Discontinuous Galerkin Finite Element Methods for the time-harmonic Maxwell equations in periodic media

Irana Denissen

August 20, 2013

Contents

Chapter 1

Introduction

In this report, it will be shown how discontinuous Galerkin (DG) finite element methods can be used to compute time-harmonic electric and magnetic fields in periodic media. A periodic medium is a medium which consists of the same unit cell repeated infinitely many times in all directions. For example, photonic crystals are periodic media which are used to guide light. In practice, photonic crystals are made with finitely many unit cells, but with enough cells that the behaviour is similar to the theoretical infinite case. Applications of photonic crystals include lasers ([17]) and solar cells ([4]).

There are two kinds of problems that we would like to be able to solve in periodic media: first, we would like to know the electric and magnetic field given a medium, a source and the time-frequency. In the second problem, we do not have a source, but we want to know for which time-frequencies the fields are evanescent in the medium and for which they are extended. For the first problem, we have to solve the time-harmonic Maxwell equations and for the second problem, we have to solve the time-harmonic Maxwell eigenproblem [16]. Both problems will be discussed in more detail in Chapter 2.

In order to solve these problems, one has to solve or approximate the time-harmonic Maxwell equations. Since it is often impossible to solve these equations analytically, a numerical method to approximate the solution is necessary.

A widely used numerical method is the finite difference method with the Yee scheme [24]. It is very popular due to the intuitivity of the discretization of the differential operators. One can either first compute a solution in the time domain and use a Fourier transformation to find the frequencies that are in the field, or directly discretize the time-harmonic Maxwell equations. However, because of the simplicity of the discretization of the differential operators, the mesh is always dimension-by-dimension, so it cannot be adapted to the structure of the media. An example of the implementation of the finite difference frequency domain discretization is given in [25].

Another type of methods are spectral methods, which are based on writing the field as a linear combination of sines and cosines, analogously to the Fourier series of a function [13]. The main disadvantage of this method is that it always produces a continuous solution and therefore converges rather slowly for media with discontinuous dielectric constants. A description of the implementation of a spectral method for photonic crystals can be found in [15].

A method that is similar, but more localized is the finite element method, in which the solution is written as a linear combination of basis functions, typically low order polynomials. These basis functions are non-zero on only a few elements, so they can enforce continuity of (components of) the solution and the resulting matrix-system is sparse. It can handle difficult geometries and can achieve high accuracy. Examples of the implementation of the finite element method for photonic crystals are [6] and [8]. The disadvantage of the standard (conforming) finite element method is that the basis functions should be chosen with care, otherwise it will always produce a continuous approximate solution, even if the solution is discontinuous. Furthermore, if one wants to know which frequencies are extended in the crystal, the finite element method does not always produce the correct results: one could find eigenvalues of the time-harmonic Maxwell equations in the numerical approximation that are not extended frequencies in the physical problem (spurious modes), see for example [5].

The method that is considered in this report is the discontinuous Galerkin finite element method, which is similar to the finite element method. The difference is that it uses basis functions that are defined on only one element, which may lead to a discontinuous approximate solution. To obtain a sensible solution, a flux is chosen to define the value of the function on the faces between elements. This allows for grids with hanging nodes, making the grid even more flexible. Furthermore, the boundary conditions do not have to be incorporated in the finite element space, but can be enforced weakly through the flux at the boundary. Also, the mass-matrix is block-diagonal, which is beneficial if one wants to solve the eigenvalue problem or the time-dependent Maxwell equations. An overview of discontinuous Galerkin methods for elliptic problems can be found in [3].

Discontinuous Galerkin methods are already successfully used for the time-harmonic Maxwell problem, see for example [20] and [11]. In [7], Buffa and Perugia have given the conditions under which a discontinuous Galerkin approximation gives a correct approximation of Maxwell's eigenvalue problem, i.e. when there are no spurious modes, the spectrum is complete and the eigenspace is not polluted and complete. They have also shown that for the most common DG-methods these conditions are satisfied.

This report is structured as follows. First, the time-harmonic Maxwell equations are described and modified such that they can be used for periodic media. Then the DG discretization of the time-harmonic Maxwell equations on a bounded domain is given and implemented. It is followed by the discretization of infinitely periodic media using a DG method. This discretization is implemented for a simple test case. Finally, some conclusions and recommendations are given.

Chapter 2

Problem setting

In this section, the model for computing the electric and magnetic fields in periodic media is described. Since we are working with electro-magnetism, a good place to start are the normalized Maxwell equations in matter in a domain $\Omega \subseteq \mathbb{R}^3$. They are given by [10]

$$\begin{aligned} \nabla \times \mathbf{E} &= \frac{\partial \mathbf{B}}{\partial t} && \text{in } \Omega, \\ \nabla \cdot \mathbf{D} &= \rho && \text{in } \Omega, \\ \nabla \times \mathbf{H} &= \frac{\partial \mathbf{D}}{\partial t} + \mathbf{J} && \text{in } \Omega, \\ \nabla \cdot \mathbf{B} &= 0 && \text{in } \Omega. \end{aligned}$$

In these equations, $\mathbf{E}$ is the electric field, $\mathbf{H}$ is the magnetic field, $\mathbf{D}$ is the electric displacement field, $\mathbf{B}$ is the magnetic induction and $\mathbf{J}$ and ρ are the electric current intensity and the electric charge density. The operator ∇ is defined as $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)^T$. It is assumed that the media are linear, so that constitutive relations can be used to eliminate the electric displacement field $\mathbf{D}$ and magnetic induction $\mathbf{B}$. These relations are given by

$$\begin{aligned} \mathbf{E} &= \epsilon \mathbf{D} && \text{in } \Omega, \\ \mathbf{B} &= \mu \mathbf{H} && \text{in } \Omega, \end{aligned}$$

where the dielectric constant $\epsilon = \epsilon(\mathbf{x})$ and electric permeability $\mu = \mu(\mathbf{x})$ are symmetric positive definite matrices. Furthermore, they are assumed to only depend on position $\mathbf{x} \in \Omega$. The next step is to restrict ourselves to time-harmonic fields. Introduce the ansatz that the electric and magnetic field are time-harmonic:

$$\begin{aligned} \mathbf{E}(\mathbf{x}, t) &= Re(e^{-i\omega t}\mathbf{E}(\mathbf{x})) && \text{in } \Omega, \\ \mathbf{H}(\mathbf{x}, t) &= Re(e^{-i\omega t}\mathbf{H}(\mathbf{x})) && \text{in } \Omega. \end{aligned}$$

In the case of periodic media, the domain extends infinitely in all directions, so it is given by $\Omega = \mathbb{R}^3$ and there are no boundary conditions. Moreover, assume there is no electric charge density, so $\rho = 0$. The resulting system is

$$\nabla \times \epsilon^{-1} \nabla \times \mathbf{H}(\mathbf{x}) - \omega^2 \mu \mathbf{H}(\mathbf{x}) = \mathbf{J} \qquad \text{in } \mathbb{R}^3, \tag{2.1}$$

$$\nabla \cdot (\mu \mathbf{H}(\mathbf{x})) = 0 \qquad \text{in } \mathbb{R}^3. \tag{2.2}$$

When the magnetic field $\mathbf{H}(\mathbf{x})$ is computed, the electric field is given by $\mathbf{E}(\mathbf{x}) = -\frac{i}{\epsilon\omega} \nabla \times \mathbf{H}(\mathbf{x})$. Note that (2.2) holds globally if and only if $\nabla \cdot \mathbf{J} = 0$, so we will not discretize this condition.

If one only wants to know which frequencies are extended in the medium, one can choose

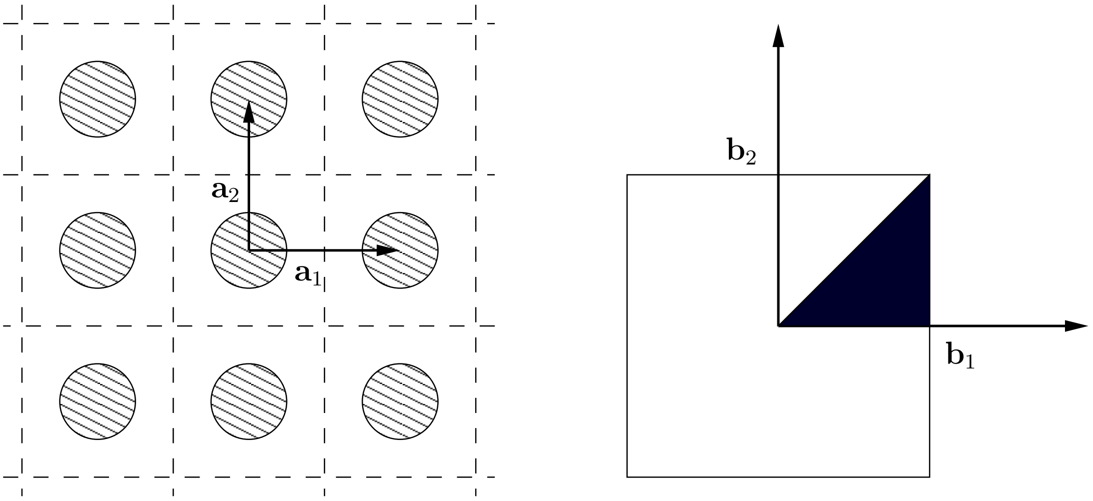


Figure 2.1: Example of a 2-dimensional periodic medium (left) and its Brillouin zone (right). The dark area in the right figure denotes the irreducible Brillouin zone.

the source to disappear, $\mathbf{J} = 0$, and solve the eigenvalue problem of finding $(\omega, \mathbf{H})$, such that (2.1) has a non-trivial solution, i.e.
Find $(\omega, \mathbf{H}) \neq (0, \mathbf{0})$ such that

$$\nabla \times \epsilon^{-1} \nabla \times \mathbf{H}(\mathbf{x}) = \omega^2 \mu \mathbf{H}(\mathbf{x}) \qquad \text{in } \mathbb{R}^3.$$

Of course, it is impossible to compute an approximate solution numerically on an infinitely large domain, so the domain somehow needs to be reduced to a finite domain. Fortunately, the periodicity of the structure can be used to take only one unit cell as the domain if some slight adaptations are made [8]. The periodicity can be expressed in terms of the dielectric constant as

$$\epsilon(\mathbf{x}) = \epsilon(\mathbf{x} + \mathbf{a}), \qquad \forall \mathbf{x} \in \mathbb{R}^3, \tag{2.3}$$

in which the vector $\mathbf{a}$ is a linear combination of the primitive lattice vectors $\mathbf{a}_i$, $i = 1, ..., d$. The primitive lattice vectors are the linearly independent vectors smallest in magnitude for which (2.3) holds, see for example Figure 2.1. Note that the number of primitive lattice vectors equals the dimension of the periodic medium. In case this dimension is smaller than three, there are also dimensions in which the medium is homogeneous. In that case

$$\epsilon(\mathbf{x}) = \epsilon(\mathbf{x} + \mathbf{a}^{\perp}), \qquad \forall \mathbf{x} \in \mathbb{R}^3,$$

in which $\mathbf{a}^{\perp} \perp \mathbf{a}_i, i = 1, ..., d$ is a vector in the direction of continuous translational symmetry. We define the vectors $\mathbf{a}_j^{\perp}, j = 1, ..., (3-d)$ such that $\|\mathbf{a}_j^{\perp}\| = 1$ and $\mathbf{a}_j^{\perp} \perp \mathbf{a}_i$ for all i and j and moreover, $\operatorname{span}(\mathbf{a}_i, \mathbf{a}_j^{\perp}) = \mathbb{R}^3$.
The primitive cell Ω is the set

$$\Omega = \{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x} = \sum_{i=1}^{d} \ell_i \mathbf{a}_i + \sum_{i=d+1}^{3} \ell_i \mathbf{a}_{4-i}^{\perp}, 0 \leq \ell_i \leq 1\}.$$

This is the domain where all the computations will be performed.
It is possible to associate reciprocal lattice vectors $\mathbf{b}_i$ to the primitive lattice vectors such that

$$\mathbf{a}_i \mathbf{b}_j = 2\pi \delta_{ij},$$

in which δ_{ij} is the Kronecker delta function.
The first Brillouin zone is now defined as the set of points which are closer to the origin than to any other point of the reciprocal lattice [13]. The irreducible Brillouin zone $\mathcal{B} \subset \mathbb{R}^d$ is the subset of the Brillouin zone which has no symmetries.
Using Bloch/Floquet theory, it can be shown that the magnetic field can be expressed as [13]

$$\mathbf{H}(\mathbf{x}) = e^{i\mathbf{k}\cdot\mathbf{x}}\mathbf{u}_{\mathbf{k}}(\mathbf{x}), \tag{2.4}$$

where $\mathbf{k} \in \mathcal{B}$ and $\mathbf{u}_{\mathbf{k}}(\mathbf{x})$ is a periodic function in the primitive cell Ω. Then the curl of the magnetic field is given by

$$\begin{aligned}
\nabla \times \mathbf{H} &= \nabla \times (e^{i\mathbf{k}\cdot\mathbf{x}}\mathbf{u}_{\mathbf{k}}) \\
&= e^{i\mathbf{k}\cdot\mathbf{x}}(\nabla \times \mathbf{u}_{\mathbf{k}}) - \mathbf{u}_{\mathbf{k}} \times \nabla e^{i\mathbf{k}\cdot\mathbf{x}} \\
&= e^{i\mathbf{k}\cdot\mathbf{x}}(\nabla \times \mathbf{u}_{\mathbf{k}}) - \mathbf{u}_{\mathbf{k}} \times (i\mathbf{k}e^{i\mathbf{k}\cdot\mathbf{x}}) \\
&= e^{i\mathbf{k}\cdot\mathbf{x}}((\nabla + i\mathbf{k}) \times \mathbf{u}_{\mathbf{k}}).
\end{aligned}$$

This gives a new operator $\nabla_{\mathbf{k}} = \nabla + i\mathbf{k}$. Inserting ansatz (2.4) into the time harmonic Maxwell equations for the magnetic field (2.1) gives

$$\begin{aligned}
\nabla \times (\epsilon^{-1}\nabla \times \mathbf{H}) - \omega^2\mu\mathbf{H} &= \mathbf{J}, \\
\nabla \times (\epsilon^{-1}\nabla \times (e^{i\mathbf{k}\cdot\mathbf{x}}\mathbf{u}_{\mathbf{k}})) - \omega^2 e^{i\mathbf{k}\cdot\mathbf{x}}\mu\mathbf{u}_{\mathbf{k}} &= \mathbf{J}, \\
\nabla \times (\epsilon^{-1}e^{i\mathbf{k}\cdot\mathbf{x}}(\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}})) - \omega^2 e^{i\mathbf{k}\cdot\mathbf{x}}\mu\mathbf{u}_{\mathbf{k}} &= \mathbf{J}, \\
e^{i\mathbf{k}\cdot\mathbf{x}}\nabla_{\mathbf{k}} \times (\epsilon^{-1}\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}}) - \omega^2 e^{i\mathbf{k}\cdot\mathbf{x}}\mu\mathbf{u}_{\mathbf{k}} &= \mathbf{J}.
\end{aligned}$$

Since $e^{i\mathbf{k}\cdot\mathbf{x}} \neq 0$ for all $\mathbf{k}$ and all $\mathbf{x}$, the system that has to be solved can be written as

$$\nabla_{\mathbf{k}} \times (\epsilon^{-1}\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}}) - \omega^2\mu\mathbf{u}_{\mathbf{k}} = \tilde{\mathbf{J}} \qquad \text{in } \Omega, \tag{2.5}$$

where Ω is the primitive cell, $\mathbf{k}$ lies in the irreducible Brillouin zone and the boundary conditions are periodic. If one wants to solve the eigenvalue problem, one has to find $(\omega, \mathbf{u}_{\mathbf{k}}) \neq (0, \mathbf{0})$ such that

$$\nabla_{\mathbf{k}} \times (\epsilon^{-1}\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}}) = \omega^2\mu\mathbf{u}_{\mathbf{k}} \qquad \text{in } \Omega.$$

First, however, an auxiliary problem will be solved, namely the time-harmonic problem on a bounded domain $\Omega \subsetneq \mathbb{R}^3$:

$$\begin{aligned}
\nabla \times \epsilon^{-1}\nabla \times \mathbf{H} - \omega^2\mu\mathbf{H} &= \mathbf{J} && \text{in } \Omega, \\
\mathbf{n} \times \mathbf{H} &= \mathbf{g} && \text{on } \Gamma_n \subset \partial\Omega,
\end{aligned}$$

and with periodic boundary conditions on $\Gamma_p = \partial\Omega \setminus \Gamma_n$. For both problems, an implementation is given for the homogeneous domain $\Omega = [0, 1]^3$ with $\epsilon = \mu = 1$.

Chapter 3

DG discretization of the time-harmonic Maxwell equations

In this chapter, it will be described how the time-harmonic Maxwell equations in a cavity will be discretized using an interior penalty discontinuous Galerkin (IP-DG) finite element method. The first section is about the domain of computation and the reference tetrahedron, which is crucial in computing the integrals resulting from the discretization. After that, the basis functions are discussed. Then the discontinuous Galerkin scheme will be derived, followed by some details on the implementation of this scheme. Finally, some results for a homogeneous medium will be given to show that this scheme has indeed the desired order of convergence.

3.1 Domain and tessellation

In this section, some notation concerning the domain and the tessellation will be given. Some extra attention is given to the reference tetrahedron, since almost all computations take place on this element.

As mentioned before, the domain is some $\Omega \subsetneq \mathbb{R}^3$ with boundary $\partial\Omega$. Consider a tessellation $\mathcal{T}_h$ that partitions Ω into a set of tetrahedra $\{K\}$. Assume that the mesh is shape-regular and that each tetrahedron is straight-sided. Let $\mathcal{F}_h, \mathcal{F}_h^i$ and $\mathcal{F}_h^n$ denote the set of all faces, the interior faces and the boundary faces respectively, where faces on a boundary with periodic boundary conditions are seen as interior faces.

It will be useful to define the tangential jump and average of a function $\mathbf{u}$ across a face f with adjacent elements K_L and K_R. They are defined respectively as

$$[\![\mathbf{u}]\!]_T = \mathbf{n}_L \times \mathbf{u}_L + \mathbf{n}_R \times \mathbf{u}_R, \qquad \{\mathbf{u}\} = (\mathbf{u}_L + \mathbf{u}_R)/2,$$

in which $\mathbf{u}_L$ and $\mathbf{u}_R$ are the values of the trace of $\mathbf{u}$ at ∂K_L and ∂K_R, respectively, and $\mathbf{n}_L$ and $\mathbf{n}_R$ are the outward pointing normal vectors of respectively K_L and K_R at the face f. On the boundary faces, this notation denotes

$$[\![\mathbf{u}]\!]_T = \mathbf{n} \times \mathbf{u}, \qquad \{\mathbf{u}\} = \mathbf{u}.$$

The reference element

To simplify the computations of element and face integrals, it is useful to define a reference element on which these integrals can be computed with standard quadrature rules. Define this reference element as the tetrahedron

$$\hat{K} = \{\boldsymbol{\xi} \in \mathbb{R}^3; -1 \leq \xi_1, \xi_2, \xi_3 \leq 1 \text{ and } \xi_1 + \xi_2 + \xi_3 \leq -1\}.$$

Its vertices are $\mathbf{v}_1 = (-1, 1, -1), \mathbf{v}_2 = (-1, -1, -1), \mathbf{v}_3 = (1, -1, -1)$ and $\mathbf{v}_4 = (-1, -1, 1)$, see Figure 3.1. For convenience, the numbering of the edges and the faces of a tetrahedron is given in Table 3.1. Note that all edges and faces of the reference tetrahedron are denoted with a hat.

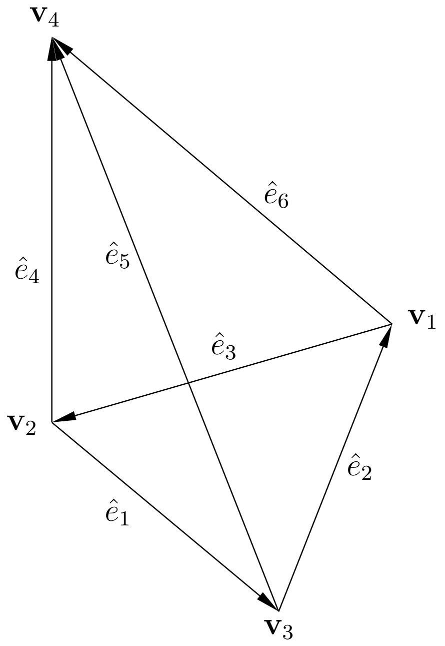


Figure 3.1: The reference tetrahedron

Edge/face	vertices
$\hat{e}_1$	$\mathbf{v}_2, \mathbf{v}_3$
$\hat{e}_2$	$\mathbf{v}_3, \mathbf{v}_1$
$\hat{e}_3$	$\mathbf{v}_1, \mathbf{v}_2$
$\hat{e}_4$	$\mathbf{v}_2, \mathbf{v}_4$
$\hat{e}_5$	$\mathbf{v}_3, \mathbf{v}_4$
$\hat{e}_6$	$\mathbf{v}_1, \mathbf{v}_4$
$\hat{f}_1$	$\mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4$
$\hat{f}_2$	$\mathbf{v}_1, \mathbf{v}_3, \mathbf{v}_4$
$\hat{f}_3$	$\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_4$
$\hat{f}_4$	$\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$

Table 3.1: Numbering of edges and faces of a tetrahedron, numbering indicates direction along edges

To map the reference tetrahedron to an arbitrary non-degenerate, straight-sided tetrahedron K, define the map $F_K : \hat{K} \to K$ [22] as

$$F_K(\boldsymbol{\xi}) = \sum_{i=1}^{4} \mathbf{x}_i \lambda_i(\boldsymbol{\xi}), \tag{3.1}$$

with $\mathbf{x}_i$ being the coordinates of the vertices of K and λ_i the barycentric coordinates of the vertices of the reference tetrahedron. The details of transforming a function on the reference tetrahedron to a function on an arbitrary straight-sided, convex, non-degenerate tetrahedron can be found in Appendix A.

3.2 Function spaces and basis functions

In this section, the function space $\mathbf{H}(\text{curl})$ is introduced, since this is the space of which the field must be an element of. Furthermore, we will define two different types of curl-conforming finite elements that can be used: the Nédélec elements described in [19] and hierarchic elements as described by Ainsworth in [2]. The advantage of the hierarchic elements is that it is much easier to determine the basis functions and so they are easier to implement. On the other hand, the Nédélec element will include less degrees of freedom, which is beneficial for the computational time.

3.2.1 The space $\mathbf{H}(\text{curl})$

In Section 3.3.3 it will turn out that the DG scheme includes an integration over the curl of the magnetic field, so a natural choice is to seek an approximation of the magnetic field in the space $\mathbf{H}(\text{curl};\Omega)$. First define the space $(L^2(\Omega))^3$ as

$$(L^2(\Omega))^3 = \left\{ \mathbf{u} = (u_1(\mathbf{x}), u_2(\mathbf{x}), u_3(\mathbf{x}))^T \ \mid \ \int_\Omega |\mathbf{u}\cdot\mathbf{u}| \ dx < \infty \right\},$$

with its norm given by

$$\|\mathbf{u}\|_{L^2(\Omega)} = \left(\int_\Omega |\mathbf{u}\cdot\mathbf{u}| \ dx \right)^{1/2}$$

The space $\mathbf{H}(\text{curl};\Omega)$ is defined as

$$\mathbf{H}(\text{curl};\Omega) = \left\{ \mathbf{u} \in (L^2(\Omega))^3 \mid \nabla \times \mathbf{u} \in (L^2(\Omega))^3 \right\}. \tag{3.2}$$

If there is no confusion possible, we may write $\mathbf{H}(\text{curl})$ instead of $\mathbf{H}(\text{curl};\Omega)$. Its norm is given by

$$\|\mathbf{u}\|_{\mathbf{H}(\text{curl};\Omega)} = \left(\|\mathbf{u}\|^2_{L^2(\Omega)} + \|\nabla \times \mathbf{u}\|^2_{L^2(\Omega)} \right)^{1/2}. \tag{3.3}$$

Because a discontinuous Galerkin finite element method is used, we would like to define a norm which incorporates the jumps of the tangential components of the magnetic field. This leads to the DG-norm

$$\|\mathbf{u}\|_{DG} = \left(\|\mathbf{u}\|^2_{L^2(\Omega)} + \|\nabla \times \mathbf{u}\|^2_{L^2(\Omega)} + \|\mathbf{h}^{-\frac{1}{2}} [\![\mathbf{u}]\!]_T\|^2_{L^2(\mathcal{F}_h)} \right)^{1/2},$$

in which $\mathbf{h} = h_f$ is the diameter of the circumscribed circle of face f.

Both types of finite elements discussed below are $\mathbf{H}(\text{curl})$-conforming, which means the global finite element space is a subspace of $\mathbf{H}(\text{curl})$ if one would use a standard finite element setting (see [2] and [18]). However, we use a discontinuous Galerkin method, which means that we do not require either the normal or tangential components to be continuous. Nonetheless, it is still beneficial to use $\mathbf{H}(\text{curl})$-conforming finite elements, for it implies that the average across any face is $\mathbf{H}(\text{curl})$-conforming and the stiffness matrix is sparser for higher-order polynomials than with standard $\mathbf{H}^1$-conforming finite elements ([21]).

