Krylov subspace methods for time modeling of photonic crystals

Bachelor assignment

Sjoerd Hack, Abel Hanse, Stephan Höcker
Tutor: M. Botchev

June 17, 2013

Contents

1 Introduction

This report investigates the performance of Krylov subspace methods with respect to their application to the time domain modeling of electromagnetic fields. In particular they are applied to photonic crystals. The results are compared with those obtained by the finite difference time domain method, or FDTD for short.
Photonic crystals are dielectric materials with permittivities that are periodic in space [13, pp. 2]. The word 'photonic' indicates that the quantity of interest is the electromagnetic field inside the material and the word 'crystal' is used by analogy with solid state physics, which deals with periodic arrangements of molecules. Solid state physics is concerned with the position of electrons whereas photonics is concerned with the electromagnetic field. In fact much of the formalism of solid state physics can be applied to photonic crystals [13, pp. 4].
Photonic crystals are usually periodic in up to three dimensions and uniform in the nonperiodic ones [13, pp. 5] and can be divided into three groups depending on the number of periodic dimensions: one, two and three dimensional crystals. Their characteristic property is that they support the propagation of electromagnetic waves but suppress waves with frequencies inside a certain range, referred to as the photonic band gap [13, pp. 2], a property that is exploited in many applications. An example is the Bragg mirror, a one-dimensional photonic crystal which consists of slabs of materials with alternating dielectric constants and perfectly reflects any incident wave with a frequency in the band gap [13, pp. 44]. Introducing defects into the structure that break up the periodicity opens op many more applications [13, pp. 5].
The locations of the band gaps in the frequency spectrum and other properties of a photonic crystal are dependent on its structure, see for example [13, pp. 46]. Because Maxwell's equations are exact on macroscopic scales and the parameters of the dielectric materials are well known it is possible given a photonic crystal to model these properties prior to actual design and manufacture. Therefore predictions from first principles can provide a valuable addition to physical experiments [13, pp. 252]. However the complicated structures of photonic crystals make it impossible to solve Maxwell's equations analytically. As a consequence, numerical techniques are used to find the band gaps, field patterns, transmission spectra and other properties of a given photonic crystal.
A distinction can be made between frequency domain methods and time domain methods. For frequency domain methods the reader is referred to appendix D of [13]. This report is concerned with time-domain techniques. These are flexible because, unlike frequency domain methods, they do not assume that the time-dependence of the electromagnetic fields is harmonic [13, pp. 259]. This means that they can be used to determine the transmission coefficients for incident electromagnetic waves with many different frequencies in a single simulation run. This can be done by simulating a photonic crystal with a pulse as a source and taking Fourier transforms [13, pp. 259] Time domain methods are also suitable for problems in which dispersion occurs [13, pp. 259].
Many numerical techniques for Maxwell's equations are based on the discretization of the spatial domain into a grid [13, pp. 259]. The structure of this grid should be suitable for the discretization of the spatial derivatives in Maxwell's equations [2, pp. 2]. In this report the successful Yee grid is used [20]. When Maxwell's equations are approximated in terms of the values at the grid points the result is a system of ordinary differential equations which must be solved in time [2, pp. 3]. Usually this is done by also discretizing the time axis and using a time stepping algorithm in which the electric and magnetic fields are updated alternatingly. This is called the FDTD algorithm.
Simulations of photonic crystal often need to be highly accurate. It turns out that the error introduced by the second order accurate spatial discretization is much higher than the error introduced by the second order time discretization. Higher order methods for space discretization do exist but are hard to implement [18, pp. 80]. Instead usually a very small spatial step size Δx is chosen. However the Courant–Friedrichs–Lewy condition implies that the time step size Δt must then be very small too [18, pp. 128] to ensure numerical stability. This is unnecessary for accuracy because the error introduced by the time discretization is small anyway. However it does make simulations of long time intervals $[0, T]$ very demanding in terms of simulation time on a computer. Therefore there is a need for fast and accurate time integration methods. This report contributes to this goal.

An alternative is to use exponential time integration schemes [2], [12]. These are methods that make use of the matrix exponential and that are unconditionally stable [2], [12]. This means that the time step size is free from the CFL restriction. In addition, the error introduced by the time discretization can be often controlled independently from the space discretization. These schemes involve a so-called matrix exponential. This is a matrix function that can be defined and computed in different ways. One way is the power series

$$\mathrm{e}^{\mathbf{A}} = \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!}.$$

Another classical definition is that, for any sufficiently smooth function $f(x)$ and a matrix $\mathbf{A}$, $\mathbf{f}(\mathbf{A})$ is a polynomial in $\mathbf{A}$ such that it interpolates $f(x)$ and possibly some of its derivatives on the spectrum of $\mathbf{A}$ (see [6], [10] for exact definitions). The matrix exponential is extremely useful in Applied Mathematics and used in applications ranging from electomagnetic modeling to network analysis and public safety [5], [9]. Accordingly, there are many ways to compute the matrix exponential [15]. A naive approach would be to diagonalize $\mathbf{A}$ by computing all its eigenvalues and eigenvectors. Unfortunately this is unfeasible for the matrices of large size. An important observation is that we actually do not need the matrix exponential as such but rather its product with a given vector, i.e. its actions. Therefore Krylov subspace methods are used. These are numerical linear algebra algorithms that are based on Gram-Schmidt orthonormalization and project the matrix exponential series into the Krylov subspace. Modifications to the Krylov subspace methods that will be explored in this report are the Arnoldi and shift-and-invert (SAI) Arnoldi methods equipped with residual based stopping criteria.

2 Maxwell's equations

In this section the macroscopic Maxwell equations are introduced. The Maxwell equations describe the interaction between the electric and magnetic fields. At the length scales of photonic crystals the macroscopic Maxwell equations are practically exact [13, pp. 252].
Furthermore it will be seen that two-dimensional electromagnetic fields can be decomposed into two sets of components that are described by two uncoupled sets of equations, the TE^z mode and the TM^z mode. An advantage is that these are easily visualized. For three-dimensional electromagnetic fields such a decomposition is not possible. However, many interesting phenomena can be demonstrated for two-dimensional photonic crystals. Therefore we shall only consider electromagnetic fields in the TM^z mode.

2.1 Vector description of Maxwell's equations

We start off with the macroscopic Maxwell equations in three dimensions and with source currents

$$\nabla \cdot \mathbf{B}(\mathbf{r},t) = 0, \qquad \frac{\partial \mathbf{B}(\mathbf{r},t)}{\partial t} = -\nabla \times \mathbf{E}(\mathbf{r},t) - \mathbf{M}'(\mathbf{r},t),$$
$$\nabla \cdot \mathbf{D}(\mathbf{r},t) = \rho, \qquad \frac{\partial \mathbf{D}(\mathbf{r},t)}{\partial t} = \nabla \times \mathbf{H}(\mathbf{r},t) - \mathbf{J}'(\mathbf{r},t).$$

Here $\mathbf{r}$ is a vector that denotes the position and t denotes the time. $\mathbf{J}'$ is the electric current density and $\mathbf{M}'$ is the nonphysical magnetic current density. $\mathbf{E}$ and $\mathbf{H}$ are the electric and the magnetic field respectively and $\mathbf{B}$ and $\mathbf{D}$ are the magetic induction field and the electric displacement current.
We adopt the constitutive relationships (relationships between the electromagnetic fields and the material parameters) that are described in [13, pp. 6–8]. These are appropiate for the test configurations that we will use to test numerical methods. The assumptions made there are summarized in the following constitutive relationship

$$\mathbf{D}(\mathbf{r},t) = \epsilon_0 \epsilon_r(\mathbf{r})\mathbf{E}(\mathbf{r},t)$$

where ϵ_r is the relative electric permittivity and ϵ_0 is the vacuum permittivity. The permittivity is the quantity that is periodically variable throughout the photonic crystal. Therefore it depends on position. The magnetic induction and magnetic field are related by

$$\mathbf{B}(\mathbf{r},t) = \mu_0 \mu_r(\mathbf{r})\mathbf{H}(\mathbf{r},t)$$

One of the assumptions on which these constitutive equations are based is that the materials are isotropic: the permittivity and permeability do not depend on the direction of the applied fields. In section (8) the anisotropic layer formulation of a perfectly matched layer will be described in which tensors appear in these equations. Most dielectric materials that are used to construct photonic crystals have a permeability that is approximately equal to the the vacuum permeability μ_0. Therefore in all simulations in this report μ_r will be set to one

$$\mu_r = 1.$$

The electric current density consists of two components. We will allow for electric losses of the form $\sigma(\mathbf{r})\mathbf{E}(\mathbf{r},t)$ and an independent source term given by $\mathbf{J}$.

$$\mathbf{J}'(\mathbf{r},t) = \sigma(\mathbf{r})\mathbf{E}(\mathbf{r},t) + \mathbf{J}(\mathbf{r},t).$$

Similarly we include magnetic losses $\sigma_m(\mathbf{r})\mathbf{H}(\mathbf{r},t)$ and an independent source term $\mathbf{M}$

$$\mathbf{M}'(\mathbf{r},t) = \sigma_m(\mathbf{r})\mathbf{H}(\mathbf{r},t) + \mathbf{M}(\mathbf{r},t)$$

Magnetic losses are not physical but they are necessary for implementing a perfectly matched layer and for the verification of the numerical methods. If we enter these terms into Maxwell's equations we obtain

$$\nabla \cdot \mu_0\mu_r(\mathbf{r})H(\mathbf{r},t) = 0, \qquad \mu_0\mu_r(\mathbf{r})\frac{\partial \mathbf{H}(\mathbf{r},t)}{\partial t} = -\nabla \times \mathbf{E}(\mathbf{r},t) - \mathbf{M}(\mathbf{r},t) - \sigma_m(\mathbf{r})\mathbf{H}(\mathbf{r},t),$$
$$\nabla \cdot \epsilon_r(\mathbf{r})\mathbf{E}(\mathbf{r},t) = \frac{\rho}{\epsilon_0}, \qquad \epsilon_0\epsilon_r(\mathbf{r})\frac{\partial \mathbf{E}(\mathbf{r},t)}{\partial t} = \nabla \times \mathbf{H}(\mathbf{r},t) - \mathbf{J}(\mathbf{r},t) - \sigma(\mathbf{r})\mathbf{E}(\mathbf{r},t). \tag{1}$$

2.2 Maxwell's equations in scalar form

Since our electric and magnetic fields are three dimensional vector fields we can decompose the fields at each point in space and time into three components along the axes of the Cartesian coordinate system. By doing so we obtain Maxwell's equations in scalar form.
Writing out the curl of the electric field in (1) yields

$$\nabla \times \mathbf{E} = \begin{vmatrix} \hat{\mathbf{x}} & \hat{\mathbf{y}} & \hat{\mathbf{z}} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ E_x & E_y & E_z \end{vmatrix} = (\frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z})\hat{\mathbf{x}} - (\frac{\partial E_z}{\partial x} - \frac{\partial E_x}{\partial z})\hat{\mathbf{y}} + (\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y})\hat{\mathbf{z}}.$$

This results in the scalar equations

$$\frac{\partial H_x}{\partial t} = \frac{1}{\mu_0\mu_r}(\frac{\partial E_y}{\partial z} - \frac{\partial E_z}{\partial y} - M_x - \sigma_m H_x),$$
$$\frac{\partial H_y}{\partial t} = \frac{1}{\mu_0\mu_r}(\frac{\partial E_z}{\partial x} - \frac{\partial E_x}{\partial z} - M_y - \sigma_m H_y),$$
$$\frac{\partial H_z}{\partial t} = \frac{1}{\mu_0\mu_r}(\frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x} - M_z - \sigma_m H_z).$$

Here all these values are defined for each point in space given by $\mathbf{r}$ and for each time t.
Writing out the curl of the magnetic field strength on the right-hand side of equation (1) gives

$$\nabla \times \mathbf{H} = \begin{vmatrix} \hat{x} & \hat{y} & \hat{z} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ H_x & H_y & H_z \end{vmatrix} = (\frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z})\hat{\mathbf{x}} - (\frac{\partial H_z}{\partial x} - \frac{\partial H_x}{\partial z})\hat{\mathbf{y}} + (\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y})\hat{\mathbf{z}}.$$

This results in:

$$\frac{\partial E_x}{\partial t} = \frac{1}{\epsilon_0\epsilon_r}(\frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} - J_x - \sigma E_x),$$
$$\frac{\partial E_y}{\partial t} = \frac{1}{\epsilon_0\epsilon_r}(\frac{\partial H_x}{\partial z} - \frac{\partial H_z}{\partial x} - J_y - \sigma E_y),$$
$$\frac{\partial E_z}{\partial t} = \frac{1}{\epsilon_0\epsilon_r}(\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - J_z - \sigma E_z).$$

The remaining two equations are Gauss's law for the electric fields and the magnetic fields

$$\nabla \cdot \epsilon_r(\mathbf{r})\mathbf{E} = \frac{\rho_f}{\epsilon_0},$$
$$\nabla \cdot \mu_r(\mathbf{r})\mathbf{H} = 0.$$

In the spatial discretization using a Yee grid it turns out [18, pp. 78] that these two equations are automatically satisfied if $\rho_f = 0$, which is the case for photonic crystals, see section (4).

2.3 TM^z mode

It is difficult to solve six equations simultaneously. However, if we assume that the EM fields are homogeneous along the z-axis, in other words if we only consider the vector fields in the xy-plane, then we can reduce the number of equations to be solved. This assumption renders all partial derivatives in the z- direction zero and as a result the scalar equations can be decomposed into two triplets of field components that only interact internally: (H_x, H_y, E_z) and (E_x, E_y, H_z).
We can split all sources and initial condition into two parts, calculate their separate solutions and subsequently superimpose both solutions to obtain the solution for all components.
Following Schneider [17] these two are called the transverse magnetic (TM^z) mode and the transverse electric (TE^z) mode. The former indicates that the magnetic field is perpendicular to the z axis, with unknowns H_x, H_y and E_z and the latter indicates that the electric field is perpendicular to the z-axis thus with unknowns E_x, E_y and H_z.
For simplicity we will only work with the TM^z mode in this report, to reduce the number of equations. However, a model for the TE^z mode can easily be deduced in an analogous fashion.
To summarize, the components of the TM^z mode are given by:

- E_z which varies only in the x-direction and y-direction.
- H_x and H_y which also only vary in the x-direction and y-direction.

For the Maxwell equations this yields:

$$\frac{\partial H_x}{\partial t}(x,y,t) = \frac{1}{\mu_0 \mu_r(x,y)}(-\frac{\partial E_z}{\partial y}(x,y,t) - M_x(x,y,t) - \sigma_m(x,y)H_x(x,y,t)),$$

$$\frac{\partial H_y}{\partial t}(x,y,t) = \frac{1}{\mu_0 \mu_r(x,y)}(\frac{\partial E_z}{\partial x}(x,y,t) - M_y(x,y,t)) - \sigma_m(x,y)H_y(x,y,t)),$$

and

$$\frac{\partial E_z}{\partial t}(x,y,t) = \frac{1}{\epsilon_0 \epsilon_r(x,y)}(\frac{\partial H_y}{\partial x}(x,y,t) - \frac{\partial H_x}{\partial y}(x,y,t) - J_z(x,y,t) - \sigma(x,y)E_z(x,y,t)).$$

3 Maxwell's equations for the TM^z mode in dimensionless form

In the previous section we have discussed Maxwell's equations with their real physical units, with given constants μ_0 ($4\pi \times 10^{-7} \frac{Vs}{Am}$) and ϵ_0 ($8.854 \times 10^{-12} \frac{F}{m}$) in SI units. Since the orders of magnitude of both the electric and magnetic fields are very small and differ significantly from each other there could be inaccuracies during computations. Therefore we use a transformation which scales the electric and magnetic field in such a way that the difference in magnitude is eliminated and all variables are rendered dimensionless.
For now we will denote all variables in Maxwell's equations with SI units with a subscript '$_s$' and the resulting dimensionless variables without a subscript. We start again with Maxwell's equations for the TM^z mode in SI units:

$$\frac{\partial H_{xs}}{\partial t_s}(x_s, y_s, t_s) = \frac{1}{\mu_0 \mu_{rs}(x_s, y_s)}[-\frac{\partial E_{zs}}{\partial y_s}(x_s, y_s, t_s) - M_{xs}(x_s, y_s, t_s) - \sigma_{ms}(x_s, y_s) H_{xs}(x_s, y_s, t_s)],$$

$$\frac{\partial H_{ys}}{\partial t_s}(x_s, y_s, t_s) = \frac{1}{\mu_0 \mu_{rs}(x_s, y_s)}[\frac{\partial E_{zs}}{\partial x_s}(x_s, y_s, t_s) - M_{ys}(x_s, y_s, t_s) - \sigma_{ms}(x_s, y_s) H_{ys}(x_s, y_s, t_s)],$$

and

$$\frac{\partial E_{zs}}{\partial t_s}(x_s, y_s, t_s) = \frac{1}{\epsilon_0 \epsilon_r(x_s, y_s)}[\frac{\partial H_{ys}}{\partial x_s}(x_s, y_s, t_s) - \frac{\partial H_{xs}}{\partial y_s}(x_s, y_s, t_s) \quad - J_z(x_s, y_s, t_s) \quad - \sigma_s(x_s, y_s) E_{zs}(x_s, y_s, t_s)].$$

Next we introduce two scalar parameters: the typical length L in meters and the typical magnetic strength H_0 in $\frac{A}{m}$. These parameters act as normal units to which the rest of the variables are scaled. They can be chosen freely but should be defined in a way which reflects the size of the grid and the magnitude of the magnetic field. Sensible choices would be the length of one side of the domain and the maximal value of the magnetic field that is reached during a simulation for H_0, if this quantity can be estimated.
We also use Z_0, the impedance of vacuum, given by $\sqrt{\frac{\mu_0}{\epsilon_0}}$ and c_0, the speed of light in vacuum, given by $\frac{1}{\sqrt{\mu_0 \epsilon_0}}$.
We transform our variabes with the following transformation as defined above

$$x = \frac{1}{L} x_s, \qquad y = \frac{1}{L} y_s, \qquad t = \frac{c_0}{L} t_s,$$

$$\sigma(x, y) = \sigma_s(x_s, y_s) Z_0 L, \quad \sigma_m(x, y) = \sigma_{ms}(x_s, y_s) Z_0 L, \quad \mu_r(x, y) = \mu_{rs}(x_s, y_s), \quad \epsilon_r(x, y) = \epsilon_{rs}(x_s, y_s)$$

$$E_z(x, y, t) = \frac{1}{H_0 Z_0} E_{zs}(x_s, y_s, t_s) \qquad H_x(x, y, t) = \frac{1}{H_0} H_{xs}(x_s, y_s, t_s)$$

$$H_y(x, y, t) = \frac{1}{H_0} H_{ys}(x_s, y_s, t_s) \qquad J(x, y, t) = \frac{L}{H_0} J_{zs}(x_s, y_s, t_s)$$

$$M_x(x, y, t) = \frac{L}{H_0 Z_0} M_{xs}(x_s, y_s, t_s) \qquad M_y(x, y, t) = \frac{L}{H_0 Z_0} M_{ys}(x_s, y_s, t_s)$$

