_\mathbf{k}(x, y) = \mathrm{e}^{i(\mathbf{k}_x x + \mathbf{k}_y y)}\, \mathbf{u}_\mathbf{k}(x, y).$$

Applying the curl curloperator on such a function looks like

$$\operatorname{curl}\operatorname{curl} \mathbf{f}_\mathbf{k} := \begin{pmatrix} \partial_y \partial_x \mathbf{f}_{\mathbf{k},y} - \partial_y \partial_y \mathbf{f}_{\mathbf{k},x} \\ -\partial_x \partial_y \mathbf{f}_{\mathbf{k},x} - \partial_x \partial_x \mathbf{f}_{\mathbf{k},y} \\ -\partial_x \partial_x \mathbf{f}_{\mathbf{k},z} - \partial_y \partial_y \mathbf{f}_{\mathbf{k},z} \end{pmatrix}.$$

In the third row the negative laplacian $-\Delta_{x,y} := -\partial_x \partial_x - \partial_y \partial_y$ remains and is applied to $\mathbf{f}_{\mathbf{k},z}$ the z-coordinate of the field. The resonance problems can be stated with the z-component of the vector fields. The problem in Definition 3 boils down to:

Definition 5 (E-field resonance problem in a two dimensional MPHC) *For* $\mathbf{k}$ *in the 2D first Brillouin zone, find* $(\omega_{\mathbf{k}}, \mathbf{E}_{\mathbf{k},z}), \omega_{\mathbf{k}} \in \mathbb{C}, \mathbf{E}_{\mathbf{k},z} \neq 0$ *such that* $\mathbf{E}_{\mathbf{k},z}$ *is Bloch periodic in two dimensions and satisfies*

$$-\Delta_{x,y}\mathbf{E}_{\mathbf{k},z}(x,y) = \omega_{\mathbf{k}}^2 \epsilon(x,y,\omega_{\mathbf{k}})\mathbf{E}_{\mathbf{k},z}(x,y), \qquad (x,y,0)^\top \in \Omega. \tag{5}$$

The second problem in Definition 4 to:

Definition 6 (H-field resonance problem in a two dimensional MPHC) *For* $\mathbf{k}$ *in the 2D first Brillouin zone, find* $(\omega_{\mathbf{k}}, \mathbf{H}_{\mathbf{k},z}), \omega_{\mathbf{k}} \in \mathbb{C}, \mathbf{H}_{\mathbf{k},z} \neq 0$ *such that* $\mathbf{H}_{\mathbf{k},z}$ *is Bloch periodic in two dimensions and satisfies*

$$-\nabla_{x,y} \cdot \epsilon(x,y,\omega_{\mathbf{k}})^{-1}\nabla_{x,y}\mathbf{H}_{\mathbf{k},z}(x,y) = \omega_{\mathbf{k}}^2 \mathbf{H}_{\mathbf{k},z}(x,y), \qquad (x,y,0)^\top \in \Omega. \tag{6}$$

The operator $\nabla_{x,y}$ is defined by $\nabla_{x,y} := (\partial_x, \partial_y)^\top$.

Solutions of Definition 5 are called transversal magnetic (TM) modes, because the magnetic field $\mathbf{H}$ is orthogonal to the z-direction and solutions of Definition 6 are called transversal electric (TE) modes [MEH02, IS01].

3 Finite elements for photonic crystals

In Section 2 we ended up with non-linear resonance problems for 2D and 3D metallic photonic crystals, for which we would like to know the resonances. The analytic calculation of them is, most of the time, not possible. This issue is solved in two steps. First the problems are discretised with the finite element method, resulting in non-linear matrix eigenvalue problems. These are still not analytically solvable. In the second step, the matrix eigenvalue problems are solved with iterative methods. The iterative methods are described in the Sections 4 and 5. The first step, discretisation into a non-linear matrix eigenvalue problem, is the focus of this section.

3.1 Finite element discretisation for 3D metallic photonic crystals

At first, the resonance problem in Definition 3 is transferred into a Hilbert space setting. The space has to contain functions defined on the unit cell Ω, that are Bloch periodic on the boundary and smooth enough to apply the curloperator. A space which holds functions with a curl, at least in a weak sense, is according to [Néd80, Néd86, Zag06]

$$\mathcal{H}(\mathrm{curl}, \Omega) := \{\mathbf{f} \in \mathcal{L}_2(\Omega, \mathbb{C})^3 : \mathrm{curl}\, \mathbf{f} \in \mathcal{L}_2(\Omega, \mathbb{C})^3\}.$$

For functions in this space only the tangential trace $\mathbf{f} \times \nu$ is defined on the boundary with ν being the outward pointing normal vector. Therefore, we can only claim tangential Bloch periodicity on the boundary. The final space is

$$\mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega) := \{\mathbf{f} \in \mathcal{H}(\mathrm{curl}, \Omega) : \mathbf{f} \times \nu \text{ Bloch periodic on } \Gamma\}.$$

Remark 4 *The Hilbert space $\mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega)$ changes with each wave vector, because the boundary condition changes. This is indicated by the subscript that the space carries. Although the weak formulation for each wave vectors looks the same it is not, because the solution space itself changes.*

We define the complex conjugation of $c \in \mathbb{C}$ by the symbol $\overline{c}$. For a vector field the complex conjugation is applied to each component. The weak formulation of Equation (3) is derived by partial integration and looks like

$$\int_\Omega \mathrm{curl}\, \mathbf{E}_{\mathbf{k}} \cdot \overline{\mathrm{curl}\, \mathbf{v}_{\mathbf{k}}}\, \mathrm{d}\mathbf{r} = \omega_{\mathbf{k}}^2 \int_\Omega \epsilon(., \omega_{\mathbf{k}}) \mathbf{E}_{\mathbf{k}} \cdot \overline{\mathbf{v}_{\mathbf{k}}}\, \mathrm{d}\mathbf{r}, \quad \forall \mathbf{v}_{\mathbf{k}} \in \mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega), \tag{7}$$

with $\mathbf{E}_{\mathbf{k}} \in \mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega)$ and the weak formulation of Equation (4) is

$$\int_\Omega \epsilon(., \omega_{\mathbf{k}})^{-1}\, \mathrm{curl}\, \mathbf{H}_{\mathbf{k}} \cdot \overline{\mathrm{curl}\, \mathbf{v}_{\mathbf{k}}}\, \mathrm{d}\mathbf{r} = \omega_{\mathbf{k}}^2 \int_\Omega \mathbf{H}_{\mathbf{k}} \cdot \overline{\mathbf{v}_{\mathbf{k}}}\, \mathrm{d}\mathbf{r}, \quad \forall \mathbf{v}_{\mathbf{k}} \in \mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega), \tag{8}$$

with $\mathbf{H}_{\mathbf{k}} \in \mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega)$.

The finite element method (FEM) uses a set of basis functions $\phi_j \in \mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega), j \in \{1, \ldots, N\}$, spanning the space $\mathcal{V}_{\mathbf{k},h} := [\phi_1, \ldots, \phi_N] \subset \mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega)$. The problem is projected on to this finite dimensional subspace.

Remark 5 *The subscript indicating the wave vector* $\mathbf{k}$ *dependency will be omitted at the FEM discretisation matrices later. It should not be forgotten that they change with each wave vector.*

The basis functions need to be elements of the Hilbert space, which means they have to be Bloch periodic. In the FEM software **Netgen/NGSolve** [Sch97, Sch14] this space $\mathcal{V}_{\mathbf{k},h}$ can be generated.

Quasi periodic geometries and spaces in Netgen/NGSolve: An example code for a 2D quasi periodic square region is illustrated in Listing 1. First, the geometry is generated. The 'periodic' keywords, in rows 4 to 7, indicate the periodicity. The parameters 'copy=bottom' tells the newly added line that it actually is a copy of the line 'bottom'. Here the orientation is important, because the mesh is copied from one line to the other. In row 9 the periodicity coefficients are specified. Note that each number corresponds to a line in the geometry sorted in the same way as the lines were added before. The first two coefficients are zero or could be anything, because they are not used. These lines in the geometry are original and not copied. The last two coefficients carry the quasi periodicity coefficients and the finite element space is generated such that the basis function are quasi periodic over the specified two lines. In the end, the finite element space is generated in row 10 with the quasi periodicity factors.

For 3D geometries the definition is a little different. An example of a simple cube in three dimensions can be seen in Listing 2. The first nine rows are standard 3D geometry generation in **Netgen/NGSolve**. To add periodic surfaces the commands in line 10 to 12 are needed. The first argument indicates the surface to copy, the second to which surface it is copied and the third specifies along

```
geo = SplineGeometry()
pnts = [ (-a/2,-a/2), (a/2,-a/2), (a/2,a/2), (-a/2,a/2) ]
pnums = [geo.AppendPoint(*p) for p in pnts]
bottom = geo.Append ( ["line", pnums[0], pnums[1]],bc="periodic")
right = geo.Append ( ["line", pnums[1], pnums[2]], bc="periodic")
geo.Append ( ["line", pnums[3], pnums[2]], leftdomain=0, rightdomain
    =1, copy=bottom, bc="periodic")
geo.Append ( ["line", pnums[0], pnums[3]], leftdomain=0, rightdomain
    =1, copy=right, bc="periodic")
mesh = Mesh(geo.GenerateMesh(maxh=h))
factors = [0,0,f1,f2]
fes = Periodic(H1(mesh,order=p,complex=True),phase=factors)
```

Listing 1 Generating a 2D quasi periodic square in **Netgen/NGSolve**.

```
left = Plane(Pnt(0,0,0),Vec(-1,0,0))
right = Plane(Pnt(1,0,0),Vec(1,0,0))
bot = Plane(Pnt(0,0,0),Vec(0,0,-1))
top = Plane(Pnt(0,0,1),Vec(0,0,1))
back = Plane(Pnt(0,0,0),Vec(0, -1, 0))
front = Plane(Pnt(0,1,0),Vec(0, 1, 0))
cube = left * right * top * bot * back * front
geo = CSGeometry()
geo.Add(cube)
geo.PeriodicSurfaces(left,right,Trafo(Vec(1,0,0)))
geo.PeriodicSurfaces(back,front,Trafo(Vec(0,1,0)))
geo.PeriodicSurfaces(bot,top,Trafo(Vec(0,0,1)))
mesh = Mesh(geo.GenerateMesh(maxh=h))
factors = [f1,f2,f3]
fes = Periodic(HCurl(mesh,order=p,complex=True),phase=factors)
```

Listing 2 Generating a 3D quasi periodic cube in **Netgen/NGSolve**.

which direction the surface is moved. This is necessary for geometries with skew faces. For example the lattice in Figure 3 has such faces. Generating the mesh and the space are almost the same as in the 2D case. The only difference is that the factors for the periodicity need to be ordered in the same way as the periodic surfaces have been added in rows 12 to 14.