3.2.2 Nédélec elements

In this section, the finite elements as defined by Nédélec in [19] will be described. They are defined on the reference domain $\hat{K}$, which is the tetrahedron with vertices $\mathbf{v}_1 = (-1,1,-1), \mathbf{v}_2 = (-1,-1,-1), \mathbf{v}_3 = (1,-1,-1)$ and $\mathbf{v}_4 = (-1,-1,1)$, see Section 3.1. Let $P_p(K)$ be the function space which consists of all real polynomials up to degree p. Then the finite element space is given by

$$R_p(\hat{K}) = (P_{p-1}(\hat{K}))^3 \oplus S_p(\hat{K}),$$

with

$$S_p(\hat{K}) = \{\mathbf{p} \in (P_p(\hat{K}))^3 \mid \mathbf{p} \text{ is homogeneous and } \boldsymbol{\xi}\cdot\mathbf{p} = 0\}.$$

There are three types of degrees of freedom, namely those associated with edges $\hat{e}$ of $\hat{K}$, those associated with faces $\hat{f}$ of $\hat{K}$ and finally degrees of freedom associated with $\hat{K}$ itself. The total number of degrees of freedom equals $\dim(R_p) = \frac{1}{2}(p+3)(p+2)p$.
Let a degree of freedom be given by $M_i(\boldsymbol{\phi})$ and the basis elements of R_p by $\{\boldsymbol{\phi}_j\}$. Then one can find the basis functions by solving $M_i(\boldsymbol{\phi}_j) = \delta_{ij}$, in which δ_{ij} is the Kronecker delta function.
First, it is described how one can find a basis for S_p, then the degrees of freedom are given and the section is concluded with an example of these concepts in case of $p = 1$. The basis functions for $p = 2$ are given in Appendix B.

Finding a basis for S_p

The first challenge when using Nédélec elements is to find a basis for the function space S_p. We will follow the approach of Gopalakrishnan in [9] for a 3-dimensional space.
Let $\boldsymbol{\alpha} \equiv (\alpha_1, \alpha_2, \alpha_3)$, define $\boldsymbol{\xi}^{\boldsymbol{\alpha}} = \xi_1^{\alpha_1}\xi_2^{\alpha_2}\xi_3^{\alpha_3}$. Furthermore, introduce the following notation:

$$\mathcal{I}(p) = \{\boldsymbol{\alpha} : 0 \leq \alpha_i \in \mathbb{Z}, \sum_{i=1}^{3} \alpha_i = p\},$$
$$\mathcal{I}_j(p) = \{\boldsymbol{\beta} \in \mathcal{I}(p) : \text{exactly } j \text{ components of } \boldsymbol{\beta} \text{ are non-zero.}\}.$$

For a $\boldsymbol{\beta} \in \mathcal{I}_j(p+1)$, define the permutation ℓ such that $\beta_{\ell(m)} > 0$ for $m \leq j$ and $\beta_{\ell(m)} = 0$ for $m > j$. Then we can define for $j > 1$

$$\mathcal{B}_p^{\boldsymbol{\beta}} = \left\{\boldsymbol{\xi}^{\boldsymbol{\beta}-\mathbf{e}_{\ell(m)}}\mathbf{e}_{\ell(m)} - \boldsymbol{\xi}^{\boldsymbol{\beta}-\mathbf{e}_{\ell(m+1)}}\mathbf{e}_{\ell(m+1)} : m = 1, 2, ..., j-1\right\},$$
$$\mathcal{B}_p^{(j)} = \bigcup_{\boldsymbol{\beta}\in\mathcal{I}_j(p+1)} \mathcal{B}_p^{\boldsymbol{\beta}},$$

in which $\mathbf{e}_1, \mathbf{e}_2$ and $\mathbf{e}_3$ are the standard basis vectors in $\mathbb{R}^3$. Then a basis of S_p is given by $\mathcal{B}_p \equiv \mathcal{B}_p^{(2)} \bigcup \mathcal{B}_p^{(3)}$. The implementation of these basis elements is straightforward: first all combinations $(\beta_1, \beta_2, \beta_3)$ are determined in which $\sum_i \alpha_i = p+1$ and either 2 or 3 α_i are non-zero. For all these combinations, compute $\mathcal{B}_p^{\boldsymbol{\beta}}$. These are the basis elements of S_p.

Degrees of freedom

For Nédélec elements, there are three types of degrees of freedom. The degrees of freedom of the first type are those associated with edges. They are given by

$$M_{\hat{e}}(\boldsymbol{\varphi}) = \left\{\int_{\hat{e}} \boldsymbol{\varphi} \cdot \hat{\boldsymbol{\tau}} q \; d\hat{s} \quad \text{for each edge } \hat{e} \text{ of } \hat{K} \text{ and for all } q \in P_{p-1}(\hat{e})\right\},$$

in which $\hat{\boldsymbol{\tau}}$ is the tangent unit vector associated with edge $\hat{e}$ in the direction of the edge as given in Table 3.1.
The second type is associated with faces. The degrees of freedom are given by

$$M_{\hat{f}}(\boldsymbol{\varphi}) = \left\{\int_{\hat{f}} (\mathbf{n} \times \boldsymbol{\varphi}) \cdot r(\mathbf{q}) \; d\hat{A} \quad \text{for each face } \hat{f} \text{ of } \hat{K} \text{ and for all } \mathbf{q} \in (P_{p-2}(\hat{T}))^2\right\},$$

in which $\hat{T}$ is the reference triangle with vertices $(-1,-1), (1,-1)$ and $(-1,1)$ and $r : \hat{T} \to \hat{f}$ is the transformation from the reference triangle to face $\hat{f}$ of the reference tetrahedron, see

Appendix A.
Finally, there are the degrees of freedom associated with the reference tetrahedron itself:

$$M_{\hat{K}}(\boldsymbol{\varphi}) = \left\{ \int_{\hat{K}} \boldsymbol{\varphi} \cdot \mathbf{q} \, d\xi \quad \text{for all } \mathbf{q} \in (P_{p-3}(\hat{K}))^3 \right\}.$$

In total there are $6p$ degrees of freedom associated with edges, $4p(p-1)$ degrees of freedom associated with faces and $p(p-1)(p-2)/2$ degrees of freedom associated with the reference tetrahedron. In total this gives $p(p+2)(p+3)/2$ degrees of freedom, which is also the dimension of the space $R_p(\hat{K})$. In Table 3.2 one can see which degrees of freedom should used when and how many there are.

Function type	**Use when**	**Number of degrees of freedom**
Edge	$p \geq 1$	$6p$
Face	$p \geq 2$	$4p(p-1)$
Interior	$p \geq 3$	$p(p-1)(p-2)/2$

Table 3.2: Number and type of degrees of freedom needed for Nédélec curl-conforming elements

Example

As an example, the above theory will be shown for the case $p = 1$.
To find a basis for S_1, first determine an expression for $\mathcal{B}_1^{(2)}$ and $\mathcal{B}_1^{(3)}$:

$$\begin{aligned}
\mathcal{B}_1^{(2)} &= \bigcup_{\boldsymbol{\beta} \in \mathcal{I}_2(2)} \mathcal{B}_1^{\boldsymbol{\beta}} \\
&= \mathcal{B}_1^{(1,1,0)} \cup \mathcal{B}_1^{(1,0,1)} \cup \mathcal{B}_1^{(0,1,1)} \\
&= \{\boldsymbol{\xi}^{(0,1,0)}\mathbf{e}_1 - \boldsymbol{\xi}^{(1,0,0)}\mathbf{e}_2\} \cup \{\boldsymbol{\xi}^{(0,0,1)}\mathbf{e}_1 - \boldsymbol{\xi}^{(1,0,0)}\mathbf{e}_3\} \cup \{\boldsymbol{\xi}^{(0,0,1)}\mathbf{e}_2 - \boldsymbol{\xi}^{(0,1,0)}\mathbf{e}_3\} \\
&= \left\{ \begin{pmatrix} \xi_2 \\ -\xi_1 \\ 0 \end{pmatrix}, \begin{pmatrix} \xi_3 \\ 0 \\ -\xi_1 \end{pmatrix}, \begin{pmatrix} 0 \\ \xi_3 \\ -\xi_2 \end{pmatrix} \right\}.
\end{aligned}$$

$\mathcal{B}_1^{(3)}$ is empty, since it is impossible to sum 3 positive integers and get a value of 2. So a basis for S_1 is given by $\mathcal{B}_1^{(2)}$. This leads to a basis for R_1 as follows:

$$\begin{aligned}
R_1 &= S_1 \oplus (P_0)^3 \\
&= \text{span} \left\{ \begin{pmatrix} \xi_2 \\ -\xi_1 \\ 0 \end{pmatrix}, \begin{pmatrix} \xi_3 \\ 0 \\ -\xi_1 \end{pmatrix} \begin{pmatrix} 0 \\ \xi_3 \\ -\xi_2 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}.
\end{aligned}$$

Substituting this basis into the degrees of freedom associated with edges of the reference tetrahedron and setting $M_{e_i}(\boldsymbol{\psi}_j) = \delta_{ij}$, one arrives at the system

$$\begin{pmatrix}
-2 & -2 & 0 & 2 & 0 & 0 \\
0 & 2 & -2 & -2 & 2 & 0 \\
-2 & 0 & 2 & 0 & -2 & 0 \\
0 & 2 & 2 & 0 & 0 & 2 \\
2 & 0 & 2 & -2 & 0 & 2 \\
-2 & 2 & 0 & 0 & -2 & 2
\end{pmatrix}
\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} = \mathbf{e}_j.$$

Solving this system leads to the functions

$$\boldsymbol{\psi}_{\hat{e}} = \lambda_i \nabla \lambda_j - \lambda_j \nabla \lambda_i,$$

in which $\hat{e}$ is the directed edge $\hat{e} = (\mathbf{v}_j, \mathbf{v}_i)$ and the barycentric coordinates λ_i are defined such that $\lambda_i(\mathbf{v}_j) = \delta_{ij}$.
Finding higher order basis functions can be done similarly by first finding a basis for S_p, computing the integrals described by the degrees of freedom and then finding the inverse of the resulting matrix.
The second order basis functions are given in Appendix B.

3.2.3 Hierarchic curl-conforming elements

In this section, the hierarchic curl-conforming elements of Ainsworth [2] will be given. The finite element space is the space

$$\hat{\mathbf{X}}_p^{\text{curl}} = \left(\mathbb{P}_p(\hat{K})\right)^3.$$

This space has dimension $(p+1)(p+2)(p+3)/2$.
The basis functions are again defined on the reference element, see Section 3.1. Before the expressions for the basis functions are given, some auxiliary functions will be defined.
The first building blocks are the barycentric coordinates. They are defined as

$$\begin{aligned}
\lambda_1 &= \frac{\xi_2 + 1}{2}, \\
\lambda_2 &= -\frac{1 + \xi_1 + \xi_2 + \xi_3}{2}, \\
\lambda_3 &= \frac{\xi_1 + 1}{2}, \\
\lambda_4 &= \frac{\xi_3 + 1}{2}.
\end{aligned}$$

Note that these functions are chosen such that λ_j disappears on face $\hat{f}_j$ for each j.
There will also be need of a scalar orthogonal basis in 1D. In this case, Legendre polynomials are chosen. They can be defined recursively as

$$\begin{aligned}
L_0(\xi) &= 1, \\
L_1(\xi) &= \xi, \\
L_p(\xi) &= \frac{2p-1}{p}\xi \; L_{p-1}(\xi) - \frac{p-1}{p} L_{p-2}(\xi), \qquad p \geq 2.
\end{aligned}$$

The first four Legendre polynomials are given by

$$\begin{aligned}
L_0(\xi) &= 1, \\
L_1(\xi) &= \xi, \\
L_2(\xi) &= \frac{3}{2}\xi^2 - \frac{1}{2}, \\
L_3(\xi) &= \frac{1}{2}\xi(5\xi^2 - 3).
\end{aligned}$$

Lastly, we need unitary normal vectors for faces and tangential vectors for edges. They are defined respectively as

$$\hat{\mathbf{n}}_i = -\frac{\nabla \lambda_i}{|\nabla \lambda_i|}, \qquad i = 1, .., 4,$$

$$\hat{\boldsymbol{\tau}}_{\hat{e}_\ell} = \mathbf{v}_j - \mathbf{v}_i, \qquad l = 1, .., 6,$$

in which $\hat{e}_\ell = (\mathbf{v}_i, \mathbf{v}_j)$ is a directed edge as given in Table 3.1. Note that $\hat{\mathbf{n}}$ denotes the outward unit normal vector. Also, following [2], we do not normalize the tangential vectors. As will be seen later, we will need the curl of all basis functions. To compute these, we use the following identities from the vector calculus [10]:

$$\nabla \times (f\mathbf{A}) = f\nabla \times \mathbf{A} - \mathbf{A} \times \nabla f, \tag{3.4}$$

$$\nabla(fg) = f\nabla g + g\nabla f, \tag{3.5}$$

in which f and g are scalar functions and $\mathbf{A} \in \mathbb{R}^3$ is a vector function.

Now we can define the basis functions. These are divided into edge functions, edge-based face functions, face bubble functions, face-based interior functions and interior bubble functions. Each type will be discussed separately.

Edge functions

For every edge, there will be defined $p+1$ edge functions, with p the maximal polynomial order of the basis functions on the element. These are constructed such that their tangential components equal the Legendre polynomials (up to a minus sign) on the corresponding edges and the tangential components disappear on the other edges. For the edge $\hat{e} = (\mathbf{v}_A, \mathbf{v}_B)$ and $\lambda_A(\mathbf{v}_A) = \lambda_B(\mathbf{v}_B) = 1$, the edge functions are defined recursively as

$$\begin{aligned}
\boldsymbol{\psi}_0^{\hat{e}} &= \lambda_B \nabla\lambda_A - \lambda_A \nabla\lambda_B, \\
\boldsymbol{\psi}_1^{\hat{e}} &= \lambda_B \nabla\lambda_A + \lambda_A \nabla\lambda_B, \\
\boldsymbol{\psi}_l^{\hat{e}} &= \frac{2l-1}{l} L_{l-1}(\lambda_B - \lambda_A)\boldsymbol{\psi}_1^{\hat{e}} - \frac{l-1}{l} L_{l-2}(\lambda_B - \lambda_A)\boldsymbol{\psi}_0^{\hat{e}}, \qquad 2 \le l \le p-1.
\end{aligned}$$

The first three edge functions for edge $\hat{e}_1 = (\mathbf{v}_2, \mathbf{v}_3)$ are given by

$$\boldsymbol{\psi}_0^{\hat{e}_1} = \frac{1}{4}\begin{pmatrix} \xi_2 + \xi_3 \\ -1 - \xi_1 \\ -1 - \xi_1 \end{pmatrix},$$

$$\boldsymbol{\psi}_1^{\hat{e}_1} = \frac{1}{4}\begin{pmatrix} -2 - 2\xi_1 - \xi_2 - \xi_3 \\ -1 - \xi_1 \\ -1 - \xi_1 \end{pmatrix},$$

$$\boldsymbol{\psi}_2^{\hat{e}_1} = \frac{1}{16}\begin{pmatrix} (6 + 6\xi_1 + 3\xi_2 + 3\xi_3)(-2 - 2\xi_1 - \xi_2 - \xi_3) - 2\xi_2 - 2\xi_3 \\ (6 + 6\xi_1 + 3\xi_2 + 3\xi_3)(-\xi_1 - 1) + 2\xi_1 + 2 \\ (6 + 6\xi_1 + 3\xi_2 + 3\xi_3)(-\xi_1 - 1) + 2\xi_1 + 2 \end{pmatrix}.$$

The curl of the basis functions can be computed with (3.4) and (3.5) by observing that $\nabla \times \nabla\lambda_i = 0$ for all $i = 1, .., 4$. The expressions that follow are

$$\begin{aligned}
\nabla \times \boldsymbol{\psi}_0^{\hat{e}} &= 2(\nabla\lambda_B \times \nabla\lambda_A), \\
\nabla \times \boldsymbol{\psi}_1^{\hat{e}} &= \mathbf{0}, \\
\nabla \times \boldsymbol{\psi}_l^{\hat{e}} &= -\frac{2l-1}{l}\frac{dL_{l-1}}{d\xi}(\lambda_B - \lambda_A)\left((\lambda_B\nabla\lambda_A + \lambda_A\nabla\lambda_B) \times \nabla(\lambda_B - \lambda_A)\right) \\
&\quad - \frac{2(l-1)}{l} L_{l-2}(\lambda_B - \lambda_A)\left(\nabla\lambda_B \times \nabla\lambda_A\right) \\
&\quad + \frac{l-1}{l}\frac{dL_{l-2}}{d\xi}(\lambda_B - \lambda_A)\left((\lambda_B\nabla\lambda_A - \lambda_A\nabla\lambda_B) \times \nabla(\lambda_B - \lambda_A)\right).
\end{aligned}$$

Edge-based face functions

For every face, there will be defined $3(p-1)$ edge-based face functions, $p-1$ for every edge. For an edge $\hat{e}$, the corresponding basis functions $\boldsymbol{\psi}_j^{\hat{f},\hat{e}}$ are constructed such that they disappear on all edges except $\hat{e}$. Furthermore, the functions equal zero on all faces except for the corresponding face and the face which also has edge $\hat{e}$, on which the tangential components disappear. For the face $\hat{f} = (\mathbf{v}_A, \mathbf{v}_B, \mathbf{v}_C)$, the edge $\hat{e} = (\mathbf{v}_A, \mathbf{v}_B)$ and $\lambda_A(\mathbf{v}_A) = \lambda_B(\mathbf{v}_B) = \lambda_C(\mathbf{v}_C) = 1$, the functions are given by

$$\boldsymbol{\psi}_l^{\hat{f},\hat{e}} = \lambda_A \lambda_B L_{p-2}(\lambda_B - \lambda_A)\hat{\mathbf{n}}_C, \qquad 2 \leq l \leq p.$$

For example, take edge $\hat{e}_1$ of face $\hat{f}_4$. Then $\lambda_A = \lambda_2, \lambda_B = \lambda_3$ and $\lambda_C = \lambda_1$ and the first two edge-based face functions are given by

$$\begin{aligned}
\boldsymbol{\psi}_0^{\hat{f}_4,\hat{e}_1} &= -\lambda_2\lambda_3 \frac{\nabla\lambda_1}{|\nabla\lambda_1|} \\
&= -\frac{1}{4}\begin{pmatrix} 0 \\ (1+\xi_1+\xi_2+\xi_3)(1+\xi_2) \\ 0 \end{pmatrix}, \\
\boldsymbol{\psi}_1^{\hat{f}_4,\hat{e}_1} &= -\lambda_2\lambda_3 L_1(\lambda_3 - \lambda_2)\frac{\nabla\lambda_1}{|\nabla\lambda_1|} \\
&= -\frac{1}{4}\begin{pmatrix} 0 \\ (1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(2+2\xi_1+\xi_2+\xi_3) \\ 0 \end{pmatrix}.
\end{aligned}$$

The curl of these basis functions can easily be computed with (3.4) and (3.5) by observing that the normal vectors are constant vectors and thus $\nabla \times \hat{\mathbf{n}}_i = 0$ for all $i = 1, .., 4$. It follows that

$$\begin{aligned}
\nabla \times \boldsymbol{\psi}_l^{\hat{f},\hat{e}} = &- \hat{\mathbf{n}}_C \times \Big(\lambda_A\lambda_B \frac{dL_{l-2}}{d\xi}(\lambda_B - \lambda_A)\nabla(\lambda_B - \lambda_A) \\
&+ \lambda_A L_{l-2}(\lambda_B - \lambda_A)\nabla\lambda_B + \lambda_B L_{l-2}(\lambda_B - \lambda_A)\nabla\lambda_A\Big).
\end{aligned}$$

Face bubble functions

For every face, one can also define basis functions that are zero on all faces and edges, except for the interior of the corresponding face. These functions are called face bubble functions, and there are $(p-2)(p-1)$ of these functions for each face. For a face $\hat{f} = (\mathbf{v}_A, \mathbf{v}_B, \mathbf{v}_C)$ and $\lambda_A(\mathbf{v}_A) = \lambda_B(\mathbf{v}_B) = \lambda_C(\mathbf{v}_C) = 1$, the face bubble functions are given by

$$\begin{aligned}
\boldsymbol{\psi}_{n_1,n_2}^{\hat{f},1} &= \lambda_A\lambda_B\lambda_C L_{n_1}(\lambda_B - \lambda_A)L_{n_2}(\lambda_A - \lambda_C)\hat{\boldsymbol{\tau}}_{(A,B)}, \\
\boldsymbol{\psi}_{n_1,n_2}^{\hat{f},2} &= \lambda_A\lambda_B\lambda_C L_{n_1}(\lambda_B - \lambda_A)L_{n_2}(\lambda_A - \lambda_C)\hat{\boldsymbol{\tau}}_{(C,A)},
\end{aligned}$$

with $0 \leq n_1, n_2, n_1 + n_2 \leq p - 3$. For face $\hat{f}_4 = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$, the first four face bubble functions are given by

$$\begin{aligned}
\boldsymbol{\psi}_{0,0}^{\hat{f}_4,1} &= \lambda_2\lambda_3\lambda_1\hat{\boldsymbol{\tau}}_{\hat{e}_1} \\
&= -\frac{1}{8}\begin{pmatrix} (1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(1+\xi_2) \\ 0 \\ 0 \end{pmatrix},
\end{aligned}$$

$$\begin{aligned}\boldsymbol{\psi}_{0,0}^{\hat{f}_4,2} &= \lambda_2\lambda_3\lambda_1\hat{\boldsymbol{\tau}}_{\hat{e}_2}\\ &= \frac{1}{8}\begin{pmatrix}0\\(1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(1+\xi_2)\\0\end{pmatrix},\end{aligned}$$

$$\begin{aligned}\boldsymbol{\psi}_{1,0}^{\hat{f}_4,1} &= \lambda_2\lambda_3\lambda_1 L_1(\lambda_2-\lambda_1)\hat{\boldsymbol{\tau}}_{\hat{e}_1}\\ &= -\frac{1}{16}\begin{pmatrix}(1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(1+\xi_2)(2+2\xi_1+\xi_2+\xi_3)\\0\\0\end{pmatrix},\end{aligned}$$

$$\begin{aligned}\boldsymbol{\psi}_{0,1}^{\hat{f}_4,1} &= \lambda_2\lambda_3\lambda_1 L_1(\lambda_1-\lambda_3)\hat{\boldsymbol{\tau}}_{\hat{e}_1}\\ &= \frac{1}{16}\begin{pmatrix}(1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(1+\xi_2)(2+\xi_1+2\xi_2+\xi_3)\\0\\0\end{pmatrix}.\end{aligned}$$

The curl of these basis functions can be computed with (3.4) and (3.5) by observing that the tangent vectors are constant vectors and thus $\nabla\times\hat{\boldsymbol{\tau}}_{\hat{e}}=0$ for all $\hat{e}=\hat{e}_1,..,\hat{e}_6$. It follows that

$$\begin{aligned}\nabla\times\boldsymbol{\psi}_{n_1,n_2}^{\hat{f},1} =& -\hat{\boldsymbol{\tau}}_{(A,B)}\times\Big(\lambda_A\lambda_B\lambda_C L_{n_1}(\lambda_B-\lambda_A)\frac{dL_{n_2}}{d\xi}(\lambda_A-\lambda_C)\nabla(\lambda_A-\lambda_C)\\ &+\lambda_A\lambda_B\lambda_C L_{n_2}(\lambda_A-\lambda_C)\frac{dL_{n_1}}{d\xi}(\lambda_B-\lambda_A)\nabla(\lambda_B-\lambda_A)\\ &+\lambda_A\lambda_B L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_C\\ &+\lambda_A\lambda_C L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_B\\ &+\lambda_B\lambda_C L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_A\Big),\\ \nabla\times\boldsymbol{\psi}_{n_1,n_2}^{\hat{f},2} =& -\hat{\boldsymbol{\tau}}_{(C,A)}\times\Big(\lambda_A\lambda_B\lambda_C L_{n_1}(\lambda_B-\lambda_A)\frac{dL_{n_2}}{d\xi}(\lambda_A-\lambda_C)\nabla(\lambda_A-\lambda_C)\\ &+\lambda_A\lambda_B\lambda_C L_{n_2}(\lambda_A-\lambda_C)\frac{dL_{n_1}}{d\xi}(\lambda_B-\lambda_A)\nabla(\lambda_B-\lambda_A)\\ &+\lambda_A\lambda_B L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_C\\ &+\lambda_A\lambda_C L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_B\\ &+\lambda_B\lambda_C L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_A\Big).\end{aligned}$$