Now we will rewrite the maxwell equations using these transformations. For readability we introduce notations $\mathbf{u}_s = (x_s, y_s, t_s)$ and $\mathbf{u} = (x, y, t)$ as well as $\mathbf{r}_s = (x_s, y_s)$ and $\mathbf{r} = (x, y)$

$$\mu_0 \mu_{rs}(\mathbf{r}_s) \frac{\partial H_{xs}}{\partial t_s}(\mathbf{u}_s) = -\frac{\partial E_{zs}}{\partial y_s}(\mathbf{u}_s) - M_{xs}(\mathbf{u}_s) - \sigma_{ms}(\mathbf{r}_s) H_{xs}(\mathbf{u}_s),$$

$$\mu_0 \mu_r(\mathbf{r}) H_0 \frac{\partial t}{\partial t_s} \frac{\partial H_x}{\partial t}(\mathbf{u}) = -H_0 Z_0 \frac{\partial y}{\partial y_s} \frac{\partial E_z}{\partial y}(\mathbf{u}) - \frac{H_0 Z_0}{L} M_x(\mathbf{u}) - \frac{H_0 Z_0}{L} \sigma_m(\mathbf{r}) H_x(\mathbf{u}),$$

and finally, using $\mu_0 c_0 = Z_0$, we obtain

$$\frac{\partial H_x}{\partial t}(x, y, t) = \frac{1}{\mu_r(x, y)}[-\frac{\partial E_z}{\partial y}(x, y, t) - M_x(x, y, t) - \sigma_m(x, y) H_x(x, y, t)]$$

In the same manner we can derive that

$$\frac{\partial H_y}{\partial t}(x,y,t) = \frac{1}{\mu_r(x,y)}[\frac{\partial E_z}{\partial x}(x,y,t) - M_y(x,y,t) - \sigma_m(x,y)H_y(x,y,t)]$$

For the time derivative of E_z we get

$$\epsilon_0\epsilon_{rs}(\mathbf{r}_s)\frac{\partial E_{zs}}{\partial t_s}(\mathbf{u}_s) = \frac{\partial H_{ys}}{\partial x_s}(\mathbf{u}_s) - \frac{\partial H_{xs}}{\partial y_s}(\mathbf{u}_s) - J(\mathbf{u}_s) - \sigma_s(\mathbf{r}_s)E_{zs}(\mathbf{u}_s)$$

$$\epsilon_0\epsilon_r(\mathbf{r})H_0Z_0\frac{\partial t}{\partial t_s}\frac{\partial E_z}{\partial t}(\mathbf{u}) = H_0\frac{\partial x}{\partial x_s}\frac{\partial H_y}{\partial x}(\mathbf{u}) - H_0\frac{\partial y}{\partial y_s}\frac{\partial H_x}{\partial y}(\mathbf{u}) - \frac{H_0}{L}J(\mathbf{u}) - H_0Z_0\sigma_s(\mathbf{r}_s)E_z(\mathbf{u})$$

Using that $\epsilon_0 c_0 = \frac{1}{Z_0}$ we get

$$\frac{\partial E_z}{\partial t}(x,y,t) = \frac{1}{\epsilon_r(x,y)}[\frac{\partial H_y}{\partial x}(x,y,t) - \frac{\partial H_x}{\partial y}(x,y,t) - J(x,y,t) - \sigma(x,y)E_z(x,y,t)].$$

For the Courant number which sets a relation between the step size in time and space of a grid and is defined as $S_c = \frac{c_0\Delta t_s}{\Delta x_s}$ we get

$$S_c = c_0\frac{\Delta t_s}{\Delta x_s} = \frac{Lc_0}{Lc_0}\frac{\Delta t}{\Delta x} = \frac{\Delta t}{\Delta x}$$

Furthermore for a frequency ω_s in SI units we get the following

$$\omega_s = \frac{2\pi}{T_s} = \frac{2\pi c_0}{TL} = \frac{c_0}{L}\omega \tag{4}$$

4 Test configurations

In this section two photonic crystal configurations are presented that will be used to test numerical methods in section (10).
As stated in section (1) photonic crystals are dielectric materials with permittivities that are periodic in space [13, pp. 2]. They are often periodic in up to three dimensions and uniform in the non periodic ones [13, pp. 5]. The configurations are required to be two-dimensional because we want to test numerical methods by simulating electromagnetic fields in TM^z mode. Photonic crystals can consist of materials with many different permittivities and can even have continuously varying permittivities. However, many interesting phenomena can already be demonstrated for photonic crystals that are built from two dielectric materials.
One way to construct a two-dimensional photonic crystal out of two dielectric materials is to take one of these materials and place cylindrical slabs of the other dielectric into it. We will call such a dielectric cylinder a rod. Since photonic crystals are periodic, the arrangement of rods must also be periodic. In other words, the rods must form a lattice. There are various ways to construct two-dimensional crystal, see for example [13, pp. 72].
There is a wide range of interesting lattices. The prototypical example of a photonic crystal is a square lattice of dielectric rods in a vacuum [13, pp. 68], but for example triangular lattices are also widely used.
Recall from section (1) that photonic crystals have band gaps, frequency ranges for which electromagnetic waves cannot propagate through the photonic crystal. For frequencies outside the band gap, electromagnetic waves propagate unhindered through the crystal and do not lose cohesion due to scattering from the constituent structures of the photonic crystal [13, pp. 2].
Photonic crystals can also have defects, irregularities within the structure of the crystal, that largely leave the periodicity intact but locally break it. This means that any mathematical methods that are based on periodicity are not strictly speaking applicable but usually the conclusions based on these methods are still valid, see for example [13, pp. 58]. We will encounter point defects and line defects in the remainder of this section as well as use them for as test cases. Examples of point defects are a single dielectric rod that is reduced in size, has a different permittivity or is absent altogether [13, pp. 87]. Line defects usually consist of a row or a column of rods that is missing or altered in some way [13, pp. 88]. As we will see, line defects can be used to guide light.
There are generally no charges or currents present in a photonic crystal [13, pp. 2]. Therefore the electromagnetic fields inside a photonic crystal are not generated by the crystal itself, that is, photonic crystals do not radiate. The electromagnetic fields inside a photonic crystal usually come from outside the photonic crystal. In most cases we will consider an electromagnetic wave with one specified frequency. However sometimes a pulse that is comprised of a wide range of frequencies is used.
We have chosen two simple settings, one with a point defect and one with a line defect, to test our simulations.
The first is a square lattice of dielectric rods, which in its center has a point defect, in the form of one column with a lower permittivity. The simulation will be initiated in a small circle in the point defect, using a sine wave with a frequency inside of the crystal's band gap. For longer simulations we expect to see patterns as described in [13, pp. 81]. A simple illustration of this setting can be seen in 1 a.
Next we will model a different square lattice of rods, containing a line defect. This will be achieved by removing the middle row of the grid, as illustrated in 1 b. The system will again be initiated in a small circle, this time in the middle of the line defect and again by using a sine wave from within the band gap. Since the wave will be unable to propagate through the lattice it will be channeled through the line defect.
The combination of multiple line defects can lead to more complex channels named waveguides and allow for control of the direction of the flow of light.

We will also consider two other settings which will not be used for numerical testing, but which do present interesting physical properties of photonic crystals.

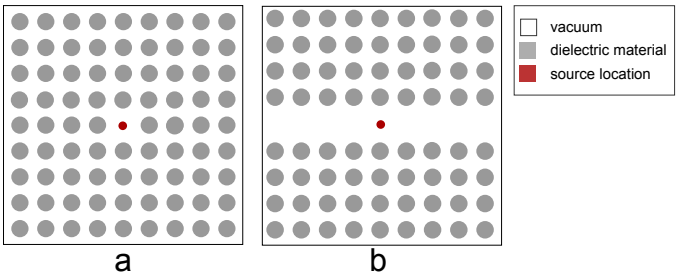

Figure 1: The two test cases

One is the waveguide bend in a square lattice as depicted in Figure 2a. This scenario consists of two line defects which intersect at a right angle to form a bend. A wave with a frequency inside the band gap will be channeled through the bend, showing that the mechanism of a waveguide is quite robust. There are also other ways to build a waveguide, such as using metallic walls [8, pp. 405] or guiding light through a dielectric fiber using the mechanism of total internal reflection [13, pp. 87]. Interestingly, using photonic crystal waveguides, light can be guided through a region that mostly consists of air [13, pp. 87]. An advantage of using a photonic crystal instead of a conventional waveguide is that bends can be easily implemented [13][pp. 87].
The other setting is a line defect inside of a triangular lattice as shown in Figure 2b. You could see the photonic crystal as having a surface, however, unlike the prototypical example of dielectric rods in air discussed above, the roles of dielectric and air have been interchanged. The photonic crystal now consists of air holes drilled into a dielectric. The advantage of a triangular lattice is that it has a band gap for any plane wave since the band gaps for electromagnetic waves in the TM^z mode and TE^z mode overlap. We will not be able to demonstrate this since we will only implement the Maxwell equations for the TM^z mode.

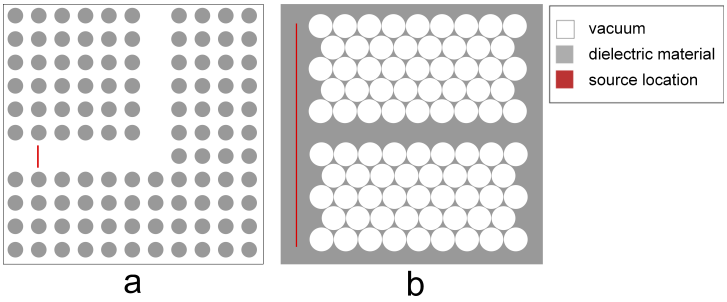

Figure 2: The extra settings test cases

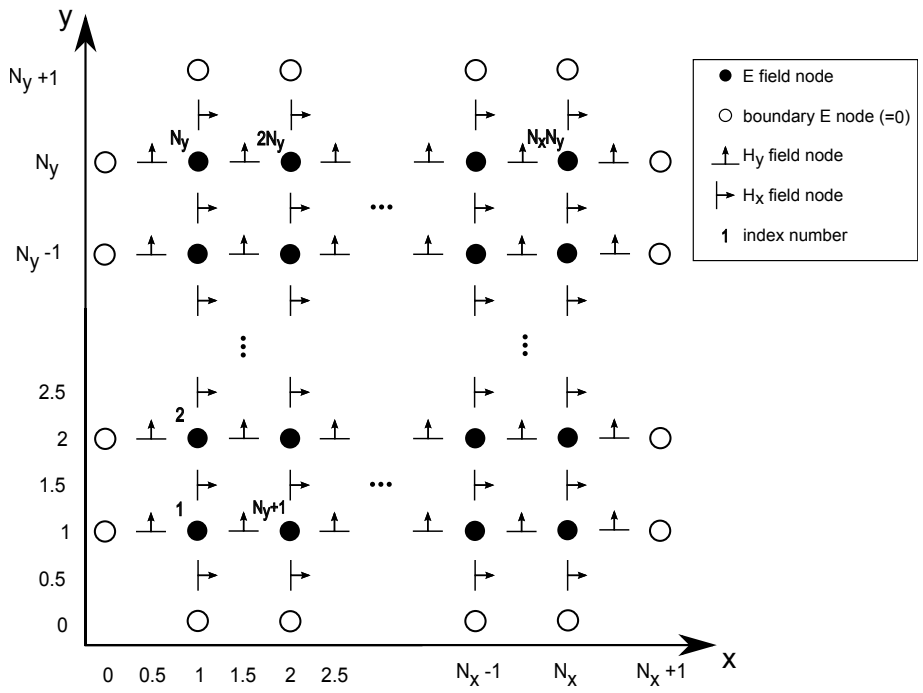

Figure 3: the TM^z Yee grid. The indices (m,n) are shown, not the spatial coordinates $(m\Delta x, n\Delta y)$.

5 Discretization

5.1 Grid

In this section the TM^z Yee grid, introduced in [20], is described.
Let $(0, l_x) \times (0, l_y) \subset \mathbb{R}^2$ be the domain that is to be discretized. Set the number of grid points N_x and N_y along the x-axis and y-axis. The spatial step sizes Δx and Δy are given by

$$\Delta x := \frac{l_x}{N_x + 1} \qquad\qquad \Delta y := \frac{l_y}{N_y + 1}.$$

An E_z node is a grid point at which a numerical approximation of the E_z component is defined. The TM^z Yee grid consists of the following sets of E_z,H_x and H_y nodes

$$\begin{aligned}
G_{E_z} &:= \{(m\Delta x, n\Delta y) : m \in \{1, \ldots, N_x\}, n \in \{1, \ldots, N_y\}\},\\
G_{H_x} &:= \{(m\Delta x, (n - \frac{1}{2})\Delta y) : m \in \{1, \ldots, N_x\}, n \in \{1, \ldots, N_y + 1\}\},\\
G_{H_y} &:= \{((m - \frac{1}{2})\Delta x, n\Delta y) : m \in \{1, \ldots, N_x + 1\}, n \in \{1, \ldots, N_y\}\},
\end{aligned}$$

see figure 3. Note the spatial staggering of the nodes.
Although E_z, H_x and H_y are only approximated on G_{E_z}, G_{H_x} and G_{H_y} respectively, the spatially discretized Maxwell equations introduced in the next section contain approximations of spatial derivatives of E_z on G_{H_x} and G_{H_y} and spatial derivatives of H_x and H_y on G_{E_z}. However these approximations are expressed in terms of the numerical E_z, H_x and H_y fields at the corresponding nodes.
Note that no nodes on the boundary have been included in this definition. The corresponding fields are set to zero in the spatially discretized Maxwell equations.

5.2 Space discretization of Maxwell's equations

In this section Faraday's equation for the H_x component is spatially discretized. Spatial discretizations of the other dimensionless Maxwell equation with the constitutive relationships from section (2) can be found in the appendix. Faraday's equation for the H_x component is given by

$$\mu_r(x,y)\frac{\partial H_x(x,y,t)}{\partial t} = -\frac{\partial E_z(x,y,t)}{\partial y} - (M_x(x,y,t) + \sigma_m(x,y)H_x(x,y,t)),$$

Central difference approximations to the derivative $\frac{\partial E_z}{\partial y}$ yield the following ODE's

$$\begin{aligned}\mu_r(m\Delta x,(n-\frac{1}{2})\Delta y)\frac{\partial H_x}{\partial t}(m\Delta x,(n-\frac{1}{2})\Delta y,t) = &-\frac{E_z(m\Delta x,n\Delta y,t)-E_z(m\Delta x,(n-1)\Delta y,t)}{\Delta y}\\ &-(M_x(m\Delta x,(n-\frac{1}{2})\Delta y,t)\\ &+\sigma_m(m\Delta x,n\Delta y)H_x(m\Delta x,n\Delta y,t)).\end{aligned}$$

As mentioned in section (5.1) any terms that refer to nodes on the boundary are set to zero.

5.3 Matrix formulation of the space-discretized Maxwell equations

Definition of vectors

The spatially discretized Maxwell equations constitute a system of ODE's that can be written in matrix form. To this end the H_x components on G_{H_x}, H_y components on G_{H_y} and E_z components on G_{E_z} are collected in vectors $\mathbf{h}_x(t)$, $\mathbf{h}_y(t)$ and $\mathbf{e}_z(t)$ respectively which are put into a vector

$$\mathbf{y}(t) := \begin{bmatrix}\mathbf{h}_x(t)\\ \mathbf{h}_y(t)\\ \mathbf{e}_z(t)\end{bmatrix}$$

The nodes are distributed over a two-dimensional domain. However the numerical field components associated with the nodes are put into vectors that have only a single index. Clearly we need to define some way of assigning indices to nodes in a two-dimensional grid. We give the node corresponding to some particular field component with the smallest x-coordinate and y-coordinate[1] the index 1 and count in the y-direction. After exhausting a column of nodes we continue with the next value of x, count in the y-direction and so on. Therefore H_x node $(m\Delta x,(n-\frac{1}{2})\Delta y)\in G_{H_x}$, H_y node $((m-\frac{1}{2})\Delta x,n\Delta y)\in G_{H_y}$ and E_z node $(m\Delta x,n\Delta y)\in G_{E_z}$ have indices

$$\begin{aligned}i &:= (m-1)(N_y+1)+n,\\ j &:= (m-1)N_y+n,\\ k &:= (m-1)N_y+n,\end{aligned}$$

in $\mathbf{h}_x(t)$, $\mathbf{h}_y(t)$ and $\mathbf{e}_z(t)$ respectively. These indices are shown next to the nodes in figure 3. The number of H_x, H_y and E_z nodes is given by

$$\begin{aligned}n_{h_x} &:= N_x(N_y+1),\\ n_{h_y} &:= (N_x+1)N_y,\\ n_{e_z} &:= N_xN_y,\end{aligned}$$

respectively. Note that the definition symbol is used because this is a definition of these variables. The number of nodes is determined by the definition of the Yee grid. So H_x node $(m\Delta x,(n-\frac{1}{2})\Delta y)\in G_{H_x}$, H_y node $((m-\frac{1}{2})\Delta x,n\Delta y)\in G_{H_y}$ and E_z node $(m\Delta x,n\Delta y)\in G_{E_z}$ have indices i, $n_{h_x}+j$ and $n_{h_x}+n_{h_y}+k$ in the vector $\mathbf{y}(t)$ respectively.

[1] Obviously this node exists in a TM^z Yee grid.