Later on the finite element method matrices $\mathrm{A}, \mathrm{A}_d, \mathrm{M}, \mathrm{M}_d \in \mathbb{C}^{N\times N}$ defined by

$$(\mathrm{A})_{i,j} := \int_\Omega \operatorname{curl}\phi_i \cdot \overline{\operatorname{curl}\phi_j}\,\mathrm{d}\mathbf{r}, \qquad (\mathrm{A}_d)_{i,j} := \int_{\Omega_d} \operatorname{curl}\phi_i \cdot \overline{\operatorname{curl}\phi_j}\,\mathrm{d}\mathbf{r},$$
$$(\mathrm{M})_{i,j} := \int_\Omega \phi_i \cdot \overline{\phi_j}\,\mathrm{d}\mathbf{r}, \qquad (\mathrm{M}_d)_{i,j} := \int_{\Omega_d} \phi_i \cdot \overline{\phi_j}\,\mathrm{d}\mathbf{r}$$

for $i, j \in \{1, \ldots, N\}$, will be needed. The finite element basis functions ϕ_j have local support and therefore, the matrices, although their dimension N may be

large, are sparse. Especially A_d and M_d have zero entries for all basis function without support in the dispersive material Ω_d.

3.1.1 Drude model discretisation

Using the Drude model for ϵ, Equation (7) and (8) look like

$$\int_\Omega \operatorname{curl} \mathbf{E_k} \cdot \overline{\operatorname{curl} \mathbf{v_k}} \, \mathrm{d}\mathbf{r} = \omega_\mathbf{k}^2 \int_\Omega \mathbf{E_k} \cdot \overline{\mathbf{v_k}} \, \mathrm{d}\mathbf{r} - \omega_\mathbf{k} \frac{\delta_p^2}{\omega_\mathbf{k} + i\gamma_p} \int_{\Omega_d} \mathbf{E_k} \cdot \overline{\mathbf{v_k}} \, \mathrm{d}\mathbf{r}, \tag{9}$$

$$\int_\Omega \operatorname{curl} \mathbf{H_k} \cdot \overline{\operatorname{curl} \mathbf{v_k}} \, \mathrm{d}\mathbf{r} + \frac{\delta_p^2}{\beta(\omega_\mathbf{k})} \int_{\Omega_d} \operatorname{curl} \mathbf{H_k} \cdot \overline{\operatorname{curl} \mathbf{v_k}} \, \mathrm{d}\mathbf{r} = \omega_\mathbf{k}^2 \int_\Omega \mathbf{H_k} \cdot \overline{\mathbf{v_k}} \, \mathrm{d}\mathbf{r}, \tag{10}$$

with $\beta(\omega_\mathbf{k}) := \omega_\mathbf{k}(\omega_\mathbf{k} + i\gamma_p) - \delta_p^2$. Using the discretisation with FEM spaces, this results in

$$\mathrm{A}u = \omega^2 \mathrm{M}u - \omega \frac{\delta_p^2}{\omega + i\gamma_p} \mathrm{M}_d u, \qquad \mathrm{A}u + \frac{\delta_p^2}{\omega(\omega + i\gamma_p) - \delta_p^2} \mathrm{A}_d u = \omega^2 \mathrm{M}u, \tag{11}$$

with $\omega \in \mathbb{C}, u \in \mathbb{C}^N \setminus \{0\}$. To calculate the band structure these non-linear matrix eigenvalue problems need to be solved for wave vector $\mathbf{k}$ in the first Brillouin zone.

3.1.2 Drude-Lorenz model discretisation

The Drude-Lorenz model is more evolved than the Drude model, luckily the same FEM matrices can be used. Applying the same steps as for the Drude model to Equation (7) lead to

$$\mathrm{A}u = \omega^2 \mathrm{M}u - \omega^2 \left(1 - \epsilon_\infty + \frac{\delta_p^2}{\alpha(\omega)} + \sum_{j=1}^{2} \delta_j \rho_j \left(\frac{\mathrm{e}^{i\beta_j}}{\alpha_j - \omega} + \frac{\mathrm{e}^{-i\beta_j}}{\overline{\alpha_j} + \omega} \right) \right) \mathrm{M}_d u, \tag{12}$$

with $\alpha_j := \delta_j - i\gamma_j$ and $\alpha(\omega) := \omega(\omega + i\gamma_p)$. The discrete eigenvalue problem for the magnetic field is not stated here, because of the structure of the electric permittivity of the Drude-Lorenz model.

3.2 Finite element discretisation for 2D metallic photonic crystals

The discretisation of 2D metallic photonic crystals follows the same steps with some slight differences. Matrix and function names from the 3D crystal will be reused. First, Equation (5) and Equation (6) are transferred into a Hilbert space setting.

A space which holds function with a gradient is

$$\mathcal{H}^1(\Omega) := \{ f \in \mathcal{L}_2(\Omega, \mathbb{C}) : \nabla f \in \mathcal{L}_2(\Omega, \mathbb{C})^2 \}.$$

For functions in this space boundary evaluation integrals are well defined. Therefore, we can claim Bloch periodicity on them. The final space is

$$\mathcal{H}^1_\mathbf{k}(\Omega) := \{ f \in \mathcal{H}^1(\Omega) : f \text{ Bloch periodic on } \Gamma \}.$$

Remark 6 *As in the case of $\mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega)$, the Hilbert space $\mathcal{H}^1_{\mathbf{k}}(\Omega)$ depends on the wave vector.*

The weak formulations in this space look like

$$\int_\Omega \nabla_{x,y} E_{\mathbf{k},z} \cdot \overline{\nabla_{x,y} v_{\mathbf{k}}} \, \mathrm{d}(x,y) = \omega_{\mathbf{k}}^2 \int_\Omega \epsilon(.,\omega_{\mathbf{k}}) E_{\mathbf{k},z} \overline{v_{\mathbf{k}}} \, \mathrm{d}(x,y), \qquad \forall v_{\mathbf{k}} \in \mathcal{H}^1_{\mathbf{k}}(\Omega),$$
$$\int_\Omega \epsilon(.,\omega_{\mathbf{k}})^{-1} \nabla_{x,y} H_{\mathbf{k},z} \cdot \overline{\nabla_{x,y} v_{\mathbf{k}}} \, \mathrm{d}(x,y) = \omega_{\mathbf{k}}^2 \int_\Omega H_{\mathbf{k},z} \overline{v_{\mathbf{k}}} \, \mathrm{d}(x,y), \quad \forall v_{\mathbf{k}} \in \mathcal{H}^1_{\mathbf{k}}(\Omega),$$

with $E_{\mathbf{k},z}, H_{\mathbf{k},z} \in \mathcal{H}^1_{\mathbf{k}}(\Omega)$. Let $\phi_j \in \mathcal{H}^1_{\mathbf{k}}(\Omega), j \in \{1, \ldots, N\}$ be the finite element basis functions and defining the matrices $\mathrm{A}, \mathrm{A}_d, \mathrm{M}, \mathrm{M}_d \in \mathbb{C}^{N \times N}$ by

$$(\mathrm{A})_{i,j} := \int_\Omega \nabla_{x,y} \phi_i \cdot \overline{\nabla_{x,y} \phi_j} \, \mathrm{d}(x,y), \qquad (\mathrm{A}_d)_{i,j} := \int_{\Omega_d} \nabla_{x,y} \phi_i \cdot \overline{\nabla_{x,y} \phi_j} \, \mathrm{d}(x,y),$$
$$(\mathrm{M})_{i,j} := \int_\Omega \phi_i \overline{\phi_j} \, \mathrm{d}(x,y), \qquad (\mathrm{M}_d)_{i,j} := \int_{\Omega_d} \phi_i \overline{\phi_j} \, \mathrm{d}(x,y),$$

for $i, j \in \{1, \ldots, N\}$. These matrices have a sparse structure for the same reason as the matrices in the 3D case. Using the Drude and the Drude-Lorentz model result in exactly the same matrix eigenvalue problems as in the 3D case, the Equations (11) and (12), with the matrices defined here.

We arrived at non-linear matrix eigenvalue problems, which we got by applying the finite element method to the non-linear resonance problems. Additionally, we noticed that, although 2D and 3D lattices have different Hilbert spaces, their according matrix eigenvalue problem structures are comparable.

4 Arnoldi with a rational linearisation

In the last section we derived non-linear matrix eigenvalue problems, more specific rational eigenvalue problems. The question is how do we solve them in an efficient way. They are large and calculating all eigenvalues will not be possible most of the time. The approach is to calculate a small amount of eigenvalues in a desirable vicinity. This is done by linearisation and applying the shift-and-invert Arnoldi solver.

Rational eigenvalue problems can be linearised in many ways [SB11]. A common approach is to transform them into polynomial problems, linearise and solve with a linear eigenvalue solver. For the linear eigenvalue solver the Arnoldi algorithm [Arn51] has been chosen. But instead of transforming it into a polynomial problem, an alternative linearisation, which is directly applied to the rational problem, is used. For the purpose of comparison also the polynomial approach has been applied to the problems.

4.1 Shift-and-invert Arnoldi method

We introduce the shift-and-invert Arnoldi algorithm for calculating a small amount of eigenpairs in the vicinity of the shift $\sigma \in \mathbb{C}$ of the generalised eigenvalue prob-

lem

$$\mathrm{A}x = \lambda \mathrm{B}x \qquad\qquad \mathrm{A}, \mathrm{B} \in \mathbb{C}^{n\times n}, n \in \mathbb{N}.$$

The Arnoldi algorithm calculates the biggest eigenvalues first. To get eigenvalues in the vicinity of σ, a shift and an inversion has to be made resulting in

$$(\mathrm{A} - \sigma\mathrm{B})^{-1}\mathrm{B}x = \mu x, \qquad\qquad \lambda = \sigma + \frac{1}{\mu}.$$

In the new eigenvalue problem the biggest eigenvalues μ correspond to the eigenvalues λ closest to σ. Let us define $\mathrm{C} := (\mathrm{A} - \sigma\mathrm{B})$.

The idea of the Arnoldi algorithm is to restrict the matrix $\mathrm{C}^{-1}\mathrm{B}$ to the Krylov subspace $\mathcal{K}_l := \{x_0, \mathrm{C}^{-1}\mathrm{B}x_0, \ldots, (\mathrm{C}^{-1}\mathrm{B})^{l-1}x_0\}$ and solve the smaller problem. Then the calculated eigenvalues are used and transformed by $\lambda = \sigma + \frac{1}{\mu}$ into approximate eigenvalues of (A, B). The vector $x_0 \in \mathbb{C}^n$ is an arbitrary starting vector.

At first, an orthonormal basis $\mathrm{V}_l := \{v_0, v_1, \ldots, v_{l-1}\}$ of $\mathcal{K}_l$ is needed and then $\mathrm{C}^{-1}\mathrm{B}$ is restricted to

$$\mathrm{T}_l = \mathrm{V}_l^T\mathrm{C}^{-1}\mathrm{B}\mathrm{V}_l \qquad\qquad \mathrm{T}_l \in \mathbb{C}^{l\times l}.$$

Next, the small eigenvalue problem for matrix T_l is solved.

The orthonormal basis is generated by orthonormalizing the next Krylov subspace vector against each preceding one, and adding it to the already existing basis. The matrix $\mathrm{C}^{-1}\mathrm{B}$ is applied to the new basis vector, or better, the system of linear equations $\mathrm{C}r_j = \mathrm{B}v_{j-1}$ is solved. The vector r_j is the next candidate for a basis vector and has to be orthogonalised. This algorithm results in the recursion

$$\begin{aligned} \mathrm{C}r_j &= \mathrm{B}v_{j-1}, \\ t_{j+1,j}v_j &= r_j - \sum_{l=0}^{j-1} t_{l+1,j}v_l, \end{aligned}$$

for $j \in \{1, \ldots, l-1\}$, with $t_{l+1,j} := v_l \cdot r_j, l \leq j-1$ the orthogonalisation coefficients and $t_{j+1,j}$ the coefficient to normalise the new vector. The scalar product is the Euclidean product in $\mathbb{C}^n$.