Face-based interior functions

Face-based interior functions must disappear on all faces except one, and on that face the tangential component must disappear. There are $(p-1)(p-2)/2$ of these functions for each face. For a face $\hat{f}=(\mathbf{v}_A,\mathbf{v}_B,\mathbf{v}_C)$ and $\lambda_A(\mathbf{v}_A)=\lambda_B(\mathbf{v}_B)=\lambda_C(\mathbf{v}_C)=1$, they are given by

$$\boldsymbol{\psi}_{n_1,n_2}^{b,\hat{f}}=\lambda_A\lambda_B\lambda_C L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\hat{\mathbf{n}}_{\hat{f}},$$

with $0\le n_1,n_2,n_1+n_2\le p-3$. For face $\hat{f}_4=(\mathbf{v}_1,\mathbf{v}_2,\mathbf{v}_3)$, the first three face-based interior functions are given by

$$\begin{aligned}\boldsymbol{\psi}_{0,0}^{b,\hat{f}_4} &= \lambda_2\lambda_3\lambda_1\hat{\mathbf{n}}_4\\ &= -\frac{1}{8}\begin{pmatrix}0\\0\\(1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(1+\xi_2)\end{pmatrix},\end{aligned}$$

$$
\begin{aligned}
\boldsymbol{\psi}^{b,\hat{f}_4}_{1,0} &= \lambda_2\lambda_3\lambda_1 L_1(\lambda_3-\lambda_2)\hat{\mathbf{n}}_4 \\
&= -\frac{1}{16}\begin{pmatrix} 0 \\ 0 \\ (1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(1+\xi_2)(2+2\xi_1+\xi_2+\xi_3) \end{pmatrix},
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\psi}^{b,\hat{f}_4}_{0,1} &= \lambda_2\lambda_3\lambda_1 L_1(\lambda_2-\lambda_1)\hat{\mathbf{n}}_4 \\
&= \frac{1}{16}\begin{pmatrix} 0 \\ 0 \\ (1+\xi_1+\xi_2+\xi_3)(1+\xi_1)(1+\xi_2)(2+\xi_1+2\xi_2+\xi_3) \end{pmatrix}.
\end{aligned}
$$

The curl of these basis functions can be computed with (3.4) and (3.5) by observing that the normal vectors are constant vectors and thus $\nabla\times\hat{\mathbf{n}}_i = 0$ for all $i = 1,..,4$. It follows that

$$
\begin{aligned}
\nabla\times\boldsymbol{\psi}^{b,\hat{f}}_{n_1,n_2} = &-\hat{\mathbf{n}}_{\hat{f}}\times\Big(\lambda_A\lambda_B\lambda_C L_{n_1}(\lambda_B-\lambda_A)\frac{dL_{n_2}}{d\xi}(\lambda_A-\lambda_C)\nabla(\lambda_A-\lambda_C) \\
&+\lambda_A\lambda_B\lambda_C L_{n_2}(\lambda_A-\lambda_C)\frac{dL_{n_1}}{d\xi}(\lambda_B-\lambda_A)\nabla(\lambda_B-\lambda_A) \\
&+\lambda_A\lambda_B L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_C \\
&+\lambda_A\lambda_C L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_B \\
&+\lambda_B\lambda_C L_{n_1}(\lambda_B-\lambda_A)L_{n_2}(\lambda_A-\lambda_C)\nabla\lambda_A\Big).
\end{aligned}
$$

Interior bubble functions

The last set of basis functions are the interior bubble functions, which are zero everywhere at the boundary of the reference tetrahedron. There are $(p-3)(p-2)(p-1)/2$ of these basis functions, and they are given by

$$
\boldsymbol{\psi}^{b}_{d,n_1,n_2,n_3} = \lambda_1\lambda_2\lambda_3\lambda_4 L_{n_1}(\lambda_1-\lambda_2)L_{n_2}(\lambda_3-\lambda_2)L_{n_3}(\lambda_4-\lambda_2)\mathbf{e}_d,
$$

in which $d\in 1,2,3$ and $0\le n_1,n_2,n_3,n_1+n_2+n_3\le p-4$.
The curl of these basis functions can be easily computed with (3.4) and (3.5) by observing that the standard basis vectors are constant vectors and thus $\nabla\times\mathbf{e}_d = 0$ for all $d = 1,..,3$. It follows that

$$
\begin{aligned}
\nabla\times\boldsymbol{\psi}^{b}_{d,n_1,n_2,n_3} = &-\mathbf{e}_d\times\Big(\lambda_1\lambda_2\lambda_3\lambda_4 L_{n_1}(\lambda_1-\lambda_2)L_{n_2}(\lambda_3-\lambda_2)\frac{dL_{n_3}}{d\xi}(\lambda_4-\lambda_2)\nabla(\lambda_4-\lambda_2) \\
&+\lambda_1\lambda_2\lambda_3\lambda_4 L_{n_1}(\lambda_1-\lambda_2)L_{n_3}(\lambda_4-\lambda_2)\frac{dL_{n_2}}{d\xi}(\lambda_3-\lambda_2)\nabla(\lambda_3-\lambda_2) \\
&+\lambda_1\lambda_2\lambda_3\lambda_4 L_{n_2}(\lambda_3-\lambda_2)L_{n_3}(\lambda_4-\lambda_2)\frac{dL_{n_1}}{d\xi}(\lambda_1-\lambda_2)\nabla(\lambda_1-\lambda_2) \\
&+\lambda_1\lambda_2\lambda_3 L_{n_1}(\lambda_1-\lambda_2)L_{n_2}(\lambda_3-\lambda_2)L_{n_3}(\lambda_4-\lambda_2)\nabla\lambda_4 \\
&+\lambda_1\lambda_2\lambda_4 L_{n_1}(\lambda_1-\lambda_2)L_{n_2}(\lambda_3-\lambda_2)L_{n_3}(\lambda_4-\lambda_2)\nabla\lambda_3 \\
&+\lambda_1\lambda_3\lambda_4 L_{n_1}(\lambda_1-\lambda_2)L_{n_2}(\lambda_3-\lambda_2)L_{n_3}(\lambda_4-\lambda_2)\nabla\lambda_2 \\
&+\lambda_2\lambda_3\lambda_4 L_{n_1}(\lambda_1-\lambda_2)L_{n_2}(\lambda_3-\lambda_2)L_{n_3}(\lambda_4-\lambda_2)\nabla\lambda_1\Big).
\end{aligned}
$$

Together, all the basis functions of this section form a hierarchic basis for the spaces $\hat{\mathbf{X}}^{\mathrm{curl}}_p$ for $p\in\mathbb{N}$ [2]. In Table 3.3 it is mentioned which functions are used when, and how many functions of that kind there are.

Function type	**Use when**	**Number of basis functions**
Edge	$p \geq 1$	$6(p+1)$
Edge-based face	$p \geq 2$	$12(p-1)$
Face bubble	$p \geq 3$	$4(p-2)(p-1)$
Face-based interior	$p \geq 3$	$2(p-2)(p-1)$
Interior bubble	$p \geq 4$	$(p-3)(p-2)(p-1)/2$

Table 3.3: Number and type of basis functions needed for hierarchic curl-conforming elements

3.3 Interior penalty discretization of the time-harmonic Maxwell equations

Consider the system

$$\nabla \times \epsilon^{-1} \nabla \times \mathbf{H} - \omega^2 \mu \mathbf{H} = \mathbf{J}, \qquad \text{in } \Omega, \tag{3.6}$$

$$\mathbf{n} \times \mathbf{H} = \mathbf{g}, \qquad \text{on } \Gamma_n \subseteq \partial\Omega, \tag{3.7}$$

and with periodic boundary conditions on $\Gamma_p = \partial\Omega \setminus \Gamma_n$. In this section, the system (3.6)-(3.7) will be discretized by an interior penalty discontinuous Galerkin (IP-DG) finite element method, analogously to the IP-DG discretization in [21]. First, the system will be written as a first order system. It will be integrated by parts over each element and summed over all elements. Then, with the help of properly defined lifting operators, the auxiliary variable can be eliminated. Finally, the IP-fluxes will be inserted to complete the IP-DG discretization.

3.3.1 Weak formulation

First, rewrite system (3.6) as a first order system:

$$\begin{aligned} \nabla \times \epsilon^{-1} \mathbf{q} - \omega^2 \mu \mathbf{H} &= \mathbf{J} && \text{in } \Omega, \\ \mathbf{q} &= \nabla \times \mathbf{H} && \text{in } \Omega, \end{aligned}$$

in which $\epsilon = \epsilon(\mathbf{x})$ and $\mu = \mu(\mathbf{x})$, so the material is assumed to be either homogeneous or inhomogeneous. Let Σ_h be the finite element space, so either

$$\Sigma_h = \left\{ \mathbf{u} \in \left(L^2(\Omega)\right)^3 \mid \mathbf{u} \text{ is a hierarchic basis function} \right\},$$

or

$$\Sigma_h = \left\{ \mathbf{u} \in \left(L^2(\Omega)\right)^3 \mid \mathbf{u} \text{ is a Nédélec basis function} \right\}.$$

Then, multiply both these equations with arbitrary test functions $\boldsymbol{\phi}, \boldsymbol{\pi} \in \Sigma_h$, integrate over each element in the domain and sum over the elements. The resulting system is

$$\begin{aligned} \int_\Omega (\nabla_h \times \epsilon^{-1} \mathbf{q}_h) \cdot \boldsymbol{\phi} \, dx - \omega^2 \mu \int_\Omega \mathbf{H}_h \cdot \boldsymbol{\phi} \, dx &= \int_\Omega \mathbf{J} \cdot \boldsymbol{\phi} \, dx, \\ \int_\Omega \mathbf{q}_h \cdot \boldsymbol{\pi} \, dx &= \int_\Omega (\nabla_h \times \mathbf{H}_h) \cdot \boldsymbol{\pi} \, dx, \end{aligned}$$

in which the operator ∇_h is defined as the element wise application of the operator ∇. Now use the following identities [10]:

$$\nabla \cdot (\mathbf{A} \times \mathbf{B}) = \mathbf{B} \cdot (\nabla \times \mathbf{A}) - \mathbf{A} \cdot (\nabla \times \mathbf{B}),$$

$$\begin{aligned}
&(\nabla \times \mathbf{A}) \cdot \mathbf{B} = \nabla \cdot (\mathbf{A} \times \mathbf{B}) + (\nabla \times \mathbf{B}) \cdot \mathbf{A},\\
&\mathbf{A} \cdot (\mathbf{B} \times \mathbf{C}) = \mathbf{B} \cdot (\mathbf{C} \times \mathbf{A}).
\end{aligned}$$

After applying Gauss' theorem the following integrals can be written as

$$\begin{aligned}
\int_\Omega \left(\nabla_h \times \epsilon^{-1}\mathbf{q}_h\right) \cdot \boldsymbol{\phi}\ dx &= \int_\Omega \nabla_h \cdot \left(\epsilon^{-1}\mathbf{q}_h \times \boldsymbol{\phi}\right) + \epsilon^{-1}\mathbf{q}_h \cdot (\nabla_h \times \boldsymbol{\phi})\ dx \\
&= \int_\Omega \epsilon^{-1}\mathbf{q}_h \cdot (\nabla_h \times \boldsymbol{\phi})\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} \left((\epsilon^{-1}\mathbf{q}_h)^* \times \boldsymbol{\phi}\right) \cdot d\mathbf{A} \\
&= \int_\Omega \epsilon^{-1}\mathbf{q}_h \cdot (\nabla_h \times \boldsymbol{\phi})\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} \left((\epsilon^{-1}\mathbf{q}_h)^* \times \boldsymbol{\phi}\right) \cdot \mathbf{n}\ dA \\
&= \int_\Omega \epsilon^{-1}\mathbf{q}_h \cdot (\nabla_h \times \boldsymbol{\phi})\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} \left(\mathbf{n} \times (\epsilon^{-1}\mathbf{q}_h)^*\right) \cdot \boldsymbol{\phi}\ dA,
\end{aligned}$$

$$\begin{aligned}
\int_\Omega (\nabla_h \times \mathbf{H}_h) \cdot \boldsymbol{\pi}\ dx &= \int_\Omega \nabla_h \cdot (\mathbf{H}_h \times \boldsymbol{\pi}) + \mathbf{H}_h \cdot (\nabla_h \times \boldsymbol{\pi})\ dx \\
&= \int_\Omega \mathbf{H}_h \cdot (\nabla_h \times \boldsymbol{\pi})\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} (\mathbf{H}_h^* \times \boldsymbol{\pi}) \cdot \mathbf{n}\ dA \\
&= \int_\Omega -\nabla_h \cdot (\mathbf{H}_h \times \boldsymbol{\pi}) + \boldsymbol{\pi} \cdot (\nabla_h \times \mathbf{H}_h)\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} (\mathbf{H}_h^* \times \boldsymbol{\pi}) \cdot \mathbf{n}\ dA \\
&= \int_\Omega (\nabla_h \times \mathbf{H}_h) \cdot \boldsymbol{\pi}\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} ((\mathbf{H}_h^* - \mathbf{H}_h) \times \boldsymbol{\pi}) \cdot \mathbf{n}\ dA \\
&= \int_\Omega (\nabla_h \times \mathbf{H}_h) \cdot \boldsymbol{\pi}\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} (\mathbf{n} \times (\mathbf{H}_h^* - \mathbf{H}_h)) \cdot \boldsymbol{\pi}\ dA.
\end{aligned}$$

In the expressions above, a * denotes the trace of the entity at the boundary of an element. Since the magnetic field $\mathbf{H}$ and auxiliary variable $\mathbf{q}$ may be discontinuous at faces, they do not have well-defined traces and one needs to define the value of these traces as a function later on. At this point, the weak formulation of (3.6) becomes:
Find $(\mathbf{H}_h, \mathbf{q}_h) \in \Sigma_h \times \Sigma_h$, such that for all functions $(\boldsymbol{\phi}, \boldsymbol{\pi}) \in \Sigma_h \times \Sigma_h$

$$\int_\Omega \epsilon^{-1}\mathbf{q}_h \cdot (\nabla_h \times \boldsymbol{\phi})\ dx - \omega^2\mu \int_\Omega \mathbf{H}_h \cdot \boldsymbol{\phi}\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} (\mathbf{n} \times (\epsilon^{-1}\mathbf{q}_h^*)) \cdot \boldsymbol{\phi}\ dA = \int_\Omega \mathbf{J} \cdot \boldsymbol{\phi}\ dx, \tag{3.8}$$

$$\int_\Omega (\nabla_h \times \mathbf{H}_h) \cdot \boldsymbol{\pi}\ dx + \sum_{K\in\mathcal{T}_h} \int_{\partial K} (\mathbf{n} \times (\mathbf{H}_h^* - \mathbf{H}_h)) \cdot \boldsymbol{\pi}\ dA = \int_\Omega \mathbf{q}_h \cdot \boldsymbol{\pi}\ dx. \tag{3.9}$$

3.3.2 Primal formulation

To get to the primal formulation, combine (3.8)-(3.9) with the identity

$$\sum_{K\in\mathcal{T}_h} \int_{\partial K} (\mathbf{n} \times \mathbf{u}) \cdot \mathbf{v}\ dA = -\int_{\mathcal{F}_h^i} \{\mathbf{u}\} \cdot [\![\mathbf{v}]\!]_T\ dA + \int_{\mathcal{F}_h^i} [\![\mathbf{u}]\!]_T \cdot \{\mathbf{v}\} dA + \int_{\mathcal{F}_h^n} (\mathbf{n} \times \mathbf{u}) \cdot \mathbf{v}\ dA,$$

which can be obtained by direct evaluation of the sums over the faces. Note that all the faces on a boundary with periodic boundary conditions are interior faces, and the other boundary faces have the boundary condition $\mathbf{n} \times \mathbf{H} = \mathbf{g}$. The results are the following integrals

$$\sum_{K\in\mathcal{T}_h} \int_{\partial K} \left(\mathbf{n} \times \epsilon^{-1}\mathbf{q}_h^*\right) \cdot \boldsymbol{\phi}\ dA = -\int_{\mathcal{F}_h^i} \left\{(\epsilon^{-1}\mathbf{q}_h)^*\right\} \cdot [\![\boldsymbol{\phi}]\!]_T\ dA + \int_{\mathcal{F}_h^i} [\![(\epsilon^{-1}\mathbf{q}_h)^*]\!]_T \cdot \{\boldsymbol{\phi}\}\ dA$$

$$+\int_{\mathcal{F}_h^n} \left(\mathbf{n}\times(\epsilon^{-1}\mathbf{q}_h)^*\right)\cdot\boldsymbol{\phi}\; dA,$$

$$\sum_{K\in\mathcal{T}_h}\int_{\partial K}(\mathbf{n}\times(\mathbf{H}_h^*-\mathbf{H}_h))\cdot\boldsymbol{\pi}\; dA = -\int_{\mathcal{F}_h^i}\{\mathbf{H}_h^*-\mathbf{H}_h\}\cdot[\![\boldsymbol{\pi}]\!]_T\; dA + \int_{\mathcal{F}_h^i}[\![\mathbf{H}_h^*-\mathbf{H}_h]\!]_T\cdot\{\boldsymbol{\pi}\}\; dA$$
$$+\int_{\mathcal{F}_h^n}(\mathbf{n}\times(\mathbf{H}_h^*-\mathbf{H}_h))\cdot\boldsymbol{\pi}\; dA.$$

Then the following forms are obtained:

$$\begin{aligned}
&\int_\Omega \epsilon^{-1}\mathbf{q}_h\cdot(\nabla_h\times\boldsymbol{\phi})\; dx - \omega^2\mu\int_\Omega \mathbf{H}_h\cdot\boldsymbol{\phi}\; dx - \int_{\mathcal{F}_h^i}\left\{(\epsilon^{-1}\mathbf{q}_h)^*\right\}\cdot[\![\boldsymbol{\phi}]\!]_T\; dA\\
&+\int_{\mathcal{F}_h^i}[\![(\epsilon^{-1}\mathbf{q}_h)^*]\!]_T\cdot\{\boldsymbol{\phi}\}\; dA + \int_{\mathcal{F}_h^n}\left(\mathbf{n}\times(\epsilon^{-1}\mathbf{q}_h)^*\right)\cdot\boldsymbol{\phi}\; dA = \int_\Omega \mathbf{J}\cdot\boldsymbol{\phi}\; dx,\\
&\int_\Omega(\nabla_h\times\mathbf{H}_h)\cdot\boldsymbol{\pi}\; dx - \int_{\mathcal{F}_h^i}\{\mathbf{H}_h^*-\mathbf{H}_h\}\cdot[\![\boldsymbol{\pi}]\!]_T\; dA + \int_{\mathcal{F}_h^i}\{\boldsymbol{\pi}\}\cdot[\![\mathbf{H}_h^*-\mathbf{H}_h]\!]_T\; dA\\
&+\int_{\mathcal{F}_h^n}(\mathbf{n}\times(\mathbf{H}_h^*-\mathbf{H}_h))\cdot\boldsymbol{\pi}\; dA = \int_\Omega \mathbf{q}_h\cdot\boldsymbol{\pi}\; dx.
\end{aligned}\tag{3.10}$$

In order to express $\mathbf{q}_h$ in terms of $\mathbf{H}_h$, define the lifting operators
$\mathcal{L}:[L^2(\mathcal{F}_h)]^3\to\Sigma_h$ and $\mathcal{R}:[L^2(\mathcal{F}_h)]^3\to\Sigma_h$ as

$$\begin{aligned}
\int_\Omega \mathcal{L}(\mathbf{u})\cdot\mathbf{v}\; dx &= \int_{\mathcal{F}_h^i}\mathbf{u}\cdot[\![\mathbf{v}]\!]_T\; dA, &&\forall\mathbf{v}\in\Sigma_h,\\
\int_\Omega \mathcal{R}(\mathbf{u})\cdot\mathbf{v}\; dx &= \int_{\mathcal{F}_h}\mathbf{u}\cdot\{\mathbf{v}\}\; dA, &&\forall\mathbf{v}\in\Sigma_h.
\end{aligned}$$

Note that by the definition of $\{\mathbf{u}\}$ and $[\![\mathbf{u}]\!]_T$ at the boundary, it holds that

$$\int_\Omega \mathcal{R}([\![\mathbf{H}_h^*-\mathbf{H}_h]\!]_T)\cdot\boldsymbol{\pi}\; dx = \int_{\mathcal{F}_h^i}\{\boldsymbol{\pi}\}\cdot[\![\mathbf{H}_h^*-\mathbf{H}_h]\!]_T\; dA + \int_{\mathcal{F}_h^n}(\mathbf{n}\times(\mathbf{H}_h^*-\mathbf{H}_h))\cdot\boldsymbol{\pi}\; dA.$$

Substitution of these lifting operators into (3.10) leads to the following expression for $\mathbf{q}_h$:

$$\mathbf{q}_h = \nabla_h\times\mathbf{H}_h - \mathcal{L}(\{\mathbf{H}_h^*-\mathbf{H}_h\}) + \mathcal{R}([\![\mathbf{H}_h^*-\mathbf{H}_h]\!]_T).$$

Substitution of this last expression then results in the formulation

$$\begin{aligned}
&\int_\Omega \epsilon^{-1}\left(\nabla_h\times\mathbf{H}_h - \mathcal{L}(\{\mathbf{H}_h^*-\mathbf{H}_h\}) + \mathcal{R}([\![\mathbf{H}_h^*-\mathbf{H}_h]\!]_T)\right)\cdot(\nabla_h\times\boldsymbol{\phi})\; dx\\
&-\omega^2\mu\int_\Omega \mathbf{H}_h\cdot\boldsymbol{\phi}\; dx - \int_{\mathcal{F}_h^i}\{(\epsilon^{-1}\mathbf{q}_h)^*\}\cdot[\![\boldsymbol{\phi}]\!]_T\; dA + \int_{\mathcal{F}_h^i}[\![(\epsilon^{-1}\mathbf{q}_h)^*]\!]_T\cdot\{\boldsymbol{\phi}\}\; dA\\
&+\int_{\mathcal{F}_h^n}\left(\mathbf{n}\times(\epsilon^{-1}\mathbf{q}_h)^*\right)\cdot\boldsymbol{\phi}\; dA = \int_\Omega \mathbf{J}\cdot\boldsymbol{\phi}\; dx.
\end{aligned}$$

Finally, we arrive at the primal formulation:
Find $\mathbf{H}_h\in\Sigma_h$, such that for all $\phi\in\Sigma_h$,

$$B(\mathbf{H}_h,\boldsymbol{\phi}) = (\mathbf{J},\boldsymbol{\phi})_\Omega,$$

in which $(\cdot,\cdot)_\Omega$ denotes the standard inner product in $[L^2(\Omega)]^2$ and the bilinear form $B(\mathbf{H}_h,\boldsymbol{\phi})$ is given by

$$
\begin{aligned}
B(\mathbf{H}_h, \boldsymbol{\phi}) = &\int_\Omega \epsilon^{-1}(\nabla_h \times \mathbf{H}_h) \cdot (\nabla_h \times \boldsymbol{\phi})\ dx - \omega^2 \mu \int_\Omega \mathbf{H}_h \cdot \boldsymbol{\phi}\ dx \\
&- \int_{\mathcal{F}_h^i} \{\mathbf{H}_h^* - \mathbf{H}_h\} \cdot [\![\epsilon^{-1}\nabla_h \times \boldsymbol{\phi}]\!]_T\ dA + \int_{\mathcal{F}_h^i} [\![\mathbf{H}_h^* - \mathbf{H}_h]\!]_T \cdot \{\epsilon^{-1}\nabla_h \times \boldsymbol{\phi}\}\ dA \\
&- \int_{\mathcal{F}_h^i} \{(\epsilon^{-1}\mathbf{q}_h)^*\} \cdot [\![\boldsymbol{\phi}]\!]_T\ dA + \int_{\mathcal{F}_h^i} [\![(\epsilon^{-1}\mathbf{q}_h)^*]\!]_T \cdot \{\boldsymbol{\phi}\}\ dA \qquad (3.11) \\
&+ \int_{\mathcal{F}_h^n} \left(\mathbf{n} \times (\epsilon^{-1}\mathbf{q}_h)^*\right) \cdot \boldsymbol{\phi}\ dA + \int_{\mathcal{F}_h^n} \left(\mathbf{n} \times (\mathbf{H}_h^* - \mathbf{H}_h)\right) \cdot (\nabla_h \times \boldsymbol{\phi})\ dA.
\end{aligned}
$$

In this expression, the numerical fluxes $\mathbf{q}_h^*$ and $\mathbf{H}_h^*$ still have to be chosen.

3.3.3 Numerical fluxes

In order to complete the discontinuous Galerkin discretization, we still need to define the numerical fluxes. For the interior penalty method, choose the numerical fluxes $\mathbf{q}_h^*$ and $\mathbf{H}_h^*$ on each interior face and face with a periodic boundary condition, $f \in \mathcal{F}_h^i$, as

$$
\begin{aligned}
\mathbf{H}_h^* &= \{\mathbf{H}_h\}, \\
(\epsilon^{-1}\mathbf{q}_h)^* &= \{\epsilon^{-1}\nabla_h \times \mathbf{H}_h\} - \alpha_f [\![\epsilon^{-1}\mathbf{H}_h]\!]_T,
\end{aligned}
$$

in which α_f is a face-dependent penalty parameter, which depends on the mesh size and polynomial order of the finite element space. Optimal choices for this parameter are given in [21].