System of ordinary differential equations

In the appendix it is shown that the spatially discretized Maxwell equations can be written in the following form

$$\begin{bmatrix} \frac{d}{dt}\mathbf{h}_x(t) \\ \frac{d}{dt}\mathbf{h}_y(t) \\ \frac{d}{dt}\mathbf{e}_z(t) \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{\mu_r,x} & 0 & 0 \\ 0 & \mathbf{P}_{\mu_r,y} & 0 \\ 0 & 0 & \mathbf{P}_{\epsilon_r} \end{bmatrix}^{-1} \left(\begin{bmatrix} -\mathbf{\Sigma}_x & \mathbf{0} & \mathbf{K} \\ \mathbf{0} & -\mathbf{\Sigma}_y & \mathbf{L} \\ -\mathbf{K}^\intercal & -\mathbf{L}^\intercal & -\mathbf{\Sigma}_z \end{bmatrix} \begin{bmatrix} \mathbf{h}_x(t) \\ \mathbf{h}_y(t) \\ \mathbf{e}_z(t) \end{bmatrix} - \begin{bmatrix} \mathbf{M}_x\,(t) \\ \mathbf{M}_y\,(t) \\ \mathbf{J}\,(t) \end{bmatrix} \right) \tag{5}$$

It is easy to see that this is a system of the form $\mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}$. Here $\mathbf{M}_x(t)$, $\mathbf{M}_y(t)$ and $\mathbf{J}(t)$ are vectors that contain the values of the magnetic and electric (free) source currents at the nodes in G_{H_x}, G_{H_y} and G_{E_z} respectively. $\mathbf{P}_{\mu_r,x}$, $\mathbf{P}_{\mu_r,y}$ and $\mathbf{P}_{\epsilon_r}$ are diagonal matrices with the relative pemeabilities at the H_x and H_y nodes and the relative permittivities at the E_z nodes as diagonal elements. $\mathbf{\Sigma}_x$, $\mathbf{\Sigma}_y$ and $\mathbf{\Sigma}_z$ are diagonal matrices with the dimensionless (magnetic) conductivities at the diagonal elements. $\mathbf{K}$ and $\mathbf{L}$ can be described using the Kronecker product

$$\mathbf{K} = -(\Delta y)^{-1}(\mathbf{I}_{N_x} \otimes \mathbf{A}_{N_y}) \qquad\qquad \mathbf{L} = (\Delta x)^{-1}(\mathbf{A}_{N_x} \otimes \mathbf{I}_{N_y}).$$

The $(n+1) \times n$ matrix $\mathbf{A}_n$ that appears in this system is given by

$$\mathbf{A}_n = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ -1 & 1 & & 0 & 0 \\ 0 & -1 & \ddots & & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & 0 & & -1 & 1 \\ 0 & 0 & \cdots & 0 & -1 \end{bmatrix}.$$

6 FDTD

In this section the FDTD method, introduced by [20], is described. Both the FDTD method and the Krylov subspace methods discretize the spatial derivatives of the electric and magnetic fields using the staggered Yee grid as described in section (5). What sets the two methods apart is that the FDTD method also discretizes the time whereas Krylov solves the system of spatially discretized Maxwell equations using matrix exponentials.
From Maxwell's equations we know that the rate of change of the electric field depends on the spatial derivatives of magnetic field and that the rate of change of the magnetic field depends on the spatial derivatives of the electric field.
The FDTD method incorporates this interaction by using a leapfrog scheme, which means that the discretized magnetic components and the electric components are defined on alternating points in time. More specifically the electric field is defined on $T_e := \{t_i : t_i = t_0 + i\Delta t, i \in \mathbb{N}\}$ and the magnetic field on $T_h := \{t_{i-0.5} : t_{i-0.5} = t_0 + (i-0.5)\Delta t, i \in \mathbb{N}\}$, provided the initial conditions are given for time t_0.
This entails that the value of the magnetic field at time $t_{i+0.5}$ is obtained by linear extrapolation using the value of $t_{i-0.5}$ and its derivative at time t_i which is related to the electric field at that time by Faraday's equation. If we implement this time discretization for the dimensionless Maxwell equations we obtain

$$H_x(m\Delta x, n\Delta y, t_{i+0.5}) = H_x(m\Delta x, n\Delta y, t_{i-0.5}) + \frac{\Delta t}{\mu_r(m\Delta x, n\Delta y)}[-\frac{1}{\Delta y}(E_z((m+0.5)\Delta x, n\Delta y, t_i)$$
$$- E_z((m-0.5)\Delta x, n\Delta y, t_i)) - M_x(m\Delta x, n\Delta y, t_i) - \sigma_m(m\Delta x, n\Delta y)H_x(m\Delta x, n\Delta y, t_i)]$$

The only problem is that H_x is not defined at time t_i in the FDTD scheme but its value is required in the conductivity term. Therefore we use linear extrapolation to approximate this value: $H_x(m\Delta x, n\Delta y, t_i) \approx (H_x(m\Delta x, n\Delta y, t_{i+0.5}) + H_x(m\Delta x, n\Delta y, t_{i-0.5}))/2$. This results in the explicit equation

$$\begin{aligned} H_x(m\Delta x, n\Delta y, t_{i+0.5}) = {} & \frac{2\mu_r(m\Delta x, n\Delta y) + \sigma_m(m\Delta x, n\Delta y)\Delta t}{2\mu_r(m\Delta x, n\Delta y) - \sigma_m(m\Delta x, n\Delta y)\Delta t} H_x(m\Delta x, n\Delta y, t_{i-0.5}) \\ & + \frac{2\Delta t}{\mu_r(m\Delta x, n\Delta y)} 2\mu_r(m\Delta x, n\Delta y) - \sigma_m(m\Delta x, n\Delta y)\Delta t \\ & [-\frac{E_z((m+0.5)\Delta x, n\Delta y, t_i) - E_z((m-0.5)\Delta x, n\Delta y, t_i)}{\Delta y} - M_x(m\Delta x, n\Delta y, t_i)] \end{aligned}$$

Conventionally the FDTD method is implemented by looping over all the nodes of the Yee-grid and updating them in the way listed above [18], [17]. However we have implemented FDTD using the matrix equations from the previous chapter.
As a result we obtain

$$\mathbf{h}_x(t_{i+0.5}) = (2\mathbf{P}_{\mu_r} + \Delta t\mathbf{\Sigma}_x)^{-1}(2\mathbf{P}_{\mu_r} - \Delta t\mathbf{\Sigma}_x)\mathbf{h}_x(t_{i-0.5}) + 2\Delta t(2\mathbf{P}_{\mu_r} + \Delta t\mathbf{\Sigma}_x)^{-1}[\mathbf{K}\mathbf{e}_z(t_i) - \mathbf{M}_x(t_i)]$$

$$\mathbf{h}_x(t_{i+0.5}) = (2\mathbf{P}_{\mu_r} + \Delta t\mathbf{\Sigma}_y)^{-1}(2\mathbf{P}_{\mu_r} - \Delta t\mathbf{\Sigma}_y)\mathbf{h}_x(t_{i-0.5}) + 2\Delta t(2\mathbf{P}_{\mu_r} + \Delta t\mathbf{\Sigma}_y)^{-1}[\mathbf{L}\mathbf{e}_z(t_i) - \mathbf{M}_y(t_i)]$$

$$\mathbf{e}_z(t_{i+1}) = (2\mathbf{P}_{\epsilon_r} + \Delta t\mathbf{\Sigma}_z)^{-1}(2\mathbf{P}_{\epsilon_r} - \Delta t\mathbf{\Sigma}_z)\mathbf{e}_z(t_i) + 2\Delta t(2\mathbf{P}_{\epsilon_r} + \Delta t\mathbf{\Sigma}_z)^{-1}[-\mathbf{K}^\intercal\mathbf{h}_x(t_{i+0.5}) - \mathbf{L}^\intercal\mathbf{h}_y(t_{i+0.5}) - \mathbf{J}(t_{i+0.5})]$$

6.1 Stability of the update equations

The update equations must be numerically stable. Roughly stability means that approximation errors are not magnified as the simulation runs. An numerically unstable algorithm tends to quickly diverge to infinite values. A necessary condition for the stability of a time discretization scheme is the Courant-Friedrichs-Lewy conditions [4], [18, pp. 128]. Roughly it assumes that during one time step of the simulation the EM fields of the analytical solution cannot propagate any further than the spatial step of the simulation. It is easy to see that it should be necessary that any part of

the initial conditions that influence a particular point in space and time in the analytical solution should also be able to propagate to this point in the numerical grid in the requisite number of time steps. Note that the Courant-Friedrichs-Lewy condition is only a necessary condition for stability and not a sufficient one. Also it is allowed that the fields propagate much faster in the numerical grid than in the physical domain.[2] Symbolically the Courant-Friedrichs-Lewy condition states for a grid where $\Delta x = \Delta y$ that

$$\frac{\Delta t}{\Delta x} = \frac{1}{\sqrt{2}}$$

where all quantities are dimensionless.

[2]Of course this will add to the error term. However in the update equation a division is performed with respect to the spatial step size. Any terms that have propagated too fast in the grid have also propagated over large distances so their size will usually be small.

7 Krylov subspace methods for exponential time integration

In this section we will introduce several Krylov subspace methods that can be used for exponential time integration. The Krylov subspace is a vector space which is defined for an n by n matrix $\mathbf{A}$ and an n-dimensional vector $\mathbf{v}$ as follows:

$$K_m(\mathbf{A}, \mathbf{v}) := \{\mathbf{v}, \mathbf{A}\mathbf{v}, \mathbf{A}^2\mathbf{v}, \ldots, \mathbf{A}^{m-1}\mathbf{v}\} \tag{7}$$

In this section we will show some methods that use the Krylov subspace to approximate the action of a matrix function of $\mathbf{A}$ on $\mathbf{v}$. In the case that the matrix function is the exponent of $\mathbf{A}$ it can be used to solve system of equations of the form

$$\frac{d\mathbf{y}}{dt} = \mathbf{A}\mathbf{y}, \quad \mathbf{y}(0) = \mathbf{v}, \tag{8}$$

where $\mathbf{y} : \mathbb{R} \to \mathbb{R}^n$ is a vector function. The analytical solution of this equation is

$$\mathbf{y}(t) = e^{\mathbf{A}t}\mathbf{v}.$$

The methods introduced will approximate the solution $\mathbf{y}(t)$ with an accuracy based on the residual of the solution. So they do not discretize in time but give an approximation of the exact solution of (8).
First the Arnoldi algorithm will be introduced which is the main algorithm which will be used in all methods. After that, methods for solving (8) will be described but also for similar equations with constant and time dependent source currrents.

7.1 The Arnoldi method

The Arnoldi method as described in [16] is an algorithm that uses the Gram-Schmidt procedure to construct an orthonormal basis for the Krylov subspace (7). The algorithm constructs a matrix $\mathbf{V}_m$ whose columns are the orthonormal basis vectors of (7). It also constructs a matrix $\mathbf{H}_{m+1,m}$ that relates $\mathbf{A}$ and $\mathbf{V}_m$ in the following way:

$$\mathbf{A}\mathbf{V}_m = \mathbf{V}_{m+1}\mathbf{H}_{m+1,m} \tag{9}$$

Here $\mathbf{V}_m$ is an orthonormal basis for the Krylov subspace of order m, $\mathbf{H}_{m+1,m}$ is upper-Hessenberg which means that if $i > j + 1$ the value of $h_{i,j} = 0$. So $\mathbf{H}_{m+1,m}$ has one diagonal below the main diagonal and below that all the values are zero. Note that this equation does not describe a recursive algorithm.
The Arnoldi algorithm that constructs these matrices is depicted in pseudo code below in (10).

```
(1)   β = ‖v‖₂
(2)   v₁ = v/β
(3)   for j = 1, 2 ... m do
(4)     w = A v_j
(5)     for i = 1, 2 ... j do
(6)       h_{i,j} = w · v_i
(7)       w = w − h_{i,j} v_i
(8)     end for
(9)     h_{j+1,j} = ‖w‖₂
(10)    if h_{j+1,j} = 0 stop
(11)    v_{j+1} = w / h_{j+1,j}
(12)  end for
```
(10)

After studying the algorithm you will notice that the matrix $\mathbf{H}_{m+1,m}$ is constructed in such a way that the following formula will hold.

$$\mathbf{A}\mathbf{v}_i = \sum_{j=1}^{i+1} h_{j,i}\mathbf{v}_j$$

The new vector $\mathbf{A}\mathbf{v}_i$ will be a linear combination of the first i orthonormal vectors out of the basis for $K_i(\mathbf{A}, \mathbf{v})$ with an additional vector $\mathbf{v}_{i+1}$ which is orthonormal to the preceding i vectors.
Let $\mathbf{h}_i$ be column i of $\mathbf{H}_{m+1,m}$, then this equation can be written in the following form.

$$\mathbf{A}\mathbf{v}_i = \mathbf{V}_{m+1}\mathbf{h}_i$$

In $\mathbf{h}_i$ now only the first $i+1$ entries can have a nonzero value so $\mathbf{H}_{m+1,m}$ is indeed upper-Hessenberg. This equation can now be generalized even further to get equation (9) as follows.

$$\mathbf{A}\mathbf{V}_m = \mathbf{A}[\mathbf{v}_1\mathbf{v}_2 \dots \mathbf{v}_m] = \mathbf{V}_{m+1}[\mathbf{h}_1\mathbf{h}_2 \dots \mathbf{h}_m] = \mathbf{V}_{m+1}\mathbf{H}_{m+1,m}$$

The matrix $\mathbf{H}_{m+1,m}$ has only zero values on row $m+1$ except for $h_{m+1,m}$, this term is the norm of $\mathbf{A}\mathbf{v}_m$ after orthogonalizing it with respect to the first m columns of $\mathbf{V}_m$. The expectation is that this term will approach zero for increasing m. This is because the columns of $\mathbf{V}_m$ are a basis for $K_m(\mathbf{A}, \mathbf{v})$ and the factor $\mathbf{A}^m\mathbf{v}$ will converge to a certain eigenvector of $\mathbf{A}$ with a big eigenvalue. This can be seen as follows, let $\mathbf{v}$ be a linear combination of eigenvectors of $\mathbf{A}$

$$\mathbf{v} = \sum_{i=1}^{k} \alpha_i \mathbf{y}_i,$$

where $\mathbf{y}_i$ are eigenvectors with corresponding eigenvalues γ_i, and α_i are constants. Then we get the following representation for $\mathbf{A}^m\mathbf{v}$

$$\mathbf{A}^m\mathbf{v} = \sum_{i=1}^{k} \alpha_i \mathbf{A}^m\mathbf{y}_i = \sum_{i=1}^{k} \alpha_i \gamma_i^m \mathbf{y}_i$$

So the eigenvector with the largest eigenvalue will become dominant and $\mathbf{A}^m\mathbf{v}$ will converge to this eigenvector. The part of $\mathbf{A}^m\mathbf{v}$ that is orthogonal with respect to the last m vectors of $K_{m+1}(\mathbf{A}, \mathbf{v})$ is therefore expected to approach zero.
Because $h_{m+1,m}$ is the only nonzero value of row $m+1$ of $\mathbf{H}_{m+1,m}$ formula (9) can easily be written into the following form.

$$\mathbf{A}\mathbf{V}_m = \mathbf{V}_m\mathbf{H}_{m,m} + h_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T \tag{11}$$

Here $\mathbf{H}_{m,m}$ is $\mathbf{H}_{m+1,m}$ without row $m+1$.
Under the assumption that $h_{m+1,m}$ will converge to zero the following equation will hold for large m.

$$\mathbf{A}\mathbf{V}_m \approx \mathbf{V}_m\mathbf{H}_{m,m} \tag{12}$$

If m is large enough, i.e. $h_{m+1,m}$ is small, this equation says that $\mathbf{A}$ times a vector out of the Krylov subspace will approximately be a vector out of the Krylov subspace again. When the algorithm breaks because $h_{m+1,m} = 0$ this equation is exact and $\mathbf{V}_m$ is also base for Krylov subspaces of higher order than m.
We will now extend this concept to functions of matrices which are defined as follows, this definition can be found in [10]. Let $[\lambda_1, \lambda_2, \dots, \lambda_q]$ be the eigenvalues of $\mathbf{A}$ let p_i be the multiplicity of λ_i and let $f(x)$ be a given function. Then a function $f(\mathbf{A})$ can be defined as a polynomial for which f and its derivatives $f^{(1)}, f^{(2)} \dots f^{(p_i-1)}$ exist for each λ_i.
So if $\mathbf{A}$ times a vector out of the Krylov subspace is approximately again a Krylov subspace vector then this concept will hold for $f(\mathbf{A})$, which is a polynomial of $\mathbf{A}$, too.

$$f(\mathbf{A})\mathbf{V}_m \approx \mathbf{V}_m f(\mathbf{H}_{m,m})$$

This can be seen simply if $f(\mathbf{A}) = \mathbf{A}^2$ as follows using (12):

$$\mathbf{A}^2\mathbf{V}_m \approx \mathbf{A}\mathbf{V}_m\mathbf{H}_{m,m} \approx \mathbf{V}_m\mathbf{H}_{m,m}^2.$$

The matrix exponent is defined as a power series in A so e^A is a polynomial and we have that:

$$e^{\mathbf{A}}\mathbf{V}_m \approx \mathbf{V}_m e^{\mathbf{H}_{m,m}}. \tag{13}$$

Some theorems about the error of this estimation are given in [11]. It turns out that this approximation is quite accurate for $m << n$.

7.2 Numerical solution of $\mathbf{y}' = \mathbf{A}\mathbf{y}$

In this chapter the Krylov subspace is used to give an approximation of the solution of the equation $\mathbf{y}' = \mathbf{A}\mathbf{y}$. Also an expression for the residual of this approximation is given.
If the initial value of $\mathbf{y}$ on $t = 0$ equals $\mathbf{v}$ then the analytical solution is as follows.

$$\mathbf{y}(t) = e^{t\mathbf{A}}\mathbf{v}$$

Now let $\mathbf{V}_m$ be an orthonormal basis for $K_m(\mathbf{A}, \mathbf{v})$ constructed by the Arnoldi algorithm. Additionally let β be the norm of $\mathbf{v}$. Then $\mathbf{v} = \beta\mathbf{V}_m\mathbf{e}_1$, where $\mathbf{e}_1$ is the first identity vector, holds. And so:

$$\mathbf{y}(t) = e^{t\mathbf{A}}\mathbf{V}_m\beta\mathbf{e}_1$$

This gives the next approximation $\mathbf{y}_m(t)$ for $\mathbf{y}(t)$:

$$\mathbf{y}(t) \approx \mathbf{y}_m(t) = \mathbf{V}_m e^{t\mathbf{H}_{m,m}}\beta\mathbf{e}_1 = \mathbf{V}_m\mathbf{u}_m(t) \tag{14}$$

With $\mathbf{u}_m(t)$ defined as:

$$\mathbf{u}_m(t) := e^{t\mathbf{H}_{m,m}}\beta\mathbf{e}_1$$

The approximation $\mathbf{y}_m(t)$ needs to estimate the differential equation so the residual $\mathbf{r}_m(t)$ is defined as follows:

$$\mathbf{r}_m(t) := \mathbf{A}\mathbf{y}_m(t) - \mathbf{y}'_m(t) \tag{15}$$

With the use of (11) and (14), (15) can be written in a simpler form.

$$\begin{array}{rcl} \mathbf{r}_m(t) & = & \mathbf{A}\mathbf{y}_m(t) - \mathbf{y}'_m(t) \\ & = & \mathbf{A}\mathbf{V}_m\mathbf{u}_m(t) - \mathbf{V}_m\mathbf{u}'_m(t) \\ & = & (\mathbf{V}_m\mathbf{H}_{m,m} + h_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T)\mathbf{u}_m(t) - \mathbf{V}_m\mathbf{H}_{m,m}\mathbf{u}_m(t) \\ & = & h_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T\mathbf{u}_m(t) \\ & = & (h_{m+1,m}\mathbf{e}_m^T\mathbf{u}_m(t))\mathbf{v}_{m+1} \\ |\mathbf{r}_m(t)| & = & |h_{m+1,m}\mathbf{e}_m^T\mathbf{u}_m(t)| \end{array} \tag{16}$$

So the residual is proportional with $h_{m+1,m}$ and the size of the m^{th} entry of $\mathbf{u}_m(t)$

7.3 Extension to $\mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}$

This section deals with the equation $\mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}$. This will be done by using Krylov subspace and the Arnoldi algorithm. The analytical solution of this differential equation is derived here.