The coefficients have not been named by chance. It turns out they are the entries of T_l at the corresponding position described by their indices. All positions of T which are not defined are zero, giving it a Hessenberg structure.

An algorithm based on these systems of equations can be seen in Algorithm 1.

If the matrix B is hermitian and positive definite Euclidian, the Euclidian scalar product can be exchanged with the by B induced scalar product. If, additionally, matrix A is symmetric, the algorithm becomes the shift-and-invert Lanczos method.

4.2 Polynomial linearisation schema

The rational eigenvalue problems in Section 3 can be transformed into polynomial eigenvalue problems by multiplying with the denominators. The general form of

Algorithm 1 Shift-and-invert Arnoldi method

Require: $\mathrm{A}, \mathrm{B} \in \mathbb{C}^{n\times n}$, Start-vector $r_0 \in \mathbb{C}^n \setminus \{0\}$

1: $t_{1,0} = (r_0 \cdot r_0)^{1/2}$
2: j $= 0$
3: **while** $j <$ maximum number of iterations **do**
4: $\quad v_j = r_j / t_{j+1,j}$
5: $\quad (\mathrm{A} - \sigma\mathrm{B}) r_{j+1} = \mathrm{B} v_j$ (solve system of linear equations)
6: $\quad l = 0$
7: $\quad$ **while** $l \leq j$ **do**
8: $\quad\quad t_{l+1,j+1} = v_l \cdot r_{j+1}$
9: $\quad\quad r_{j+1} = r_{j+1} - t_{l+1,j+1} v_j$
10: $\quad t_{j+2,j+1} = (r_{j+1} \cdot r_{j+1})^{1/2}$
11: $\quad j = j + 1$
12: $\quad$ Compute the eigenvalues of T_j and the corresponding error bounds.
13: $\quad$ Stop if enough eigenvalues have been found.

polynomial eigenvalue problems look like: Find pairs (ω, u) with $\omega \in \mathbb{C}, u \in \mathbb{C}^N \setminus \{0\}$ such that

$$\mathrm{T}(\omega)u = \sum_{j=0}^{k} \omega^j \mathrm{A}_j u = 0, \qquad \mathrm{A}_j \in \mathbb{C}^{N\times N}.$$

The polynomial eigenvalue problem is linearised by introducing new vectors $y_0 := u, y_j := \omega y_{j-1}, j \in \{1, \ldots, k-1\}$. Substituting u with the vectors y_j leads to the system of equations

$$\mathrm{A}_0 y_0 + \omega \sum_{j=1}^{k} \mathrm{A}_j y_{j-1} = 0.$$

Combining this system of equations with the definition of the vectors y_j results in the bigger linear eigenvalue problem $\tilde{\mathrm{A}}\tilde{x} = \omega\tilde{\mathrm{B}}\tilde{x}$ with $\tilde{x} := (y_0, \ldots, y_{k-1})^\top$ and the matrices

$$\tilde{\mathrm{A}} := \begin{pmatrix} \mathrm{A}_0 & 0 & \cdots & 0 \\ 0 & \mathrm{I} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathrm{I} \end{pmatrix}, \qquad \tilde{\mathrm{B}} := \begin{pmatrix} -\mathrm{A}_1 & -\mathrm{A}_2 & \ldots & -\mathrm{A}_k \\ \mathrm{I} & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ 0 & 0 & \mathrm{I} & 0 \end{pmatrix}.$$

The shift-and-invert Arnoldi method needs the with σ shifted and then inverted matrix $(\tilde{\mathrm{A}} - \sigma\tilde{\mathrm{B}})^{-1}$ at each iteration step, as can be seen in Algorithm 1 Line 5.

Usually, the inversion of a matrix k-times larger is more expensive, but the lower, right part of the shifted matrix can be analytically inverted:

$$\begin{pmatrix} \mathrm{I} & 0 & \cdots & \cdots & 0 \\ -\sigma\mathrm{I} & \mathrm{I} & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathrm{I} & 0 \\ 0 & \cdots & 0 & -\sigma\mathrm{I} & \mathrm{I} \end{pmatrix}^{-1} = \begin{pmatrix} \mathrm{I} & 0 & \cdots & \cdots & 0 \\ \sigma\mathrm{I} & \mathrm{I} & \ddots & \ddots & \vdots \\ \sigma^2\mathrm{I} & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathrm{I} & 0 \\ \sigma^{k-1}\mathrm{I} & \cdots & \sigma^2\mathrm{I} & \sigma\mathrm{I} & \mathrm{I} \end{pmatrix}.$$

An ideal case to apply the Schur complement

$$\begin{pmatrix} \mathrm{C}_{11} & \mathrm{C}_{12} \\ \mathrm{C}_{21} & \mathrm{C}_{22} \end{pmatrix}^{-1} = \begin{pmatrix} \mathrm{I} & 0 \\ -\mathrm{C}_{22}^{-1}\mathrm{C}_{21} & \mathrm{I} \end{pmatrix} \begin{pmatrix} (\mathrm{C}_{11} - \mathrm{C}_{12}\mathrm{C}_{22}^{-1}\mathrm{C}_{21})^{-1} & 0 \\ 0 & \mathrm{C}_{22}^{-1} \end{pmatrix} \begin{pmatrix} \mathrm{I} & -\mathrm{C}_{12}\mathrm{C}_{22}^{-1} \\ 0 & \mathrm{I} \end{pmatrix}$$

with matrices $\mathrm{C}_{11} := \mathrm{A}_0 + \sigma\mathrm{A}_1$, $\mathrm{C}_{12} := \begin{pmatrix} \sigma\mathrm{A}_2 & \sigma\mathrm{A}_3 & \dots & \sigma\mathrm{A}_k \end{pmatrix}$ and

$$\mathrm{C}_{21} := \begin{pmatrix} -\sigma\mathrm{I} \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \qquad \mathrm{C}_{22} := \begin{pmatrix} \mathrm{I} & 0 & \cdots & \cdots & 0 \\ -\sigma\mathrm{I} & \mathrm{I} & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathrm{I} & 0 \\ 0 & \cdots & 0 & -\sigma\mathrm{I} & \mathrm{I} \end{pmatrix}.$$

Only systems of linear equations with the matrix $\mathrm{T}(\sigma) = \mathrm{C}_{11} - \mathrm{C}_{12}\mathrm{C}_{22}^{-1}\mathrm{C}_{21}$ have to be numerically solved.

4.3 Rational linearisation schema for the Drude model in electric form

The form of the matrix eigenvalue problem with the Drude model is independent of the problem dimension. Therefore, a general schema for both cases can be developed. The eigenvalue problem for the Drude model in electric form

$$\mathrm{A}u = \omega^2\mathrm{M}u - \omega\frac{\delta_p^2}{\omega + i\gamma_p}\mathrm{M}_d u,$$

can be linearised by introducing new vectors $v := \omega u$ and $x := \frac{1}{\omega + i\gamma_p}u$. The variable x is an unusual substitution, but with it the rational part is directly linearised. Otherwise, the whole system of equations would have to be multiplied by the denominator. Substituting the vectors leads to

$$\mathrm{A}u = \omega\mathrm{M}v - \omega\delta_p^2\mathrm{M}_d x.$$

Combining the new system of equations with the definitions of v and x leads to the bigger linear eigenvalue problem $\tilde{\mathrm{A}}\tilde{x} = \omega\tilde{\mathrm{B}}\tilde{x}$ with the matrices

$$\tilde{\mathrm{A}} := \begin{pmatrix} \mathrm{A} & 0 & 0 \\ 0 & \mathrm{I} & 0 \\ \mathrm{I} & 0 & -i\gamma_p\mathrm{I} \end{pmatrix}, \qquad \tilde{\mathrm{B}} := \begin{pmatrix} 0 & \mathrm{M} & -\delta_p^2\mathrm{M}_d \\ \mathrm{I} & 0 & 0 \\ 0 & 0 & \mathrm{I} \end{pmatrix}$$

and the vector $\tilde{x} := (u, v, x)^\top$.

As in the polynomial linearisation the lower right part of the shifted matrix can be analytically inverted

$$\begin{pmatrix} \mathrm{I} & 0 \\ 0 & -(\sigma + i\gamma_p)\mathrm{I} \end{pmatrix}^{-1} = \begin{pmatrix} \mathrm{I} & 0 \\ 0 & -\frac{1}{\sigma + i\gamma_p}\mathrm{I} \end{pmatrix}.$$

```
v0 = vec[0].CreateVector()
v0.data = M*vec[1] - op*op*Md*vec[2]
v1 = vec[1].CreateVector()
v1.data = vec[0]
v2 = vec[2].CreateVector()
v2.data = vec[2]

v0.data += sig*M*v1 + sig*op*op/(sig + 1J*gp)*Md*v2

v0.data = invS*v0
v2.data = (-1)/(sig + 1J*gp)*v2

v1.data += sig*v0
v2.data += 1/(sig + 1J*gp)*v0
```

Listing 3 Applying $(\tilde{\mathrm{A}} - \sigma\tilde{\mathrm{B}})^{-1}\tilde{\mathrm{B}}$ to a vector in **Netgen/NGSolve**

Applying the Schur complement results in

$$
\begin{aligned}
(\tilde{\mathrm{A}} - \sigma\tilde{\mathrm{B}})^{-1} &= \begin{pmatrix} \mathrm{A} & -\sigma\mathrm{M} & \sigma\delta_p^2\mathrm{M}_d \\ -\sigma\mathrm{I} & \mathrm{I} & 0 \\ \mathrm{I} & 0 & -(\sigma + i\gamma_p)\mathrm{I} \end{pmatrix}^{-1} \\
&= \begin{pmatrix} \mathrm{I} & 0 & 0 \\ \sigma\mathrm{I} & \mathrm{I} & 0 \\ \frac{1}{\sigma+i\gamma_p}\mathrm{I} & 0 & \mathrm{I} \end{pmatrix} \begin{pmatrix} \mathrm{S}(\sigma)^{-1} & 0 & 0 \\ 0 & \mathrm{I} & 0 \\ 0 & 0 & -\frac{1}{\sigma+i\gamma_p}\mathrm{I} \end{pmatrix} \begin{pmatrix} \mathrm{I} & \sigma\mathrm{M} & \sigma\frac{\delta_p^2}{\sigma+i\gamma_p}\mathrm{M}_d \\ 0 & \mathrm{I} & 0 \\ 0 & 0 & \mathrm{I} \end{pmatrix},
\end{aligned}
$$

with the matrix valued function $\mathrm{S}(\omega) := \mathrm{A} - \omega^2\mathrm{M} + \omega\frac{\delta_p^2}{\omega+i\gamma_p}\mathrm{M}_d, \omega \in \mathbb{C}$. Note that the original matrix eigenvalue problem is $\mathrm{S}(\omega)u = 0$.