The interior face integrals in (3.11) now equal

$$
\begin{aligned}
&\int_{\mathcal{F}_h^i} \{\mathbf{H}_h^* - \mathbf{H}_h\} \cdot [\![\epsilon^{-1}\nabla_h \times \boldsymbol{\phi}]\!]_T\ dA = 0, \\
&\int_{\mathcal{F}_h^i} [\![\mathbf{H}_h^* - \mathbf{H}_h]\!]_T \cdot \{\epsilon^{-1}\nabla_h \times \boldsymbol{\phi}\}\ dA = -\int_{\mathcal{F}_h^i} [\![\mathbf{H}_h]\!]_T \cdot \{\epsilon^{-1}\nabla_h \times \boldsymbol{\phi}\}\ dA,
\end{aligned}
$$

$$
\begin{aligned}
&\int_{\mathcal{F}_h^i} \{(\epsilon^{-1}\mathbf{q}_h)^*\} \cdot [\![\boldsymbol{\phi}]\!]_T\ dA = \int_{\mathcal{F}_h^i} \{\epsilon^{-1}\nabla_h \times \mathbf{H}_h\} \cdot [\![\boldsymbol{\phi}]\!]_T\ dA - \int_{\mathcal{F}_h^i} \alpha_f [\![\epsilon^{-1}\mathbf{H}_h]\!]_T \cdot [\![\boldsymbol{\phi}]\!]_T\ dA, \\
&\int_{\mathcal{F}_h^i} [\![(\epsilon^{-1}\mathbf{q}_h)^*]\!]_T \cdot \{\boldsymbol{\phi}\}\ dA = 0.
\end{aligned}
$$

At boundaries with boundary conditions $\mathbf{n} \times \mathbf{H} = \mathbf{g}$, choose the numerical flux as

$$
\begin{aligned}
\mathbf{n} \times \mathbf{H}_h^* &= \mathbf{g}, \\
(\epsilon^{-1}\mathbf{q}_h)^* &= \epsilon^{-1}\nabla_h \times \mathbf{H}_h - \alpha_f \epsilon^{-1}(\mathbf{n} \times \mathbf{H}_h) + \alpha_f \epsilon^{-1}\mathbf{g}.
\end{aligned}
$$

The boundary face integrals in (3.11) are now equal to

$$
\begin{aligned}
\int_{\mathcal{F}_h^n} \left(\mathbf{n} \times (\epsilon^{-1}\mathbf{q}_h)^*\right) \cdot \boldsymbol{\phi}\ dA = &-\int_{\mathcal{F}_h^n} (\mathbf{n} \times \boldsymbol{\phi}) \cdot \epsilon^{-1} (\nabla_h \times \mathbf{H}_h)\ dA \\
&+ \int_{\mathcal{F}_h^n} \alpha_f (\mathbf{n} \times \boldsymbol{\phi}) \cdot \epsilon^{-1} (\mathbf{n} \times \mathbf{H}_h)\ dA \\
&- \int_{\mathcal{F}_h^n} \alpha_f \epsilon^{-1} \mathbf{g} \cdot (\mathbf{n} \times \boldsymbol{\phi})\ dA,
\end{aligned}
$$

$$\int_{\mathcal{F}_h^n} (\mathbf{n} \times (\mathbf{H}_h^* - \mathbf{H}_h)) \cdot (\nabla_h \times \boldsymbol{\phi})\ dA = \int_{\mathcal{F}_h^n} (\boldsymbol{g} - (\mathbf{n} \times \mathbf{H}_h)) \cdot (\nabla_h \times \boldsymbol{\phi})\ dA.$$

The interior penalty discontinuous Galerkin method is now given by:
Find $\mathbf{H}_h \in \Sigma_h$, such that for all $\boldsymbol{\phi} \in \Sigma_h$,

$$B^{IP}(\mathbf{H}_h, \boldsymbol{\phi}) = \mathcal{J}^{IP}(\boldsymbol{\phi}), \tag{3.12}$$

in which the bilinear form $B^{IP}(\mathbf{H}_h, \boldsymbol{\phi})$ is given by

$$\begin{aligned} B^{IP}(\mathbf{H}_h, \boldsymbol{\phi}) = & \int_\Omega \epsilon^{-1}(\nabla_h \times \mathbf{H}_h) \cdot (\nabla_h \times \boldsymbol{\phi})\ dx - \omega^2 \int_\Omega \mu \mathbf{H}_h \cdot \boldsymbol{\phi}\ dx \\ & - \int_{\mathcal{F}_h} [\![\mathbf{H}_h]\!]_T \cdot \{\epsilon^{-1}\nabla_h \times \boldsymbol{\phi}\}\ dA - \int_{\mathcal{F}_h} \{\epsilon^{-1}\nabla_h \times \mathbf{H}_h\} \cdot [\![\boldsymbol{\phi}]\!]_T\ dA \\ & + \int_{\mathcal{F}_h} \alpha_f [\![\epsilon^{-1}\mathbf{H}_h]\!]_T \cdot [\![\boldsymbol{\phi}]\!]_T\ dA \end{aligned}$$

and the linear form $\mathcal{J}^{IP}(\boldsymbol{\phi})$ by

$$\begin{aligned} \mathcal{J}^{IP}(\boldsymbol{\phi}) = & \int_\Omega \mathbf{J} \cdot \boldsymbol{\phi}\ dx + \int_{\mathcal{F}_h^n} \alpha_f \epsilon^{-1} \mathbf{g} \cdot (\mathbf{n} \times \boldsymbol{\phi})\ dA \\ & - \int_{\mathcal{F}_h^n} \mathbf{g} \cdot (\nabla_h \times \boldsymbol{\phi})\ dA. \end{aligned}$$

3.4 Implementation

In this section, it will be discussed how the IP-DG scheme of last section can be implemented for a domain $\Omega = [0,1]^3$. Let either be $\Gamma_n = \partial\Omega$ or $\Gamma_n = \emptyset$, so the boundary conditions are either all of the type $\mathbf{n} \times \mathbf{H} = \mathbf{g}$ or there are only periodic boundary conditions.
To implement the IP-DG scheme, the following steps should be taken:

1. Make a tessellation of the domain Ω,
2. Convert the integrals of (3.12) to a linear system,
3. Solve the resulting linear system.

These points will be treated in the rest of this section. At the end of the section, a more detailed overview of the algorithm will be given.

3.4.1 Construction of the mesh

For the basic tests of the numerical discretization we first consider a cubic domain of size $[0,1] \times [0,1] \times [0,1]$. To generate a tetrahedral mesh for this cube, first divide the cube into $(n_x \times n_y \times n_z)$ bricks, which are divided in $(2 \times 2 \times 2)$ smaller bricks, which all have the same size. Then each of these small bricks is divided into five tetrahedra, as is done in [20], p. 52, such that the faces match exactly (a zigzag-pattern).

The first step is to define the positions of all vertices. Since all vertices are corners of the small bricks, this can be done with three for-loops over the coordinates x, y and z. This leads to a $(n_{ver} \times 3)$ array, where n_{ver} is the number of vertices. On every row i, there are the x, y and z-coordinates of vertex i in the first, second and third column, respectively.

Then the elements are generated. Compute for every tetrahedron the coordinates of the vertices and determine the indices of the vertices with the help of the array of vertex coordinates. This results into a $(N_{elt} \times 4)$ array, where N_{elt} is the number of elements. On every row i, the vertex numbers of Element i are put in the first, second, third and fourth column. For example, if element 1 consists of Vertices $2, 5, 10$ and 1, then the first row of this array is given by [2 5 10 1].

The last step is computing the faces. To do this, loop over the elements and over the faces of the specific element and then first compute the sum of the coordinates of these vertices. If the coordinates are not on the list of faces yet, add this face to the list of faces, this element is K_L. If the coordinates are already on the list, then this face already exists and this is element K_R of this particular face. If there are only periodic boundary conditions, one should take the coordinates modulo 1 to make sure that the elements that are next to each other across the boundary, are also next to each other in the list of faces.

It is also useful to save the relative position of the face on the tetrahedron, for example the face with the first three vertices of an element has relative position 4 (see Table 3.1). Finally, save the orientation and reflection flag of both adjacent elements, which are defined in [22], p.171. So in the end, the faces are represented in two arrays, one for the interior faces and one for the boundary faces. The one for the interior faces has size $(n_{int} \times 11)$, where n_{int} is the number of interior faces. The first three columns of the array contain the vertex numbers of the face, the next four columns are for the "left" element K_L: first the element number, then the relative position and finally two columns for the orientation flag. The last four columns are for the "right" element K_R, with the same division as with the left element. The array for the boundary faces has size $(n_{bound} \times 7)$, where n_{bound} is the number of boundary faces. Again, the first three columns are the vertex numbers, the fourth column contains the element number, the fifth column contains the relative position of the face and the last two columns contain the flag.

So in the end, the mesh is represented in four arrays: one that connects the vertex numbers to the vertex coordinates, one that connects the element numbers to the vertex numbers and two arrays with information about the faces.

3.4.2 Computation of transformations

In this step, the transformations F_K are computed for every element. This can be done with the barycentric coordinates λ_i of the reference tetrahedron (see Section 3.2.3) and the coordinates of the vertices of element K, call them $\mathbf{x}_i$. The transformation is given by

$$F_K(\boldsymbol{\xi}) = \sum_{i=1}^{4} \mathbf{x}_i \lambda_i(\boldsymbol{\xi}).$$

However, it is more beneficial to express the transformation as $F_K(\boldsymbol{\xi}) = B_K\boldsymbol{\xi} + b_K$, since it is easier to determine the inverse transformation and to determine the Jacobian matrix. Fortunately, it is not difficult to express the transformation as a linear system:

$$\begin{aligned} F_K\left((0,0,0)^T\right) =& b_K, \\ F_K\left((1,0,0)^T\right) =& B_K^1 + b_K, \\ F_K\left((0,1,0)^T\right) =& B_K^2 + b_K, \\ F_K\left((0,0,1)^T\right) =& B_K^3 + b_K, \end{aligned}$$

in which B_K^j is the j^{th} column of B_K.
The Jacobian matrix is now given by B_K and the inverse transformation is given by

$$F_K^{-1}(\mathbf{x}) = B_K^{-1}\mathbf{x} - B_K^{-1}b_K.$$

3.4.3 Computation of local matrices

The next step is to compute the local matrices that will in the end be assembled into a global matrix. In order to do that, first the solution will be expressed as a linear combination of the basis functions, then (3.12) will be expressed as a linear system in the coefficients and finally the local matrices can be computed with Gauss quadrature.

Substitution

To determine the linear system for (3.12), we substitute for each element $K \in \mathcal{T}_h$

$$\mathbf{H}_h(\mathbf{x})|_K = \sum_{j=1}^{n_p} u_{j,K} \boldsymbol{\psi}_{j,K}(\mathbf{x}),$$

in which $u_{j,K} \in \mathbb{R}$ are coefficients and $\boldsymbol{\psi}_{j,K}(\mathbf{x}) = \left[(dF_K)^{-T}\boldsymbol{\psi}_j\right] \circ F_K^{-1}$ are the basis functions transformed from the reference element $\hat{K}$ to element K (see Section 3.1 and Appendix A). A simple substitution shows that the curl is a linear operator, so

$$\nabla_h \times \mathbf{H}_h(\mathbf{x})|_K = \sum_{j=1}^{n_p} u_{j,K} \nabla_h \times \boldsymbol{\psi}_{j,K}(\mathbf{x}).$$

If one also substitutes $\boldsymbol{\phi}(\mathbf{x}) = \boldsymbol{\psi}_{i,K}(\mathbf{x})$, the contributions to (3.12) that integrate over the elements become

$$\begin{aligned}
\int_\Omega \epsilon^{-1}(\nabla_h \times \mathbf{H}_h) \cdot (\nabla_h \times \boldsymbol{\phi}) \; dx &= \sum_{j=1}^{n_p} u_{j,K} \int_K \epsilon^{-1}(\nabla_h \times \boldsymbol{\psi}_{j,K}) \cdot (\nabla_h \times \boldsymbol{\psi}_{i,K}) \; dx \\
&= \mathcal{S}_K \mathbf{u}_K, \\
\int_\Omega \mu \mathbf{H}_h \cdot \boldsymbol{\phi} \; dx &= \sum_{j=1}^{n_p} u_{j,K} \int_K \mu \boldsymbol{\psi}_{j,K} \cdot \boldsymbol{\psi}_{i,K} \; dx \\
&= \mathcal{M}_K \mathbf{u}_K.
\end{aligned}$$

The contributions with integrals over the faces become

$$\begin{aligned}
&\int_{\mathcal{F}_h} \llbracket \mathbf{H}_h \rrbracket_T \cdot \{\epsilon^{-1} \nabla_h \times \boldsymbol{\phi}\} + \{\epsilon^{-1} \nabla_h \times \mathbf{H}_h\} \cdot \llbracket \boldsymbol{\phi} \rrbracket_T \; dA \\
&= \sum_{f \in \partial K} \int_f \llbracket \mathbf{H} \rrbracket_T \cdot \{\epsilon^{-1}(\nabla_h \times \boldsymbol{\phi}_{i,K})\} + \{\epsilon^{-1}(\nabla_h \times \mathbf{H})\} \cdot \llbracket \boldsymbol{\phi}_{i,K} \rrbracket_T \; dA \\
&= \sum_{\substack{f \in \partial K, \\ K = K_L}} \int_f \left((\mathbf{n}_L \times \mathbf{H}^L) + (\mathbf{n}_R \times \mathbf{H}^R)\right) \cdot \frac{1}{2} \left(\left(\epsilon^{-1}\right)^L (\nabla_h \times \boldsymbol{\phi}_{i,K})\right) \; dA \\
&\quad + \sum_{\substack{f \in \partial K, \\ K = K_R}} \int_f \left((\mathbf{n}_L \times \mathbf{H}^L) + (\mathbf{n}_R \times \mathbf{H}^R)\right) \cdot \frac{1}{2} \left(\left(\epsilon^{-1}\right)^R (\nabla_h \times \boldsymbol{\phi}_{i,K})\right) \; dA \\
&\quad + \sum_{\substack{f \in \partial K \\ K = K_L}} \int_f \frac{1}{2} \left(\left(\epsilon^{-1}\right)^L (\nabla_h \times \mathbf{H}^L) + \left(\epsilon^{-1}\right)^R (\nabla_h \times \mathbf{H}^R)\right) \cdot \left(\mathbf{n}_L \times \boldsymbol{\phi}_{i,K}\right) \; dA \\
&\quad + \sum_{\substack{f \in \partial K \\ K = K_R}} \int_f \frac{1}{2} \left(\left(\epsilon^{-1}\right)^L (\nabla_h \times \mathbf{H}^L) + \left(\epsilon^{-1}\right)^R (\nabla_h \times \mathbf{H}^R)\right) \cdot \left(\mathbf{n}_R \times \boldsymbol{\phi}_{i,K}\right) \; dA \\
&= \sum_{\substack{f \in \partial K, \\ K = K_L}} \sum_{j=1}^{n_p} u_{j,K_L} \int_f \frac{1}{2} (\mathbf{n}_L \times \boldsymbol{\phi}_{j,K_L}) \cdot \left(\epsilon^{-1}\right)^L (\nabla_h \times \boldsymbol{\phi}_{i,K}) \; dA
\end{aligned}$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f \frac{1}{2}\left(\epsilon^{-1}\right)^L(\nabla_h\times\boldsymbol{\phi}_{j,K_L})\cdot(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \frac{1}{2}(\mathbf{n}_R\times\boldsymbol{\phi}_{j,K_R})\cdot\left(\epsilon^{-1}\right)^L(\nabla_h\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \frac{1}{2}\left(\epsilon^{-1}\right)^R(\nabla_h\times\boldsymbol{\phi}_{j,K_R})\cdot(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f \frac{1}{2}(\mathbf{n}_L\times\boldsymbol{\phi}_{j,K_L})\cdot\left(\epsilon^{-1}\right)^R(\nabla_h\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f \frac{1}{2}\left(\epsilon^{-1}\right)^L(\nabla_h\times\boldsymbol{\phi}_{j,K_L})\cdot(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \frac{1}{2}(\mathbf{n}_R\times\boldsymbol{\phi}_{j,K_R})\cdot\left(\epsilon^{-1}\right)^R(\nabla_h\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \frac{1}{2}\left(\epsilon^{-1}\right)^L(\nabla_h\times\boldsymbol{\phi}_{j,K_R})\cdot(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})\,dA.$$

$$\int_{\mathcal{F}_h}\alpha_f[\![\epsilon^{-1}\mathbf{H}_h]\!]_T\cdot[\![\boldsymbol{\phi}]\!]_T$$

$$=\sum_{\substack{f\in\partial K,\\ K=K_L}}\int_f\alpha_f\left(\left(\epsilon^{-1}\right)^L(\mathbf{n}_L\times\mathbf{H}^L)+\left(\epsilon^{-1}\right)^R(\mathbf{n}_R\times\mathbf{H}^R)\right)\cdot(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_R}}\int_f\alpha_f\left(\left(\epsilon^{-1}\right)^L(\mathbf{n}_L\times\mathbf{H}^L)+\left(\epsilon^{-1}\right)^R(\mathbf{n}_R\times\mathbf{H}^R)\right)\cdot(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})\,dA$$

$$=\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f\alpha_f\left(\epsilon^{-1}\right)^L(\mathbf{n}_L\times\boldsymbol{\phi}_{j,K_L})\cdot(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f\alpha_f\left(\epsilon^{-1}\right)^R(\mathbf{n}_R\times\boldsymbol{\phi}_{j,K_R})\cdot(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f\alpha_f\left(\epsilon^{-1}\right)^L(\mathbf{n}_L\times\boldsymbol{\phi}_{j,K_L})\cdot(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})\,dA$$

$$+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f\alpha_f\left(\epsilon^{-1}\right)^R(\mathbf{n}_R\times\boldsymbol{\phi}_{j,K_R})\cdot(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})\,dA.$$

In the expressions above, K_L and K_R are the elements adjacent to face f. Define the matrices $\mathcal{S}_K,\mathcal{M}_K,\mathcal{A}_{CD}$ and $\mathcal{B}_{CD}$, $C,D\in\{K_L,K_R\}$ as

$$\mathcal{S}_K^{ij}=\int_K\epsilon^{-1}(\nabla_h\times\boldsymbol{\psi}_{j,K})\cdot(\nabla_h\times\boldsymbol{\psi}_{i,K})\,dx,$$

$$
\begin{aligned}
\mathcal{M}_K^{ij} &= \int_K \mu \boldsymbol{\psi}_{j,K} \cdot \boldsymbol{\psi}_{i,K}\; dx,\\
\mathcal{A}_{CD}^{ij} &= \frac{1}{2}\int_f \left(\epsilon^{-1}\right)^C (\mathbf{n}_D \times \boldsymbol{\psi}_{j,K_D}) \cdot (\nabla_h \times \boldsymbol{\psi}_{i,K_C})\; dA,\\
\mathcal{B}_{CD}^{ij} &= \int_f \left(\epsilon^{-1}\right)^D (\mathbf{n}_D \times \boldsymbol{\psi}_{j,K_D}) \cdot (\mathbf{n}_C \times \boldsymbol{\psi}_{i,K_C})\; dA,
\end{aligned}
$$

and define the right-hand side vectors

$$
\begin{aligned}
\mathcal{G}_f &= \int_f \epsilon^{-1}\mathbf{g} \cdot (\mathbf{n} \times \boldsymbol{\phi}),\\
\mathcal{H}_f &= \int_f \epsilon^{-1}\mathbf{g} \cdot (\nabla_h \times \boldsymbol{\phi}).
\end{aligned}
$$

At the end of this section it will be shown how these matrices and vectors can be incorporated in the global matrices.

Gauss quadrature

Both on the reference tetrahedron $\hat{K}$ and the reference triangle $\hat{T}$, which has vertices $(-1,1),(1,-1)$ and $(-1,1)$, Gauss quadrature is used to compute the integrals over this domain. Let w_ℓ be the Gauss weights and $\boldsymbol{\xi}_\ell$ the Gauss quadrature points, then any integral can be approximated with

$$
\begin{aligned}
\int_{\hat{K}} f(\boldsymbol{\xi})\; d\xi &\approx \sum_\ell f(\boldsymbol{\xi}_l) w_\ell,\\
\int_{\hat{T}} f(\boldsymbol{\xi})\; d\xi &\approx \sum_\ell f(\boldsymbol{\xi}_l) w_\ell.
\end{aligned}
$$

The number and location of the quadrature points depend on domain type and the desired accuracy. A list of Gauss quadrature points $\boldsymbol{\xi}_\ell$ and weights w_ℓ can be found in [22], Chapter 4. The main advantage is that Gauss quadrature can compute integrals over polynomials exactly using less points than for example Newton-Cotes quadrature. For example, if one wants to integrate a fourth order polynomial exactly over the reference triangle, one would need 15 points using Newton-Cotes quadrature, but only 6 points in Gauss quadrature.

Computing the integrals

In this paragraph, it will be discussed how the integrals discussed earlier in this section can be computed. Generally, the integrands should first be expressed in terms of the basis functions on the reference element and the transformation $F_K : \hat{K} \to K$, for details see Appendix A. After that, these integrals can be computed with Gauss quadrature.
Define the Jacobian matrix dF_K of the transformation $F_K : \hat{K} \to K$ as

$$
(dF_K)_{lm} = \frac{\partial (F_K)_l}{\partial \xi_m}.
$$

In practice, this is the matrix B_K, see Section 3.4.2. The entries of the matrices $\mathcal{M}_K$ and $\mathcal{S}_K$ can be expressed as

$$
\begin{aligned}
\mathcal{M}_K^{ij} &= \int_{\widehat{K}} \mu(F_K)(dF_K)^{-T}\boldsymbol{\psi}_j \cdot (dF_K)^{-T}\boldsymbol{\psi}_i |\det(dF_K)|\; d\xi\\
&= \sum_\ell \mu(F_K(\boldsymbol{\xi}_\ell) \left((dF_K)^{-T}\boldsymbol{\psi}_j(\boldsymbol{\xi}_\ell)\right) \cdot \left((dF_K)^{-T}\boldsymbol{\psi}_i(\boldsymbol{\xi}_\ell)\right) |\det(dF_K)| w_\ell,
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{S}_K^{ij} &= \int_{\widehat{K}} \epsilon^{-1}(F_K) \left(\frac{1}{\det(dF_K)} dF_K \hat{\nabla} \times \boldsymbol{\psi}_j \right) \cdot \left(\frac{1}{\det(dF_K)} dF_K \hat{\nabla} \times \boldsymbol{\psi}_i \right) |\det(dF_K)| \; d\xi \\
&= \int_{\widehat{K}} \epsilon^{-1}(F_K) \frac{\operatorname{sign}(\det(dF_K))}{\det(dF_K)} (dF_K \hat{\nabla} \times \boldsymbol{\psi}_j) \cdot (dF_K \hat{\nabla} \times \boldsymbol{\psi}_i) \; d\xi \\
&= \sum_{\ell} \epsilon^{-1}(F_K(\boldsymbol{\xi}_\ell)) \frac{1}{|\det(dF_K)|} \left(dF_K \hat{\nabla} \times \boldsymbol{\psi}_j(\boldsymbol{\xi}_\ell) \right) \cdot \left(dF_K \hat{\nabla} \times \boldsymbol{\psi}_i(\boldsymbol{\xi}_\ell) \right) w_\ell,
\end{aligned}
$$

where we used the transformations as described in Appendix A. An expression for the face integrals $\mathcal{A}_{CD}$ and $\mathcal{B}_{CD}$ can be found by

$$
\begin{aligned}
\mathcal{A}_{CD}^{ij} =& \frac{1}{2} \int_{\hat{T}} \left(\frac{(dF_{K_D})^{-T} \hat{\mathbf{n}}_D}{|(dF_{K_D})^{-T} \hat{\mathbf{n}}_D|} \times (dF_{K_D})^{-T} \boldsymbol{\psi}_j(r_D(\boldsymbol{\eta})) \right) \cdot \left(\frac{1}{\det(dF_{K_C})} dF_{K_C} \hat{\nabla} \times \boldsymbol{\psi}_i(r_C(\boldsymbol{\eta})) \right) \\
& \epsilon^{-1}(F_{K_C}(r_C(\boldsymbol{\eta})) \; |\det(dF_{K_L})| |(dF_{K_L})^{-T} \hat{\mathbf{n}}_L| |r_{L,\eta_1} \times r_{L,\eta_2}| \; d\eta,
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{B}_{CD}^{ij} =& \int_{\hat{T}} \left(\frac{(dF_{K_D})^{-T} \hat{\mathbf{n}}_D}{|(dF_{K_D})^{-T} \hat{\mathbf{n}}_D|} \times (dF_{K_D})^{-T} \boldsymbol{\psi}_j(r_D(\boldsymbol{\eta})) \right) \cdot \left(\frac{(dF_{K_C})^{-T} \hat{\mathbf{n}}_C}{|(dF_{K_C})^{-T} \hat{\mathbf{n}}_C|} \times (dF_{K_C})^{-T} \boldsymbol{\psi}_i(r_C(\boldsymbol{\eta})) \right) \\
& \epsilon^{-1}(F_{K_D}(r_D(\boldsymbol{\eta})) \; |\det(dF_{K_L})| |(dF_{K_L})^{-T} \hat{\mathbf{n}}_L| |r_{L,\eta_1} \times r_{L,\eta_2}| \; d\eta,
\end{aligned}
$$

in which the transformation $r_j(\boldsymbol{\eta}) : \hat{T} \to \hat{f}_j$ is defined in Appendix A. The vectors $\mathcal{G}_K$ and $\mathcal{H}_K$ can be computed as

$$
\begin{aligned}
\mathcal{G}_f^i =& \int_{\hat{T}} \epsilon^{-1}(F_K(r(\boldsymbol{\eta}))) \; \mathbf{g}(F_K(r(\boldsymbol{\eta}))) \cdot \left(\frac{(dF_K)^{-T} \hat{\mathbf{n}}}{|(dF_K)^{-T} \hat{\mathbf{n}}|} \times (dF_K)^{-T} \boldsymbol{\psi}_i(r(\boldsymbol{\eta})) \right) \\
& |\det(dF_K)| |(dF_K)^{-T} \hat{\mathbf{n}}| |r_{\eta_1} \times r_{\eta_2}| \; d\eta,
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{H}_f^i =& \int_{\hat{T}} \epsilon^{-1}(F_K(r(\boldsymbol{\eta}))) \; \mathbf{g}(F_K(r(\boldsymbol{\eta}))) \cdot \left(\frac{1}{\det(dF_K)} dF_K \hat{\nabla} \times \boldsymbol{\psi}_i(r(\boldsymbol{\eta})) \right) \\
& |\det(dF_K)| |(dF_K)^{-T} \hat{\mathbf{n}}| |r_{\eta_1} \times r_{\eta_2}| \; d\eta.
\end{aligned}
$$

At last, we have to compute the entries of the right-hand side $\mathcal{J}_K^i = (\mathbf{J}, \boldsymbol{\psi}_{i,K})$ for every element K. This can be done by

$$
\mathcal{J}_K^i = \int_{\hat{K}} (\mathbf{J}(F_K(\boldsymbol{\xi}))) \cdot \left((dF_K)^{-T} \boldsymbol{\psi}_i \right) |\det(dF_K)| \; d\xi.
$$

3.4.4 Overview of the algorithm

Standard problem

In order to implement the IP-DG method for the standard problem, the following steps should be taken:

1. Generate a mesh,
2. Define the basis functions on the reference tetrahedron and compute their value at Gauss points,
3. Compute the element matrices $\mathcal{M}_K, \mathcal{S}_K, \mathcal{A}_{CD}$ and $\mathcal{B}_{CD}$ for $C, D \in \{L, R\}$ for every element $K \in \mathcal{T}_h$ and face $f \in \mathcal{F}_h$,
4. Compute $\mathcal{J}_K, \mathcal{G}_f$ and $\mathcal{H}_f$ for every element $K \in \mathcal{T}_h$ and face $f \in \mathcal{F}_h^n$,

5. Assemble the global matrix $[M]$ and the righthand side $[F]$ with a loop over the elements and a loop over the faces as below,
6. Solve the linear system $[M]\mathbf{u} = [F]$ with Matlab's \-operator,
7. For every element, extract the solution $\mathbf{H}_h(\mathbf{x})|_K = \sum_{j=1}^{n_p} u_{j,K}\boldsymbol{\psi}_{j,K}(\mathbf{x})$,
8. Plot and post-process the solution $\mathbf{H}_h(\mathbf{x})$.