$$\begin{array}{rcl} \mathbf{y}' & = & \mathbf{A}\mathbf{y} + \mathbf{b} \\ (\mathbf{y} + \mathbf{A}^{-1}\mathbf{b})' & = & \mathbf{A}(\mathbf{y} + \mathbf{A}^{-1}\mathbf{b}) \\ \mathbf{y}(t) & = & e^{t\mathbf{A}}(\mathbf{v} + \mathbf{A}^{-1}\mathbf{b}) - \mathbf{A}^{-1}\mathbf{b} \end{array}$$

A problem of this solution is that the action of two matrix functions has to be calculated and that $\mathbf{A}^{-1}$ might not exist. This can be avoided by the ϕ-operator as in [2]. This operator is an iteration on e^x and its definition is given here.

$$\phi_{i+1}(x) := \frac{\phi_i(x) - \phi_i(0)}{x} \quad \phi_0 := e^x$$

$$\phi_1(x) = \frac{e^x - 1}{x} = 1 + \frac{x}{2!} + \frac{x^2}{3!} + \dots$$

With the use of this operator we can rewrite the analytical solution of $\mathbf{y}' = \mathbf{A}\mathbf{y} + \mathbf{b}$ to the following form.

$$y(t) = t\phi_1(t\mathbf{A})(\mathbf{A}\mathbf{v} + \mathbf{b}) + \mathbf{v} \tag{17}$$

This solution can be approximated in the same way as in the preceding chapter using the Arnoldi algorithm. This will be done on the Krylov space $K_m(\mathbf{A}, \mathbf{A}\mathbf{v} + \mathbf{b})$. If β is chosen as the norm of $\mathbf{A}\mathbf{v} + \mathbf{b}$ then $\mathbf{y}(t)$ can be approximated as follows.

$$\mathbf{y}(t) \approx \mathbf{y}_m(t) = \mathbf{V}_m t\phi_1(t\mathbf{H}_{m,m})\beta\mathbf{e}_1 + \mathbf{v} = \mathbf{v} + \mathbf{V}_m\mathbf{u}_m(t)$$

With $\mathbf{u}_m(t)$ defined as:

$$\mathbf{u}_m(t) := t\phi_1(t\mathbf{H}_{m,m})\beta\mathbf{e}_1$$

The advantage of using ϕ_1 instead of the matrix exponential is that the effect of $\mathbf{A}^{-1}$ on $\mathbf{b}$ does not have to be calculated. This would be additional work and this representation is also defined if $\mathbf{A}^{-1}$ does not exist. The approximation of $\phi_1(\mathbf{H}_{m+1,m})$ is just slightly different from the approximation of the matrix exponential so this does not pose a problem.
The residual of this approximation is defined as $\mathbf{r}_m(t) := \mathbf{A}\mathbf{y}_m(t) - \mathbf{y}'_m(t) + \mathbf{b}$. To transform this expression into an easier one it will come in handy to calculate the derivative of $\mathbf{u}_m(t)$ first. This will be done by rewriting this expression to a matrix exponential form.

$$\begin{aligned} \mathbf{u}_m(t) &= \mathbf{H}_{m,m}^{-1}(\mathbf{e}^{t\mathbf{H}_{m,m}} - \mathbf{I}_m)\beta\mathbf{e}_1 \\ \mathbf{u}'_m(t) &= e^{t\mathbf{H}_{m,m}}\beta\mathbf{e}_1 \\ &= \mathbf{H}_{m,m}\mathbf{u}_m(t) + \beta\mathbf{e}_1 \end{aligned}$$

Now a formula for the residual will be derived.

$$\begin{aligned} \mathbf{r}_m(t) &= \mathbf{A}\mathbf{y}_m(t) - \mathbf{y}'_m(t) + \mathbf{b} \\ &= \mathbf{A}\mathbf{v} + \mathbf{A}\mathbf{V}_m\mathbf{u}_m(t) - \mathbf{V}_m\mathbf{H}_{m,m}\mathbf{u}_m(t) - \mathbf{V}_m\beta\mathbf{e}_1 + \mathbf{b} \\ &= \mathbf{A}\mathbf{v} + (\mathbf{V}_m\mathbf{H}_{m,m} + h_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T)\mathbf{u}_m(t) - \mathbf{V}_m\mathbf{H}_{m,m}\mathbf{u}_m(t) - \mathbf{b} - \mathbf{A}\mathbf{v} + \mathbf{b} \\ &= (h_{m+1,m}\mathbf{e}_m^T\mathbf{u}_m(t))\mathbf{v}_{m+1} \\ |\mathbf{r}_m(t)| &= |h_{m+1,m}\mathbf{e}_m^T\mathbf{u}_m(t)| \end{aligned} \tag{18}$$

This is the same expression as the residual of the method introduced in section 7.2.

7.4 The SAI method

The Arnoldi method uses a basis for $K_m(\mathbf{A}, \mathbf{v})$, a disadvantage here is that this subspace will approximate the eigenvectors with bigger eigenvalues better. This is contra productive because the smaller eigenvalues will be of more value in the approximation of $\mathbf{y}(t)$ as t gets larger. This is because every eigenvalue can be associated with the exponent of that eigenvalue times t in the matrix exponential, when the real part of an eigenvalue is larger than another it will approach zero faster. In this context large can be understood as a bigger negative value. When the Krylov method approximates these large eigenvalues better this is not efficient, the shift and invert (SAI) method is developed to counter this effect.
The SAI method uses the Arnoldi algorithm on the matrix $(\mathbf{I} - \gamma\mathbf{A})^{-1}$ instead of $\mathbf{A}$. This is the matrix $\mathbf{A}$ with a shift in the main diagonal and then inverted. Applying the Arnoldi algorithm on $K_m((\mathbf{I} - \gamma\mathbf{A})^{-1}, \mathbf{v})$ gives the next expression.

$$(\mathbf{I} - \gamma\mathbf{A})^{-1}\tilde{\mathbf{V}}_m = \tilde{\mathbf{V}}_m\tilde{\mathbf{H}}_{m,m} + \tilde{h}_{m+1,m}\tilde{\mathbf{v}}_{m+1}\mathbf{e}_m^T$$

This can be rewritten in the following way:

$$\begin{array}{lcll} \tilde{\mathbf{V}}_m\tilde{\mathbf{H}}_{m,m}^{-1} & = & (\mathbf{I}-\gamma\mathbf{A})\tilde{\mathbf{V}}_m & +(\mathbf{I}-\gamma\mathbf{A})\tilde{h}_{m+1,m}\tilde{\mathbf{v}}_{m+1}\mathbf{e}_m^T\tilde{\mathbf{H}}_{m,m}^{-1} \\ \tilde{\mathbf{V}}_m\frac{1}{\gamma}(\mathbf{I}-\tilde{\mathbf{H}}_{m,m}^{-1}) & = & \mathbf{A}\tilde{\mathbf{V}}_m & -\frac{1}{\gamma}(\mathbf{I}-\gamma\mathbf{A})\tilde{h}_{m+1,m}\tilde{\mathbf{v}}_{m+1}\mathbf{e}_m^T\tilde{\mathbf{H}}_{m,m}^{-1} \\ \mathbf{A}\tilde{\mathbf{V}}_m & = & \tilde{\mathbf{V}}_m\frac{1}{\gamma}(\mathbf{I}-\tilde{\mathbf{H}}_{m,m}^{-1}) & +\frac{1}{\gamma}(\mathbf{I}-\gamma\mathbf{A})\tilde{h}_{m+1,m}\tilde{\mathbf{v}}_{m+1}\mathbf{e}_m^T\tilde{\mathbf{H}}_{m,m}^{-1} \end{array}$$

We now define $\mathbf{H}_{m,m} := \frac{1}{\gamma}(\mathbf{I}-\tilde{\mathbf{H}}_{m,m}^{-1})$ and $\mathbf{V}_m := \tilde{\mathbf{V}}_m$.
With this $\mathbf{H}_{m,m}$ we can approximate the solution of $\mathbf{y}' = \mathbf{A}\mathbf{y}+\mathbf{b}$ similar to the last two chapters. There are some changes to the approximation procedure. The most important one is that the Arnoldi algorithm has to construct a basis for $K_m((\mathbf{I}-\gamma\mathbf{A})^{-1},\mathbf{v})$. In order to do this the product $\mathbf{w} = (\mathbf{I}-\gamma\mathbf{A})^{-1}\mathbf{v}_i$ has to be calculated m times. When the size of $\mathbf{A}$ is very large the calculation of the action of $(\mathbf{I}-\gamma\mathbf{A})^{-1}$ is a very demanding task. That is why solving these systems iteratively is a good option. For more details we refer to [2].
In addition, the expression of the residual in the SAI method changes slightly, this can be derived similar to and .

$$\begin{array}{lcl} \mathbf{r}_m(t) & = & \frac{1}{\gamma}(\mathbf{I}-\gamma\mathbf{A})\tilde{h}_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T\tilde{\mathbf{H}}_{m,m}^{-1}\mathbf{u}_m(t) \\ & = & \frac{\tilde{h}_{m+1,m}}{\gamma}(\mathbf{e}_m^T\tilde{\mathbf{H}}_{m,m}^{-1}\mathbf{u}_m(t))(\mathbf{I}-\gamma\mathbf{A})\mathbf{v}_{m+1} \end{array}$$

This expression is the same for the case of approximating $\mathbf{y}' = \mathbf{A}\mathbf{y}$ with the use of $e^{t\mathbf{H}_{m,m}}$ as the case of approximating the solution of $\mathbf{y}' = \mathbf{A}\mathbf{y}+\mathbf{b}$ with the use of $\phi_1(t\mathbf{H}_{m,m})$.

The SAI method and eigenvalues

The SAI method is designed for matrices with stiff spectrum, because the normal Krylov approximation is known to converge slowly for these types of matrices. A stiff spectrum means that the matrix has eigenvalues with both very small and large real part. The problem of a stiff spectrum is that the Krylov subspace will converge to the larger eigenvalues. These eigenvalues are of less importance when approximating a solution with the matrix exponential because they converge to zero faster than the smaller eigenvalues. A measurement for the spreading of the eigenvalues is:

$$\frac{\Re(\lambda_n)-\Re(\lambda_1)}{\Re(\lambda_2)-\Re(\lambda_1)}$$

Where the distinct real parts of the eigenvalues λ_i are sorted in increasing order. So this quantity gives the quotient between the largest gap between two eigenvalues and the gap between the smallest eigenvalues.
Now we will look what will happen to the eigenvalues of $\mathbf{A}$ if the SAI algorithm is applied. Suppose $\mathbf{y}_i$ is an eigenvector of $\mathbf{A}$ with corresponding eigenvalue λ_i, also suppose that the eigenvalues can be complex valued.

$$\begin{array}{lcl} \mathbf{A}\mathbf{y}_i & = & \lambda_i\mathbf{y}_i \\ \gamma\mathbf{A}\mathbf{y}_i & = & \gamma\lambda_i\mathbf{y}_i \\ (\mathbf{I}-\gamma\mathbf{A})\mathbf{y}_i & = & (1-\gamma\lambda_i)\mathbf{y}_i \\ (\mathbf{I}-\gamma\mathbf{A})^{-1}\mathbf{w}_i & = & \frac{1}{1-\gamma\lambda_i}\mathbf{w}_i \end{array}$$

In the last step the fact is used that if β is an eigenvalue of $\mathbf{B}$, then β^{-1} is an eigenvalue of $\mathbf{B}^{-1}$. Here $\mathbf{w}_i$ is an eigenvector corresponding to the eigenvalue $\frac{1}{1+\gamma\lambda_i}$. Also if $\mathbf{A}$ is nonsingular it has an eigendecomposition and only nonzero eigenvalues. Then $\mathbf{A}$ can be represented as $\mathbf{T}\mathbf{D}\mathbf{T}^{-1}$, with $\mathbf{T}$ the eigenbasis of $\mathbf{A}$ and $\mathbf{D}$ containing the corresponding eigenvalues. Then we have:

$$(\mathbf{I}-\gamma\mathbf{A})^{-1} = (\mathbf{T}(\mathbf{I}-\gamma\mathbf{D})\mathbf{T}^{-1})^{-1} = \mathbf{T}(\mathbf{I}-\gamma\mathbf{D})^{-1}\mathbf{T}^{-1}$$

So the vector $\mathbf{w}_i$ corresponds to column i of $\mathbf{T}$.
Seen the physical context of $\mathbf{A}$ in the Maxwell equations the $\mathbf{A}$ has to be negative semi definite,

therefore all the eigenvalues of $\mathbf{I} - \gamma\mathbf{A}$ are nonzero hence it is nonsingular.
When $\lambda_i = x + yi$ is an eigenvalue of $\mathbf{A}$ then the new eigenvalue becomes:

$$\frac{1 - \gamma x + \gamma y i}{(1 - \gamma x)^2 + (\gamma y)^2}$$

Both the real part and the absolute value of the imaginary part of the new eigenvalue lie between 0 and 1. So the spreading of the eigenvalues will probably be smaller and so the system will converge faster. For more information about the effect of this transformation on the eigenvalues we refer to [19].
For matrices without stiff spectrum the normal Krylov method can usually better be used. This is because the advantages of the faster converging can be countered by the additional computational work that is needed.

7.5 Practical issues when using Krylov subspace methods

In this section some issues for the practical use of the Krylov subspace methods introduced above will be discussed. A useful stopping criterion will be introduced, and methods for restarting the algorithm will be discussed for speeding up the algorithm.

Stopping criterion

One of the important issues when using Krylov subspace algorithms is how long you continue expanding the Krylov basis. If you continue expanding if you already can make a good approximation it is a waste of computational work. Conversely, if the dimension of the basis is too small the approximation is not good enough. So a good stopping criterion is needed. In this report we will use residual based stopping criterions. We have already derived the residual of both the normal Krylov subspace method and the SAI method. Before giving a stopping criterion based on the residual we will first explain how the residual is correlated to error. The error ϵ is the difference between the exact solution and the approximation, the residual is the error of the approximation or a measure of how good the approximation solves the original equation. So when the equation $\mathbf{y}' = \mathbf{A}\mathbf{y}$ is used, the error and residual are defined as follows:

$$\mathbf{e}_m(t) := \mathbf{y}(t) - \mathbf{y}_m(t), \quad \mathbf{r}_m(t) := \mathbf{A}\mathbf{y}_m(t) - \mathbf{y}'_m(t),$$

here m is the dimension of the Krylov subspace. Now the derivative of the equation of the error can be rewritten into a form that correlates it to the residual.

$$\begin{aligned} \mathbf{e}'_m(t) &= \mathbf{y}'(t) - \mathbf{y}'_m(t) \\ &= \mathbf{A}\mathbf{y}(t) + \mathbf{A}\mathbf{y}_m(t) - \mathbf{A}\mathbf{y}_m(t) - \mathbf{y}'_m(t) \\ &= \mathbf{A}(\mathbf{y}(t) - \mathbf{y}_m(t)) + \mathbf{r}_m(t) \\ &= \mathbf{A}\mathbf{e}_m(t) + \mathbf{r}_m(t) \end{aligned}$$

So the error is an inhomogeneous differential equation in $\mathbf{A}$ so evaluating this term is even more difficult then estimating $\mathbf{y}(t)$. At $t = 0$ we know the exact solution so the residual and the error are zero at that point. So if the inhomogeneous term $\mathbf{r}_m(t)$ is small the error will be small too. [3] gives a lot of information on the estimation of this error term.
Because the residual is closely related to the error it can be used as a stopping criterion for the estimation of the matrix exponential. The following measure will be used as a stopping criterion while expanding the Krylov basis:

$$\frac{\|\mathbf{r}_m(t)\|}{\|\mathbf{v}\|} = \frac{\|\mathbf{r}_m(t)\|}{\beta}$$

At every step of the Arnoldi algorithm this measure, the relative residual, can be evaluated. Because the residual can be very small at a certain time, the L_2 norm can be estimated over several time steps in the interval $(0, t]$.

Choosing the proper t value

Assume we have to find $\mathbf{y}(t)$ of (8) for $t \in [0, T]$. Then one can plot the CPU time needed to approximate $\mathbf{y}(t)$ against t.
In this plot one would see a convex shaped function. So taking a large step will take more CPU time then taking two smaller steps to arrive at the same t. This suggests it may be efficient to restart the Krylov algorithms to reduce the CPU time. Restarting is the process of first estimating $\mathbf{y}(t_1) = e^{\mathbf{A}t_1}v$ and then using $\mathbf{y}(t_1)$ to approximate the next time step to t_2 with $\mathbf{y}(t_2) = e^{\mathbf{A}(t_2-t_1)}\mathbf{y}(t_1)$ using $K_m(\mathbf{A}, \mathbf{y}(t_1))$. This can be done for finitely many time steps. If the step size is chosen in an efficient way this process can be used to speed up the estimation of the matrix exponent using Krylov subspace methods.
When one calculates the fraction of CPU time and t one can choose the minimum value of this fraction. The t value corresponding to this minimum will use the least CPU time compared to the simulation step it takes, this is an optimal t value. This optimal t value can be used to simulate (8) over larger time gaps by taking time steps.

Complex eigenvalues

If σ and σ_m as introduced in section 5 are equal to zero, the matrix $\mathbf{A}$ is equal to a diagonal matrix times a skew-symmetric matrix. A skew-symmetric matrix has purely imaginary eigenvalues. $\mathbf{A}$ is similar to this skew-symmetric matrix and so has the same eigenvalues, thus $\mathbf{A}$ has purely imaginary eigenvalues. Also if σ and σ_m are nonzero the system will still have complex eigenvalues but this time the oscillations, that result from the imaginary part, will be damped.
Complex eigenvalues always come in pairs with their conjugates. So when the matrix $\mathbf{H}$ is calculated with an odd dimension one of the eigenvalues of $\mathbf{H}$ has to be a wrong representation of the eigenvalues of $\mathbf{A}$. Therefore it is a good choice to consider only the even Krylov dimensions. This will save the CPU time of evaluating the relative residual for the odd dimensions.