Although a big eigenvalue problem is created by rational linearisation, only an inverse with the original problem size has to be calculated. The rest is handled by matrix vector multiplications. The implementation of applying $(\tilde{\mathrm{A}} - \sigma\tilde{\mathrm{B}})^{-1}\tilde{\mathrm{B}}$ to the last Krylov vector can be seen in Listing 3.

The modified shift-and-invert Arnoldi has vectors of three times the finite element space size. Therefore, calculating scalar products is three times more expensive. As is the whole orthogonalisation process. The cost to calculate the eigenvalues of the small matrix T_l stays the same. Applying $(\tilde{\mathrm{A}} - \sigma\tilde{\mathrm{B}})^{-1}\tilde{\mathrm{B}}$ requires four more matrix vector multiplications and three vector scalings. In the matter of storage, one additional matrix has to be stored and each vector is three times the usual size.

4.4 Rational linearisation schema for the Drude model in magnetic form

There also exists a linearisation for the magnetic form

$$\mathrm{A}u + \frac{\delta_p^2}{\omega(\omega + i\gamma_p) - \delta_p^2}\mathrm{A}_d u = \omega^2 \mathrm{M}u,$$

which yields the same benefits as the one for the electric form. At first, the denominator has to be factorised into $\omega(\omega + i\gamma_p) - \delta_p^2 = (\omega - \omega_1)(\omega - \omega_2)$ and introducing new vectors $v := \omega u$, $x := \frac{1}{\omega - \omega_1}u$, $y := \frac{1}{\omega - \omega_2}x$ leads to

$$\mathrm{A}u + \delta_p^2 \mathrm{A}_d y = \omega \mathrm{M}v,$$

which in combination with the definition of v, x, y has the form of the linear eigenvalue problem

$$\begin{pmatrix} \mathrm{A} & 0 & 0 & \delta_p^2\mathrm{A}_d \\ 0 & \mathrm{I} & 0 & 0 \\ \mathrm{I} & 0 & \omega_1\mathrm{I} & 0 \\ 0 & 0 & \mathrm{I} & \omega_2\mathrm{I} \end{pmatrix} \begin{pmatrix} u \\ v \\ x \\ y \end{pmatrix} = \omega \begin{pmatrix} 0 & \mathrm{M} & 0 & 0 \\ \mathrm{I} & 0 & 0 & 0 \\ 0 & 0 & \mathrm{I} & 0 \\ 0 & 0 & 0 & \mathrm{I} \end{pmatrix} \begin{pmatrix} u \\ v \\ x \\ y \end{pmatrix}.$$

This problem is four times the original size. As before the lower, right part of the shifted matrix can be analytically inverted and the Schur complement applied.

4.5 Rational linearisation schema for the Drude-Lorentz model in electric form

Finally, let us have a look at the Drude-Lorentz model and its linearisation.

For the Drude-Lorentz model in electric form,

$$\mathrm{A}u = \omega^2 \mathrm{M}u - \omega^2 \left(1 - \epsilon_\infty + \frac{\delta_p^2}{\alpha(\omega)} - \sum_{j=1}^{2} \delta_j \rho_j \left(\frac{\mathrm{e}^{i\beta_j}}{\alpha_j - \omega} + \frac{\mathrm{e}^{-i\beta_j}}{\overline{\alpha_j} + \omega}\right)\right) \mathrm{M}_d u,$$

a couple more vectors are needed. With the new vectors $v = \omega u, (\omega + i\gamma_p)x = u, (\delta_j - \omega - i\gamma_j)y_j = v, (\delta_j + \omega + i\gamma_j)z_j = v, j \in \{1, 2\}$ the system of equations looks like

$$\mathrm{A}u = \omega\left(\mathrm{M} - (1 - \epsilon_\infty)\,\mathrm{M}_d\right)v + \omega\delta_p^2 \mathrm{M}_d x - \omega \sum_{j=1}^{2} \delta_j \rho_j \left(\mathrm{e}^{i\beta_j}\,\mathrm{M}_d y_j + \mathrm{e}^{-i\beta_j}\,\mathrm{M}_d z_j\right).$$

In combination with the definition of v, x, y_1, y_2, z_1, z_2, it is a linear eigenvalue problem seven times the original size. As in the Drude model, the Schur complement can be used for this linear eigenvalue problem as well.

We have seen a nice property of the rational linearisation. We only need to invert a matrix of the original problem size to be able to apply the Arnoldi method. The drawback is the orthogonalisation has to be done on the bigger vectors.

Remark 7 *The shown linearisation can be applied to every rational eigenvalue problem, because the lower, right part of the linear problem is always analytically invertible. A quick glance at it reveals that, although it looks like a big matrix, the actual dimension is exactly the amount of used help vectors, which should be relatively small and just a small matrix has to be analytically inverted to apply the Schur complement trick.*

Although the explicit dependency on the wave vector **k** has been omitted in this sections, the matrices depend on it.

5 Alternative methods

Apart from the polynomial and rational Arnoldi, there are other solvers for rational eigenvalue problems. One such solver is introduced in [HLM16]. It is a Newton-type method applied to the 3D metallic photonic crystal, as seen in Figure 3. The same crystal has been used for tests in this thesis.

The second solver, Beyn's method [Bey12], belongs to a new family of eigenvalue solvers. These solvers are based upon contour integrals in the complex plane.

5.1 A Newton-type method with non-equivalence deflation

A short introduction to the main cornerstones of this method will be given here. More information can be found in [HLM16].

The algorithm is constructed to calculate one eigenpair at a time for non-linear eigenvalue problems of the form

$$\mathrm{A}x = \omega^2 \mathrm{B}(\omega)x, \qquad \mathrm{A} \in \mathbb{C}^{n\times n}, \forall\omega \in \mathbb{C}, \mathrm{B}(\omega) \in \mathbb{C}^{n\times n},$$

where matrix $\mathrm{B}(\omega)$ is diagonal. Note that the non-linearity is only in matrix $\mathrm{B}(\omega)$. In each step the eigenpair with the smallest positive real part eigenvalue is calculated and the problem deflated such that the already converged eigenvalue is transformed into an infinite eigenvalue, to not disturb further eigenvalue searches. The deflated eigenvalue problem looks like

$$\mathrm{A}x = \omega\tilde{\mathrm{B}}(\omega)x,$$

with the new matrix $\tilde{\mathrm{B}}(\omega)$.

To find the next eigenvalue a Newton-type method is introduced. For a given ω the authors consider the general eigenvalue problem

$$\beta\mathrm{A}x = \tilde{\mathrm{B}}(\omega)x,$$

with β and x depending on ω.

Remark 8 *In the Newton-type method, different eigenvalue problems are stated, with dependance on ω. Note that ω is fixed in these eigenvalue problems. β is the desired eigenvalue.*

If $\beta(\omega) = \omega^{-1}$ would hold, then (β^{-1}, x) would be the new eigenpair. Therefore a root of the function

$$f(\omega) := \beta(\omega) - \omega^{-1}$$

is needed.

A Newton iteration is applied. In each iteration a eigenvalue problem has to be solved, but it has an eigenspace for the eigenvalue zero. This eigenspace influences the convergence badly. The authors solved this issue by applying a second deflation by projecting away from the kernel of the matrix A. For the Newton iteration the inverse $\tilde{\mathrm{B}}(\omega)^{-1}$ is needed. $\tilde{\mathrm{B}}(\omega)$ changes with each deflation, but the inverse can be calculated by using the inverse of the original matrix $\mathrm{B}(\omega)$.

Remark 9 *The matrix* $\mathrm{B}(\omega)$ *is diagonal, because of the finite difference discretisation schema. The inverse can be analytically calculated.*

We note the only needed inverse is that of $\mathrm{B}(\omega)$, which is a diagonal matrix. To solve the arising standard eigenvalue problems, they used either the Jacobi-Davison method or the Shift-invert residual Arnoldi method.

One issue still remained, the Newton convergence depends on the start value and only with a good start value, a fast convergence can be assured. To calculate such a start value the non-linear Arnoldi or the non-linear Jacobi Davidson has been used. Whenever the Newton iteration did not converge a new start value has been computed.

5.2 Beyn's method

Beyn's method, introduced in [Bey12] by Wolf-Jürgen Beyn, is an eigenvalue solver based on contour integrals in the complex plane. It belongs to the new class of moment-based eigenvalue solvers [ST07]. This section is a short overview of this method and based on [Bey12]. Beyn's method can be applied to non-linear eigenvalue problems of the form

$$\mathrm{T}(\omega)v = 0, \qquad v \in \mathbb{C}^m, v \neq 0, \omega \in \mathcal{M} \subset \mathbb{C},$$

with $\mathcal{M}$ a simply connected domain and $\mathrm{T} : \mathcal{M} \to \mathbb{C}^{m \times m}$ holomorphic.

Remark 10 *The setting in this section including Keldysh's theorem can be generalised to holomorphic Fredholm operators. For details see [MM03, Chapter 1].*

The pair (ω, v), such that $\mathrm{T}(\omega)v = 0$, is called an eigenpair. The main point of the algorithm is that the inverse $\mathrm{T}(\omega)^{-1}$ is meromorphic in $\mathcal{M}$ and in the vicinity $\mathcal{U}$ of a pole $\lambda \in \mathcal{M}$ it has a Laurent expansion of the form

$$\mathrm{T}(\omega)^{-1} = \sum_{j=-\kappa}^{\infty} \mathrm{S}_j(\omega - \lambda)^j, \qquad \mathrm{S}_j \in \mathbb{C}^{m \times m},$$

with κ being the order of the pole. For simplicity, the algorithm will be explained for an eigenvalue problem with strictly simple eigenvalues. For simple eigenvalues the order of the pole is always $\kappa = 1$.

The matrix S_{-1} is unknown for the operator T, but according to Keldysh's theorem [MM03, Chapter 1], it can be represented with the left eigenvector u and the right eigenvector v satisfying

$$T(\lambda)v = 0, u^*T(\lambda) = 0, \qquad u^*T'(\lambda)v = 1.$$

Then the inverse has the form

$$T(\omega)^{-1} = \frac{1}{\omega - \lambda} vu^* + R(\omega), \qquad \omega \in \mathcal{U} \setminus \{\lambda\},$$

with a holomorphic function $R : \mathcal{U} \to \mathbb{C}^{m\times m}$ in $\mathcal{U}$. The pole of the Laurent series is expressed in left and right eigenvectors and the remaining rest is holomorphic.

With more than one eigenvalue in $\mathcal{U}$ the inverse has the similar form,

$$T(\omega)^{-1} = \sum_{n=1}^{k} \frac{1}{\omega - \lambda_n} v_n u_n^* + R(\omega), \qquad \omega \in \mathcal{U} \setminus \{\lambda_1, \ldots, \lambda_k\},$$

with u_n, v_n left and right eigenvectors to λ_n satisfying $u_n^* T'(\lambda_n) v_n = 1$.

Applying the residue theorem from complex analysis to a contour $\mathcal{C}$ in $\mathcal{U}$, which encloses the eigenvalues of interest, leads to

$$\frac{1}{2\pi i} \int_{\mathcal{C}} f(\omega) T(\omega)^{-1} d\omega = \sum_{n=1}^{k} f(\lambda_n) v_n u_n^*, \tag{13}$$

for any in $\mathcal{U}$ holomorphic scalar function.