The assembly of the global matrix can be summarized as:
initialize $[M] = zeros(n_p \cdot N_{elt}, n_p \cdot N_{elt})$ and $[F] = zeros(n_p \cdot N_{elt}, 1)$. Then loop over the elements, and update

$$\begin{aligned}[M]_{KK} &\leftarrow \mathcal{S}_K - \omega^2\mathcal{M}_K\\ [F]_K &\leftarrow \mathcal{J}_K.\end{aligned}$$

Next, loop over the interior faces and the faces with periodic boundary conditions and update

$$\begin{aligned}[M]_{LL} &\leftarrow -\mathcal{A}_{LL} - \mathcal{A}_{LL}^T + \alpha_f\mathcal{B}_{LL},\\ [M]_{LR} &\leftarrow -\mathcal{A}_{LR} - \mathcal{A}_{RL}^T + \alpha_f\mathcal{B}_{LR},\\ [M]_{RL} &\leftarrow -\mathcal{A}_{RL} - \mathcal{A}_{LR}^T + \alpha_f\mathcal{B}_{RL},\\ [M]_{RR} &\leftarrow -\mathcal{A}_{RR} - \mathcal{A}_{RR}^T + \alpha_f\mathcal{B}_{RR}.\end{aligned}$$

Finally, loop over the boundary faces $f \in \mathcal{F}_h^n$ and update

$$\begin{aligned}[M]_{KK} &\leftarrow -\mathcal{A}_{KK} - \mathcal{A}_{KK}^T + \alpha_f\mathcal{B}_L L,\\ [F]_K &\leftarrow \alpha_F\mathcal{G}_f - \mathcal{H}_f.\end{aligned}$$

Solving the resulting system leads to the desired coefficients $u_{j,K}$. The approximate solution $\mathbf{H}_h$ then can be found using the expression

$$\mathbf{H}_h(\mathbf{x}) = \sum_{j=1}^{n_p} u_{j,K}\boldsymbol{\psi}_{j,K}(\mathbf{x}).$$

Eigenvalue problem

In order to implement the IP-DG method for the eigenvalue problem, the following steps should be taken:

1. Generate a mesh,
2. Define the basis functions on the reference tetrahedron and compute their value at Gauss points,
3. Compute the element matrices $\mathcal{M}_K, \mathcal{S}_K, \mathcal{A}_{CD}$ and $\mathcal{B}_{CD}$ for $C, D \in \{L, R\}$ for every element $K \in \mathcal{T}_h$ and face $f \in \mathcal{F}_h$,
4. Compute $\mathcal{G}_f$ and $\mathcal{H}_f$ for every element $K \in \mathcal{T}_h$ and face $f \in \mathcal{F}_h^n$,
5. Assemble the global matrices $[S]$ and $[M]$ with a loop over the elements and a loop over the faces as below,
6. Solve the generalized eigenproblem $[S]\mathbf{u} = \lambda[M]\mathbf{u}$ with Matlab's "eigs"-function and remove eigenvalues with value zero,

7. Display and post-process the eigenfunctions λ_h.

To construct the matrices $[S]$ and $[M]$, first initialize $[S] = zeros(n_p \cdot N_{elt}, n_p \cdot N_{elt})$ and $[M] = zeros(n_p \cdot N_{elt}, n_p \cdot N_{elt})$. Then loop over the elements and update

$$\begin{aligned}[S]_{KK} &\leftarrow \mathcal{S}_K, \\ [M]_{KK} &\leftarrow \mathcal{M}_K.\end{aligned}$$

Next, loop over the interior faces, and update

$$\begin{aligned}[S]_{LL} &\leftarrow -\mathcal{A}_{LL} - \mathcal{A}_{LL}^T + \alpha_f \mathcal{B}_{LL}, \\ [S]_{LR} &\leftarrow -\mathcal{A}_{LR} - \mathcal{A}_{RL}^T + \alpha_f \mathcal{B}_{LR}, \\ [S]_{RL} &\leftarrow -\mathcal{A}_{RL} - \mathcal{A}_{LR}^T + \alpha_f \mathcal{B}_{RL}, \\ [S]_{RR} &\leftarrow -\mathcal{A}_{RR} - \mathcal{A}_{RR}^T + \alpha_f \mathcal{B}_{RR}.\end{aligned}$$

Finally, loop over the boundary faces $f \in \mathcal{F}_h^n$ and update

$$[S]_{KK} \leftarrow -\mathcal{A}_{KK} - \mathcal{A}_{KK}^T + \alpha_f \mathcal{B}_L L.$$

Some results of both the standard problem and the eigenvalue problem will be given in next section.

3.5 Results

In this section, some results will be presented. First we will present the results of the standard time-harmonic Maxwell problem in a homogeneous medium, then the results of the eigenvalue problem are presented.

3.5.1 Standard problem

Consider the time-harmonic Maxwell equations with $\omega^2 = 1$. As test case, the homogeneous unit cube $\Omega = [0,1]^3$ with $\epsilon_r = \mu_r = 1$ is chosen. All matrices are computed with the hierarchic elements as described in Section 3.2.3.
Assume that the field at the boundary satisfies $\mathbf{n} \times \mathbf{H} = 0$ at $\partial\Omega$. The source term is chosen to be

$$\mathbf{J}_1(x,y,z) = (2\pi^2 - 1) \begin{pmatrix} \sin(\pi y)\sin(\pi z) \\ \sin(\pi z)\sin(\pi x) \\ \sin(\pi x)\sin(\pi y) \end{pmatrix},$$

so that the exact solution is given by

$$\mathbf{H}(x,y,z) = \begin{pmatrix} \sin(\pi y)\sin(\pi z) \\ \sin(\pi z)\sin(\pi x) \\ \sin(\pi x)\sin(\pi y) \end{pmatrix}.$$

The optimal convergence rate for this problem is $O(h^{p+1})$ in the L^2-norm and $O(h^p)$ in the DG-norm [12]. From [21], we know that the optimal penalty parameter $\alpha_{f,0}$ equals

$$\alpha_{f,0} = \frac{1}{2h_f} + \frac{2}{3}(p+1)(p+3)\left(\frac{S(f)}{V(K_L)} + \frac{S(f)}{V(K_R)}\right),$$

in which h_f is the diameter of the circumscribed circle of the face, p is the maximum polynomial order of the basis functions, $S(f)$ is the area of the face and $V(K)$ is the volume of element K. Since the expressions for computing the optimal penalty parameter for each face is rather involved and the parameter can be changed a bit without influencing the stability ([21]), the penalty parameter is chosen to be

$$\alpha_f = \frac{2(p+1)(p+3)}{\min(dx, dy, dz)}.$$

The computations are performed in Matlab, in which solving the linear system is done with Matlab's \-operator. The results for the first four hierarchic elements are displayed in Table 3.4. As can be seen in the table, the convergence results are of the optimal order. It can also be seen in the table that higher order polynomials are more efficient than lower order polynomials for this problem.

To test the problem with periodic boundary conditions, we look at

$$\mathbf{J}_2(x, y, z) = (8\pi^2 - 1)\begin{pmatrix} \sin(2\pi y)\sin(2\pi z) \\ \sin(2\pi z)\sin(2\pi x) \\ \sin(2\pi x)\sin(2\pi y) \end{pmatrix},$$

so that the exact solution is given by

$$\mathbf{H}(x, y, z) = \begin{pmatrix} \sin(2\pi y)\sin(2\pi z) \\ \sin(2\pi z)\sin(2\pi x) \\ \sin(2\pi x)\sin(2\pi y) \end{pmatrix}.$$

The results are displayed in Table 3.5. Again, the convergence results are of the correct order. The errors are a bit larger for this problem, since we now have a full period of the sines instead of only half a period. The results for source $\mathbf{J}_2$ with boundary conditions $\mathbf{n} \times \mathbf{H} = 0$ are similar to the results with the periodic boundary conditions.

3.5.2 Eigenvalue problem

As a test case, we again consider the homogeneous unit cube $\Omega = [0, 1]^3$ with $\epsilon = \mu = 1$ and the hierarchic elements of Section 3.2.3. The eigenvalues ω^2 can be written as [21]

$$\omega^2 = \pi^2(l^2 + m^2 + n^2),$$

where $l, m, n \in \mathbb{N}\backslash_{\{0\}}$ and $lm + ln + mn > 0$. If l, m and n are all non-zero, the eigenvalue has multiplicity 2. All the eigenfunctions are smooth. From [7] we know that the optimal convergence rate for the eigenvalues equals $O(h^{2p})$, since the solution is smooth and the IP-DG method is a hermitian method.

The penalty parameter is chosen the same as before, so

$$\alpha_f = \frac{2(p+1)(p+3)}{\min(dx, dy, dz)}.$$

To find the eigenvalues, Matlab's iterative eigenvalue solver "eigs" is used. The results are displayed in Tables 3.6 - 3.9. As can be seen in these tables, the convergence order seems to converge to $2p$, except for the finest grid of $p = 4$. However, the error in the eigenvalues are so small that rounding errors due to working at double precision (ϵ =2.22e-16) and the rounding errors in the Gauss-quadrature rules are relevant. Therefore we believe that this method works correctly.

	$\|\mathbf{H}-\mathbf{H}_h\|_{L^2}$	Order	$\|\mathbf{H}-\mathbf{H}_h\|_{DG}$	Order	DOFs
$p=1$					
$N_{elt}=5$	5.2230e-01		3.5999e+00		60
$N_{elt}=40$	1.6789e-01	1.64	1.8115e+00	0.99	480
$N_{elt}=320$	5.0805e-02	1.72	9.4992e-01	0.93	3840
$N_{elt}=2560$	1.3575e-02	1.90	4.8244e-01	0.98	30720
$N_{elt}=20480$	3.4829e-03	1.96	2.4253e-01	0.99	245760
$p=2$					
$N_{elt}=5$	1.5189e-01		1.4179e00		150
$N_{elt}=40$	2.8098e-02	2.43	4.9225e-01	1.53	1200
$N_{elt}=320$	3.8554e-03	2.87	1.3069e-01	1.91	9600
$N_{elt}=2560$	4.9803e-04	2.95	3.3236e-02	1.98	76800
$N_{elt}=20480$	6.3073e-05	2.98	8.3513e-03	1.99	614400
$p=3$					
$N_{elt}=5$	6.9848e-02		8.4323e-01		300
$N_{elt}=40$	4.3311e-03	4.01	9.8644e-02	3.10	2400
$N_{elt}=320$	2.8585e-04	3.92	1.2857e-02	2.94	19200
$N_{elt}=2560$	1.8143e-05	3.98	1.6236e-03	2.99	1228800
$p=4$					
$N_{elt}=5$	2.7999e-03		1.3739e-01		525
$N_{elt}=40$	5.6767e-04	2.30	1.5582e-02	3.14	4200
$N_{elt}=320$	1.8830e-05	4.91	1.0171e-03	3.94	33600
$N_{elt}=2560$	6.0128e-07	4.97	6.4228e-05	3.99	268800

Table 3.4: Convergence of IP-DG method on tetrahedral meshes with source $\mathbf{J}_1$ and boundary conditions $\mathbf{n}\times\mathbf{H}=0$ at $\partial\Omega$.

	$\|\mathbf{H}-\mathbf{H}_h\|_{L^2}$	Order	$\|\mathbf{H}-\mathbf{H}_h\|_{DG}$	Order
$p=1$				
$N_{elt}=40$	6.0205e-01		7.1879e+00	
$N_{elt}=320$	1.9396e-01	1.63	3.6752e+00	0.97
$N_{elt}=2560$	5.3682e-02	1.85	1.9196e+00	0.94
$N_{elt}=20480$	1.3796e-02	1.96	9.7027e-01	0.98
$p=2$				
$N_{elt}=40$	1.8277e-01		2.9173e+00	
$N_{elt}=320$	3.0013e-02	2.61	9.9472e-01	1.55
$N_{elt}=2560$	3.9812e-03	2.91	2.6285e-01	1.92
$p=3$				
$N_{elt}=40$	8.5117e-02		1.6541e+00	
$N_{elt}=320$	4.5800e-03	4.22	1.9650e-01	3.07
$N_{elt}=2560$	2.9255e-04	3.97	2.5690e-02	2.94
$p=4$				
$N_{elt}=40$	1.4872e-02		2.7565e-01	
$N_{elt}=320$	5.8974e-04	4.66	3.0982e-02	3.15
$N_{elt}=2560$	1.9182e-05	4.94	2.0287e-03	3.93

Table 3.5: Convergence of IP-DG method on tetrahedral meshes with source $\mathbf{J}_2$ and periodic boundary conditions at $\partial\Omega$.

Analytical	$\lvert\lambda - \lambda_h\rvert$						
	$N_{elt} = 40$	$N_{elt} = 320$	Order	$N_{elt} = 2560$	Order	$N_{elt} = 20480$	Order
$2\pi^2$	2.1693e+00	7.8772e-01	1.46	2.1519e-01	1.87	5.5713e-02	1.95
$2\pi^2$	2.1693e+00	7.8772e-01	1.46	2.1519e-01	1.87	5.5713e-02	1.95
$2\pi^2$	2.1693e+00	7.8772e-01	1.46	2.1519e-01	1.87	5.5713e-02	1.95
$3\pi^2$	6.3406e+00	1.7056e+00	1.89	4.6926e-01	1.86	1.2180e-01	1.95
$3\pi^2$	6.3406e+00	1.7056e+00	1.89	4.6926e-01	1.86	1.2180e-01	1.95
$5\pi^2$	5.1191e+00	4.6910e+00	0.13	1.3150e+00	1.83	3.4025e-01	1.95
$5\pi^2$	5.1191e+00	4.6910e+00	0.13	1.3150e+00	1.83	3.4025e-01	1.95
$5\pi^2$	5.1191e+00	4.6910e+00	0.13	1.3150e+00	1.83	3.4025e-01	1.95
$5\pi^2$	2.4528e+01	4.6918e+00	2.39	1.3152e+00	1.83	3.4026e-01	1.95
$5\pi^2$	2.4528e+01	4.6918e+00	2.39	1.3152e+00	1.83	3.4026e-01	1.95

Table 3.6: Convergence of the IP-DG method with $p = 1$ and $\mathbf{n} \times \mathbf{H} = 0$ at $\partial\Omega$.

Analytical	$\lvert\lambda - \lambda_h\rvert$				
	$N_{elt} = 40$	$N_{elt} = 320$	Order	$N_{elt} = 2560$	Order
$2\pi^2$	2.1142e-01	1.5778e-02	3.74	1.0696e-03	3.88
$2\pi^2$	2.1142e-01	1.5778e-02	3.74	1.0696e-03	3.88
$2\pi^2$	2.1142e-01	1.5778e-02	3.74	1.0696e-03	3.88
$3\pi^2$	2.0040e-01	5.2232e-02	1.94	3.5881e-03	3.86
$3\pi^2$	2.0040e-01	5.2232e-02	1.94	3.5881e-03	3.86
$5\pi^2$	1.3339e+00	2.1779e-01	2.61	1.5320e-02	3.83
$5\pi^2$	1.3339e+00	2.1779e-01	2.61	1.5320e-02	3.83
$5\pi^2$	1.3339e+00	2.1779e-01	2.61	1.5320e-02	3.83
$5\pi^2$	2.6659e+00	2.1782e-01	3.61	1.5320e-02	3.83
$5\pi^2$	2.6659e+00	2.1782e-01	3.61	1.5320e-02	3.83

Table 3.7: Convergence of the IP-DG method with $p = 2$ and $\mathbf{n} \times \mathbf{H} = 0$ at $\partial\Omega$.

Analytical	$\lvert\lambda - \lambda_h\rvert$				
	$N_{elt} = 40$	$N_{elt} = 320$	Order	$N_{elt} = 2560$	Order
$2\pi^2$	8.3182e-03	1.6457e-04	5.66	2.7419e-06	5.91
$2\pi^2$	8.3182e-03	1.6457e-04	5.66	2.7422e-06	5.91
$2\pi^2$	8.3182e-03	1.6457e-04	5.66	2.7433e-06	5.91
$3\pi^2$	7.1576e-02	8.2898e-04	6.43	1.4109e-05	5.88
$3\pi^2$	7.1576e-02	8.2898e-04	6.43	1.4109e-05	5.88
$5\pi^2$	2.5029e-01	5.5273e-03	5.50	9.6445e-05	5.84
$5\pi^2$	2.5029e-01	5.5273e-03	5.50	9.6445e-05	5.84
$5\pi^2$	2.5029e-01	5.5273e-03	5.50	9.6445e-05	5.84
$5\pi^2$	2.6950e-01	5.5303e-03	5.61	9.6456e-05	5.84
$5\pi^2$	2.6950e-01	5.5303e-03	5.61	9.6456e-05	5.84

Table 3.8: Convergence of the IP-DG method with $p = 3$ and $\mathbf{n} \times \mathbf{H} = 0$ at $\partial\Omega$.

Analytical	$\|\lambda - \lambda_h\|$				
	$N_{elt} = 40$	$N_{elt} = 320$	Order	$N_{elt} = 2560$	Order
$2\pi^2$	2.4102e-04	1.0756e-06	7.81	4.3582e-08	4.63
$2\pi^2$	2.4102e-04	1.0758e-06	7.81	4.3582e-08	4.63
$2\pi^2$	2.4102e-04	1.0768e-06	7.81	4.3618e-08	4.63
$3\pi^2$	2.1587e-04	7.2257e-06	4.90	2.9534e-07	4.61
$3\pi^2$	2.1587e-04	7.2263e-06	4.90	2.9534e-07	4.61
$5\pi^2$	1.3896e-02	8.3946e-05	7.37	3.4878e-06	4.59
$5\pi^2$	1.3896e-02	8.3946e-05	7.37	3.4878e-06	4.59
$5\pi^2$	1.3896e-02	8.3946e-05	7.37	3.4878e-06	4.59
$5\pi^2$	1.5931e-02	8.3981e-05	7.57	3.4885e-06	4.59
$5\pi^2$	1.5931e-02	8.3981e-05	7.57	3.4885e-06	4.59

Table 3.9: Convergence of the IP-DG method with $p = 4$ and $\mathbf{n} \times \mathbf{H} = 0$ at $\partial\Omega$.

Chapter 4

Discretization of Maxwell-equations for periodic media

Now that it is known how to use discontinuous Galerkin methods for the time-harmonic Maxwell problem in a cavity, we will continue to the goal of this project, namely how to approximate the magnetic field/eigenvalues in a periodic medium. Since the magnetic field is expanded in Bloch-modes, it will be beneficial to adapt the basis functions accordingly. This will be done in Section 4.1. After that, the IP-DG discretization will be derived in Section 4.2, followed by some details on how to implement that discretization in Section 4.3. Finally, in Section 4.4, some results will be given for the simplest case, namely an infinite homogeneous domain.

4.1 Modified basis functions

In this section, it will be discussed how to modify the basis functions in order to approximate $\mathbf{u_k}$ instead of $\mathbf{H}$. To do that, first observe that $\mathbf{H}(\mathbf{x}) = e^{i\mathbf{k}\cdot\mathbf{x}}\mathbf{u_k}(\mathbf{x})$. Let $\{\boldsymbol{\psi}\}$ form a basis for $\mathbf{H}_h$, in this case the hierarchic basis functions as described in Section 3.2.3, and let $\{\boldsymbol{\phi}\}$ form a basis for $\mathbf{u_k}_h$. Then we can express the fields in terms of a linear combination of their basis functions:

$$\mathbf{H}_h(\mathbf{x}) = \sum_j u_{j,K}\boldsymbol{\psi}_{j,K}(\mathbf{x}),$$

$$\mathbf{u_k}_h(\mathbf{x}) = \sum_j u_{j,K}\boldsymbol{\phi}_{j,K}(\mathbf{x}).$$

If one now uses $\mathbf{H}(\mathbf{x}) = e^{i\mathbf{k}\cdot\mathbf{x}}\mathbf{u_k}(\mathbf{x})$, then the relation between the basis functions is

$$\boldsymbol{\phi}_{j,K} = e^{-i\mathbf{k}\cdot\mathbf{x}}\boldsymbol{\psi}_{j,K}.$$

Note that this adaption takes place on the tetrahedra in physical space.
A second step in modifying the basis functions is shifting the exponential with some function-dependent shift, [8]

$$\boldsymbol{\phi}_{j,K} = e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s})}\boldsymbol{\psi}_{j,K},$$

in which the shifts $\mathbf{s}$ can be chosen as one wants. The values for the shifts used in this project will be given in Section 4.3.6. These shifts make it easier to find a solution if one already knows the solution for the same problem with slightly different $\mathbf{k}$. Note that

$$\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{j,K} = \nabla \times (e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s})}\boldsymbol{\psi}_{j,K}) + (i\mathbf{k} \times (e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s})}\boldsymbol{\psi}_{j,K})$$

$$= e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s})}(\nabla \times \boldsymbol{\psi}_{j,K}),$$

so these functions are convenient if one wants to use the operator $\nabla_{\mathbf{k}}$. Normally, we would like to write the basis functions in terms of the coordinates of the reference tetrahedron, but it will turn out that this is not needed for the computations later on. In the next section, Maxwell's equation for periodic media will be discretized.