7.6 Solving ODEs using the Krylov subspace

The Krylov subspace can also be used for solving ODEs of the following form:

$$\mathbf{y}' = \mathbf{A}\mathbf{y} + \alpha(t)\mathbf{g}, \quad \mathbf{y}(0) = 0$$

The vector $\mathbf{g}$ is a constant here. The method described below is introduced in [1]. This method is based on reducing the error of an approximation $\mathbf{y}_k(t)$ to $\mathbf{y}(t)$. This residual is refined by solving small systems of ODEs generated by the Arnoldi algorithm. The residual $\mathbf{r}_k(t)$ is defined as:

$$\mathbf{r}_k(t) = \mathbf{A}\mathbf{y}_k + \alpha(t)\mathbf{g} - \mathbf{y}_k'$$

We claim that $\mathbf{r}_k(t) = \rho(t)\bar{\mathbf{r}}_k$ with $\bar{\mathbf{r}}_k$ a constant vector. This will be shown later. Let $\mathbf{y}_0 = 0$ then $\mathbf{r}_0(t) = \alpha(t)\mathbf{g}$.
Now use the Arnoldi algorithm on the space $K_m(\mathbf{A}, \mathbf{r}_k)$ to get:

$$\mathbf{A}\mathbf{V}_m = \mathbf{V}_m\mathbf{H}_{m,m} + h_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T$$

Now let $\mathbf{e}_k(t) = \mathbf{y}(t) - \mathbf{y}_k(t)$, then solving the next system will give the exact solution of $\mathbf{e}_k(t)$:

$$\mathbf{e}_k'(t) = \mathbf{A}\mathbf{e}_k(t) + \rho_k(t)\bar{\mathbf{r}}_k \quad \mathbf{e}(0) = 0 \tag{19}$$

A derivation of the ODE with a similar error term is given in chapter 7.5. We want to estimate $\mathbf{e}_k(t)$ by $\tilde{\mathbf{e}}_k(t) = \mathbf{V}_m\mathbf{u}(t)$. Putting this into equation (19) we get:

$$\begin{array}{lcl} \mathbf{V}_m\mathbf{u}'(t) & = & \mathbf{A}\mathbf{V}_m\mathbf{u}(t) + \rho_k(t)\bar{\mathbf{r}}_k \\ \mathbf{V}_m^T\mathbf{V}_m\mathbf{u}'(t) & = & \mathbf{V}_m^T((\mathbf{V}_m\mathbf{H}_{m,m} + h_{m+1,m}\mathbf{v}_{m+1}\mathbf{e}_m^T)\mathbf{u}(t) + \rho_k(t)\bar{\mathbf{r}}_k) \\ \mathbf{u}'(t) & = & \mathbf{H}_{m,m}\mathbf{u}(t) + \mathbf{e}_1\rho_k(t) \end{array}$$

This small system is easily solvable with a standard ODE solver. Now we get that $\mathbf{y}_{k+m} = \mathbf{y}_k + \mathbf{V}_m\mathbf{u}(t)$ and can derive the next residual term $\mathbf{r}_{k+m}(t)$.

$$\begin{aligned}
\mathbf{r}_{k+m}(t) &= \mathbf{A}\mathbf{y}_{k+m} + \alpha(t)\mathbf{g} - \mathbf{y}'_{k+m} \\
&= \mathbf{A}\mathbf{y}_k + \mathbf{A}\mathbf{V}_m\mathbf{u}(t) + \alpha(t)\mathbf{g} - \mathbf{y}'_k - \mathbf{V}_m\mathbf{u}'(t) \\
&= \rho_k(t)\bar{\mathbf{r}}_k + \mathbf{A}\mathbf{V}_m\mathbf{u}(t) - \mathbf{V}_m(\mathbf{H}_{m,m}\mathbf{u}(t) + \rho_k(t)\mathbf{e}_1) \\
&= (h_{m+1,m}\mathbf{e}_m^T\mathbf{u}(t))\mathbf{v}_{m+1}
\end{aligned}$$

So $\bar{\mathbf{r}}_{k+m} = \mathbf{v}_{m+1}$ and $\rho_{k+m}(t) = h_{m+1,m}\mathbf{e}_m^T\mathbf{u}(t)$.
This procedure can also be applied to $K_m((\mathbf{I} - \gamma\mathbf{A})^{-1}, \bar{\mathbf{r}}_k)$ in this case the residual term would change slightly into:

$$\mathbf{r}_{k+m}(t) = (\frac{\tilde{h}_{m+1,m}}{\gamma}\mathbf{e}_m^T\tilde{\mathbf{H}}_{m,m}^{-1}\mathbf{u}(t))(\mathbf{I} - \gamma\mathbf{A})\mathbf{v}_{m+1}$$

The procedure can be repeated until the residual is small enough.

8 Perfectly matched layer boundaries

In this section the perfectly matched layer boundary is introduced. Recall that in section 5 the boundary nodes are set to zero. This models a perfect electric conductor, in which $\mathbf{E} = \mathbf{0}$ [17, pp. 25] and which perfectly reflects incident electromagnetic waves [8, pp. 396]. This is inappropriate for domains which model a region of free space or a chunk of a photonic crystal. The most advanced method for modeling the suitable absorbing boundaries is the perfectly matched layer [17, pp. 305]. It separates the domain and the PEC boundary by a conducting material which attenuates the reflected electromagnetic waves [8, pp. 392]. Jumps or gradients in conductivity usually result in some reflection [8, pp. 396] but the perfectly matched layer is constructed so that this does not occur [7].
Perfectly matched layers can be seen as an analytic continuation [14]. There are several formulations of perfectly matched layers of which the uniaxial anisoptropic medium, introduced in [7], is used in this report. Its name derives from the anisotropic (tensor valued) permittivities and permeabilities that are used in the layer. These make it necessary to use an alternative set of ODEs. The permittivities and permeabilities are set so that analytically there is no reflection. However the discretization does result in some reflection. This is minimized by ramping up the conductivity gradually.
We shall now describe the alternative set of ODEs. The boundary is divided into two regions, as depicted in figure 4, one of which has a surface perpendicular to the x-axis and the other to y-axis. We use the approach of [7] to describe these alternative ODEs, which are most succinctly

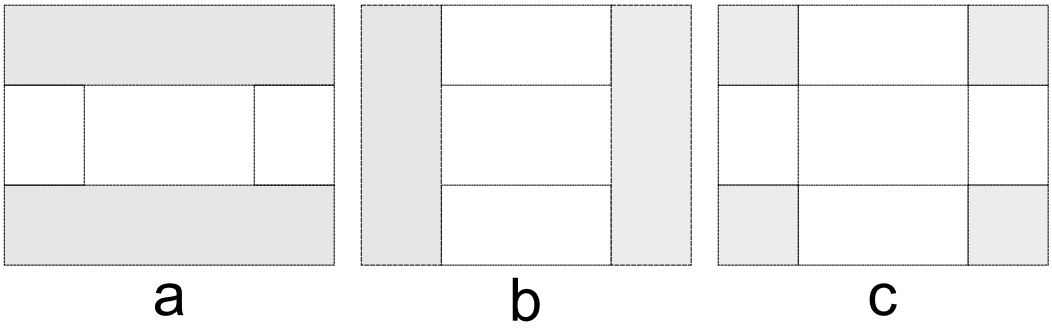

Figure 4: this figure shows three regions of the layer. a depicts the y-boundary, b the x-boundary and c the cornerpoints, the regions where the x- and y-boundary overlap.

expressed in the frequency domain. We will first show an example of how the time domain and frequency domain descriptions of Maxwell's equations are related. We start off with Ampere's equation in the absense of source currents

$$\frac{\partial E_z}{\partial t} = \frac{1}{\epsilon_r}[\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - \sigma E_z]$$

Next we assume that E_z can be written as the real part of $E_z(x,y)e^{i\omega t}$ which results in

$$\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} = i\omega\epsilon_r(1 + \frac{\sigma}{i\omega\epsilon_r})E_z$$

According to [7] no reflection occurs at the interface between the domain and the PML layer if the three equations of the TM_z mode are given in the PML layer by

$$\begin{bmatrix} -\frac{\partial E_z}{\partial y} \\ \frac{\partial E_z}{\partial x} \\ \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{s_y}{s_x} & & \\ & \frac{s_x}{s_y} & \\ & & s_x s_y \end{bmatrix} \begin{bmatrix} i\omega\mu_r H_x \\ i\omega\mu_r H_y \\ i\omega\epsilon_r E_z \end{bmatrix}$$

Here s_x is associated with the x-boundary and s_y with the y-boundary as depicted in figure 4 and they assume the following values

$$s_x = \begin{cases} 1 + \frac{\sigma_x}{\mu_r i\omega} & \text{if } (x,y) \in \text{x-boundary} \\ 1 & \text{if } (x,y) \notin \text{x-boundary} \end{cases} \qquad s_y = \begin{cases} 1 + \frac{\sigma_y}{\mu_r i\omega} & \text{if } (x,y) \in \text{y-boundary} \\ 1 & \text{if } (x,y) \notin \text{y-boundary} \end{cases}$$

σ_x and σ_y are associated with their boundaries and must be of the form that they are constant along the y or x direction respectively. We furthermore have made the choice to set $\mu_r = \epsilon_r = 1$. This is will not influence the test domain since the boundary is reflectionless. The equations are physical in the sense that they satisfy Maxwell's equations but a material with these constitutive parameters has not been constructed yet [7]. In the areas where s_x or s_y are 1, namely the areas outside the x-boundary, y-boundary and cornerpoints for H_x, H_y and E_z respectively, the equations have the same form as in section 2. At all other points the equations have a more complicated structure. It is hard to transform them directly into the time domain. Therefore the dummy variables G_x, G_y and D_z are introduced. They are defined as follows

$$G_x = \frac{s_y}{s_x} H_x \qquad G_y = \frac{s_x}{s_y} H_y \qquad D_z = s_x E_z.$$

The Maxwell equations become

$$-\frac{\partial E_z}{\partial y} = \mu_r \frac{\partial G_x}{\partial t} \qquad \frac{\partial E_z}{\partial x} = \mu_r \frac{\partial G_y}{\partial t} \qquad \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} = \epsilon_r \frac{\partial D_z}{\partial t} + \sigma_y D_z \tag{20}$$

and we obtain the following set of auxiliary differential equations

$$\mu_r \frac{\partial G_x}{\partial t} + \sigma_x G_x = \mu_r \frac{\partial H_x}{\partial t} + \sigma_y H_x \tag{21a}$$

$$\mu_r \frac{\partial G_y}{\partial t} + \sigma_y G_y = \mu_r \frac{\partial H_y}{\partial t} + \sigma_x H_y \tag{21b}$$

$$\epsilon_r \frac{\partial D_z}{\partial t} = \epsilon_r \frac{\partial E_z}{\partial t} + \sigma_x E_z \tag{21c}$$

8.1 Matrix description of alternative set of ODEs

In this section the matrix description of the spatially discretized Maxwell equations in the PML layer is given. We set $\epsilon_r = 1$ and $\mu_r = 1$. The spatially discretized Maxwell equations (20) are given by

$$\frac{\partial \mathbf{g}_x}{\partial t} = \mathbf{K}\mathbf{e}_z \tag{22a}$$

$$\frac{\partial \mathbf{g}_y}{\partial t} = \mathbf{L}\mathbf{e}_z \tag{22b}$$

$$\frac{\partial \mathbf{d}_z}{\partial t} = [-\mathbf{K}^\intercal \quad -\mathbf{L}^\intercal]\mathbf{h} - \mathbf{\Sigma}_z^y \mathbf{d}_z \tag{22c}$$

and the spatially discretized auxiliary differential equations (21) are given by

$$\begin{aligned} \frac{\partial \mathbf{h}_x}{\partial t} &= \frac{\partial \mathbf{g}_x}{\partial t} + \mathbf{\Sigma}_x^x \mathbf{g}_x - \mathbf{\Sigma}_x^y \mathbf{h}_x \\ &= \mathbf{K}\mathbf{e}_z + \mathbf{\Sigma}_x^x \mathbf{g}_x - \mathbf{\Sigma}_x^y \mathbf{h}_x \end{aligned} \tag{23a}$$

$$\begin{aligned} \frac{\partial \mathbf{h}_y}{\partial t} &= \frac{\partial \mathbf{g}_y}{\partial t} + \mathbf{\Sigma}_y^y \mathbf{g}_y - \mathbf{\Sigma}_y^x \mathbf{h}_y \\ &= \mathbf{L}\mathbf{e}_z + \mathbf{\Sigma}_y^y \mathbf{g}_x - \mathbf{\Sigma}_y^x \mathbf{h}_y \end{aligned} \tag{23b}$$

$$\begin{aligned} \frac{\partial \mathbf{e}_z}{\partial t} &= \frac{\partial \mathbf{d}_z}{\partial t} - \mathbf{\Sigma}_z^x \mathbf{e}_z \\ &= [-\mathbf{K}^\intercal \quad -\mathbf{L}^\intercal]\mathbf{h} - \mathbf{\Sigma}_z^y \mathbf{d}_z - \mathbf{\Sigma}_z^x \mathbf{e}_z. \end{aligned} \tag{23c}$$

Here $\mathbf{\Sigma}_j^i$ is the diagonal matrix of σ_i's defined on the G_{H_x}, G_{H_y} and G_{E_z} grids for $j = x, y, z$. The next step is to introduce a new $\mathbf{y}$

$$\mathbf{y} = \begin{bmatrix} \mathbf{h}_x & \mathbf{h}_y & \mathbf{e}_z & \mathbf{g}_x & \mathbf{g}_y & \mathbf{d}_z \end{bmatrix}^\top$$

The matrix for $\mathbf{y}'$ can be constructed using (22) and (23):

$$\mathbf{y}' = \begin{bmatrix} -\mathbf{\Sigma}_x^y & & \mathbf{K} & \mathbf{\Sigma}_x^x & & \\ & -\mathbf{\Sigma}_y^x & \mathbf{L} & & \mathbf{\Sigma}_y^y & \\ -\mathbf{K}^\intercal & -\mathbf{L}^\intercal & -\mathbf{\Sigma}_z^x & & & -\mathbf{\Sigma}_z^y \\ & & \mathbf{K} & & & \\ & & \mathbf{L} & & & \\ -\mathbf{K}^\intercal & -\mathbf{L}^\intercal & & & & -\mathbf{\Sigma}_z^y \end{bmatrix} \mathbf{y}$$

This equation holds for all grids, but the size of the matrix can be significantly reduced if the variables $\mathbf{g}_x$, $\mathbf{g}_y$ and $\mathbf{d}_z$ are only defined for the points on the x-boundary, y-boundary and cornerpoints respectively. The other nodes are updated using the standard Maxwell equations.

8.2 PML for FDTD

Discretizing both the spatial and temporal derivatives in (20), using notation similar to section 6, we obtain the following difference equations

$$\mathbf{g}_x(t_{i+\frac{1}{2}}) = \mathbf{g}_x(t_{i-\frac{1}{2}}) + \Delta t \mathbf{K} \mathbf{e}_z(t_i)$$

$$\mathbf{g}_y(t_{i+\frac{1}{2}}) = \mathbf{g}_y(t_{i-\frac{1}{2}}) + \Delta t \mathbf{L} \mathbf{e}_z(t_i)$$

$$\mathbf{d}_z(t_{i+1}) = (2 + \Delta t \mathbf{\Sigma}_z^y)^{-1}(2 - \Delta t \mathbf{\Sigma}_z^y)\mathbf{d}_z(t_i) + 2\Delta t(2 + \Delta t \mathbf{\Sigma}_z^y)^{-1}[-\mathbf{K}^\intercal \quad -\mathbf{L}^\intercal]\mathbf{h}(t_{i+\frac{1}{2}})$$

and discretizing (21) results in

$$\begin{aligned} \mathbf{h}_x(t_{i+\frac{1}{2}}) &= (2 + \Delta t \mathbf{\Sigma}_x^y)^{-1}(2 - \Delta t \mathbf{\Sigma}_x^y)\mathbf{h}_x(t_{i-\frac{1}{2}}) \\ &\quad + (2 + \Delta t \mathbf{\Sigma}_x^y)^{-1}[(2 + \Delta t \mathbf{\Sigma}_x^x)\mathbf{g}_x(t_{i+\frac{1}{2}}) - (2 - \Delta t \mathbf{\Sigma}_x^x)\mathbf{g}_x(t_{i-\frac{1}{2}})] \end{aligned}$$

$$\begin{aligned} \mathbf{h}_y(t_{i+\frac{1}{2}}) &= (2 + \Delta t \mathbf{\Sigma}_y^x)^{-1}(2 - \Delta t \mathbf{\Sigma}_y^x)\mathbf{h}_y(t_{i-\frac{1}{2}}) \\ &\quad + (2 + \Delta t \mathbf{\Sigma}_y^x)^{-1}[(2 + \Delta t \mathbf{\Sigma}_y^y)\mathbf{g}_y(t_{i+\frac{1}{2}}) - (2 - \Delta t \mathbf{\Sigma}_y^y)\mathbf{g}_y(t_{i-\frac{1}{2}})] \end{aligned}$$

$$\mathbf{e}_z(t_{i+1}) = (2 + \Delta t \mathbf{\Sigma}_z^x)^{-1}(2 - \Delta t \mathbf{\Sigma}_z^x)\mathbf{e}_z(t_i) + 2(2 + \Delta t \mathbf{\Sigma}_z^x)^{-1}[\mathbf{d}_z(t_{i+1}) - \mathbf{d}_z(t_i)].$$

Again we have set $\epsilon_r = \mu_r = 1$.

9 Verification of FDTD

To verify our model we use source currents to determine an analytical solution as described in appendix 2. We use two different tests: First we determine the order of convergence for both spatial and temporal refinement. Next we calculate the order of convergence for only temporal refinement and calculate both the order of convergence as well as the relative error. Note that for the temporal order of convergence we cannot use the analytical solution as a reference since the total error will still converge towards the spatial error.
Both tests have been done on a 1×1 grid. We have used a solution that is a superposition of two spatial patterns $f(x,y) = x(x-1)y(y-1)$ and $g(x,y) = \sin(\pi x)\sin(\pi y)$. Corresponding to the former spatial pattern we use the functions $\alpha_x(t) = e^{-t}$,$\alpha_y(t) = \frac{1}{1+t}$ and $\alpha_z(t) = \sin(t)$ for H_x, H_y and E_z respectively. For the latter we use $\beta_x(t) = \frac{1}{2}$,$\beta_y(t) = \ln(t+1)$ and $\beta_z(t) = \cos(t)$. Thus we have an analytical solution of the form:

$$H_x(x,y,t) = \sin(\pi x)\cos(\pi y)e^{-t} + x(x-1)(2y-1)\frac{1}{2}$$

$$H_y(x,y,t) = \cos(\pi x)\sin(\pi y)\frac{1}{1+t} + (2x-1)y(y-1)\ln(t+1)$$

$$E_z(x,y,t) = \sin(\pi x)\sin(\pi y)\sin(t) + x(x-1)y(y-1)\cos(t)$$

The tests are done for $\epsilon_r = 2$ and $\mu_r = 3$.
For the first test we have used $\Delta t = \frac{\Delta x}{\sqrt{2}}$. For a number of nodes N per side this results in spatial step sizes given by $\Delta x = \Delta y = \frac{1}{N+1}$. We then compared the solutions for different step sizes at time $t = \frac{20}{201\sqrt{2}}$. As can be seen in figure 5 we get a second order of convergence towards the analytical solution.