With this information, an eigenvalue solver can be proposed. We will need the matrices $V := (v_1, \ldots, v_k), U := (u_1, \ldots, u_k) \in \mathbb{C}^{m\times k}$. They store the left and right eigenvectors. In [ST07] methods with higher moments are introduced. Beyn's method only needs the first two moments:

$$A_0 := \frac{1}{2\pi i} \int_{\mathcal{C}} T(\omega)^{-1} \hat{V} d\omega \in \mathbb{C}^{m\times l},$$
$$A_1 := \frac{1}{2\pi i} \int_{\mathcal{C}} \omega T(\omega)^{-1} \hat{V} d\omega \in \mathbb{C}^{m\times l}.$$

The matrix $\hat{V} \in \mathbb{C}^{m\times l}, k \le l \le m$ is randomly chosen such that $U^*\hat{V}$ has rank k, which also implies full rank for U. The space spanned by the columns of $\hat{V}$ is also revered to as ansatz space. Matrix V should also have full rank k.

Remark 11 *For linear eigenvalue problems the rank conditions on* U *and* V *are trivial. For non-linear eigenvalues this does not hold, eigenvectors to different eigenvalues may be linear dependent.*

The current forms of A_0 and A_1 are not helpful, but with Equation (13) they have the form,

$$A_0 = \sum_{n=1}^{k} v_n u_n^* \hat{V} = VU^*\hat{V}, \tag{14}$$

$$A_1 = \sum_{n=1}^{k} \lambda_n v_n u_n^* \hat{V} = V\Lambda U^*\hat{V}, \qquad \Lambda := \mathrm{diag}(\lambda_1, \ldots, \lambda_k). \tag{15}$$

Matrix A_1 has the eigenvalues stored, but they and the matrices U, V are unknown. There is a way to extract the eigenvalues from these matrices.

First, a singular value decomposition of A_0 in reduced form

$$V U^* \hat{V} = A_0 = V_0 \Sigma_0 U_0^*$$

is calculated, where $V_0 \in \mathbb{C}^{m \times k}, \Sigma_0 := \operatorname{diag}(\sigma_1, \ldots, \sigma_k), U_0 \in \mathbb{C}^{l \times k}, V_0^* V_0 = I_k, U_0^* U_0 = I_k$. The rank of A_0 is k, hence A_0 has singular values

$$\sigma_1 \geq \cdots \geq \sigma_k > 0 = \sigma_{k+1} = \cdots = \sigma_l.$$

Because of the rank condition earlier the range of A_0, V and V_0 is the same. Therefore, there exists a regular matrix $S \in \mathbb{C}^{k \times k}$ such that

$$V = V_0 S, \qquad S := V_0^* V. \tag{16}$$

V_0 spans the same space as V, which is the eigenspace. Combining Equation (14) and (16) leads to

$$V_0 S U^* \hat{V} = V_0 \Sigma_0 U_0^* \iff U^* \hat{V} = S^{-1} \Sigma_0 U_0^*.$$

With this, $U^* \hat{V}$ in Equation (15) can be eliminated, obtaining

$$V_0^* A_1 = S \Lambda U^* \hat{V} = S \Lambda S^{-1} \Sigma_0 U_0^* \iff S \Lambda S^{-1} = V_0^* A_1 U_0 \Sigma_0^{-1}.$$

The right hand matrix can be calculated from the matrices A_0, A_1 and has the same eigenvalues as Λ, which are desired. By calculating the eigenvalues of it, the algorithm is concluded.

Remark 12 *The matrices* A_0 *and* A_1 *hold all the necessary information to calculate the eigenvalues. The main goal of the algorithm is to make these informations accessible through a singular value decomposition.*

Computing A_0 **and** A_1 There is still the question of how to calculate the contour integrals for A_0 and A_1. The discrete integrals cannot be calculated, a numerical quadrature rule has to be applied. It turns out that the composite rectangle rule for the quadrature is good enough. For a contour given by a 2π-periodic smooth parametrisation $\psi(t)$ the approximation of A_0 looks like

$$A_{0,N_c} := \frac{1}{i N_c} \sum_{j=0}^{N_c - 1} T(\psi(t_j))^{-1} \hat{V} \psi'(t_j)$$

and the approximation for A_1 is

$$A_{1,N_c} := \frac{1}{i N_c} \sum_{j=0}^{N_c - 1} T(\psi(t_j))^{-1} \hat{V} \psi(t_j) \psi'(t_j).$$

The integration nodes are taken equidistant $t_j := \frac{2j\pi}{N_c}, j \in \{0, \ldots, N_c\}$. For each integration node l systems of linear equations with the vectors in $\hat{V}$ have to be solved.

The computational effort of the algorithm increases with l the rank of $\hat{\mathrm{V}}$. In theory, l can be set to $l = k$. The algorithm has been introduced for $k \leq m$. In non-linear eigenvalue problems it can occur that there are more eigenvalues than the matrix has dimension. Usage of more than the first two moments solves this issue. More details can be found in [Bey12].

In this section, two different solver for non-linear eigenvalues have been introduced. The first was for a very specific problem and combined different methods to tackle the issues of this problem. The second algorithm is very different in this perspective. It can handle a wide range of non-linear problems.

5.3 Comparison

We collected all necessary tools to handle the arising rational eigenvalue problem. Four different eigenvalue solvers were introduced. In this section we are going to compare them with each other.

Beyn's method Beyn's method is based on contour integrals. It is not an iterative eigenvalue solver. The contour integrals are calculated once and the eigenvalues extracted. Theoretically, it calculates all eigenvalues in the contour and none on the outside. It separates the eigenvalues in the complex plane. The solver has to be restarted if the results are not satisfying. Convergence can only be achieved by increasing the number of quadrature points. An increase of the ansatz space is not enough. On the other side, Beyn's method can handle a variety of non-linear problems. With this property it stands out amongst the introduced solvers.

To the subject of computational costs. The contour integrals have to be calculated. For that a number of systems of linear equations at quadrature points in the complex plane have to be solved. If a factorisation can be applied, then for each quadrature point a factorisation has to be calculated and used. In consequence, N_c factorisations have to be generated and with each l systems of linear equations solved. If no factorisation is possible N times l systems of linear equations have to be solved. The advantage is that all those systems of equations can be handled in parallel, making the algorithm optimal for parallelisation. After that a singular value decomposition of the matrix A_0 is needed.

Newton-type method with non-equivalence deflation The Newton-type eigenvalue solver proposed in [HLM16] is adapted to a very special case of non-linear eigenvalue problems. Those which have the non-linearity in a diagonal matrix, but this is at the same time the advantage of the method. Not a single system of linear equations has to be solved numerically, because the matrices can be analytically inverted. The disadvantage is in the Newton method itself. Good start values have to be calculated to ensure convergence to the desired eigenvalues. If the eigenvalues are clustered, then it may not be possible to ensure convergence to the correct eigenvalue. Additionally, the eigenvalues have to be computed one after another.

Rational and polynomial Arnoldi There is not a large difference between the rational and polynomial Arnoldi. The convergence for both is comparable, as will be seen later. Only at roots of denominators the polynomial Arnoldi behaves differently. Both calculate eigenvalues closest to the shift. Compared to Beyn's method they are not as versatile, but they are iterative solvers and work till convergence. Additionally, the ratio of eigenvalue accuracy to needed system of linear equations solves is much better.

Disadvantage is that the Arnoldi algorithm is sequential, the systems of linear equations cannot be solved parallel at the same time and also the orthogonalisation gets more expensive with each iteration step and might at some point outweigh the system of equations solve. If a factorisation is applied, Arnoldi has the advantage that this factorisation can be used in each iteration step, making the algorithm very efficient in this case. If a factorisation is not possible, the systems of linear equations have to be solved iteratively one after the other. In this regard Beyn's method is better.

Compared to the Newton-type method, it is more stable in finding a large amount of eigenvalues. Another advantage of it is, that a-priori and a-posteriori error estimators for the Arnoldi method may be applied.

6 Numerical results

We want to see if the finite element framework in combination with rational linearisation and Arnoldi eigenvalue solver can handle metallic photonic crystals. The rational Arnoldi from Section 4 is applied to a 2D and a 3D metallic photonic crystal and the band structure along specific points is calculated. In these band structure diagrams only the real part of the eigenvalues are plotted. We are going to compare them to the results of a polynomial linearisation.

6.1 2D metallic photonic crystal

First, the square lattice of circular cross-section cylinders, see Figure 2, a 2D example. The same crystal has been analysed in [MEH02, IS01]. As in the referenced papers the Drude model is applied to this lattice.

The interesting region of the first Brillouin zone is spanned by the points $\Gamma := (0,0)^\top, X := (\frac{1}{2a},0)^\top, M := (\frac{1}{2a},\frac{1}{2a})^\top$. It is marked with red lines in Figure 4a. Thirty equidistant wave vectors $\mathbf{k}$ have been chosen on each of these lines. The Drude model constants have been set to $\delta_p = 1, \gamma_p = 10^{-2}$ and the lattice constant $a = 2\pi$ has been chosen. For the finite element method **Netgen/NGSolve** was used, the mesh can be seen in Figure 5.

Earlier it was stated that two dimensional MPHCs have two different types of resonances, transversal magnetic and transversal electric ones.

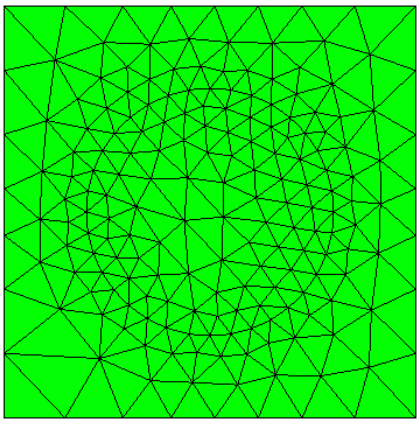


Figure 5: Finite element mesh generated in **Netgen/NGSolve** and refined on the metallic surface. The length of its border is $a = 2\pi$ and the radius of the circle $r = 0.3a$. The maximal triangulation diameter is $h_{max} = 1$ and the maximal diameter on the material boundary $h_{bnd} = \frac{1}{3}h_{max}$.

6.1.1 Transversal magnetic resonances

Equation (9) defines the transversal magnetic resonances. Applying the rational Arnoldi and the polynomial Arnoldi to it, the band structure for the TM modes looks like Figure 6. Note that only the real part of ω is used in band structure diagrams. They have been computed by the polynomial and the rational Arnoldi, with the shift $\sigma = 0.9$ and finite element order $p = 4$ on the mesh given in Figure 5. In each solver $l = 150$ iterations have been made per wave vector $\mathbf{k}$. To the eye, both solvers generate the same band structure.

6.1.2 Transversal electric resonances

The transversal electric modes are defined through Equation (10) and have an interesting mathematical and numerical effect. Especially interesting is the inverse electric permittivity, because it has a singular point in the spectral region of interest. This leads to a very curious effect resulting in the band structure seen in Figure 7. The band structure is a combination of two separate runs with each the polynomial and the rational Arnoldi. In the first run, the shift has been set to $\sigma = 0.4$ and in the second run $\sigma = 0.85$. The finite element order was $p = 4$ on the mesh given in Figure 5 and each solver made $l = 150$ iterations per wave vector $\mathbf{k}$.