4.2 DG discretization of photonic crystals

As with the time-harmonic Maxwell's equations in Chapter 3, the equation

$$\nabla_{\mathbf{k}} \times (\epsilon^{-1}(\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}})) - \omega^2 \mu \mathbf{u}_{\mathbf{k}} = \tilde{\mathbf{J}} \qquad \text{in } \Omega,$$

with periodic boundary conditions has to be discretized in order to find an approximate solution. In order to discretize the system, start again by writing the system as a first order system:

$$\begin{aligned} \nabla_{\mathbf{k}} \times \epsilon^{-1}\mathbf{q} - \omega^2 \mu \mathbf{u}_{\mathbf{k}} &= \tilde{\mathbf{J}} && \text{in } \Omega, \\ \nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}} &= \mathbf{q} && \text{in } \Omega. \end{aligned}$$

Formulate this as an integral problem by taking the inner product of both functions with test functions $(\boldsymbol{\phi}, \boldsymbol{\pi})$:

$$\int_\Omega (\nabla_{\mathbf{k}} \times (\epsilon^{-1}\mathbf{q}_h)) \cdot \overline{\boldsymbol{\phi}} \, dx - \omega^2 \int_\Omega \mu \mathbf{u}_{\mathbf{k}h} \cdot \overline{\boldsymbol{\phi}} \, dx = \int_\Omega \tilde{\mathbf{J}} \cdot \overline{\boldsymbol{\phi}} \, dx, \tag{4.1}$$

$$\int_\Omega (\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{\boldsymbol{\pi}} \, dx = \int_\Omega \mathbf{q}_h \cdot \overline{\boldsymbol{\pi}} \, dx. \tag{4.2}$$

The next step is to use Equation (4.2) to find an expression for $\mathbf{q}_h$ in terms of $\mathbf{u}_{\mathbf{k}h}$ and its flux $\mathbf{u}_{\mathbf{k}h}^*$, then manipulate Equation (4.1), such that this $\mathbf{q}_h$ can be inserted again. Insert this $\mathbf{q}$ and choose the fluxes $\mathbf{q}_h^*$ and $\mathbf{u}_{\mathbf{k}h}^*$. The resulting bilinear form in $\mathbf{u}_{\mathbf{k}h}$ and $\boldsymbol{\phi}$ can then be used for the computations.
Start with manipulating the left-hand side of (4.2) by integrating by parts twice as in Section 3.3.1:

$$\begin{aligned} \int_\Omega (\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{\boldsymbol{\pi}} \, dx &= \int_\Omega (\nabla_h \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{\boldsymbol{\pi}} + (i\mathbf{k} \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{\boldsymbol{\pi}} \, dx \\ &= \int_\Omega (\nabla_h \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{\boldsymbol{\pi}} \\ &\quad + (i\mathbf{k} \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{\boldsymbol{\pi}} \, dx + \sum_{K \in \mathcal{T}_h} \int_{\partial K} (\mathbf{n} \times (\mathbf{u}_{\mathbf{k}h}^* - \mathbf{u}_{\mathbf{k}h})) \cdot \overline{\boldsymbol{\pi}} \, dA \\ &= \int_\Omega (\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{\boldsymbol{\pi}} \, dx - \int_{\mathcal{F}_h} \{\mathbf{u}_{\mathbf{k}h}^* - \mathbf{u}_{\mathbf{k}h}\} \cdot [\![\overline{\boldsymbol{\pi}}]\!]_T \, dA \\ &\quad + \int_{\mathcal{F}_h} [\![\mathbf{u}_{\mathbf{k}h}^* - \mathbf{u}_{\mathbf{k}h}]\!]_T \cdot \{\overline{\boldsymbol{\pi}}\} \, dA. \end{aligned}$$

Define the lifting operators
$\mathcal{L} : [L^2(\mathcal{F}_h)]^3 \to \Sigma_h$ and $\mathcal{R} : [L^2(\mathcal{F}_h)]^3 \to \Sigma_h$ as

$$\int_\Omega \mathcal{L}(\mathbf{u}) \cdot \mathbf{v} \, dx = \int_{\mathcal{F}_h^i} \mathbf{u} \cdot [\![\mathbf{v}]\!]_T \, dA, \qquad \forall \mathbf{v} \in \Sigma_h,$$

$$\int_{\Omega} \mathcal{R}(\mathbf{u}) \cdot \mathbf{v}\, dx = \int_{\mathcal{F}_h} \mathbf{u} \cdot \{\mathbf{v}\}\, dA, \qquad \forall \mathbf{v} \in \Sigma_h.$$

Then $\mathbf{q}$ can be expressed as

$$\mathbf{q}_h = \nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h} - \mathcal{L}(\{\mathbf{u}_{\mathbf{k}h}^* - \mathbf{u}_{\mathbf{k}h}\}) + \mathcal{R}([\![\mathbf{u}_{\mathbf{k}h}^* - \mathbf{u}_{\mathbf{k}h}]\!]_T). \tag{4.3}$$

Now we will look again at Equation (4.1). We want to manipulate it such that (4.3) can be inserted easily. In order to do that, integrate by parts and recall the manipulations of Section 3.3.1.

$$\begin{aligned}
\int_{\Omega} (\nabla_{\mathbf{k}} \times \epsilon^{-1}\mathbf{q}_h) \cdot \overline{\boldsymbol{\phi}}\, dx &= \int_{\Omega} (\nabla_h \times \epsilon^{-1}\mathbf{q}_h) \cdot \overline{\boldsymbol{\phi}} + (i\mathbf{k} \times \epsilon^{-1}\mathbf{q}_h) \cdot \overline{\boldsymbol{\phi}}\, dx \\
&= \int_{\Omega} \epsilon^{-1}\mathbf{q}_h \cdot (\nabla_{\mathbf{k}} \times \overline{\boldsymbol{\phi}}) + (\overline{\boldsymbol{\phi}} \times i\mathbf{k}) \cdot \epsilon^{-1}\mathbf{q}\, dx \\
&+ \sum_{K \in \mathcal{T}_h} \int_{\partial K} \Big(\mathbf{n} \times \big(\epsilon^{-1}\mathbf{q}_h\big)^*\Big) \cdot \overline{\boldsymbol{\phi}}\, dA \\
&= \int_{\Omega} \epsilon^{-1}\mathbf{q}_h \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi})} + \epsilon^{-1}\mathbf{q}_h \cdot \overline{(i\mathbf{k} \times \boldsymbol{\phi})}\, dx \\
&+ \sum_{K \in \mathcal{T}_h} \int_{\partial K} (\mathbf{n} \times \big(\epsilon^{-1}\mathbf{q}_h\big)^* \cdot \overline{\boldsymbol{\phi}}\, dA \\
&= \int_{\Omega} \epsilon^{-1}\mathbf{q}_h \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi})}\, dx - \int_{\mathcal{F}_h} \{\big(\epsilon^{-1}\mathbf{q}_h\big)^*\} \cdot \overline{[\![\boldsymbol{\phi}]\!]_T}\, dA \\
&+ \int_{\mathcal{F}_h} [\![\big(\epsilon^{-1}\mathbf{q}_h\big)^*]\!]_T \cdot \overline{\{\boldsymbol{\phi}\}}\, dA.
\end{aligned}$$

In this expression, $\mathbf{q}$ can be substituted easily. The resulting system is

$$\begin{aligned}
&\int_{\Omega} (\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi})}\, dx \\
&- \int_{\mathcal{F}_h} \{\mathbf{u}_{\mathbf{k}h}^* - \mathbf{u}_{\mathbf{k}h}\} \cdot [\![\overline{\epsilon^{-1}\nabla_{\mathbf{k}} \times \boldsymbol{\phi}}]\!]_T\, dA + \int_{\mathcal{F}_h} [\![\mathbf{u}_{\mathbf{k}h}^* - \mathbf{u}_{\mathbf{k}h}]\!]_T \cdot \{\overline{\epsilon^{-1}\nabla_{\mathbf{k}} \times \boldsymbol{\phi}}\}\, dA \\
&- \int_{\mathcal{F}_h} \{\big(\epsilon^{-1}\mathbf{q}_h\big)^*\} \cdot [\![\overline{\boldsymbol{\phi}}]\!]_T\, dA + \int_{\mathcal{F}_h} [\![\big(\epsilon^{-1}\mathbf{q}_h\big)^*]\!]_T \cdot \{\overline{\boldsymbol{\phi}}\}\, dA - \omega^2 \int_{\Omega} \mu \mathbf{u}_{\mathbf{k}h} \cdot \overline{\boldsymbol{\phi}}\, dx = \int_{\Omega} \tilde{\mathbf{J}} \cdot \overline{\boldsymbol{\phi}}\, dx.
\end{aligned}$$

The following fluxes will be used on any face $f \in \mathcal{F}_h$:

$$\begin{aligned}
\mathbf{u}_{\mathbf{k}h}^* &= \{\mathbf{u}_{\mathbf{k}h}\}, \\
\big(\epsilon^{-1}\mathbf{q}_h\big)^* &= \{\epsilon^{-1}\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h}\} - \alpha_f [\![\epsilon^{-1}\mathbf{u}_{\mathbf{k}h}]\!]_T,
\end{aligned}$$

in which α_f is a face-dependent penalty parameter which depends on the mesh size and polynomial order. The interior penalty discontinuous Galerkin method is then given by: Find $\mathbf{u}_{\mathbf{k}h} \in \Sigma_h$, such that for all $\boldsymbol{\phi} \in \Sigma_h$,

$$s_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) - a_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) + b_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) - \omega^2 m(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) = j(\boldsymbol{\phi}), \tag{4.4}$$

in which

$$\begin{aligned}
s_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) &= \int_{\Omega} \epsilon^{-1}(\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi})}\, dx, \\
a_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) &= \int_{\mathcal{F}_h} [\![\mathbf{u}_{\mathbf{k}h}]\!]_T \cdot \{\overline{\epsilon^{-1}(\nabla_{\mathbf{k}} \times \boldsymbol{\phi})}\} + \{\epsilon^{-1}(\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}h})\} \cdot [\![\overline{\boldsymbol{\phi}}]\!]_T\, dA,
\end{aligned}$$

$$
\begin{aligned}
b_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) &= \int_{\mathcal{F}_h} \alpha_f [\![\epsilon^{-1}\mathbf{u}_{\mathbf{k}h}]\!]_T \cdot [\![\overline{\boldsymbol{\phi}}]\!]_T \; dA, \\
m(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}) &= \int_{\Omega} \mu \mathbf{u}_{\mathbf{k}h} \cdot \overline{\boldsymbol{\phi}} \; dx, \\
j(\boldsymbol{\phi}) &= \int_{\Omega} \tilde{\mathbf{J}} \cdot \overline{\boldsymbol{\phi}} \; dx.
\end{aligned}
$$

4.3 Implementation

In this section, some implementation details for the IP-DG method for periodic media will be given. Since the problem is very similar to the problem without Bloch-modes, we only describe the parts that are different. So the mesh-generation and numerical integration will not be discussed in this section. Instead we start with writing system (4.4) as a linear system. After that, some details with respect to the adapted basis functions will be given. Finally, the implementation in a homogeneous medium is discussed.

4.3.1 Substitution

The first step to rewrite system (4.4) as a linear system is to substitute

$$
\mathbf{u}_{\mathbf{k}}|_K = \sum_{j=1}^{n_p} u_{j,K} \boldsymbol{\phi}_{j,K}(\mathbf{x}, \mathbf{k})
$$

in system (4.4). Next, we construct a linear system of equations for the DG discretization, so we would like to have a square matrix $[S]$, in which every row stands for a test function and every column stands for a basis function, and a vector $[J]$, such that in the end we have to solve $[S]\mathbf{u} = [J]$, in which u is the vector of coefficients $u_{j,K}$. If we are solving the eigenvalue problem, we would like to have matrices $[S]$ and $[M]$ such that $[S]\mathbf{u} = \omega^2[M]\mathbf{u}$. Since we have n_p basis functions for every element, the matrices will consist of $N_{elt} \times N_{elt}$ blocks of size $n_p \times n_p$. As before, with a block $[S]^{CD}$ we mean the C^{th} block of matrix $[S]$ vertically and the D^{th} block of $[S]$ horizontally. In the rest of the section, the bilinear forms from the last section will be converted to a matrix-system.

Bilinear forms for the elements

In this section, the bilinear forms that include element integrals will be expanded for every test function $\boldsymbol{\phi}_{i,K}$, so that these can be written as a linear combination of the coefficients $u_{j,K}$. The entries for the mass- and stiffness matrix can be computed by

$$
\begin{aligned}
m(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}_{i,K}) &= \int_K \mu \mathbf{u}_{\mathbf{k}} \cdot \overline{\boldsymbol{\phi}_{i,K}} \; dx \\
&= \int_K \left(\sum_{j=1}^{n_p} u_{j,K} \boldsymbol{\phi}_{j,K} \right) \cdot \overline{\boldsymbol{\phi}_{i,K}} \; dx \\
&= \sum_{j=1}^{n_p} u_{j,K} \int_K \boldsymbol{\phi}_{j,K} \cdot \overline{\boldsymbol{\phi}_{i,K}} \; dx, \\
s_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}_{i,K}) &= \int_K \epsilon^{-1} (\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})} \; dx \\
&= \int_K \epsilon^{-1} \left(\nabla_{\mathbf{k}} \times \left(\sum_{j=1}^{n_p} u_{j,K} \boldsymbol{\phi}_{j,K} \right) \right) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})} \; dx
\end{aligned}
$$

$$= \sum_{j=1}^{n_p} u_{j,K} \int_K \epsilon^{-1}(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{j,K}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})}\, dx.$$

A natural choice to represent these equations are the element matrices $\tilde{\mathcal{M}}_K$ and $\tilde{\mathcal{S}}_K$, in which

$$\begin{aligned} \tilde{\mathcal{M}}_K^{ij} &= \int_K \mu \boldsymbol{\phi}_{j,K} \cdot \overline{\boldsymbol{\phi}_{i,K}}\, dx, \\ \tilde{\mathcal{S}}_K^{ij} &= \int_K \epsilon_K^{-1}(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{j,K}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})}\, dx. \end{aligned}$$

To insert these local matrices into the global matrices, loop over the elements $K \in \mathcal{T}_h$ and perform for the standard problem

$$[S]^{KK} \leftarrow \tilde{\mathcal{S}}_K - \omega^2 \tilde{\mathcal{M}}_K,$$

and for the eigenproblem

$$\begin{aligned} [S]^{KK} &\leftarrow \tilde{\mathcal{S}}_K, \\ [M]^{KK} &\leftarrow \tilde{\mathcal{M}}_K. \end{aligned}$$

Bilinear forms for the faces

In this section, the bilinear forms that include integrals over the faces will be discussed. For convenience, every face is said to have an element on the left, which is denoted by K_L, and an element on the right, K_R. The choice for which element is left and which is right is arbitrary and in practice the element with the lowest index will be the left one. Note that by assumption, the dielectric constant ϵ_K and the basis functions $\boldsymbol{\psi}_{j,K}$ from Section 3.2 are real for all elements. The bilinear form $a_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}_{i,K})$ can be represented by

$$\begin{aligned} a_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}}, \boldsymbol{\phi}_{i,K}) &= \sum_{f \in \partial K} \int_f [\![\mathbf{u}_{\mathbf{k}}]\!]_T \cdot \{\overline{\epsilon^{-1}(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})}\} + \{\epsilon^{-1}(\nabla_{\mathbf{k}} \times \mathbf{u}_{\mathbf{k}})\} \cdot [\![\overline{\boldsymbol{\phi}_{i,K}}]\!]_T\, dA \\ &= \sum_{\substack{f \in \partial K, \\ K = K_L}} \int_f \left((\mathbf{n}_L \times {\mathbf{u}_{\mathbf{k}}}^L) + (\mathbf{n}_R \times {\mathbf{u}_{\mathbf{k}}}^R)\right) \cdot \overline{\frac{1}{2}\left(\epsilon_L^{-1}(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})\right)}\, dA \\ &\quad + \sum_{\substack{f \in \partial K, \\ K = K_R}} \int_f \left((\mathbf{n}_L \times {\mathbf{u}_{\mathbf{k}}}^L) + (\mathbf{n}_R \times {\mathbf{u}_{\mathbf{k}}}^R)\right) \cdot \overline{\frac{1}{2}\left(\epsilon_R^{-1}(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})\right)}\, dA \\ &\quad + \sum_{\substack{f \in \partial K \\ K = K_L}} \int_f \frac{1}{2}\left(\epsilon_L^{-1}(\nabla_{\mathbf{k}} \times {\mathbf{u}_{\mathbf{k}}}^L) + \epsilon_R^{-1}(\nabla_{\mathbf{k}} \times {\mathbf{u}_{\mathbf{k}}}^R)\right) \cdot \overline{\left(\mathbf{n}_L \times \boldsymbol{\phi}_{i,K}\right)}\, dA \\ &\quad + \sum_{\substack{f \in \partial K \\ K = K_R}} \int_f \frac{1}{2}\left(\epsilon_L^{-1}(\nabla_{\mathbf{k}} \times {\mathbf{u}_{\mathbf{k}}}^L) + \epsilon_R^{-1}(\nabla_{\mathbf{k}} \times {\mathbf{u}_{\mathbf{k}}}^R)\right) \cdot \overline{\left(\mathbf{n}_R \times \boldsymbol{\phi}_{i,K}\right)}\, dA \\ &= \sum_{\substack{f \in \partial K, \\ K = K_L}} \sum_{j=1}^{n_p} u_{j,K_L} \int_f \frac{1}{2}\epsilon_L^{-1}(\mathbf{n}_L \times \boldsymbol{\phi}_{j,K_L}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})}\, dA \\ &\quad + \sum_{\substack{f \in \partial K, \\ K = K_L}} \sum_{j=1}^{n_p} u_{j,K_L} \int_f \frac{1}{2}\epsilon_L^{-1}(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{j,K_L}) \cdot \overline{(\mathbf{n}_L \times \boldsymbol{\phi}_{i,K})}\, dA \\ &\quad + \sum_{\substack{f \in \partial K, \\ K = K_L}} \sum_{j=1}^{n_p} u_{j,K_R} \int_f \frac{1}{2}\epsilon_L^{-1}(\mathbf{n}_R \times \boldsymbol{\phi}_{j,K_R}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K})}\, dA \end{aligned}$$

$$
\begin{aligned}
&+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \frac{1}{2}\epsilon_R^{-1}(\nabla_{\mathbf{k}}\times\boldsymbol{\phi}_{j,K_R})\cdot\overline{(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})}\,dA\\
&+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f \frac{1}{2}\epsilon_R^{-1}(\mathbf{n}_L\times\boldsymbol{\phi}_{j,K_L})\cdot\overline{(\nabla_{\mathbf{k}}\times\boldsymbol{\phi}_{i,K})}\,dA\\
&+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f \frac{1}{2}\epsilon_L^{-1}(\nabla_{\mathbf{k}}\times\boldsymbol{\phi}_{j,K_L})\cdot\overline{(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})}\,dA\\
&+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \frac{1}{2}\epsilon_R^{-1}(\mathbf{n}_R\times\boldsymbol{\phi}_{j,K_R})\cdot\overline{(\nabla_{\mathbf{k}}\times\boldsymbol{\phi}_{i,K})}\,dA\\
&+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \frac{1}{2}\epsilon_R^{-1}(\nabla_{\mathbf{k}}\times\boldsymbol{\phi}_{j,K_R})\cdot\overline{(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})}\,dA.
\end{aligned}
$$

From this last expression, it follows how this bilinear form can be implemented for all test functions: for every face define the matrices

$$
\tilde{\mathcal{A}}^{ij}_{CD}=\int_f \frac{1}{2}\epsilon_C^{-1}(\mathbf{n}_D\times\boldsymbol{\phi}_{j,K_D})\cdot\overline{(\nabla_{\mathbf{k}}\times\boldsymbol{\phi}_{i,K_C})}\,dA,
$$

for $C,D\in\{L,R\}$ and $i,j=1,...,n_p$. This results in four small, square matrices for every face. To add them to the global matrix, perform for every face

$$
\begin{aligned}
[S]^{LL} &\leftarrow -\tilde{\mathcal{A}}_{LL}-\tilde{\mathcal{A}}^*_{LL},\\
[S]^{LR} &\leftarrow -\tilde{\mathcal{A}}_{LR}-\tilde{\mathcal{A}}^*_{RL},\\
[S]^{RL} &\leftarrow -\tilde{\mathcal{A}}_{RL}-\tilde{\mathcal{A}}^*_{LR},\\
[S]^{RR} &\leftarrow -\tilde{\mathcal{A}}_{RR}-\tilde{\mathcal{A}}^*_{RR},
\end{aligned}
$$

in which the * stands for the hermitian transpose of the matrix.
Doing the same for $b_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}},\boldsymbol{\phi}_{i,K})$ leads to

$$
\begin{aligned}
b_{\mathbf{k}}(\mathbf{u}_{\mathbf{k}},\boldsymbol{\phi}_{i,K}) &= \sum_{f\in\partial K}\int_f \alpha_f[\![\epsilon^{-1}\mathbf{u}_{\mathbf{k}}]\!]_T\cdot\overline{[\![\boldsymbol{\phi}]\!]}_T\,dA\\
&=\sum_{\substack{f\in\partial K,\\ K=K_L}}\int_f \alpha_f\left((\mathbf{n}_L\times\epsilon_L^{-1}\mathbf{u_k}^L)+(\mathbf{n}_R\times\epsilon_R^{-1}\mathbf{u_k}^R)\right)\cdot\overline{(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})}\,dA\\
&\quad+\sum_{\substack{f\in\partial K,\\ K=K_R}}\int_f \alpha_f\left((\mathbf{n}_L\times\epsilon_L^{-1}\mathbf{u_k}^L)+(\mathbf{n}_R\times\epsilon_R^{-1}\mathbf{u_k}^R)\right)\cdot\overline{(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})}\,dA\\
&=\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f \alpha_f\epsilon_L^{-1}(\mathbf{n}_L\times\boldsymbol{\phi}_{j,K_L})\cdot\overline{(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})}\,dA\\
&\quad+\sum_{\substack{f\in\partial K,\\ K=K_L}}\sum_{j=1}^{n_p} u_{j,K_R}\int_f \alpha_f\epsilon_R^{-1}(\mathbf{n}_R\times\boldsymbol{\phi}_{j,K_R})\cdot\overline{(\mathbf{n}_L\times\boldsymbol{\phi}_{i,K})}\,dA\\
&\quad+\sum_{\substack{f\in\partial K,\\ K=K_R}}\sum_{j=1}^{n_p} u_{j,K_L}\int_f \alpha_f\epsilon_L^{-1}(\mathbf{n}_L\times\boldsymbol{\phi}_{j,K_L})\cdot\overline{(\mathbf{n}_R\times\boldsymbol{\phi}_{i,K})}\,dA
\end{aligned}
$$

$$+ \sum_{\substack{f \in \partial K, \\ K = K_R}} \sum_{j=1}^{n_p} u_{j,K_R} \int_f \alpha_f \epsilon_R^{-1} (\mathbf{n}_R \times \boldsymbol{\phi}_{j,K_R}) \cdot \overline{(\mathbf{n}_R \times \boldsymbol{\phi}_{i,K})} \, dA.$$

So, for every face, define the matrices

$$\tilde{\mathcal{B}}_{CD}^{ij} = \int_f \alpha_f \epsilon_D^{-1} (\mathbf{n}_D \times \boldsymbol{\phi}_{j,K_D}) \cdot \overline{(\mathbf{n}_C \times \boldsymbol{\phi}_{i,K})} \, dA,$$

for $C, D \in \{L, R\}$ and $i, j = 1, ..., n_p$. Again this results in four small, square matrices. To add them to the global matrix, perform for every face

$$\begin{aligned} [S]^{LL} &\leftarrow \tilde{\mathcal{B}}_{LL}, \\ [S]^{LR} &\leftarrow \tilde{\mathcal{B}}_{LR}, \\ [S]^{RL} &\leftarrow \tilde{\mathcal{B}}_{RL}, \\ [S]^{RR} &\leftarrow \tilde{\mathcal{B}}_{RR}. \end{aligned}$$

Now that we know which matrices should be computed, the next sections will be devoted to computing these small element- and face matrices.

4.3.2 Element matrices

In this section, the expressions for the element matrices $\tilde{\mathcal{M}}_K$ and $\tilde{\mathcal{S}}_K$ will be simplified for the modified basis functions. Starting with the mass-matrix, we have

$$\begin{aligned} \tilde{\mathcal{M}}_K^{ij} &= \int_K \mu \boldsymbol{\phi}_{j,K} \cdot \overline{\boldsymbol{\phi}_{i,K}} \, dx \\ &= \int_K \mu e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s}_j)} \boldsymbol{\psi}_{j,K} \cdot \overline{e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s}_i)} \boldsymbol{\psi}_{i,K}} \, dx \\ &= \int_K \mu e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i)} \boldsymbol{\psi}_{j,K} \cdot \overline{\boldsymbol{\psi}_{i,K}} \, dx \\ &= e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i)} \int_K \mu \boldsymbol{\psi}_{j,K} \cdot \boldsymbol{\psi}_{i,K} \, dx. \end{aligned}$$

Note that $\boldsymbol{\psi}$ is a real-valued function, so $\boldsymbol{\psi} = \overline{\boldsymbol{\psi}}$.
So to compute the element mass-matrices, compute the element mass matrices with the standard basis functions and then multiply them with the exponent term with the appropriate shifts. The computation of the element stiffness matrix is equivalent:

$$\begin{aligned} \tilde{\mathcal{S}}_K^{ij} &= \int_\Omega \epsilon^{-1} (\nabla \times \boldsymbol{\phi}_{j,K}) \cdot \overline{(\nabla \times \boldsymbol{\phi}_{i,K})} \, dx \\ &= \int_K \epsilon^{-1} (\nabla_{\mathbf{k}} \times e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s}_j)} \boldsymbol{\psi}_{j,K}) \cdot \overline{(\nabla_{\mathbf{k}} \times e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s}_i)} \boldsymbol{\psi}_{i,K})} \, dx \\ &= \int_K \epsilon^{-1} e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i)} (\nabla \times \boldsymbol{\psi}_{j,K}) \cdot \overline{(\nabla \times \boldsymbol{\psi}_{i,K})} \, dx \\ &= e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i)} \int_K \epsilon^{-1} (\nabla \times \boldsymbol{\psi}_{j,K}) \cdot (\nabla \times \boldsymbol{\psi}_{i,K}) \, dx. \end{aligned}$$

So the computation of the element matrices in case of periodic media is just the computation of the standard element matrices, multiplied by some exponent term.

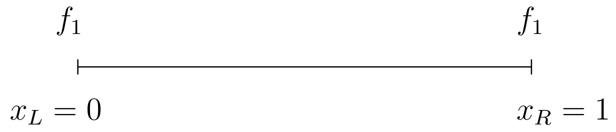


Figure 4.1: Boundary face for a 1D-domain

4.3.3 Face matrices

In this section, the face matrices $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$ will be simplified for the modified basis functions. Here, the expressions for the interior faces and the faces at the boundary are not the same, since $\mathbf{x}$ has two different values for each point at a face at the boundary. To show this, it is easiest to first look at one dimension, for example, look at the sitation in Figure 4.1. As one can see, face f_1 is a boundary face. In case of periodic boundary conditions, the left element for this face is the most left element, K_1, and the right element is the most right element, K_N. For K_1, the value of x at this face equals 0. For K_N, the value of x at this face equals 1. So define $x_L = 0$ to be the value of x of the left element and $x_R = 1$ the value of x of the right element.