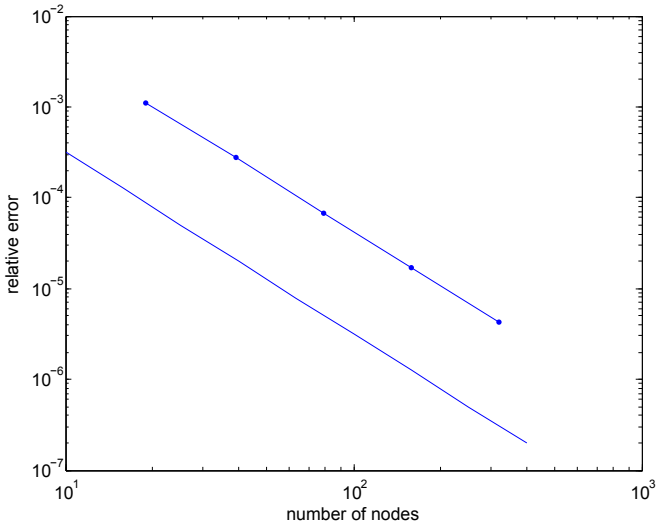

Figure 5: Plot of number of nodes along one side of the grid vs. relative error. As a reference we included a graph of the form $a10^{-2N}$. Since the plot of relative errors is parallel to the reference line we conclude that there is second order convergence

For the second test we have used 100×100 nodes. As can be seen in 6 we have a second order convergence.

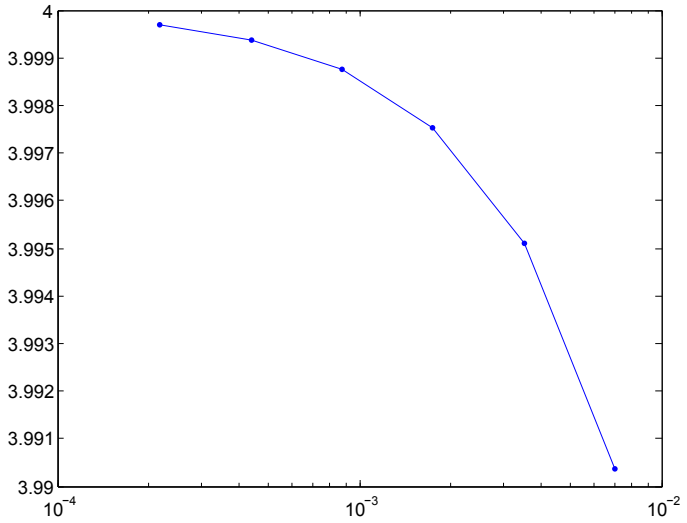

Figure 6: Plot of convergence factor vs. time step. The convergence factor approaches 4 thus indicating that we have a second order convergence

The relative error in respect to the analytical solution is given by the following values:

k	0	1	2	3	4	5	6	7
rel. error ($\times 10^{-5}$)	0.8419	0.7798	0.7643	0.7604	0.7594	0.7592	0.7591	0.7591

Here Δt is given by $\frac{1}{2^k 101\sqrt{2}}$. From this we conclude that the error created by spatial discretization is dominant. This justifies the choice of as large as possible time steps.
From now on we will assume the correctness of the FDTD method and now we can construct accurate reference solutions for testing the Krylov methods.

method	Δt	$\frac{\|\mathbf{e}_{ref}-\mathbf{e}\|}{\|\mathbf{e}_{ref}\|}$	$\frac{\|\mathbf{h}_{ref}-\mathbf{h}\|}{\|\mathbf{h}_{ref}\|}$	CPU (s)
FDTD	1.25×10^{-4}	3.22×10^{-4}	1.86×10^{-4}	76
	6.25×10^{-5}	7.97×10^{-5}	4.61×10^{-5}	152
	3.13×10^{-5}	2.00×10^{-5}	1.15×10^{-5}	304
	1.56×10^{-5}	5.00×10^{-6}	2.90×10^{-6}	610
Krylov		3.03×10^{-7}		230

Table 1: Results for point defect on 275×275 grid.

10 Results

In this section we present the results of simulations of the test cases. The test cases are described in section (4). The implementation of PML for Krylov still lacks second order convergence. Therefore we have not included numerical test results for the FDTD and Krylov PML implementation.
Recall that for Krylov to work properly the functions that describe the sources have to be of the form $J(x, y, t) = f(t)g(x, y)$, see section (7.6).

Rectangular lattice with a point defect

This test case consists of a square lattice with 7×7 dielectric rods. Recall from section (3) that when transforming the Maxwell equations into dimensionless form a typical length L needs to be chosen to which the other variables are scaled. We set L equal to the length of the physical domain. So the dimensionless domain has length 1. N_x denotes the number of grid points along the x-axis. The resulting spatial step size is given by $\Delta x = \frac{1}{Nx+1}$.
The lattice constant [13, pp. 32] is given by $a = 25$ nodes. Each 'box' contains a dielectric rod with radius $r = 0.38a\Delta x$ and with permittivity $\epsilon_r = 8.9$. Furthermore we include a boundary of $b = 50$ nodes on all sides. The resulting grid size is therefore 275×275 nodes.
The defect consists of the middle rod which has the same radius as the other rods, but a different relative permability $\epsilon_r = 6.052$. The regions in between the rods have the vacuum permittivity ϵ_0.
The system is initiated in the middle rod by an additive dimensionless source given by spatial component $g(x, y) = e^{(-2000((x/L-0.5)^2+(y/L-0.5)^2))}$ and time component $f(t) = 400\cos(\omega t)$. Note that the source has a spatial component because we have decided to make it a sharply peaked function instead of true point source in order to make the numerical discretization of the source more smooth. The scalar in front of the cosine has been chosen in such a way that the resulting fields will not exceed a value of 1, which is possible since the specific magnetic field strength H_0 can be chosen arbitrarily, see section (3). This could be demonstrated by transforming the dimensionless fields back into the physical fields, however, we do not show it in this report.
To get a frequency inside of the band gap of the rectangular lattice we choose a frequency inside the band gaps determined in [13, pp. 68]: $\omega_s \approx \frac{0.42 \cdot 2\pi c_0}{a_s}$. Here a_s denotes the lattice constant in SI units. Using the methods of section (3) this frequency can be transformed into dimensionless form. Combining $a_s = \frac{a}{L} = \frac{a}{N_x+1}$, (4) and ω_s yields

$$\omega = \frac{2\pi 0.42(N_x + 1)}{\hat{a}}.$$

Note that $\hat{a}$ is the length of a side of a 'box' in grid points, not its dimensionless length. A rough schematic of this scenario is included below in figure 7.
The Krylov subspace methods and FDTD-scheme are compared at time $t = 0.80$. A reference solution is calculated using the FDTD scheme with second order convergence and with $\Delta t \approx 6.67 \times 10^{-6}$, the results are listed in table 1. The data in the last two rows of the FDTD results have been calculated directly based on the assumption of second order convergence and does not come from a simulation. This has been done because the Krylov time integration already turned out to be faster and more accurate at that point. $\mathbf{e}_{\text{ref}}$ and $\mathbf{e}$ are respectively the reference solution for the electric field and the computed electric field, $\mathbf{h}_{\text{ref}}$ and $\mathbf{h}$ are the same for the

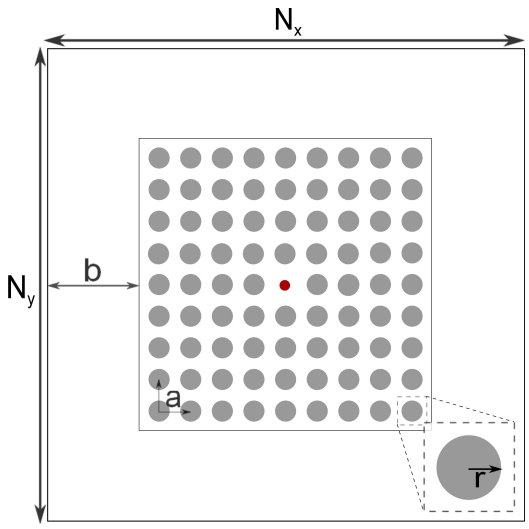

Figure 7: square lattice with point defect. The grey circles indicate rods of dielectric material, with $\epsilon_r \neq 1$. The white regions have the vacuum permittivity ϵ_0. N_x and N_y are the number of grid points along the x-axis and y-axis respectively. b is the width of the boundary layer. a is the distance between the centers of adjacent rods. r is the radius of a rod. The point defect consists of a missing rod at the center of the lattice. The source is intialized at the center of the lattice.

method	Δt	$\frac{\|\mathbf{e}_{ref}-\mathbf{e}\|}{\|\mathbf{e}_{ref}\|}$	$\frac{\|\mathbf{h}_{ref}-\mathbf{h}\|}{\|\mathbf{h}_{ref}\|}$	CPU (s)
FDTD	2.50×10^{-4}	9.08×10^{-4}	5.74×10^{-4}	383
	1.25×10^{-4}	2.29×10^{-4}	1.48×10^{-4}	784
	6.25×10^{-5}	5.65×10^{-5}	3.68×10^{-5}	1558
Krylov		2.33×10^{-7}		1340

Table 2: Results for point defect on 550×550 grid.

magnetic field. The relative error of the Krylov solution is calculated over both the electrical and the magnetic field. For the FDTD method this has been done for the electric and magnetic fields separately. The resulting relative errors cannot be added to obtain the relative error over $\mathbf{y}$. However, this is an upper bound. Of course each of the relative errors over $\mathbf{e}$ and $\mathbf{h}$ is a lower bound for the relative error over $\mathbf{y}$. The Krylov solution is calculated over 10 timesteps with $\Delta t = 0.08$. The method used is slightly more general than the one used in section (7.6). It is described in [1]. In this case the vector values of the function $\mathbf{g}(t)$ belong to a two-dimensional subspace in $\mathbb{R}^{n_{h_x}+n_{h_y}+n_{e_z}}$. The estimation is restarted every 20 Krylov subspace dimensions, and the tolerance is set to 10^{-8}.
The same test is done for the same grid structure but with $a = 50$, $b = 100$ and a grid size of 550×550. The reference solution is computed with the same time step $\Delta t \approx 6.67 \times 10^{-6}$. The results are listed in table 2. Here the Krylov solution is calculated with the same parameters as described above.

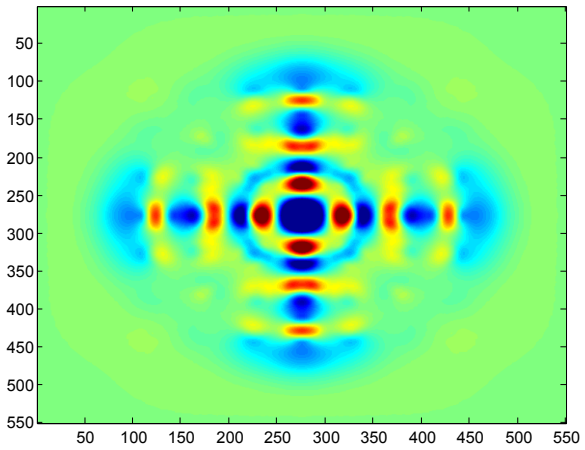

Figure 8: The resulting field pattern of the electric field in the first scenario. Positive values are red, negative values are depicted as blue. We see an evanescent [13, pp ./52] wave, however the system has not reached a steady state yet.

Line defect

For our line defect we have used two sets of square lattices with 15×9 dielectric rods. Recall from section (5) that this means that there are 9 rows along the y-axis and 15 columns along the x-axis. However the 5th of the 9 lines is missing. This means that we are modeling a line defect. The specific length L has again been chosen to be equal to the length of the physical domain along the x-axis. This means that the dimensionless length of the domain is given by $L := 1$. The lattice constant is given by $a = 20$ nodes, the radius is given by $r = 0.2a\Delta x$ and $\epsilon_r = 8.9$. Around the domain a boundary with a width of a single node is defined.
This results in a grid size of 302×182 nodes. We initiate the system with the similar pulse as in the previous test configuration with $g(x, y) = e^{(-((x-0.5)^2+(y\cdot\frac{L_x}{L_y}-0.5)^2)/10)}$. Here L_x denotes the physical length in x-direction and L_y the same in the y-direction. The time component is given by $f(t) = 400 \cdot \cos(\frac{2\pi 0.37(N_x+1)}{a}t)$ using a bandgap frequency determined by [13, pp. 68]. A rought schematic of this scenario is included below in figure 9. Snapshots are taken at time $t = \frac{225}{303\sqrt{2}}$, a reference solution is calculated using FDTD with $\Delta t = 1.139 \times 10^{-6}$. The results are listed in table 3. Here the Krylov algorithm is carried out in 8 timesteps and restarted every 20 dimensions. The first result of the Krylov solution is with a tolerance of 10^{-8}, the second one with a tolerance of 1.

method	Δt	$\frac{\|\mathbf{e}_{ref}-\mathbf{e}\|}{\|\mathbf{e}_{ref}\|}$	CPU (s)
FDTD	5.83×10^{-4}	3.10×10^{-3}	21
	2.92×10^{-4}	7.70×10^{-4}	42
	1.46×10^{-4}	1.92×10^{-4}	84
	7.29×10^{-5}	4.81×10^{-5}	168
	3.64×10^{-5}	1.20×10^{-5}	336
Krylov		3.24×10^{-8}	253
		1.63×10^{-5}	233

Table 3: results for line defect on 302 grid

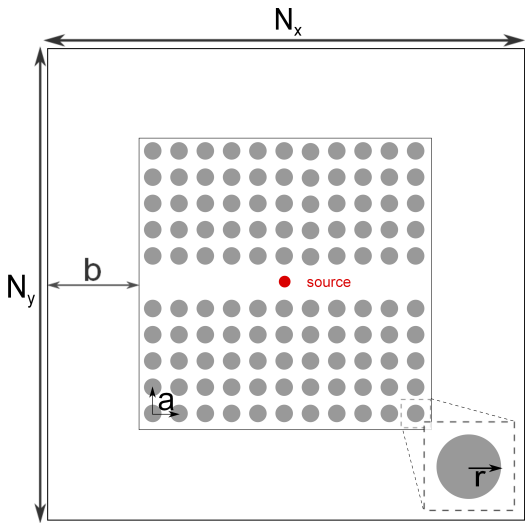

Figure 9: square lattice with a line defect. The grey circles denote rods of dielectric material, with $\epsilon_r \neq 1$. The white regions have the vacuum permittivity ϵ_0. N_x and N_y are the number of grid points along the x-axis and y-axis respectively. b is the width of the boundary layer. a is the distance between the centers of adjacent rods. r is the radius of the rod. The line defect consists of a missing line of rods. The source is intialized along a section of the line defect.

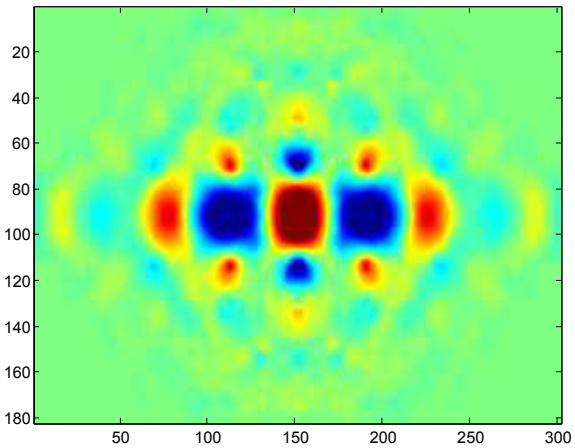

Figure 10: The resulting electric field of the line defect in a square lattice. We can see that the waves propagate only in the waveguide. In the crystal they decay exponentially.

Other scenarios

The other two settings have been included without further analysis and technical discription, thus only with visual results.
Both settings have been calculated using a sine wave with a frequency in the band gap. Also, in both cases a PML boundary has been included, which allows for longer simulations.
Figure 11 shows the result of a waveguide bend. Figure 12 depicts the triangular lattice with a line defect.

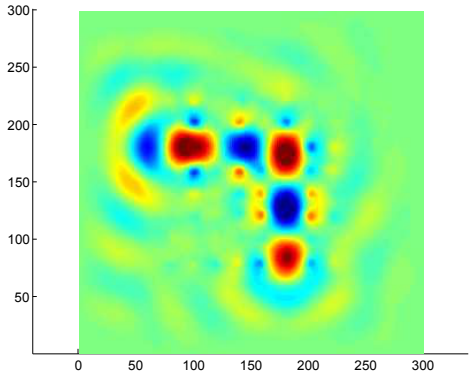

Figure 11: The resulting electric field of a waveguide bend. It can be seen that the transmission in the waveguide is almost not hindered by the bend.

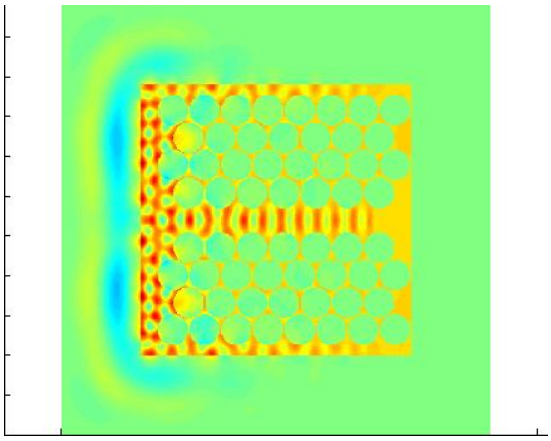

Figure 12: Electric field pattern of the triangular lattice with a line defect. The dielectric material is shown in this picture in orange. Note that the waves propagate only in the line defect and around the crystal.

11 Conclusions

The purpose of this research is to compare Krylov subspace methods and the FDTD scheme for time-domain simulations of photonic crystals. In order to do this we have discretized Maxwell's equations using the Yee grid to get a spatial discretization of the form:

$$\begin{bmatrix} \mathbf{h}'(t) \\ \mathbf{e}'(t) \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{h}(t) \\ \mathbf{e}(t) \end{bmatrix} + \alpha(t)\mathbf{g},$$

where the vector $\mathbf{h}$ denotes the magnetic field and the vector $\mathbf{e}$ denotes the electric field.
We have compared the time integration of this equation for a few different scenarios using both the FDTD scheme and Krylov subspace methods.
In tables 1, 2 and 3 you can see the results of FDTD and Krylov simulations of a few different scenarios and for a number of grid sizes. It is clear from this data that Krylov subspace methods are way more efficient than FDTD if a high accuracy is needed. In table 3 it can be seen that FDTD can be faster when there are no high demands on the accuracy. We know from section 9 that in the case of a Yee grid space discretization the error is dominated by the spatial error. In simulations based on the Yee grid there is therefore no real need for high accuracy in time integration. Krylov subspace methods allow one for a slightly higher time discretization error to perform the time integration much faster.
One of the disadvantages of Krylov subspace methods in comparison to the FDTD scheme is that they need some tuning before working properly. There are several parameters that can be chosen like the restarting time and restarting dimension. These are parameters that can greatly influence the simulation speed and, although there are systematic methods to choose their proper values, this requires additional effort. In comparison the only input parameter that the FDTD scheme requires is Δt, and if it is chosen small enough then the FDTD method is guaranteed to work.
On the other hand, a drawback of the FDTD scheme is that it does not provide an error estimate, i.e. several simulations with different time step sizes are needed to estimate the error of the time discretization. On the other hand, the Krylov methods do provide the residual value. The residual can be seen as a backward error and a reliable estimate of the true unknown error [3]. Apart from the standard Yee scheme with reflecting boundary conditions we have also implemented a perfectly matched layer absorbing boundary condition. It was implemented correctly with second order convergence for the FDTD scheme, however time limitations of this project did not allow us to get it working properly for the Krylov subspace methods. This could be a subject for further research.