In the band structure diagram a cluster of flat bands can be seen around $\omega = 0.7$. A more detailed view of this interesting part is highlighted in Figure 8 and Figure 9. These band structures have been computed by the polynomial and the rational Arnoldi with the shift $\sigma = 0.695$ and finite element order $p = 4$.

The fact that they are flat lines means that they are independent of the Bloch boundary condition. These resonance functions are localised on the surface of the metallic rod and are, so called, plasmon resonances. A couple of theme can be seen in Figure 10. They correspond to waves that travel around the metallic rod. It is notable that the wavelengths of these resonances along the metallic

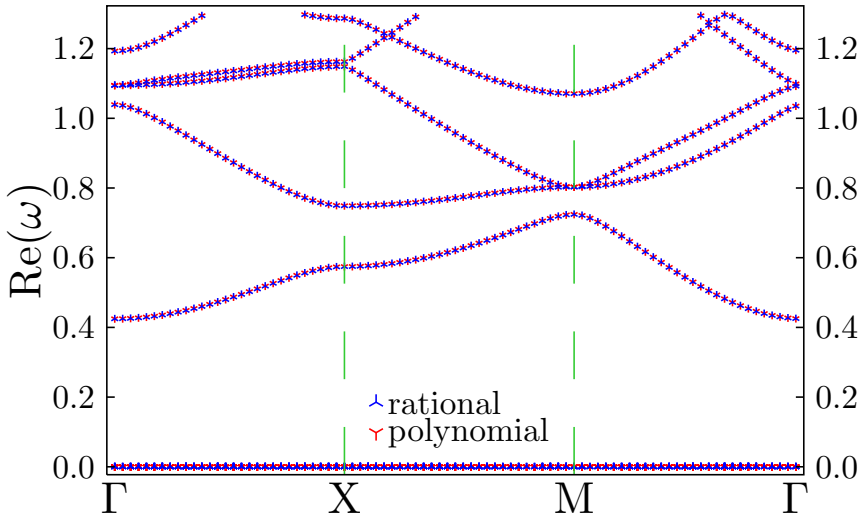


Figure 6: Transversal magnetic band structure of a square lattice with metallic cylinders as dispersive material, computed by polynomial and rational Arnoldi. Two band gaps can be seen.

surface are very short and grow shorter. This is challenging for the finite element method, because the eigenfunctions need to be properly resolved. The solution is a local refined mesh around the metallic surface, as can be seen in Figure 5. A global refinement is unnecessary, because the eigenfunctions are very smooth away from the surface and a local refinement is therefore more efficient yielding the same results.

In the vicinity of the interesting wavelength resonances with very high frequencies along the metallic surface exist. With a finite mesh size, those cannot be resolved properly, resulting in fractions of modes. A couple of these fractions can be seen in Figure 11. These correspond to eigenpairs of the matrix eigenvalue problem, which converged in the iterative solvers, but are not resonances of the continuous problem. Further refining the mesh does not resolve this issue.

Although more resonance functions with higher frequency are resolved, there are always fractions of resonance functions with even higher frequencies. These resonances are not well separated from the others, see Figure 8 and Figure 9, and can not be filtered by the eigenvalue, but only by the resonance function. Another property of them is their local support making them independent of the Bloch boundary, meaning they will appear as a line in the band structure.

Additionally, a flat line at around one can be seen in Figure 7. Only the polynomial solver generates this line. It is exactly at the root of the denominator and in the polynomial solver the system of equations is multiplied with it. In the end, the system of equations is multiplied with zero for this eigenvalue and wrong resonances are generated.

6.2 3D metallic photonic crystal

Finally, we are at a complex three dimensional problem in the Hilbert space $\mathcal{H}_{\mathbf{k}}(\mathrm{curl}, \Omega)$. As 3D lattice the diamond structured cubic lattice with spheres

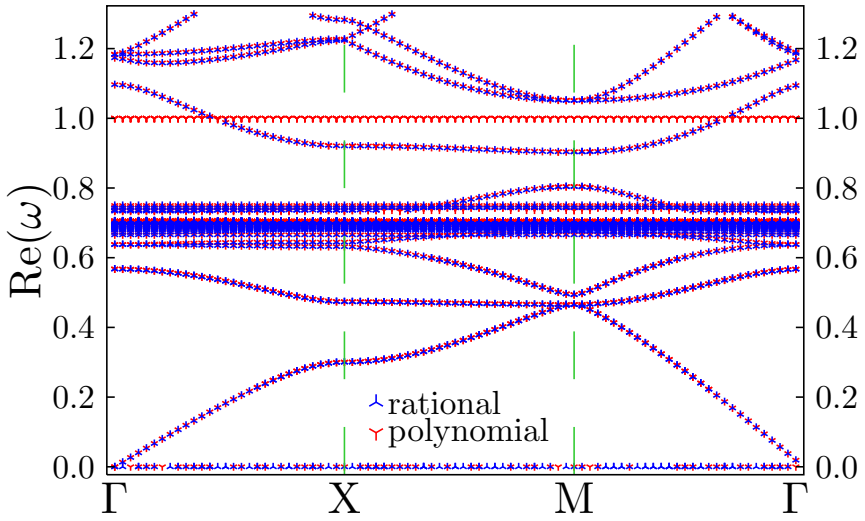


Figure 7: Transversal electric band structure of a square lattice with metallic cylinders as dispersive material.

connected by ellipsoids, as in [HLM16], has been used. Its unit cell can be seen in Figure 3. It is spanned by the lattice translation vectors

$$\mathbf{a}_1 := \frac{a}{\sqrt{2}}(1,0,0)^\top, \quad \mathbf{a}_2 := \frac{a}{\sqrt{2}}\left(\frac{1}{2}, \frac{\sqrt{3}}{2}, 0\right)^\top, \quad \mathbf{a}_3 := \frac{a}{\sqrt{2}}\left(\frac{1}{2}, \frac{1}{2\sqrt{3}}, \sqrt{\frac{2}{3}}\right)^\top.$$

The vectors

$$\mathbf{b}_1 := \frac{2}{a}\left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{6}}, -\frac{1}{2\sqrt{3}}\right)^\top, \mathbf{b}_2 := \frac{2}{a}\left(0, \sqrt{\frac{2}{3}}, -\frac{1}{2\sqrt{3}}\right)^\top, \mathbf{b}_3 := \frac{2}{a}\left(0, 0, \frac{\sqrt{3}}{2}\right)^\top$$

span the reciprocal space. These come very handy in defining the first Brillouin zone, or better the interesting points in it. The lattice constant is $a = 2\pi$, the radius of the spheres is $r = 0.08a$ and the minor axis of the connecting spheroids is $s = 0.06a$. The permittivity parameters are $\delta_p = \frac{10\pi}{a}, \gamma_p = \frac{2\pi}{14500}, \Omega_1 = \frac{2\pi}{470}, \Omega_2 = \frac{2\pi}{325}\gamma_1 = \frac{2\pi}{1900}, \gamma_2 = \frac{2\pi}{1060}, \epsilon_\infty = 1.54, A_1 = 1.27, A_2 = 1.1, \beta_1 = -\frac{\pi}{4}$ and $\beta_2 = -\frac{\pi}{4}$.
Its Brillouin zone points are defined through the dual vectors $\mathbf{b}_i$ as following: $X := \frac{1}{2}(\mathbf{b}_1 + \mathbf{b}_3), U := \frac{1}{8}(5\mathbf{b}_1 + 2\mathbf{b}_2 + 5\mathbf{b}_3), L := \frac{1}{2}(\mathbf{b}_1 + \mathbf{b}_2 + \mathbf{b}_3), \Gamma := (0,0,0)^\top, W := \frac{1}{2}(2\mathbf{b}_1 + \mathbf{b}_2 + 3\mathbf{b}_3)$ and $K := \frac{1}{8}(3\mathbf{b}_1 + 3\mathbf{b}_2 + 6\mathbf{b}_3)$. For this crystal both the Drude and the Drude-Lorentz model have been applied to the electric field Equation (7).

6.2.1 Drude model

Let us start with the Drude model. It can be linearised in different ways, by multiplication and then applying a polynomial approach, or with a rational approach as can be seen in Section 4. The calculated band structure for the electric field formulation in Equation (9) can be seen in Figure 12. The rational and polynomial Arnoldi deliver the same visual results. The Arnoldi solver has been started with a shift $\sigma = 1$ and 50 iteration have been done for each wave vector $\mathbf{k}$. This problem is comparable to the transversal magnetic modes in the 2D example.

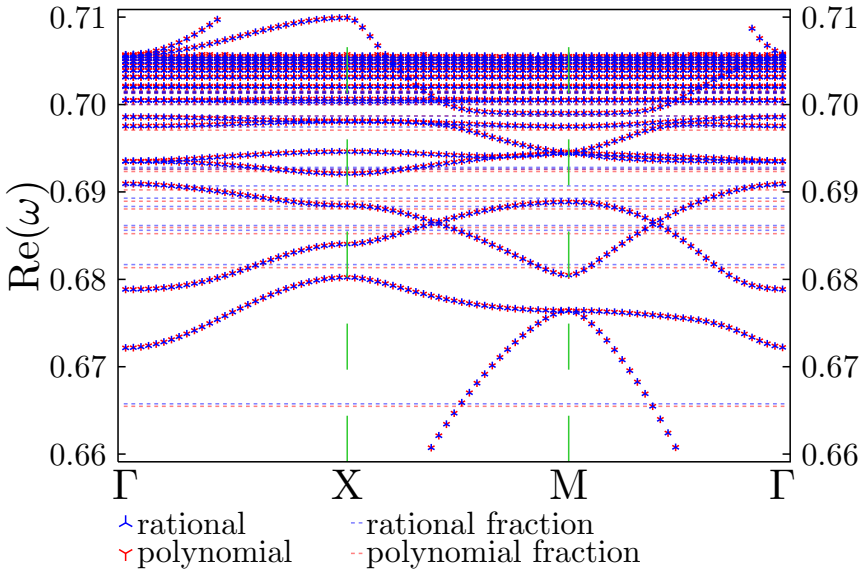


Figure 8: In depth view of the transversal electric band structure of a square lattice with metallic rods as dispersive material.

6.2.2 Drude-Lorentz model

The band structure for the Drude-Lorentz model looks similar to the band structure of the Drude model. Only small differences in the resonances appear. The first 16 resonances for both models have been calculated for the wave vector $\mathbf{k} = \frac{3}{7}X$ and can be seen in Table 1.

In [HLM16] the same problem was solved and a couple of reference values are stated. In the paper, the authors find a cluster of resonances around $\omega = 1.3$, for both the Drude and the Drude-Lorentz model. We did not find these resonances. The Band structure for the first six resonances are optically the same.

The calculations were done with a finite element order $p = 3$ and mesh size $h = 0.3$ resulting in $n_{dof} = 532272$ degrees of freedom.

6.3 Convergence comparison

We compared rational and polynomial Arnoldi in band structures, but this is only an visual comparison and unsatisfying. We are going to take a closer look. They are both measured in the 2D lattice problem setup from Section 6.1 at the wave vector $\mathbf{k} = \Gamma$.

We are going to analyse convergence toward the discretised eigenvalues of the examples. For this, reference values for the matrix eigenvalue problems are needed. It would be possible to use the rational or polynomial Arnoldi solver with very high iteration numbers to calculate them, but this way mistakes made in the linearisation would not be highlighted. To solve this issue, the reference values have been computed by the different eigenvalue solver in Section 5.2. This is a solver which can be directly applied to the non-linear problem without a linearisation.