This can be generalized for higher dimensions. Introduce the notation $\mathbf{x}_L$ and $\mathbf{x}_R$ for the value of $\mathbf{x}$ at the face on the left element and the face on the right element respectively. So $\mathbf{x}_R = \mathbf{x}_L + \mathbf{a}$, for some $\mathbf{a}$ dependent on the size of the domain.
First compute the face integrals for interior faces:

$$\begin{aligned}
\tilde{\mathcal{A}}^{ij}_{CD} &= \int_f \frac{1}{2}\epsilon_C^{-1}(\mathbf{n}_D \times \boldsymbol{\phi}_{j,K_D}) \cdot \overline{(\nabla_{\mathbf{k}} \times \boldsymbol{\phi}_{i,K_C})}\, dA \\
&= \int_f \frac{1}{2}\epsilon_C^{-1}(\mathbf{n}_D \times e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s}_j)}\boldsymbol{\psi}_{j,K_D}) \cdot \overline{(\nabla_{\mathbf{k}} \times e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s}_i)}\boldsymbol{\phi}_{i,K_C})}\, dA \\
&= \frac{1}{2} e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i)} \int_f \epsilon_C^{-1}(\mathbf{n}_D \times \boldsymbol{\psi}_{j,K_D}) \cdot (\nabla \times \boldsymbol{\psi}_{i,K_C})\, dA,
\end{aligned}$$

$$\tilde{\mathcal{B}}^{ij}_{CD} = \alpha_f\; e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i)} \int_f \epsilon_D^{-1}(\mathbf{n}_D \times \boldsymbol{\psi}_{j,K_D}) \cdot (\mathbf{n}_C \times \boldsymbol{\psi}_{i,K_C})\, dA.$$

For boundary faces, the expressions for $\tilde{\mathcal{A}}_{LL}, \tilde{\mathcal{A}}_{RR}, \tilde{\mathcal{B}}_{LL}$ and $\tilde{\mathcal{B}}_{RR}$ are the same as their respective expressions at interior faces. The face integrals across the boundaries are given by

$$\begin{aligned}
\tilde{\mathcal{A}}^{ij}_{LR} &= \int_f \frac{1}{2}\epsilon_L^{-1} \left(\mathbf{n}_R \times e^{-i\mathbf{k}\cdot(\mathbf{x}_R-\mathbf{s}_j)}\boldsymbol{\psi}_{j,K_R}\right) \cdot \overline{\left(\nabla_{\mathbf{k}} \times e^{-i\mathbf{k}\cdot(\mathbf{x}_L-\mathbf{s}_i)}\boldsymbol{\psi}_{i,K_L}\right)}\, dA \\
&= \int_f \frac{1}{2}\epsilon_L^{-1} \left(\mathbf{n}_R \times e^{-i\mathbf{k}\cdot(\mathbf{x}_L+\mathbf{a}-\mathbf{s}_j)}\boldsymbol{\psi}_{j,K_R}\right) \cdot \overline{\left(\nabla_{\mathbf{k}} \times e^{-i\mathbf{k}\cdot(\mathbf{x}_L-\mathbf{s}_i)}\boldsymbol{\psi}_{i,K_L}\right)}\, dA \\
&= \int_f \frac{1}{2}\epsilon_L^{-1} e^{-i\mathbf{k}\cdot(\mathbf{x}_L+\mathbf{a}-\mathbf{s}_j-\mathbf{x}_L+\mathbf{s}_i)} \left(\mathbf{n}_R \times \boldsymbol{\psi}_{j,K_R}\right) \cdot \overline{\left(\nabla \times \boldsymbol{\psi}_{i,K_L}\right)}\, dA \\
&= \frac{1}{2} e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i-\mathbf{a})} \int_f \epsilon_L^{-1} \left(\mathbf{n}_R \times \boldsymbol{\psi}_{j,K_R}\right) \cdot \left(\nabla \times \boldsymbol{\psi}_{i,K_L}\right)\, dA,
\end{aligned}$$

$$\begin{aligned}
\tilde{\mathcal{A}}^{ij}_{RL} &= \int_f \frac{1}{2}\epsilon_R^{-1} \left(\mathbf{n}_L \times e^{-i\mathbf{k}\cdot(\mathbf{x}_L-\mathbf{s}_j)}\boldsymbol{\psi}_{j,K_L}\right) \cdot \overline{\left(\nabla_{\mathbf{k}} \times e^{-i\mathbf{k}\cdot(\mathbf{x}_R-\mathbf{s}_i)}\boldsymbol{\psi}_{i,K_R}\right)}\, dA \\
&= \frac{1}{2} e^{i\mathbf{k}\cdot(\mathbf{s}_j-\mathbf{s}_i+\mathbf{a})} \int_f \epsilon_R^{-1} \left(\mathbf{n}_L \times \boldsymbol{\psi}_{j,K_L}\right) \cdot \left(\nabla \times \boldsymbol{\psi}_{i,K_R}\right)\, dA,
\end{aligned}$$

$$\tilde{\mathcal{B}}^{ij}_{LR} = \int_f \alpha_f \epsilon_R^{-1} \left(\mathbf{n}_R \times e^{-i\mathbf{k}\cdot(\mathbf{x}_R-\mathbf{s}_j)}\boldsymbol{\psi}_{j,K_R}\right) \cdot \overline{\left(\mathbf{n}_L \times e^{-i\mathbf{k}\cdot(\mathbf{x}_L-\mathbf{s}_i)}\boldsymbol{\psi}_{i,K_L}\right)}\, dA$$

$$= \alpha_f \ e^{i\mathbf{k}\cdot(\mathbf{s}_j - \mathbf{s}_i - \mathbf{a})} \int_f \alpha_f \epsilon_R^{-1} \left(\mathbf{n}_R \times \boldsymbol{\psi}_{j,K_R}\right) \cdot \left(\mathbf{n}_L \times \boldsymbol{\psi}_{i,K_L}\right) \ dA,$$

$$\tilde{\mathcal{B}}_{RL}^{ij} = \int_f \alpha_f \epsilon_L^{-1} \left(\mathbf{n}_L \times e^{-i\mathbf{k}\cdot(\mathbf{x}_L - \mathbf{s}_j)} \boldsymbol{\psi}_{j,K_L}\right) \cdot \overline{\left(\mathbf{n}_R \times e^{-i\mathbf{k}\cdot(\mathbf{x}_R - \mathbf{s}_i)} \boldsymbol{\psi}_{i,K_R}\right)} \ dA$$

$$= \alpha_f \ e^{i\mathbf{k}\cdot(\mathbf{s}_j - \mathbf{s}_i + \mathbf{a})} \int_f \alpha_f \epsilon_L^{-1} \left(\mathbf{n}_L \times \boldsymbol{\psi}_{j,K_L}\right) \cdot \left(\mathbf{n}_R \times \boldsymbol{\psi}_{i,K_R}\right) \ dA.$$

4.3.4 Right-hand side

The only term that is left is the integral over the source term multiplied with a basis function. This can be expressed as follows:

$$\begin{aligned} j(\boldsymbol{\phi}_{i,K}) &= \int_\Omega \tilde{\mathbf{J}} \cdot \overline{\boldsymbol{\phi}_{i,K}} \ dx \\ &= \int_K \tilde{\mathbf{J}} \cdot \overline{e^{-i\mathbf{k}\cdot(\mathbf{x}-\mathbf{s}_i)} \boldsymbol{\psi}_{i,K}} \ dx \\ &= \int_{\hat{K}} e^{-i\mathbf{k}\cdot(F_K(\boldsymbol{\xi}) - \mathbf{s}_i)} \mathbf{J}(F_K(\boldsymbol{\xi})) \cdot (dF_K)^{-T} \boldsymbol{\psi} |\det(dF_K)| \ dx. \end{aligned} \tag{4.5}$$

4.3.5 Eigenvalue problem

In the eigenvalue problem, we want to compute the smallest eigenvalues for many different values of $\mathbf{k}$. Since the matrices can be constructed by first constructing the matrices for $\mathbf{k} = 0$ and then multiplying every entry with an exponent term, we first compute and save the matrices computed for $\mathbf{k} = 0$. From [13] we know that we only have to compute the eigenvalues for $\mathbf{k}$ on the edges of the irreducible Brillouin zone in order to see whether there is a band gap or not, so the next step is to vary $\mathbf{k}$ over the edges of the irreducible Brillouin zone with small steps. Then the global matrices are computed and finally, the first 10 non-zero eigenvalues are computed. In our case we use a shift-and-invert algorithm to find eigenvalues with value greater than 0, although it would be better to develop an algorithm to get rid of the kernel of the curl-operator before solving the linear eigenvalue problem.

4.3.6 Implementation for a homogeneous medium

In this section, it will be discussed how to implement the DG-discretization for the 0-dimensional periodic medium by adapting the code of the standard Maxwell solver. The computations are done on the domain $\Omega = [0,1]^3$.

Computing the shifts

For the shifts, we choose to define a shift for each "type" of basis functions. Given an edge $e = (\mathbf{v}_i, \mathbf{v}_j)$, a face $f = (\mathbf{v}_i, \mathbf{v}_j, \mathbf{v}_l)$ or an element $K = (\mathbf{v}_i, \mathbf{v}_j, \mathbf{v}_l, \mathbf{v}_m)$ The shifts are defined as

- **Edge functions**: $\mathbf{s_e} = \frac{1}{2}(\mathbf{v}_i + \mathbf{v}_j)$
- **Edge-based face functions**: $\mathbf{s_{ef}} = \frac{1}{5}(2\mathbf{v}_i + 2\mathbf{v}_j + \mathbf{v}_l)$
- **Face bubble functions**: $\mathbf{s_{fb}} = \frac{1}{3}(\mathbf{v}_i + \mathbf{v}_j + \mathbf{v}_l)$
- **Face-based interior functions**: $\mathbf{s_{fi}} = \frac{1}{7}(2\mathbf{v}_i + 2\mathbf{v}_j + 2\mathbf{v}_l + \mathbf{v}_m)$
- **Interior bubble functions**: $\mathbf{s_{ib}} = \frac{1}{4}(\mathbf{v}_i + \mathbf{v}_j + \mathbf{v}_l + \mathbf{v}_m)$

The use of the shifts is optional. If the shifts are not used, the results are the same as with this shifts. These shifts can be used as a preconditioner and to estimate the eigenvectors, in order to find a good starting vector. Both of these applications are not further investigated in this project.

Adapting the right-hand side

First of all, the boundary conditions are periodic on all boundaries. The basis functions now contain an exponent term, so the integration of the source is a bit different. The transformation of the integrals of the source to an integral over the reference element is given in (4.5).

Adapting the matrices

All element and face matrices can be obtained by multiplying them with a complex exponent. Since this exponent only depends on the basis functions, this will be done as a last step before solving the system, namely after the assembly of the global matrix.

Retrieving the magnetic field

As a result, one will find the DG coefficient vector $\mathbf{u}$. From here, one should be careful to include the exponent term when computing $\mathbf{u_k}|_K = \sum_{j=1}^{n_p} u_{j,K}\boldsymbol{\phi}_{j,K}(\mathbf{x},\mathbf{k})$. The magnetic field can now be found by computing $\mathbf{H}(\mathbf{x}) = e^{i\mathbf{k}\cdot\mathbf{x}}\mathbf{u_k}(\mathbf{x})$.

4.4 Results

In this section, some results for the infinitely large homogeneous domain are given. First, we will look at the problem with source and check whether it works correctly. After that, the eigenvalue problem will be solved.

4.4.1 Standard problem

In this section, some results for the standard problem with Bloch-modes in a homogeneous medium wit $\epsilon = \mu = 1$ are given. We want to test for a periodic field $\mathbf{u_k}$, so we choose $\Omega = [0,1]^3$ and

$$\mathbf{u_k}(x,y,z) = \begin{pmatrix} \sin(2\pi y)\sin(2\pi z) \\ \sin(2\pi z)\sin(2\pi x) \\ \sin(2\pi x)\sin(2\pi y) \end{pmatrix}.$$

Computing the source $\mathbf{J} = \nabla_\mathbf{k} \times \nabla_\mathbf{k} \times \mathbf{u_k} - \omega^2\mathbf{u_k}$ and choosing $\omega^2 = 1$ leads to the source $\tilde{\mathbf{J}} = (\tilde{J}^{(1)}, \tilde{J}^{(2)}, \tilde{J}^{(3)})^T$, with the components of $\tilde{\mathbf{J}}$ given by

$$\begin{aligned}
\tilde{J}^{(1)} = {} & (8\pi^2 - 1)\sin(2\pi y)\sin(2\pi z) - 4\pi i\ k_y\cos(2\pi y)\sin(2\pi z) - 4\pi i\ k_z\cos(2\pi z)\sin(2\pi y) \\
& + 2\pi i\ k_y\cos(2\pi x)\sin(2\pi z) + 2\pi i\ k_z\cos(2\pi x)\sin(2\pi y) - k_yk_x\sin(2\pi x)\sin(2\pi z) \\
& - k_zk_x\sin(2\pi x)\sin(2\pi y) + k_y^2\sin(2\pi y)\sin(2\pi z) + k_z^2\sin(2\pi y)\sin(2\pi z),
\end{aligned}$$

$$\begin{aligned}
\tilde{J}^{(2)} = {} & (8\pi^2 - 1)\sin(2\pi x)\sin(2\pi z) - 4\pi i\ k_x\cos(2\pi x)\sin(2\pi z) - 4\pi i\ k_z\cos(2\pi z)\sin(2\pi x) \\
& + 2\pi i\ k_x\cos(2\pi y)\sin(2\pi z) + 2\pi i\ k_z\cos(2\pi y)\sin(2\pi x) - k_yk_x\sin(2\pi y)\sin(2\pi z) \\
& - k_zk_y\sin(2\pi x)\sin(2\pi y) + k_x^2\sin(2\pi x)\sin(2\pi z) + k_z^2\sin(2\pi x)\sin(2\pi z),
\end{aligned}$$

$$\tilde{J}^{(3)} = (8\pi^2 - 1)\sin(2\pi y)\sin(2\pi x) - 4\pi i\ k_y\cos(2\pi y)\sin(2\pi x) - 4\pi i\ k_x\cos(2\pi x)\sin(2\pi y)$$

	$\|\mathbf{u_k} - \mathbf{u_{k}}_h\|_{L^2}$	Order	$\|\mathbf{u_k} - \mathbf{u_{k}}_h\|_{DG}$	Order
$p = 1$				
$N_{elt} = 40$	6.0382e-01		7.1285e+00	
$N_{elt} = 320$	1.9534e-01	1.63	3.7531e+00	0.93
$N_{elt} = 2560$	5.4780e-02	1.83	1.9634e+00	0.93
$p = 2$				
$N_{elt} = 40$	1.7623e-01		3.1010e+00	
$N_{elt} = 320$	3.1270e-02	2.49	1.0430e+00	1.57
$N_{elt} = 2560$	4.1910e-03	2.90	2.7653e-01	1.92
$p = 3$				
$N_{elt} = 40$	7.8843e-02		1.7102e+00	
$N_{elt} = 320$	4.9380e-03	4.00	2.1352e-01	3.00
$N_{elt} = 2560$	3.1975e-04	3.95	2.8026e-02	2.93
$p = 4$				
$N_{elt} = 40$	2.1295e-02		3.5137e-01	
$N_{elt} = 320$	6.7123e-04	4.99	3.5095e-02	3.32
$N_{elt} = 2560$	2.1835e-05	4.94	2.3089e-03	3.93

Table 4.1: Convergence of IP-DG method on tetrahedral meshes with source $\tilde{\mathbf{J}}$ and $\mathbf{k} = (1, -1/2, 1/4)$.

$$+ 2\pi i\ k_y \cos(2\pi z)\sin(2\pi x) + 2\pi i\ k_x \cos(2\pi z)\sin(2\pi y) - k_y k_z \sin(2\pi z)\sin(2\pi x)$$
$$- k_z k_x \sin(2\pi z)\sin(2\pi y) + k_y^2 \sin(2\pi y)\sin(2\pi x) + k_x^2 \sin(2\pi y)\sin(2\pi x).$$

The penalty parameter is again chosen to be

$$\alpha_f = \frac{2(p+1)(p+3)}{\min(dx, dy, dz)}.$$

The solution for $\mathbf{u_k}$ is smooth, so we expect again a convergence rate of $O(h^{p+1})$. In Table 4.1, the results for the wave vector $\mathbf{k} = (1, -1/2, 1/4)$ are given. As can be seen in this table, the convergence rate is correct.

The results for different wave vectors are similar, although the errors become bigger when $\mathbf{k}$ gets larger, since the Gauss-quadrature rules are not very suitable for high-oscillating functions. One could use other integration rules, such as Filon's formula [1], but these rules are outside the scope of this project. If one would like to use these integrals and still use the Gauss quadrature rules, one could for example make the domain bigger, so that the Brillouin zone becomes smaller and therefore the values of the wave vector that needs to be checked are smaller.

4.4.2 Eigenvalue problem

In this section, some results for Bloch mode computations in a homogeneous medium are presented. The domain is given by $\Omega = [0, 1]^3$ and $\epsilon = \mu = 1$ for the entire domain. The penalty parameter is chosen to be

$$\alpha_f = \frac{2(p+1)(p+3)}{\min(dx, dy, dz)}.$$

For the polynomial order, we choose $p = 4$. This results in the fastest convergence, since the eigenfunctions are smooth for this problem. In Section 3.5.2, it can be seen that the relative error of the eigenvalues for $\mathbf{k} = 0$ are about 10^{-7} for a mesh of 320 elements, so this

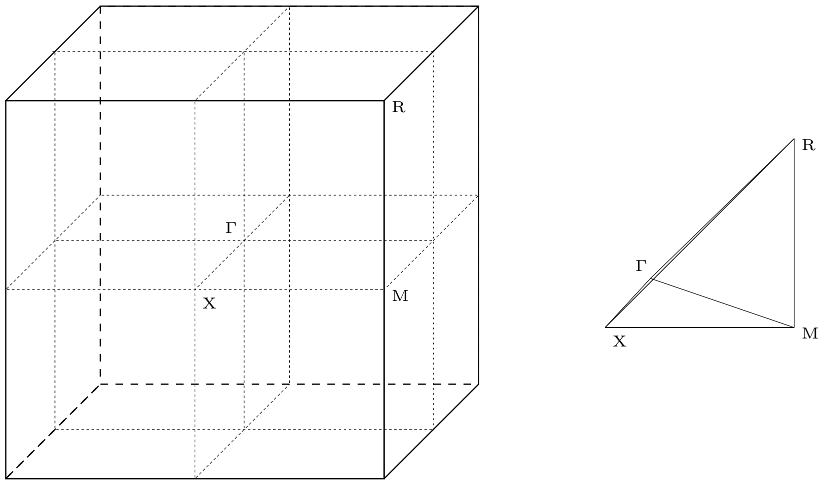


Figure 4.2: Brillouin zone (left) and reduced Brillouin zone (right)

mesh should be small enough to compute sufficiently accurate eigenvalues for all values of $\mathbf{k}$.

The Brillouin zone is the cube $\mathcal{B} = [-\pi, \pi]^3$, and the irreducible Brillouin zone is the tetrahedron with vertices $\Gamma = (0, 0, 0)^T$, X= $(\pi, 0, 0)^T$, M= $(\pi, \pi, 0)^T$ and R= $(\pi, \pi, \pi)^T$, see Figure 4.2. The first ten non-zero eigenvalues for $\mathbf{k}$ on the edges of the irreducible Brillouin zone with increments of $\pi/25$ are computed. To find the eigenvalues, Matlab's iterative eigensolver "eigs" is used with as starting vector the first eigenvector of the $\mathbf{k}$ from the previous step is used. Unfortunately, it is only possible to give one starting vector for all eigenvectors, instead of a starting vector for each eigenvector.

The eigenvalues generated with the discontinuous Galerkin finite element method are plotted in Figure 4.3. These eigenvalues are compared with the results of Bulovyatov, [8], in which the eigenvalues are computed with a curl-conforming finite element method, and with the MIT photonic bands (MPB) software by Johnson and Joannapoulos, [14], in which the eigenvalues are computed with a plane-wave method. The bands are displayed in Figure 4.4.

For the MPB-software, we used the default settings with 24 interior points on each edge, the same number as we used for our software to generate Figure 4.3. The results can be seen in Figure 4.4a. The MPB-software generates almost the same bands as our DG-software, with the exception that it generates some peaks and the multiplicity of two bands change between $\mathbf{k} = (\pi, 0, 0)^T$ and $\mathbf{k} = (\pi, \pi, 0)^T$. Since the bands of eigenvalues should be smooth when $\mathbf{k}$ is smooth ([13]), this is probably a flaw in the software and the eigenvalues are most likely computed incorrectly at these points.

Looking at the band-structure of Bulovyatov in Figure 4.4b, we first have to observe that he only uses 3 interior points on each edge of the irreducible Brillouin zone, and plotted smooth curves through these points afterward. He used first order curl-conforming basis functions on 4096 cubic elements. Note that using first order basis functions is less efficient than higher order functions when dealing with smooth eigenfunctions, see Section 3.5.1.

We see that at the points where the eigenvalues are computed, the eigenvalues are the same as our eigenvalues, but that the behavior of the bands outside these points is slighly different. Since $\mathbf{k}$ is not smooth at the corners of the reduced Brillouin zone, we do not expect the bands to be smooth at these points. Also, the first band has a larger value than the second band at some intervals, which contradicts the definition of how the bands are numbered. Thus, it is likely that the interpolation of the bands is done incorrectly and that they should not be smooth at $\mathbf{k}$ =X and $\mathbf{k}$ =M.

All in all, we believe that our code produces the right results, even though the graph of

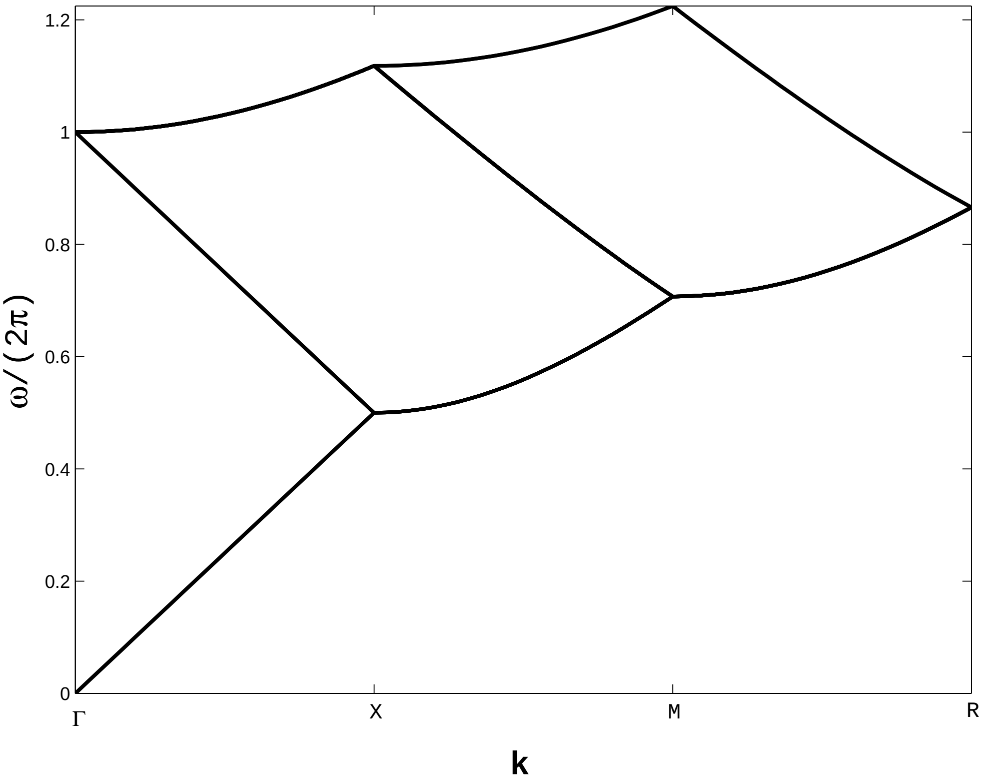
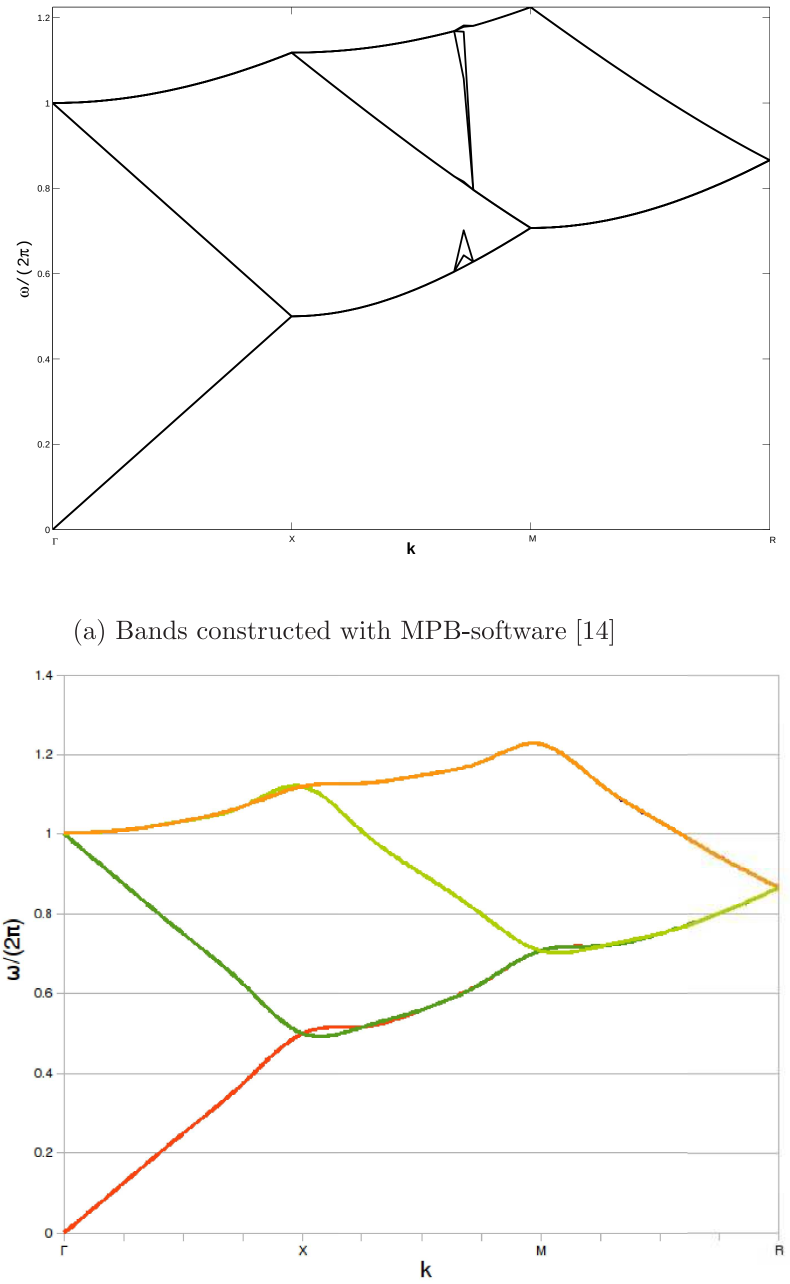


Figure 4.3: Band structure for a homogeneous medium

the bands is different from the graphs produced by other methods.