1 Appendix: detailed space discretization of Maxwell's equations and derivation of the matrix formulation of the resulting system

In this section a detailed description is given of the space discretization of the Maxwell equations. Furthermore the matrix description of the space-discretized Maxwell equations (5) is derived.

1.1 Space discretization of Maxwell's equations

The dimensionless Maxwell equations for the TM^z mode with the constitutive equations of section 2 are given by

$$\mu_r(x,y)\frac{\partial H_x(x,y,t)}{\partial t} = -\frac{\partial E_z(x,y,t)}{\partial y} - (M_x(x,y,t) + \sigma_m(x,y)H_x(x,y,t)), \tag{26}$$

$$\mu_r(x,y)\frac{\partial H_y(x,y,t)}{\partial t} = \frac{\partial E_z(x,y,t)}{\partial x} - (M_y(x,y,t) + \sigma_m(x,y)H_y(x,y,t)), \tag{27}$$

$$\epsilon_r(x,y)\frac{\partial E_z(x,y,t)}{\partial t} = \frac{\partial H_y(x,y,t)}{\partial x} - \frac{\partial H_x(x,y,t)}{\partial y} - (J(x,y,t) + \sigma(x,y)E_z(x,y,t)). \tag{28}$$

Discretization of Faraday's equation for the H_x component for nodes not near the boundary

In this section central differences are applied to Faraday's equation for the H_x component (26) on the set G_{H_x} of H_x nodes. This cannot be done for all H_x nodes in the same way because for some H_x nodes the standard central difference expression refers to missing E_z nodes on the boundary. This is the case for H_x nodes that are located half a spatial step Δy above the bottom boundary of the domain or below the top boundary

$$B_{b,H_x} := \{(m\Delta x, \frac{1}{2}\Delta y) : m \in \{1,...,N_x\}\} \subset G_{H_x},$$
$$B_{t,H_x} := \{(m\Delta x, (N_y + \frac{1}{2})\Delta y) : m \in \{1,...,N_x\}\} \subset G_{H_x}.$$

The remainder of H_x nodes, which are not located next to the boundary of the domain, is given by the proper subset

$$I_{H_x} := \{(m\Delta x, (n - \frac{1}{2})\Delta y) : m \in \{1,...,N_x\}, n \in \{2,...,N_y\}\} \subset G_{H_x}.$$

Central difference approximations to the derivative $\frac{\partial E_z}{\partial y}$ in Faraday's equation (26) at the H_x nodes in I_{H_x} yield

$$\begin{aligned}\mu_r(m\Delta x, (n-\frac{1}{2})\Delta y)\frac{dH_x}{dt}(m\Delta x, (n-\frac{1}{2})\Delta y, t) = &-\frac{E_z(m\Delta x, n\Delta y, t) - E_z(m\Delta x, (n-1)\Delta y, t)}{\Delta y} \\ &- (M_x(m\Delta x, (n-\frac{1}{2})\Delta y, t) \\ &+ \sigma_m(m\Delta x, (n-\frac{1}{2})\Delta y)H_x(m\Delta x, (n-\frac{1}{2})\Delta y, t)),\end{aligned} \tag{29}$$

for $m \in \{1,...,N_x\}$ and $n \in \{2,...,N_y\}$. Central difference approximations are second order accurate. This fact is derived using Taylor series in textbooks in numerical analysis.

Discretization of Faraday's equation for the H_x component near the boundary of the domain

Suppose that the boundary of the domain is a perfect electric conductor. Then for H_x nodes next to the bottom boundary of the domain (the set B_{b,H_x}) we obtain using central differences and setting to zero boundary nodes

$$\mu_r(m\Delta x, \frac{1}{2}\Delta y)\frac{dH_x}{dt}(m\Delta x, \frac{1}{2}\Delta y, t) = -\frac{E_z(m\Delta x, \Delta y, t)}{\Delta y} - (M_x(m\Delta x, \frac{1}{2}\Delta y, t) + \sigma_m(m\Delta x, \frac{1}{2}\Delta y)H_x(m\Delta x, \frac{1}{2}\Delta y, t)), \tag{30}$$

where $m \in \{1, ..., N_x\}$, and for H_x nodes next to the top boundary (the set B_{t,H_x}) we obtain

$$\mu_r(m\Delta x, (N_y + \frac{1}{2})\Delta y)\frac{dH_x}{dt}(m\Delta x, (N_y + \frac{1}{2})\Delta y, t) = \frac{E_z(m\Delta x, N_y\Delta y, t)}{\Delta y} - (M_x(m\Delta x, (N_y + \frac{1}{2})\Delta y, t) + \sigma_m(m\Delta x, (N_y + \frac{1}{2})\Delta y)H_x(m\Delta x, (N_y + \frac{1}{2})\Delta y, t)), \tag{31}$$

where $m \in \{1, ..., N_x\}$.

Discretization of Faraday's equation for the H_y component

In this section central differences are applied to Faraday's equation for the H_y component (27) on the set G_{H_y} of H_y nodes. Similar to the case for H_x this cannot be done automatically because the boundary H_y have been set to zero. This is the case for the subset of leftmost H_y nodes and the subset of rightmost H_y nodes

$$B_{l,H_y} := \{(\frac{1}{2}\Delta x, n\Delta y) : n \in \{1, ..., N_y\}\} \subset G_{H_y},$$
$$B_{r,H_y} := \{((N_x + \frac{1}{2})\Delta x, n\Delta y) : n \in \{1, ..., N_y\}\} \subset G_{H_y}.$$

The proper subset of H_y nodes that do not lie next to the boundary is given by

$$I_{H_y} := \{((m - \frac{1}{2})\Delta x, n\Delta y) : m \in \{2, ..., N_x\}, n \in \{1, ..., N_y\}\} \subset G_{H_y}.$$

Central difference approximations of $\frac{\partial E_z}{\partial x}$ in Faraday's equation (27) on I_{H_y} yield

$$\mu_r((m - \frac{1}{2})\Delta x, n\Delta y)\frac{dH_y}{dt}((m - \frac{1}{2})\Delta x, n\Delta y, t) = \frac{E_z(m\Delta x, n\Delta y, t) - E_z((m-1)\Delta x, \Delta y, t)}{\Delta x} - (M_y((m - \frac{1}{2})\Delta x, n\Delta y, t) + \sigma_m((m - \frac{1}{2})\Delta x, n\Delta y)H_x((m - \frac{1}{2})\Delta x, n\Delta y, t)) \tag{32}$$

for $m \in \{2, ..., N_x\}$ and $n \in \{1, ..., N_y\}$. Similar to the case for H_x, equation (32) cannot be used for H_y nodes in $G_{H_y} \setminus I_{H_y}$. For the leftmost H_y nodes (B_{l,H_y}) we have

$$\mu_r(\frac{1}{2}\Delta x, n\Delta y)\frac{dH_y}{dt}(\frac{1}{2}\Delta x, n\Delta y, t) = \frac{E_z(\Delta x, n\Delta y, t)}{\Delta x} - (M_y(\frac{1}{2}\Delta x, n\Delta y, t) + \sigma_m(\frac{1}{2}\Delta x, n\Delta y)H_y(\frac{1}{2}\Delta x, n\Delta y, t), \tag{33}$$

for $n \in \{1, ..., N_y\}$, and for the rightmost nodes (B_{r,H_y})

$$\mu_r((N_x + \frac{1}{2})\Delta x, n\Delta y)\frac{dH_y}{dt}((N_x + \frac{1}{2})\Delta x, n\Delta y, t) = -\frac{E_z(N_x\Delta x, n\Delta y, t)}{\Delta x} - (M_y((N_x + \frac{1}{2})\Delta x, n\Delta y, t) + \sigma_m((N_x + \frac{1}{2})\Delta x, n\Delta y, t)H_y((N_x + \frac{1}{2})\Delta x, n\Delta y, t), \tag{34}$$

for $n \in \{1, ..., N_y\}$.

Discretization of Ampere's equation

Central difference approximations of the spatial derivatives in equation (28) for the set G_{E_z} of E_z nodes yield

$$\epsilon_r(m\Delta x, n\Delta y)\frac{dE_z}{dt}(m\Delta x, n\Delta y, t) = \frac{H_y((m + \frac{1}{2})\Delta x, n\Delta y, t) - H_y((m - \frac{1}{2})\Delta x, n\Delta y, t)}{\Delta x} - \frac{H_x(m\Delta x, (n + \frac{1}{2})\Delta y, t) - H_x(m\Delta x, (n - \frac{1}{2})\Delta y, t)}{\Delta y} - (J(m\Delta x, n\Delta y) + \sigma_r(m\Delta x, n\Delta y)E_z(m\Delta x, n\Delta y, t)) \tag{35}$$

for $m \in \{1, ..., N_x\}$ and $n \in \{1, ..., N_y\}$. The central difference expressions are defined for all E_z nodes and we do not need to make any modifications.

1.2 Matrix formulation of the space-discretized Maxwell equations

Definition of vectors

The system of ODE's (29) to (35) can be written in matrix form. To this end the H_x components on G_{H_x}, H_y components on G_{H_y} and E_z components on G_{E_z} are collected in vectors $\mathbf{h}_x(t)$, $\mathbf{h}_y(t)$ and $\mathbf{e}_z(t)$ respectively which are put into a vector

$$\mathbf{y}(t) := \begin{bmatrix} \mathbf{h}_x(t) \\ \mathbf{h}_y(t) \\ \mathbf{e}_z(t) \end{bmatrix}$$

The nodes are distributed over a two-dimensional domain. However the numerical field components associated with the nodes are put into vectors that have only a single index. Clearly we need to define some way of assigning indices to nodes in a two-dimensional grid. We give the node corresponding to some particular field component with the smallest x-coordinate and y-coordinate[3] the index 1 and count in the y-direction. After exhausting a column of nodes we continue with the next value of x, count in the y-direction and so on. Therefore H_x node $(m\Delta x, (n - \frac{1}{2})\Delta y) \in G_{H_x}$, H_y node $((m - \frac{1}{2})\Delta x, n\Delta y) \in G_{H_y}$ and E_z node $(m\Delta x, n\Delta y) \in G_{E_z}$ have indices

$$\begin{aligned} i &:= (m - 1)(N_y + 1) + n, \\ j &:= (m - 1)N_y + n, \\ k &:= (m - 1)N_y + n, \end{aligned}$$

in $\mathbf{h}_x(t)$, $\mathbf{h}_y(t)$ and $\mathbf{e}_z(t)$ respectively. The number of H_x, H_y and E_z nodes is given by

$$n_{h_x} := N_x(N_y + 1),$$

[3]Obviously this node exists in a TM^z Yee grid.

$$n_{h_y} := (N_x + 1)N_y,$$
$$n_{e_z} := N_x N_y,$$

respectively. Note that the definition sign has been used to define the symbols, the number of grid points follows from the definition of the Yee grid. So H_x node $(m\Delta x, (n - \frac{1}{2})\Delta y) \in G_{H_x}$, H_y node $((m - \frac{1}{2})\Delta x, n\Delta y) \in G_{H_y}$ and E_z node $(m\Delta x, n\Delta y) \in G_{E_z}$ have indices i, $n_{h_x} + j$ and $n_{h_x} + n_{h_y} + k$ in the vector $\mathbf{y}(t)$ respectively.

System of ordinary differential equations

We will show that the system of ODE's (29) to (35) can be written in the following form

$$\begin{bmatrix} \frac{d}{dt}\mathbf{h}_x(t) \\ \frac{d}{dt}\mathbf{h}_y(t) \\ \frac{d}{dt}\mathbf{e}_z(t) \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{\mu_r,x} & 0 & 0 \\ 0 & \mathbf{P}_{\mu_r,y} & 0 \\ 0 & 0 & \mathbf{P}_{\epsilon_r} \end{bmatrix}^{-1} \left(\begin{bmatrix} -\mathbf{\Sigma}_x & \mathbf{0} & \mathbf{K} \\ \mathbf{0} & -\mathbf{\Sigma}_y & \mathbf{L} \\ -\mathbf{K}^\intercal & -\mathbf{L}^\intercal & -\mathbf{\Sigma}_z \end{bmatrix} \begin{bmatrix} \mathbf{h}_x(t) \\ \mathbf{h}_y(t) \\ \mathbf{e}_z(t) \end{bmatrix} - \begin{bmatrix} \mathbf{M}_x(t) \\ \mathbf{M}_y(t) \\ \mathbf{J}(t) \end{bmatrix} \right) \tag{36}$$

Here $\mathbf{M}_x(t)$, $\mathbf{M}_y(t)$ and $\mathbf{J}(t)$ are vectors that contain the values of the magnetic and electric (free) source currents at the nodes in G_{H_x}, G_{H_y} and G_{E_z} respectively. $\mathbf{P}_{\mu_r,x}$, $\mathbf{P}_{\mu_r,y}$ and $\mathbf{P}_{\epsilon_r}$ are diagonal matrices with the relative pemeabilities at the H_x and H_y nodes and the relative permittivities at the E_z nodes as diagonal elements. $\mathbf{\Sigma}_x$, $\mathbf{\Sigma}_y$ and $\mathbf{\Sigma}_z$ are diagonal matrices with the dimensionless (magnetic) conductivities at the diagonal elements. In the sections that follow we shall construct $\mathbf{K}$ and $\mathbf{L}$ using the Kronecker product.

Matrix description for central differences in one dimension

In this section we will make a start with the derivation of the matrices $\mathbf{K}$ and $\mathbf{L}$ that are used in the system of ODE's (36). Let $F : [0, L] \mapsto \mathbb{R}$ be a differentiable function. Let the number N of grid points be given. The spatial step size is given by

$$\Delta x = \frac{L}{N + 1}.$$

Let the set of grid points be

$$P := \{m\Delta x : m \in \{1, ..., N\}\}.$$

Central differences can be applied to the derivative of F at points in between grid points

$$\frac{dF}{dx}((m + \frac{1}{2})\Delta x) \approx \frac{F((m + 1)\Delta x) - F(m\Delta x)}{\Delta x},$$

for $m \in \{0, ..., N\}$. Let $F(0) = F((N + 1)\Delta x) = 0$. Let $\mathbf{f} \in \mathbb{R}^{N+1}$ be a vector that contains the values of F at all half-way points in between grid points and in between the first and last grid point and the boundary of the domain. The following matrix-vector product approximates the derivative of $\mathbf{f}$

$$\frac{d\mathbf{f}}{dx} = \begin{bmatrix} \frac{dF}{dx}(\frac{1}{2}\Delta x) \\ \frac{dF}{dx}(\frac{3}{2}\Delta x) \\ \frac{dF}{dx}(\frac{5}{2}\Delta x) \\ \vdots \\ \frac{dF}{dx}((N - \frac{1}{2})\Delta x) \\ \frac{dF}{dx}((N + \frac{1}{2})\Delta x) \end{bmatrix} \approx (\Delta x)^{-1} \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ -1 & 1 & & 0 & 0 \\ 0 & -1 & \ddots & & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & 0 & & -1 & 1 \\ 0 & 0 & \cdots & 0 & -1 \end{bmatrix} \begin{bmatrix} F(\Delta x) \\ F(2\Delta x) \\ \vdots \\ F((N - 1)\Delta x) \\ F(N\Delta x) \end{bmatrix} \tag{37}$$

We will use the symbol $\mathbf{A}_n$ for a matrix of the form that appears in this matrix-vector product. It has dimensions $(n + 1) \times n$. In this case $n = N_x$.

Matrix description of central difference approximations to $\frac{\partial E_z}{\partial y}$

In this section we will continue with the derivation of the matrices $\mathbf{K}$ and $\mathbf{L}$ that are used in the system of ODE's (36). We will first derive the matrix $\mathbf{K}$. You can see in the system that the matrix $\mathbf{K}$ relates the vector $\frac{d\mathbf{h}_x}{dt}$ to the vector $\mathbf{e}_z$. This is a matrix that contains the central difference approximations (29), (30) and (31) at the set G_{H_x} of H_x nodes to the $\frac{\partial E_z}{\partial y}$ derivative that appears in Faraday's equation for the H_x component (26). The discussion in the previous section can be applied to the nodes in the set G_{H_x}. This discussion shows that the central difference approximations to $\frac{\partial E_z}{\partial y}$ at nodes in a column along the y-axis of the grid only depend on values of E_z at nodes in this column and not in any other column. Hence the central difference approximations to $\frac{\partial E_z}{\partial y}$ at all nodes in G_{H_x} can be determined separately for each column of H_x nodes. These are one-dimensional problems that can be solved as in the previous section. As a result the matrix that approximates $\frac{\partial E_z}{\partial y}$ at all H_x nodes consists of N_x blocks (for the N_x columns of H_x nodes) of $\mathbf{A}_{N_y}$ as defined under equation (37) centered around the diagonal. The diagonal structure ensures that there is no interaction between different columns. The matrices $\mathbf{A}_{N_y}$ have dimensions $(N_y+1)\times N_y$ because each column of H_x nodes contains N_y+1 H_x nodes, however the central difference approximations at these nodes depend on only N_y nonzero E_z nodes, because the boundaries of the grid have been set to zero. Here is the equation

$$\frac{\partial \mathbf{e}_z}{\partial y}(t) := \begin{bmatrix} \frac{\partial}{\partial y}\mathbf{e}_{z,1}(t) \\ \frac{\partial}{\partial y}\mathbf{e}_{z,2}(t) \\ \vdots \\ \frac{\partial}{\partial y}\mathbf{e}_{z,N_x-1}(t) \\ \frac{\partial}{\partial y}\mathbf{e}_{z,N_x}(t) \end{bmatrix} \approx (\Delta y)^{-1} \begin{bmatrix} \mathbf{A}_{N_y} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{N_y} & & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & \vdots \\ \mathbf{0} & \mathbf{0} & & \mathbf{A}_{N_y} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{A}_{N_y} \end{bmatrix} \mathbf{e}_z(t)$$

where

$$\frac{\partial}{\partial y}\mathbf{e}_{z,i}(t) := \begin{bmatrix} \frac{\partial E_z}{\partial y}(i\Delta x, \frac{1}{2}\Delta y, t) \\ \frac{\partial E_z}{\partial y}(i\Delta x, \frac{3}{2}\Delta y, t) \\ \vdots \\ \frac{\partial E_z}{\partial y}(i\Delta x, (N_y+\frac{1}{2})\Delta y, t) \end{bmatrix}.$$

The matrix in this matrix-vector product is the matrix $\mathbf{K}$ in equation (36) multiplied by $-\Delta y$. The derivative in the symbol $\frac{\partial \mathbf{e}_z}{\partial y}$ should be seen as part of the notation. Only the derivatives inside the vectors $\frac{\partial}{\partial y}\mathbf{e}_{z,i}$ are true derivatives. It is a vector of derivatives evaluated at the G_{H_x} grid points, not a derivative of a vector. $\mathbf{0}$ represents a $(N_y+1)\times N_y$ zero matrix. We can summarize the matrix description of central differences to $\frac{\partial E_z}{\partial y}$ using the Kronecker product

$$\frac{\partial \mathbf{e}_z}{\partial y}(t) \approx (\Delta y)^{-1}(I_{N_x} \otimes A_{N_y})\mathbf{e}_z(t).$$

We will not yet express $\frac{d\mathbf{h}_x}{dt}(t)$ in terms of $\mathbf{e}_z(t)$ but we will save this for a later section.