In each example, the data for rational, polynomial Arnoldi and the reference values have been calculated with exactly the same finite element discretisation.

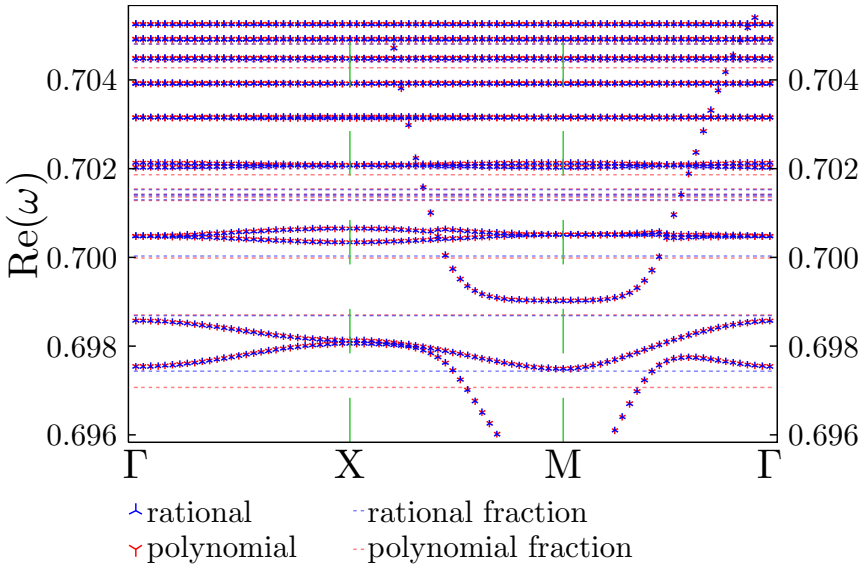


Figure 9: In depth view of the transversal electric band structure of a square lattice with metallic rods as dispersive material.

6.3.1 Convergence for TM modes

The finite element order has been set to $p = 4$ and the mesh size to $h = 1$. The first step generates reference values with Beyn's method. Five reference values for the smallest eigenvalues have been calculate. For each value the method has been applied anew, with a circle as contour closely around the specific eigenvalue. The calculated reference values are cumulated in Table 2. The following convergence test for the eigenvalue solvers is based upon convergence toward these values.

In each iteration step both solvers deliver an approximate eigenvalue and the relative error of these develop as can be seen in Figure 13. The figure illustrates the convergence of the first five eigenvalues with positive real part. Both solvers show the same rate of convergence.

6.3.2 Convergence for TE modes

The transversal electric modes showed a strange behaviour. There exist plasmon frequencies and a clustered region of eigenvalues. We could test the convergence at eigenvalues far away from this cluster, but we will not. As before, reference values have been calculated with Beyn's method, but it is quite impossible to separate the eigenvalues in this region by circles. A brute force approach has been chosen. Only one contour containing the whole region has been used. The circles middle point was $mp = 0.7$ and its radius $r = 0.1$. In this circle, a lot of eigenvalues were found. To be precise, Beyn's method calculated 142 eigenvalues from which many were fraction eigenvalues. Only five not fraction eigenvalues have been chosen as reference values for the convergence test and can be seen in Table 3.

In each iteration step, both solvers deliver an approximate eigenvalue and the relative error of these develop as can be seen in Figure 14. The figure illustrates the convergence for five eigenvalues to the reference values. From all calculated

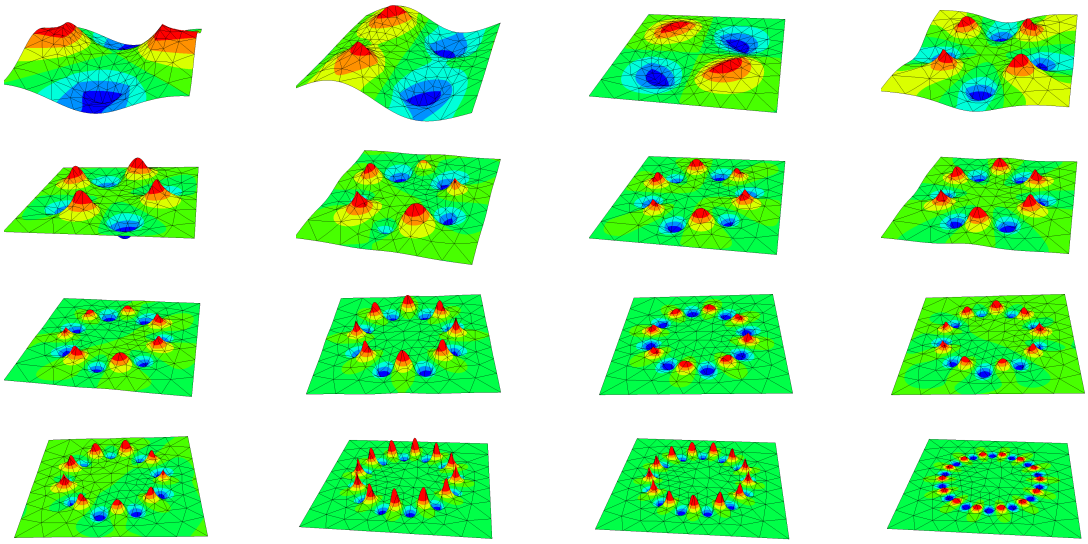


Figure 10: Transversal electric resonance functions at wave vector $\mathbf{k} = \Gamma$ for the 2D lattice given in Figure 2. The plasmon resonance functions are ordered with increasing resonance.

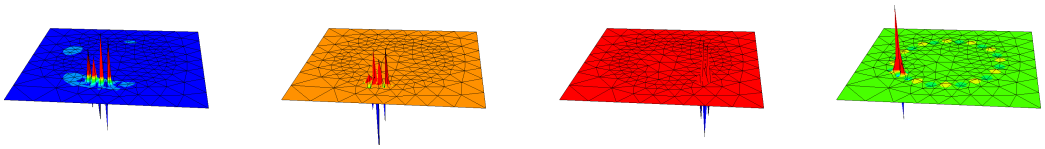


Figure 11: Transversal electric fraction resonance functions at wave vector $\mathbf{k} = \Gamma$ for the lattice given in Figure 2.

eigenvalues in each iteration, the closest one has been taken as the approximation for the respective solver. Again, they show the same rate of convergence.

In the matter of convergence, both linearisations show the same behaviour. The iterations until convergence are almost the same.

7 Conclusion

In this thesis the rational Arnoldi algorithm has been formulated by combining rational linearisation, Schur decomposition and shift-and-invert Arnoldi. The main purpose was to state a matrix eigenvalue solver which is fast, reliable and can handle any rational eigenvalue problem. The rational linearisation has been compared to the polynomial linearisation. Band structures for two dimensional and three dimensional metallic photonic crystals have been calculated and compared to solutions from other papers. Additionally mathematically interesting transversal electric modes, which are challenging for the used finite element method, have been analysed and the convergence rates for rational and polynomial Arnoldi have been compared.

From the tests we concluded that the rational linearisation is as good as the polynomial linearisation and the shift-and-invert Arnoldi solver shows the same

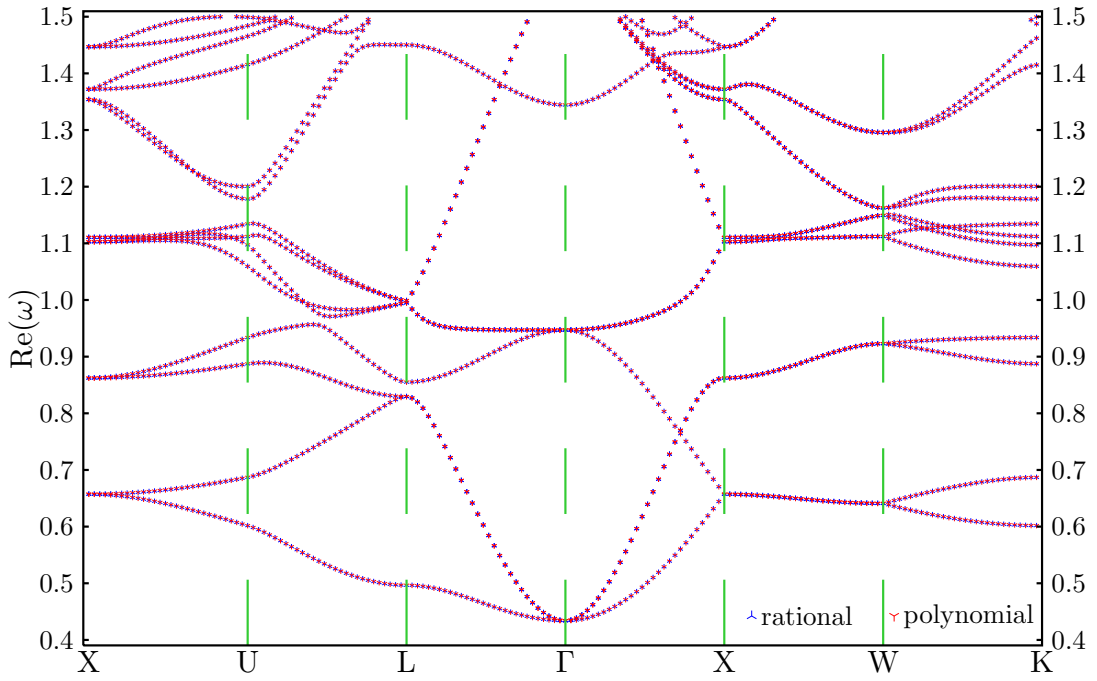


Figure 12: Band structure diagram for a diamond structured lattice with connection spheroids, size $a = 2\pi$, with the Drude model and the electric field formulation. Solved by rational and polynomial Arnoldi, finite element order $p = 2$ and mesh size $h = 0.3$.

convergence. The rational linearisation has the advantage for computing eigenvalues around the roots of the denominators, where the polynomial linearisation may produce wrong eigenvalues.

Compared to the Newton-type solver introduced in [HLM16], the rational Arnoldi is easier to implement, works with the Arnoldi a-posteriori and a-priori error estimates and can be applied to every type of rational eigenvalue problem. Additionally, it has the advantage of calculating a multitude of eigenpairs at once.

Another big advantage is that the usually big systems of linear equations in linearised eigenvalue problems can be reduced to systems of linear equations of the same size as the original rational matrix eigenvalue size.

All these properties make the rational Arnoldi method a fast and reliable rational eigenvalue solver.

8 Future prospects

There are many ways to improve the introduced eigenvalue solver. For example, in the algorithm only systems of linear equations of the original problem size have to be solved numerically reducing the computational costs. The issue of the three times more expensive orthogonalisation still stands and should be handled in a more efficient way.

One could also invest some time into the fraction resonances of the transversal electric example. They cannot be filtered by the linear eigenvalue solver. A post

	Drude model	Drude-Lorentz model
μ_1	$0.65675903638 - 0.00007751022i$	$0.65593337614 - 0.00024711616i$
μ_2	$0.65697377973 - 0.00007746811i$	$0.65639174526 - 0.00024724518i$
μ_3	$0.86184052633 - 0.00004180947i$	$0.86086029148 - 0.00020300389i$
μ_4	$0.86186928038 - 0.00004181119i$	$0.86086907548 - 0.00020298735i$
μ_5	$1.10246492272 - 0.00001578311i$	$1.10595885996 - 0.00047921043i$
μ_6	$1.10247044276 - 0.00001578066i$	$1.10600675273 - 0.00047958820i$
μ_7	$1.10972368440 - 0.00006618165i$	$1.10156906633 - 0.00011597750i$
μ_8	$1.10974830362 - 0.00006618999i$	$1.10157791710 - 0.00011622408i$
μ_9	$1.35372928977 - 0.00001178461i$	$1.35237742754 - 0.00012708095i$
μ_{10}	$1.35374564981 - 0.00001178273i$	$1.35238877944 - 0.00012709179i$
μ_{11}	$1.37176054343 - 0.00001154696i$	$1.37036405283 - 0.00012740894i$
μ_{12}	$1.37183556470 - 0.00001154595i$	$1.37052504910 - 0.00012760363i$
μ_{13}	$1.44662475251 - 0.00002242831i$	$1.44333905164 - 0.00027838356i$
μ_{14}	$1.44666440357 - 0.00002240315i$	$1.44335588464 - 0.00027790251i$
μ_{15}	$1.54790395724 - 0.00011036558i$	$1.52872762977 - 0.00146856847i$
μ_{16}	$1.54839966307 - 0.00011032100i$	$1.52938546458 - 0.00146893979i$

Table 1: First 16 eigenvalues of the 3D example at wave vector $\frac{3}{7}X$.

	Transversal magnetic
μ_0	$0.42463251715047362067 - 0.00307862191626115042i$
μ_1	$1.03915857553789936496 - 0.00031144787450729239i$
μ_2	$1.09449573834616176171 - 0.00056426930890006140i$
μ_3	$1.09449574420785955553 - 0.00056426932180721158i$
μ_4	$1.19296512079353855817 - 0.00110969096317216354i$

Table 2: Reference values for the first five eigenvalues at wave vector Γ in the 2D transversal magnetic example.

processing might be possible. An approach could be an error estimator which checks the smoothness of the tangential derivation on the metallic surface.

Another approach may be to have a look at other fully non-linear eigenvalue solvers. For example Beyn's method or the non-linear Feast algorithm [GMP18]. The non-linear Feast algorithm is a new method, which combines the contour integral approach in Beyn's method with common iterative eigenvalue solvers.

Finally, the territory of larger eigenvalue problems may be explored. With it comes a smaller discretisation error and the possibility for more complex and interesting applications. But also the necessity to apply an iterative system of linear equations solver, because factorisations are not feasible anymore. Not even for the original matrix size. While being at the subject of iterative system of linear equations solvers, the influence of inexact solves on eigenvalue solvers may be explored.

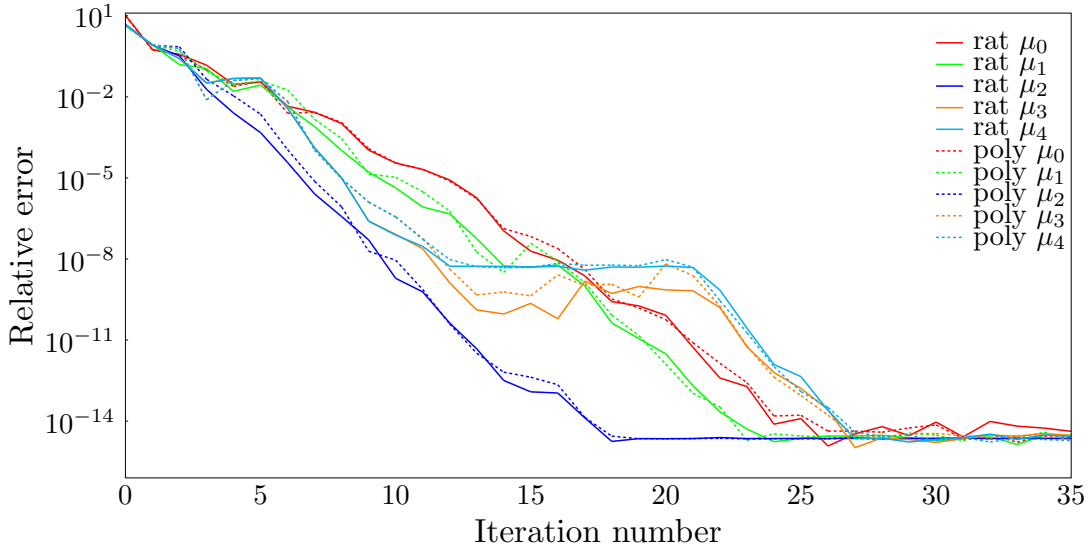


Figure 13: Convergence in iterations for the first five TM modes with positive real part, at wave vector Γ. The relative error $\epsilon = \left|\frac{\mu_i - \mu_{ref,i}}{\mu_{ref,i}}\right|$ is plotted over the iteration number.

	Transversal electric
μ_0	$0.63778866426117453159 - 0.00351199656565298873i$
μ_1	$0.67884275659042558893 - 0.00459978824916659149i$
μ_2	$0.69094160873694443481 - 0.00470948628252337853i$
μ_3	$0.70212655269656787382 - 0.00492824109492175783i$
μ_4	$0.70314885136315641301 - 0.00494305888291618738i$

Table 3: Reference values for five eigenvalues at wave vector Γ in the 2D transversal electric example.

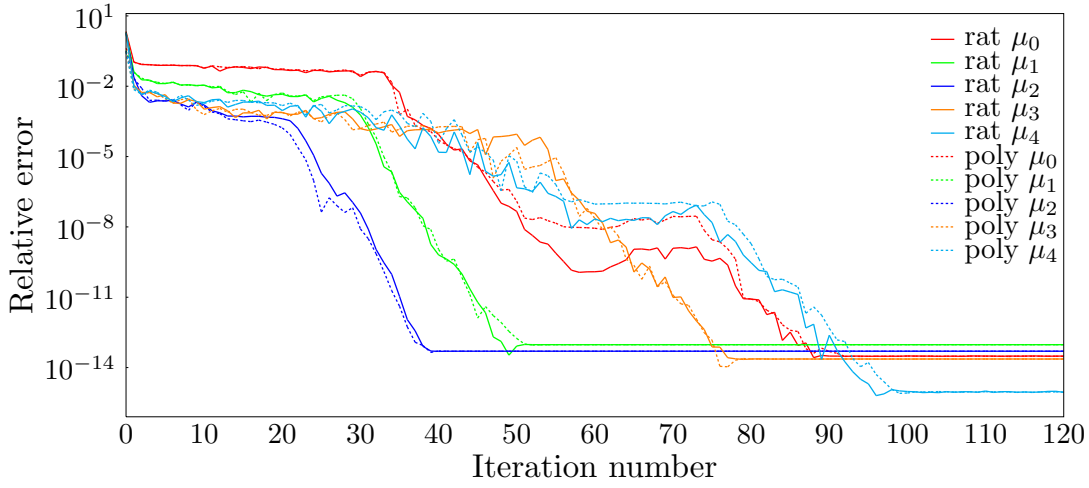


Figure 14: Convergence in iterations for TE modes next to the reference values in Table 3 at wave vector Γ. The relative error $\epsilon = \left|\frac{\mu_i - \mu_{ref,i}}{\mu_{ref,i}}\right|$ is plotted over the iteration number.

Acronyms

PHC photonic crystal

MPHC metallic photonic crystal

TM transversal magnetic

TE transversal electric

FEM finite element method

Listings

List of Figures

List of Tables

References

[Arn51] W. E. Arnoldi. The principle of minimized iteration in the solution of the matrix eigenvalue problem. *Quart. Appl. Math.*, 9:17–29, 1951.

[Bey12] W.-J. Beyn. An integral method for solving nonlinear eigenvalue problems. *Linear Algebra Appl.*, 436(10):3839–3863, 2012.

[ELRM06] P. G. Etchegoin, E. C. Le Ru, and M. Meyer. An analytic model for the optical properties of gold. *The Journal of Chemical Physics*, 125(16):164705, 2006.

[ELRM07] P. G. Etchegoin, E. C. Le Ru, and M. Meyer. Erratum: An analytic model for the optical properties of gold [j. chem. phys. 125, 164705 (2006)]. *The Journal of Chemical Physics*, 127(18):189901, 2007.

[FYS+13] T. Fu, Z. Yang, Z. Shi, F. Lan, D. Li, and X. Gao. Dispersion properties of a 2d magnetized plasma metallic photonic crystal. *Physics of Plasmas*, 20(2):023109, 2013.

[GMP18] B. Gavin, A. Miedlar, and E. Polizzi. FEAST eigensolver for nonlinear eigenvalue problems. *J. Comput. Sci.*, 27:107–117, 2018.

[HLM16] T.-M. Huang, W.-W. Lin, and V. Mehrmann. A Newton-type method with nonequivalence deflation for nonlinear eigenvalue problems arising in photonic crystal modeling. *SIAM J. Sci. Comput.*, 38(2):B191–B218, 2016.

[IS01] T. Ito and K. Sakoda. Photonic bands of metallic systems. ii. features of surface plasmon polaritons. *Phys. Rev. B*, 64:045117, Jul 2001.

[JJWM08] J. D. Joannopoulos, S. G. Johnson, J. N. Winn, and R. D. Meade. *Photonic Crystals: Molding the Flow of Light (Second Edition).* Princeton University Press, 2 edition, 2008.

[MEH02] E. Moreno, D. Erni, and C. Hafner. Band structure computations of metallic photonic crystals with the multiple multipole method. *Phys. Rev. B*, 65:155120, Apr 2002.

[MM03] R. Mennicken and M. Möller. *Non-self-adjoint boundary eigenvalue problems*, volume 192 of *North-Holland Mathematics Studies*. North-Holland Publishing Co., Amsterdam, 2003.

[Néd80] J.-C. Nédélec. Mixed finite elements in $\mathbf{R}^3$. *Numer. Math.*, 35(3):315–341, 1980.

[Néd86] J.-C. Nédélec. A new family of mixed finite elements in $\mathbf{R}^3$. *Numer. Math.*, 50(1):57–81, 1986.

[SB11] Y. Su and Z. Bai. Solving rational eigenvalue problems via linearization. *SIAM J. Matrix Anal. Appl.*, 32(1):201–216, 2011.

[SC10] W. Setyawan and S. Curtarolo. High-throughput electronic band structure calculations: Challenges and tools. *Computational Materials Science*, 49(2):299 – 312, 2010.

[Sch97] J. Schöberl. Netgen - an advancing front 2d/3d-mesh generator based on abstract rules. *Comput.Visual.Sci*, 1:41–52, 1997.

[Sch14] J. Schöberl. C++11 implementation of finite elements in ngsolve. Preprint 30/2014, Institute for Analysis and Scientific Computing, TU Wien, 2014.

[ST07] T. Sakurai and H. Tadano. CIRR: a Rayleigh-Ritz type method with contour integral for generalized eigenvalue problems. *Hokkaido Math. J.*, 36(4):745–757, 2007.

[Zag06] S. Zaglemayr. *High Order Finite Element Methods for Electromagnetic Field Computation.* PhD thesis, Universität Linz, 2006.