Matrix description of central difference approximations to $\frac{\partial E_z}{\partial x}$

We will now derive the matrix $\mathbf{L}$. You can see in the system of ODE's (36) that the matrix $\mathbf{L}$ relates the vector $\frac{d\mathbf{h}_y}{dt}(t)$ to the vector $\mathbf{e}_z(t)$. This is a matrix that contains the central difference approximations 32 to 34 at the set G_{H_y} of H_y nodes to the $\frac{\partial E_z}{\partial x}$ derivative that appears in Faraday's equation for the H_y component (27). The matrix that we obtain again looks a bit like the one-dimensional matrix A_n. The difference is that H_y nodes corresponding to a single row in the

x-direction are spread N_y elements apart instead of 1 in the vector $\mathbf{h}_y$. The central difference approximations to the $\frac{\partial E_z}{\partial x}$ derivative at the first row of H_y nodes (thus for $n = 1$) are given by

$$\frac{\partial \mathbf{e}_{z,1}}{\partial x} := \begin{bmatrix} \frac{\partial}{\partial x}E_z(\frac{1}{2}\Delta x, \Delta y, t) \\ 0 \\ \vdots \\ 0 \\ \frac{\partial}{\partial x}E_z(\frac{3}{2}\Delta x, \Delta y, t) \\ 0 \\ \vdots \\ 0 \\ \frac{\partial}{\partial x}E_z(\frac{5}{2}\Delta x, \Delta y, t) \\ 0 \\ \vdots \\ \vdots \\ 0 \\ \frac{\partial}{\partial x}E_z((N_x - \frac{1}{2})\Delta x, \Delta y, t) \\ 0 \\ \vdots \\ 0 \\ \frac{\partial}{\partial y}E_z((N_x + \frac{1}{2})\Delta x, \Delta y, t) \\ 0 \\ \vdots \\ 0 \end{bmatrix} \approx (\Delta x)^{-1} \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ -\mathbf{R}_1 & \mathbf{R}_1 & & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{R}_1 & \ddots & & \vdots \\ \vdots & & \ddots & \mathbf{R}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & & -\mathbf{R}_1 & \mathbf{R}_1 \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & -\mathbf{R}_1 \end{bmatrix} \mathbf{e}_z(t) \tag{38}$$

where $\mathbf{0}$ is a $N_y \times N_y$ zero matrix and $\mathbf{R}_1$ is a $N_y \times N_y$ matrix (there are N_y rows of H_y nodes and N_y nonzero E_z nodes in each row) defined as

$$\mathbf{R}_1 := \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

The vector $\frac{\partial \mathbf{e}_{z,1}}{\partial x}$ has length n_{h_y} and its indexing has been performed in the same way as in $\mathbf{h}_y$. All elements corresponding to nodes not in the first row of G_{H_y} have been set to zero. For the second row of H_x nodes we obtain a similar pattern. The higher y-value results in a shift downwards and to the right in the matrix $\mathbf{R}_1$, which results in a matrix $\mathbf{R}_2$ that is used in the same way and has the same dimensions as $\mathbf{R}_1$. $\mathbf{R}_2$ is defined as

$$\mathbf{R}_2 := \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 1 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

We add the resulting vector $\frac{\partial \mathbf{e}_{z,2}}{\partial x}$ to the vector $\frac{\partial \mathbf{e}_{z,1}}{\partial x}$. The resulting sum can be expressed in terms of $\mathbf{e}_z(t)$ by replacing $\mathbf{R}_1$ in equation (38) by $\mathbf{R}_1 + \mathbf{R}_2$. This replaces the zero values at the

elements corresponding to the second row of H_y nodes with central difference approximations to the derivative $\frac{\partial E_z}{\partial x}$ at the second row of H_y nodes. We can do the same thing for all rows. If we add the vectors $\frac{\partial \mathbf{e}_{z,1}}{\partial x}, \frac{\partial \mathbf{e}_{z,2}}{\partial x}, \cdots, \frac{\partial \mathbf{e}_{z,N_y}}{\partial x}$ and the matrices $\mathbf{R}_1, \mathbf{R}_2, \cdots, \mathbf{R}_{N_y}$ (there are N_y rows of H_y nodes) for all rows then we get a a vector $\frac{\partial \mathbf{e}_z}{\partial x}$ and a matrix $\mathbf{R}$ that equal to the identity matrix $\mathbf{I}_{N_y}$ with dimensions $N_y \times N_y$. The central difference approximation of $\frac{\partial E_z}{\partial x}$ can then be more tersely described using the kronecker product

$$\frac{\partial \mathbf{e}_z}{\partial x}(t) = (\Delta x)^{-1}(A_{N_x} \otimes I_{N_y})\mathbf{e}_z(t)$$

The matrix $A_{N_x} \otimes I_{N_y}$ that appears in this matrix-vector product is the matrix $\mathbf{L}$ multiplied by Δx in the system of ODE's (36). $\frac{d\mathbf{h}_y}{dt}$ will be expressed in terms of $\mathbf{e}_z$ in a later section.

Matrix description of central difference approximations to $\frac{\partial H_y}{\partial x}$ and $\frac{\partial H_x}{\partial y}$

In the preceding sections we have derived the structure of the matrices $\mathbf{K}$ and $\mathbf{L}$. In the system of ODE's we proposed that the central difference approximation to the derivatives $\frac{\partial H_y}{\partial x}$ and $\frac{\partial H_x}{\partial y}$ that appear in Ampere's equation can be described by matrices $-\mathbf{K}^\intercal$ and $-\mathbf{L}^\intercal$. These relate the vector $\frac{d\mathbf{e}_z}{dt}(t)$ to the vectors $\mathbf{h}_x(t)$ and $\mathbf{h}_y(t)$. Consider the first column (with $m = 1$) of E_z nodes. Again using the results for the one-dimensional case we obtain the following matrix description of the central difference approximation to the derivative $\frac{\partial H_x}{\partial y}(t)$ on the first column of E_z nodes

$$\frac{\partial \mathbf{h}_{x,1}}{\partial y}(t) = \begin{bmatrix} \frac{\partial}{\partial y}H_x(\Delta x, \Delta y, t) \\ \frac{\partial}{\partial y}H_x(\Delta x, 2\Delta y, t) \\ \vdots \\ \frac{\partial}{\partial y}H_x(\Delta x, (N_y-1)\Delta y, t) \\ \frac{\partial}{\partial y}H_x(\Delta x, N_y\Delta y, t) \end{bmatrix} \approx (\Delta y)^{-1} \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & & 0 & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & 0 & & -1 & 1 & 0 \\ 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix} \tilde{\mathbf{h}}_x(t)$$

Here $\tilde{\mathbf{h}}_x$ is the first column of H_x nodes. Note that we are approximating derivatives of $H_{x,ex}$ at E_z nodes but that these approximations are expressed in terms of numerical H_x values at the H_x nodes. The matrix that appears in this matrix-vector product has dimensions $N_y \times (N_y + 1)$ (there are N_y E_z nodes and $N_y + 1$ H_x nodes in each column). Note that it is equal to $-\mathbf{A}_{N_y}^\intercal$. The derivation of the matrix description of the central difference approximations to the derivative $\frac{\partial H_x}{\partial y}$ on all E_z nodes is very similar to the derivation of the matrix $\mathbf{K}$ so we will not repeat it here. The result is

$$\frac{\partial \mathbf{h}_x}{\partial y}(t) = (\Delta y)^{-1}(\mathbf{I}_{N_x} \otimes (-\mathbf{A}_{N_y})^\intercal)\mathbf{h}_x(t).$$

The derivation of the matrix description of the central difference approximations to the derivative $\frac{\partial H_y}{\partial x}$ is very similar to the derivation of the matrix $\mathbf{L}$ so we will again not repeat it here. The result is

$$\frac{\partial \mathbf{h}_y}{\partial x}(t) = (\Delta x)^{-1}((-\mathbf{A}_{N_x})^\intercal \otimes \mathbf{I}_{N_y})\mathbf{h}_y(t).$$

We have now shown that the system of ODE's can be written in the form (36).

Matrix description of central difference approximations to the Maxwell equations using the Kronecker product

In this section we will describe the central difference approximations to the Maxwell equations using the Kronecker product. They are given by

$$\frac{d\mathbf{h}_x(t)}{dt} = (\mathbf{P}_{\mu_r,x})^{-1}(-(\Delta y)^{-1}(\mathbf{I}_{N_x} \otimes \mathbf{A}_{N_y})\mathbf{e}_z(t) - \mathbf{M}_x(t) - \mathbf{\Sigma}_x\mathbf{h}_x(t))$$

$$
\begin{aligned}
&= (\mathbf{P}_{\mu_r,x})^{-1}(\mathbf{K}\mathbf{e}_z(t) - \mathbf{M}_x(t) - \mathbf{\Sigma}_x\mathbf{h}_x(t)),\\
\frac{d\mathbf{h}_y(t)}{dt} &= (\mathbf{P}_{\mu_r,y})^{-1}((\Delta x)^{-1}(\mathbf{A}_{N_x}\otimes\mathbf{I}_{N_y})\mathbf{e}_z(t) - \mathbf{M}_y(t) - \mathbf{\Sigma}_y\mathbf{h}_y)\\
&= (\mathbf{P}_{\mu_r,y})^{-1}(\mathbf{L}\mathbf{e}_z(t) - \mathbf{M}_y(t) - \mathbf{\Sigma}_y\mathbf{h}_y),\\
\frac{d\mathbf{e}_z(t)}{dt} &= (\mathbf{P}_{\epsilon_r})^{-1}((\Delta x)^{-1}((-\mathbf{A}_{N_x})^\intercal\otimes\mathbf{I}_{N_y})\mathbf{h}_y(t) - (\Delta y)^{-1}(\mathbf{I}_{N_x}\otimes(-\mathbf{A}_{N_y})^\intercal)\mathbf{h}_x(t) -\\
&- \mathbf{J}(t) - \mathbf{\Sigma}_z\mathbf{e}_z(t))\\
&= (\mathbf{P}_{\epsilon_r})^{-1}(-\mathbf{K}^\intercal\mathbf{h}_x(t) - \mathbf{L}^\intercal\mathbf{h}_y(t)) - \mathbf{J}(t) - \mathbf{\Sigma}_z\mathbf{e}_z(t)).
\end{aligned}
$$

2 Using source currents to generate arbitrary field patterns

In this section we would like to determine the set of source terms necessary for an arbitrary electromagnetic pattern. The only assumptions necessary is that the time and spatial component of the field are seperable. We start off with the dimensionless Maxwell equations for the TM_z mode:

$$\mu_r \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y} - M_x \tag{39a}$$

$$\mu_r \frac{\partial H_y}{\partial t} = \frac{\partial E_z}{\partial x} - M_y \tag{39b}$$

$$\epsilon_r \frac{\partial E_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - J \tag{39c}$$

Solving these equations for M_x, M_y and E_z would essentially yield the source current needed to obtain the analytical solution to arbitrary H_x,H_y,E_z, at least if our initial conditions match these values. Thus if we implement the source currents in our discretized model we should obtain the wanted field patterns.
With our discretized model we have that:

$$h^{t+0.5} = h^{t-0.5} + \Delta t(A_h e_z^t + v_h^t) \tag{40a}$$

$$e^{t+1} = e^t + \Delta t(A_e h^{t+0.5} + v_e^{t+0.5}) \tag{40b}$$

with $h^q = (h_x^q, h_y^q)$.
Using 39 and 40 it is clear that we need to relate v_h and v_e to the source currents M_x, M_y, E_z, since the matrices A_h and A_e take care of the Maxwell equations without source current and conductivities

$$v_{h_x}^t(i) = -\frac{M_x}{\mu_r}(x(i), y(i), t) \tag{41a}$$

$$v_{h_y}^t(i) = -\frac{M_y}{\mu_r}(x(i), y(i), t) \tag{41b}$$

$$v_e^t(i) = -\frac{J}{\epsilon_r}(x(i), y(i), t) \tag{41c}$$

Here i denotes the i'th index in the vector and x(i) and y(i) it's coordinates. The v_h is given by $(v_{h_x}; v_{h_y})$ Now we use the equation of the courant number without dimensions:

$$\Delta t = S_c \Delta x \tag{42}$$

this results in:

$$\Delta t v_{h_x}(i,t) = S_c \Delta x(\frac{\partial E_z}{\mu_r \partial y} + \frac{\partial H_x}{\partial t})(x(i), y(i), t) \tag{43a}$$

$$\Delta t v_{h_y}(i,t) = S_c \Delta x(-\frac{\partial E_z}{\mu_r \partial x} + \frac{\partial H_y}{\partial t})(x(i), y(i), t) \tag{43b}$$

$$\Delta t v_e(i,t) = S_c \Delta x(\frac{\partial E_z}{\partial t} + \frac{\partial H_x}{\epsilon_r \partial y} - \frac{\partial H_y}{\epsilon_r \partial x})(x(i), y(i), t) \tag{43c}$$

2.1 Analytical solution for harmonic spatial patterns

we want to derive an analytical solution for any given wave pattern defined by

$$H_x = H_{x_0} b_x(t) \tag{44a}$$

$$H_y = H_{y_0} b_y(t) \tag{44b}$$

$$E_z = E_{z_0} a(t) \tag{44c}$$

where:

$$H_{x_0} = \sin(\pi x)\cos(\pi y) \tag{45a}$$

$$H_{y_0} = \cos(\pi x)\sin(\pi y) \tag{45b}$$

$$E_{z_0} = \sin(\pi x)\sin(\pi y) \tag{45c}$$

We choose a rectangular 1×1 grd with a number of N evenly distributed nodes (excluding the boundary) as our domain. Thus the analytical solution satisfies the conditions for a superconductor boundary.
Substituting 44 into 43 and using that $\Delta x = 1/(N+1)$ we get:

$$\Delta t v_{h_x}(i,t) = \frac{S_c}{(N+1)} H_{x_0}(x(i), y(i))(\frac{\pi}{\mu_r} a(t) + b'_x(t)) \tag{46a}$$

$$\Delta t v_{h_y}(i,t) = \frac{S_c}{(N+1)} H_{y_0}(x(i), y(i))(-\frac{\pi}{\mu_r} a(t) + b'_y(t)) \tag{46b}$$

$$\Delta t v_e(i,t) = \frac{S_c}{(N+1)} E_{z_0}(x(i), y(i))(a'(t) + \frac{\pi}{\epsilon_r}(b_y(t) - b_x(t))) \tag{46c}$$

2.2 Analytical solution for polynomial spatial patterns

To prevent the solution being an eigenvalue we determine a second solution of the form:

$$H_x = x(x-1)(2y-1)\beta_x(t) \tag{47a}$$

$$H_y = (2x-1)y(y-1)\beta_y(t) \tag{47b}$$

$$E_z = x(x-1)y(y-1)\alpha(t) \tag{47c}$$

We use the same grid and therefore get a solution:

$$\Delta t v_{h_x}(i,t) = \frac{S_c}{(N+1)} x(i)(x(i)-1)(2y(i)-1)(\frac{1}{\mu_r}\alpha(t) + \beta'_x(t)) \tag{48a}$$

$$\Delta t v_{h_y}(i,t) = \frac{S_c}{(N+1)} (2x(i)-1)y(i)(y(i)-1)(-\frac{1}{\mu_r}\alpha(t) + \beta'_y(t)) \tag{48b}$$

$$\Delta t v_e(i,t) = \frac{S_c}{(N+1)} (x(i)(x(i)-1)y(i)(y(i)-1)\alpha'(t) + \frac{2}{\epsilon_r}(\beta_x(t)x(i)(x(i)-1) - \beta_y(t)y(i)(y(i)-1))) \tag{48c}$$

Since the solutions given in 46 and 47 can be super imopsed also every linear combination of solutions for different a, b, α, β will satisfy the equations, if we use the same linear combination for our v_h and v_e's.

Bibliography

[1] M. A. Botchev. A block Krylov subspace time-exact solution method for linear ODE systems. Memorandum 1973, Department of Applied Mathematics, University of Twente, Enschede, January 2012.

[2] M. A. Botchev. A short guide to exponential Krylov subspace time integration for Maxwell's equations. Memorandum 1992, Department of Applied Mathematics, University of Twente, Enschede, September 2012.

[3] M. A. Botchev, V. Grimm, and M. Hochbruck. Residual, restarting, and richardson iteration for the matrix exponential. *SIAM Journal on Scientific Computing*, 35(3):A1376–A1397, 2013.

[4] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, 100(1):32–74, 1928.

[5] E. Estrada and D. J. Higham. Network properties revealed through matrix functions. *SIAM review*, 52(4):696–714, 2010.

[6] F. Gantmacher. The theory of matrices, vol. i. *Chelsea, New York*, 220, 1959.

[7] S. D. Gedney. An anisotropic perfectly matched layer-absorbing medium for the truncation of FDTD lattices. *Antennas and Propagation, IEEE Transactions on*, 44(12):1630–1639, 1996.

[8] D. J. Griffiths and R. College. *Introduction to electrodynamics*, volume 3. prentice Hall New Jersey, 1999.

[9] D. J. Higham, P. Grindrod, and E. Estrada. People who read this article also read...: Part i. *SIAM News*, 44(1), 2011.

[10] N. Higham. *Functions of Matrices: Theory and Computation*. SIAM e-books. Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2008.

[11] M. Hochbruck and C. Lubich. On Krylov subspace approximations to the matrix exponential operator. *SIAM Journal on Numerical Analysis*, 34(5):1911–1925, 1997.

[12] M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numerica*, 19(1):209–286, 2010.

[13] J. D. Joannopoulos, S. G. Johnson, J. N. Winn, and R. D. Meade. *Photonic crystals: molding the flow of light*. Princeton university press, 2008.

[14] S. G. Johnson. Notes on perfectly matched layers (pmls). *Lecture notes, Massachusetts Institute of Technology, Massachusetts*, 2008.

[15] C. Moler and C. Van Loan. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM review*, 45(1):3–49, 2003.

[16] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, 2003.

[17] J. B. Schneider. Understanding the finite-difference time-domain method. www.eecs.wsu.edu/ schneidj/ufdtd, 2010.

[18] A. Taflove and S. C. Hagness. *Computational electrodynamics*, volume 160. Artech house Boston, 2000.

[19] J. van den Eshof and M. Hochbruck. Preconditioning lanczos approximations to the matrix exponential. *SIAM Journal on Scientific Computing*, 27(4):1438–1457, 2006.

[20] K. Yee. Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media. *Antennas and Propagation, IEEE Transactions on*, 14(3):302–307, 1966.