(a) Bands constructed with MPB-software [14]

(b) Band structure of Bulovyatov [8]

Figure 4.4: Band structures resulting from other methods

Chapter 5

Conclusion and outlook

In this project, the goal was to develop an algorithm to approximate the Maxwell-eigenvalue problem in periodic media using a discontinuous Galerkin method. This algorithm was developed in small steps: first we looked at the time-harmonic Maxwell equations in a bounded domain. For this problem, the derivation of the discontinuous Galerkin method is given in more detail than one normally finds in papers on this subject. It is implemented for a homogeneous medium and tested against known results.

After that, the same procedure was repeated for the time-harmonic Maxwell-equations in periodic media using Bloch mode expansions. The algorithm gives the results with the proper rate of convergence based on analytical results, so it provides a good basis for later work on this subject.

The next step would be to implement the algorithms for more complex media and to find ways to improve the performance of the algorithm. This improvement can be done in several ways. For example, one could discretize the transversility condition $\nabla \cdot \mathbf{H} = 0$ or filter out the null-space of the curl-operator from the eigenspace. Another way is to develop a fast eigensolver and preconditioner for this specific type of problems.

In order to deal with singularities at material interfaces and sharp corners and edges, hp-adaption will be important (h-adaption: local refinement of the mesh, p-adaption: element wise adjustment of the polynomial order), including a theoretical analysis of the convergence of the numerical algorithm and a posteriori error estimation to control the adaptation algorithm.

Also, it would be beneficial to be able to include real-world features such as media in which the unit cell is repeated only finitely many times, including proper inflow and outflow boundary conditions. Moreover, the production process of periodic media can give some perturbations in the media. It would be interesting to investigate the influence of these pertubations.

Appendix A

Transformation of functions and integrals

In this appendix, it is discussed how functions and integrals can be transformed from the reference tetrahedron $\hat{K}$ to any tetrahedral element K in physical space.

A.1 Transformation of functions

We start by defining the transformation that maps the reference tetrahedron onto an arbitrary element. This transformation $F_K : \hat{K} \to K$ is given by

$$F_K(\boldsymbol{\xi}) = \sum_{i=1}^{4} \mathbf{x}_i \lambda_i(\boldsymbol{\xi}),$$

with $\mathbf{x}_i$ being the coordinates of the vertices of K and λ_i the barycentric coordinates of the vertices of the reference tetrahedron. Since this is an affine transformation, it can be written as

$$\mathbf{x} = B_K \boldsymbol{\xi} + \mathbf{b}_K.$$

The Jacobian matrix dF_K of this transformation is defined as

$$(dF_K)_{lm} = \frac{\partial (F_K)_l}{\partial \xi_m},$$

so for this transformation $dF_K = B_K$ and $(dF_K)^{-T} = B_K^{-T}$. Now a scalar function $\hat{p}(\boldsymbol{\xi})$ can be transformed as

$$p(F_K(\boldsymbol{\xi})) = \hat{p}(\boldsymbol{\xi}),$$

or, equivalently,

$$p(\mathbf{x}) = \hat{p}(F_K^{-1}(\mathbf{x})).$$

Using the chain rule for differentiation, the gradient of $p \in H^1(\Omega)$ can now be expressed as

$$(dF_K)^T \nabla p \circ F_K = \hat{\nabla} \hat{p},$$

or equivalently

$$\nabla p \circ F_K = (dF_K)^{-T} \hat{\nabla} \hat{p},$$

in which $\hat{\nabla}$ is the gradient with respect to the coordinates ξ_1, ξ_2 and ξ_3 in the reference space.

Since $\nabla \times \nabla p = 0$ for all $p \in H^1(\Omega)$, it is clear that $\nabla p \in H(\text{curl}, K)$ and $\hat{\nabla}\hat{p} \in H(\text{curl}, \hat{K})$. This suggests that any $\hat{\mathbf{u}} \in H(\text{curl}, \hat{K})$ can be transformed to a $\mathbf{u} \in H(\text{curl}, K)$ via

$$\mathbf{u} \circ F_K = (dF_K)^{-T}\hat{\mathbf{u}}.$$

Substituting the expression $\mathbf{x} = B_K\boldsymbol{\xi} + \mathbf{b}_K$ then gives the following relations between $\mathbf{u}$ and $\hat{\mathbf{u}}$:

$$\begin{aligned}\hat{\mathbf{u}}(\boldsymbol{\xi}) &= B_K^T\mathbf{u}\left(B_K\boldsymbol{\xi} + \mathbf{b}_K\right),\\ \mathbf{u}(\mathbf{x}) &= B_K^{-T}\hat{\mathbf{u}}\left(B_K^{-1}\left(\mathbf{x} - \mathbf{b}_K\right)\right).\end{aligned}$$

Now the expressions for the normal vectors follow immediately from this expression, namely, if $\hat{\mathbf{n}}$ is the outward unit normal vector to $\hat{K}$ at $\boldsymbol{\xi} \in \partial\hat{K}$, then

$$\begin{aligned}\mathbf{n}\left(F_K\left(\boldsymbol{\xi}\right)\right) &= \frac{(dF_K)^{-T}\hat{\mathbf{n}}}{|(dF_K)^{-T}\hat{\mathbf{n}}|}(\boldsymbol{\xi})\\ &= \frac{B_K^{-T}\hat{\mathbf{n}}}{|B_K^{-T}\hat{\mathbf{n}}|}(\boldsymbol{\xi})\end{aligned}$$

is the outward unit normal vector to K at $\mathbf{x} \in \partial K$.
The last expression we need is one that relates $\nabla \times \mathbf{u}$ to $\hat{\nabla} \times \hat{\mathbf{u}}$. This expression is given by

$$(\nabla \times \mathbf{u}) \circ F_K = \frac{1}{\det(dF_K)} dF_K \hat{\nabla} \times \hat{\mathbf{u}}.$$

A complete derivation of this expression is given at p.78 of [18].

A.2 Transformation of integrals

If an integral is defined on (the boundary of) an element K, it can be beneficial to transform the integral to (the boundary of) the reference element $\hat{K}$ and compute the integral on the reference element. Given an affine mapping $F_K : \hat{K} \to K$ and Jacobian matrix dF_K, then the integral of a scalar function over an element is given by [23]

$$\int_K p(\mathbf{x})\ dx = \int_{\hat{K}} \hat{p}\left(\boldsymbol{\xi}\right) |\det(dF_K)|\ d\xi.$$

To integrate over a face, one wants to transform the reference triangle $\hat{T}$ with vertices $\mathbf{v}_{t,1} = (-1,-1), \mathbf{v}_{t,2} = (1,-1), \mathbf{v}_{t,3} = (-1,1)$ and local coordinates η_1, η_2 to the face with vertices $\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3$. This transformation is performed in two steps: first the reference triangle will be transformed to a face of the reference element with the transformation $r_i : \hat{T} \to \hat{f}_i$, then this face will be transformed to the physical face with the transformation $F_K : \hat{K} \to K$ as defined before.
The transformation $\mathbf{r}_i$ is defined as

$$r_1(\boldsymbol{\eta}) = \begin{pmatrix}\eta_1\\-1\\\eta_2\end{pmatrix}, \qquad r_3(\boldsymbol{\eta}) = \begin{pmatrix}-1\\\eta_1\\\eta_2\end{pmatrix},$$

$$r_2(\boldsymbol{\eta}) = \begin{pmatrix}-\eta_1 - \eta_2 - 1\\\eta_1\\\eta_2\end{pmatrix}, \qquad r_4(\boldsymbol{\eta}) = \begin{pmatrix}\eta_1\\\eta_2\\-1\end{pmatrix}.$$

An integral over a face f can be transformed to an integral over face $\hat{f}_i$ of the reference tetrahedron by [18]

$$\int_f p \; dA = \int_{\hat{f}_j} \hat{p} |\det(dF_K)||(dF_K)^{-T}\hat{\mathbf{n}}_j| \; d\hat{A},$$

in which $\hat{p} = p \circ F_K$. To integrate this over a the reference triangle, we have

$$\int_{\hat{f}_j} \hat{p} \; d\hat{A} = \int_{\hat{T}} \tilde{p} |r_{j,\eta_1} \times r_{j,\eta_2}| \; d\eta,$$

in which $\tilde{p} = \hat{p} \circ r_j(\boldsymbol{\eta})$.
So an integral of the form $\int_f \boldsymbol{\psi}_1 \cdot \boldsymbol{\psi}_2 \; dA$ can be written as

$$\begin{aligned}\int_f \boldsymbol{\psi}_1 \cdot \boldsymbol{\psi}_2 \; dA &= \int_{\hat{f}_j} (dF_K)^{-T}\hat{\boldsymbol{\psi}}_1 \cdot (dF_K)^{-T}\hat{\boldsymbol{\psi}}_2 |\det(dF_K)||(dF_K)^{-T}\hat{\mathbf{n}}_j| \; d\hat{A} \\ &= \int_{\hat{T}} \Big((dF_K)^{-T}\hat{\boldsymbol{\psi}}_1 \cdot (dF_K)^{-T}\hat{\boldsymbol{\psi}}_2 \Big) \circ r_j(\boldsymbol{\eta}) |\det(dF_K)||(dF_K)^{-T}\hat{\mathbf{n}}_j||r_{j,\eta_1} \times r_{j,\eta_2}| \; d\eta.\end{aligned}$$

Note that the expression $|\det(dF_K)||(dF_K)^{-T}\hat{\mathbf{n}}_j||r_{j,\eta_1} \times r_{j,\eta_2}|$ gives the proportion between the area of the physical element and the area of the reference triangle. So if we have to compute an integral with terms on both adjacent faces, then it is sufficient to transform the functions to their equivalent reference face, then compute the expression in terms of $\boldsymbol{\xi}$ and finally transforming these functions to the reference triangle with their respective transformations r_L and r_R. Then the integral can be computed by multiplying this last term with the ratio between the reference triangle and the physical phase and integrate over $\boldsymbol{\eta}$.
With the help of these transformations, all the integrals of Sections 3.4.3 and 4.3.1 can be computed.

Appendix B

Nédélec basis functions

The basis functions for the second order Nédélec elements are given by

$$\boldsymbol{\phi}^{e_1,0} = \begin{pmatrix} \frac{3}{4}\xi_2 + \frac{3}{4}\xi_3 + \xi_2\xi_3 + \frac{1}{2}\xi_2^2 + \frac{1}{2}\xi_3^2 \\ -\frac{1}{2}\xi_2 - \frac{1}{2}\xi_3 - \frac{1}{2}\xi_1\xi_2 - \frac{1}{2}\xi_1\xi_3 - \frac{3}{4}\xi_1 - \frac{3}{4} \\ -\frac{1}{2}\xi_2 - \frac{1}{2}\xi_3 - \frac{1}{2}\xi_1\xi_2 - \frac{1}{2}\xi_1\xi_3 - \frac{3}{4}\xi_1 - \frac{3}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_2,0} = \begin{pmatrix} -\frac{1}{2}\xi_1 - \frac{3}{4}\xi_2 - \frac{1}{2}\xi_1\xi_2 - \frac{1}{2}\xi_2^2 - \frac{1}{4} \\ \frac{3}{4}\xi_1 + \frac{1}{2}\xi_2 + \frac{1}{2}\xi_1\xi_2 + \frac{1}{2}\xi_1^2 + \frac{1}{4} \\ 0 \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_3,0} = \begin{pmatrix} \frac{1}{2}\xi_1 + \frac{3}{4}\xi_2 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_1\xi_2 + \frac{1}{2}\xi_2\xi_3 + \frac{3}{4} \\ -\frac{3}{4}\xi_1 - \frac{3}{4}\xi_3 - \xi_1\xi_3 - \frac{1}{2}\xi_1^2 - \frac{1}{2}\xi_3^2 \\ \frac{1}{2}\xi_1 + \frac{3}{4}\xi_2 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_1\xi_2 + \frac{1}{2}\xi_2\xi_3 + \frac{3}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_4,0} = \begin{pmatrix} -\frac{1}{2}\xi_2 - \frac{3}{4}\xi_3 - \frac{1}{2}\xi_1\xi_3 - \frac{1}{2}\xi_2\xi_3 - \frac{1}{2}\xi_1 - \frac{3}{4} \\ -\frac{1}{2}\xi_2 - \frac{3}{4}\xi_3 - \frac{1}{2}\xi_1\xi_3 - \frac{1}{2}\xi_2\xi_3 - \frac{1}{2}\xi_1 - \frac{3}{4} \\ \frac{3}{4}\xi_1 + \frac{3}{4}\xi_2 + \xi_1\xi_2 + \frac{1}{2}\xi_1^2 + \frac{1}{2}\xi_2^2 \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_5,0} = \begin{pmatrix} -\frac{3}{4}\xi_3 - \frac{1}{2}\xi_1\xi_3 - \frac{1}{2}\xi_3^2 - \frac{1}{4} \\ 0 \\ \frac{3}{4}\xi_1 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_1\xi_3 + \frac{1}{2}\xi_1^2 + \frac{1}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_6,0} = \begin{pmatrix} 0 \\ -\frac{1}{2}\xi_2 - \frac{3}{4}\xi_3 - \frac{1}{2}\xi_2\xi_3 - \frac{1}{2}\xi_3^2 - \frac{1}{4} \\ \frac{3}{4}\xi_2 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_2\xi_3 + \frac{1}{2}\xi_2^2 + \frac{1}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_1,1} = \begin{pmatrix} -\frac{1}{2}\xi_1 - \frac{5}{4}\xi_2 - \frac{5}{4}\xi_3 - \xi_1\xi_2 - \xi_1\xi_3 - \xi_2\xi_3 - \frac{1}{2}\xi_2^2 - \frac{1}{2}\xi_3^2 - \frac{1}{2} \\ \frac{7}{4}\xi_1 + \frac{1}{2}\xi_2 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_1\xi_2 + \frac{1}{2}\xi_1\xi_3 + \xi_1^2 + \frac{3}{4} \\ \frac{7}{4}\xi_1 + \frac{1}{2}\xi_2 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_1\xi_2 + \frac{1}{2}\xi_1\xi_3 + \xi_1^2 + \frac{3}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_2,1} = \begin{pmatrix} \frac{1}{2}\xi_1 - \frac{1}{4}\xi_2 + \frac{1}{2}\xi_1\xi_2 - \frac{1}{2}\xi_2^2 + \frac{1}{4} \\ \frac{1}{2}\xi_2 - \frac{1}{4}\xi_1 + \frac{1}{2}\xi_1\xi_2 - \frac{1}{2}\xi_1^2 + \frac{1}{4} \\ 0 \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_3,1} = \begin{pmatrix} -\frac{7}{4}\xi_2 - \frac{1}{2}\xi_3 - \frac{1}{2}\xi_1\xi_2 - \frac{1}{2}\xi_1 - \frac{1}{2}\xi_2\xi_3 - \xi_2^2 - \frac{3}{4} \\ \frac{5}{4}\xi_1 + \frac{1}{2}\xi_2 + \frac{5}{4}\xi_3 + \xi_1\xi_2 + \xi_1\xi_3 + \xi_2\xi_3 + \frac{1}{2}\xi_1^2 + \frac{1}{2}\xi_3^2 + \frac{1}{2} \\ -\frac{7}{4}\xi_2 - \frac{1}{2}\xi_3 - \frac{1}{2}\xi_1\xi_2 - \frac{1}{2}\xi_1 - \frac{1}{2}\xi_2\xi_3 - \xi_2^2 - \frac{3}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_4,1} = \begin{pmatrix} \frac{1}{2}\xi_1 + \frac{1}{2}\xi_2 + \frac{1}{2}\xi_1\xi_3 + \frac{1}{2}\xi_2\xi_3 + \xi_3^2 + \frac{3}{4} \\ \frac{1}{2}\xi_1 + \frac{1}{2}\xi_2 + \frac{7}{4}\xi_3 + \frac{1}{2}\xi_1\xi_3 + \frac{1}{2}\xi_2\xi_3 + \xi_3^2 + \frac{3}{4} \\ -\frac{5}{4}\xi_1 - \frac{5}{4}\xi_2 + \frac{7}{4}\xi_3 - \frac{1}{2}\xi_3 - \xi_1\xi_2 - \xi_1\xi_3 - \xi_2\xi_3 - \frac{1}{2}\xi_1^2 - \frac{1}{2}\xi_2^2 - \frac{1}{2} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_5,1} = \begin{pmatrix} \frac{1}{2}\xi_1 - \frac{1}{4}\xi_3 + \frac{1}{2}\xi_1\xi_3 - \frac{1}{2}\xi_3^2 + \frac{1}{4} \\ 0 \\ \frac{1}{2}\xi_3 - \frac{1}{4}\xi_1 + \frac{1}{2}\xi_1\xi_3 - \frac{1}{2}\xi_1^2 + \frac{1}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{e_6,1} = \begin{pmatrix} 0 \\ \frac{1}{2}\xi_2 - \frac{1}{4}\xi_3 + \frac{1}{2}\xi_2\xi_3 - \frac{1}{2}\xi_3^2 + \frac{1}{4} \\ \frac{1}{2}\xi_3 - \frac{1}{4}\xi_2 + \frac{1}{2}\xi_2\xi_3 - \frac{1}{2}\xi_2^2 + \frac{1}{4} \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_1,0} = \begin{pmatrix} -\frac{1}{2}\xi_2 - \frac{3}{2}\xi_3 - \xi_1\xi_3 - \frac{1}{2}\xi_2\xi_3 - \xi_1 - \frac{1}{2}\xi_3^2 - \\ -\frac{1}{2}\xi_1 - \frac{1}{2}\xi_3 - \frac{1}{2}\xi_1\xi_3 - \frac{1}{2} \\ \frac{3}{2}\xi_1 + \xi_2 + \frac{1}{2}\xi_3 + \xi_1\xi_2 + \frac{1}{2}\xi_1\xi_3 + \xi_1^2 + \frac{1}{2} \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_1,1} = \begin{pmatrix} -\xi_2 - \frac{3}{2}\xi_3 - \frac{1}{2}\xi_1 - \frac{1}{2}\xi_1\xi_3 - \xi_2\xi_3 - \xi_3^2 - \frac{1}{2} \\ \frac{1}{2}\xi_1 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_1\xi_3 + \frac{1}{2} \\ \frac{3}{2}\xi_1 + \frac{1}{2}\xi_2 + \xi_3 + \frac{1}{2}\xi_1\xi_2 + \xi_1\xi_3 + \frac{1}{2}\xi_1^2 + 1 \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_2,0} = \begin{pmatrix} \frac{\sqrt{3}}{2}\xi_2 + \frac{\sqrt{3}}{2}\xi_3 + \frac{\sqrt{3}}{2}\xi_2\xi_3 + \frac{\sqrt{3}}{2} \\ -\frac{\sqrt{3}}{10}\xi_1 - \frac{\sqrt{3}}{10}\xi_3 - \frac{\sqrt{3}}{10}\xi_1\xi_3 - \frac{\sqrt{3}}{10} \\ -\frac{2\sqrt{3}}{5}\xi_1 - \frac{2\sqrt{3}}{5}\xi_2 - \frac{2\sqrt{3}}{5}\xi_1\xi_2 - \frac{2\sqrt{3}}{5} \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_2,1} = \begin{pmatrix} -\frac{\sqrt{3}}{2}\xi_3 - \frac{\sqrt{3}}{2}\xi_2\xi_3 - \frac{\sqrt{3}}{2}\xi_2 - \frac{\sqrt{3}}{2} \\ \frac{2\sqrt{3}}{5}\xi_1 + \frac{2\sqrt{3}}{5}\xi_3 + \frac{2\sqrt{3}}{5}\xi_1\xi_3 + \frac{2\sqrt{3}}{5} \\ \frac{\sqrt{3}}{10}\xi_1 + \frac{\sqrt{3}}{10}\xi_2 + \frac{\sqrt{3}}{10}\xi_1\xi_2 + \frac{\sqrt{3}}{10} \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_3,0} = \begin{pmatrix} \frac{1}{2}\xi_2 + \frac{1}{2}\xi_3 + \frac{1}{2}\xi_2\xi_3 + \frac{1}{2} \\ \frac{1}{2}\xi_1 + \xi_2 + \frac{3}{2}\xi_3 + \frac{1}{2}\xi_1\xi_3 + \xi_2\xi_3 + \frac{1}{2}\xi_3^2 + \\ -\xi_1 - \frac{3}{2}\xi_2 - \frac{1}{2}\xi_3 - \xi_1\xi_2 - \frac{1}{2}\xi_2\xi_3 - \xi_2^2 - \frac{1}{2} \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_3,1} = \begin{pmatrix} -\frac{1}{2}\xi_2 - \frac{1}{2}\xi_3 - \frac{1}{2}\xi_2\xi_3 - \frac{1}{2} \\ \xi_1 + \frac{1}{2}\xi_2 + \frac{3}{2}\xi_3 + \xi_1\xi_3 + \frac{1}{2}\xi_2\xi_3 + \xi_3^2 + \frac{1}{2} \\ -\frac{1}{2}\xi_1 - \frac{3}{2}\xi_2 - \xi_3 - \frac{1}{2}\xi_1\xi_2 - \xi_2\xi_3 - \frac{1}{2}\xi_2^2 - 1 \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_4,0} = \begin{pmatrix} \xi_1 + \frac{3}{2}\xi_2 + \frac{1}{2}\xi_3 + \xi_1\xi_2 + \frac{1}{2}\xi_2\xi_3 + \frac{1}{2}\xi_2^2 + \\ -\frac{3}{2}\xi_1 - \frac{1}{2}\xi_2 - \xi_3 - \frac{1}{2}\xi_1\xi_2 - \xi_1\xi_3 - \xi_1^2 - \frac{1}{2} \\ \frac{1}{2}\xi_1 + \frac{1}{2}\xi_2 + \frac{1}{2}\xi_1\xi_2 + \frac{1}{2} \end{pmatrix},$$

$$\boldsymbol{\phi}^{f_4,1} = \begin{pmatrix} \frac{1}{2}\xi_1 + \frac{3}{2}\xi_2 + \xi_3 + \frac{1}{2}\xi_1\xi_2 + \xi_2\xi_3 + \xi_2^2 + \frac{1}{2} \\ -\xi_2 - \frac{1}{2}\xi_3 - \xi_1\xi_2 - \frac{1}{2}\xi_1\xi_3 - \frac{3}{2}\xi_1 - \frac{1}{2}\xi_1^2 - 1 \\ -\frac{1}{2}\xi_2 - \frac{1}{2}\xi_1 - \frac{1}{2}\xi_1\xi_2 - \frac{1}{2} \end{pmatrix}.$$

The edges and faces are given in Figure B.1 and Table B.1.

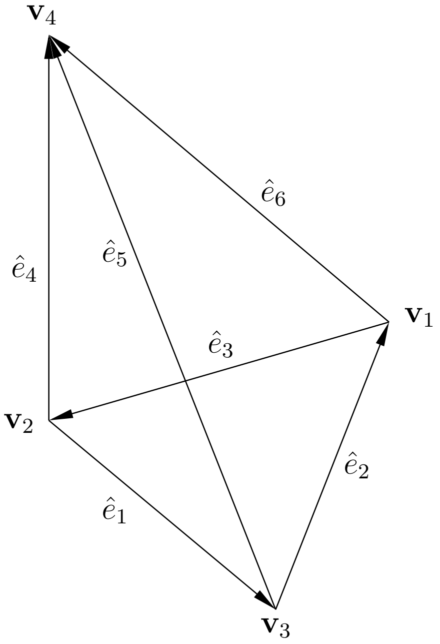


Figure B.1: The reference tetrahedron

Edge/face	vertices
$\hat{e}_1$	$\mathbf{v}_2, \mathbf{v}_3$
$\hat{e}_2$	$\mathbf{v}_3, \mathbf{v}_1$
$\hat{e}_3$	$\mathbf{v}_1, \mathbf{v}_2$
$\hat{e}_4$	$\mathbf{v}_2, \mathbf{v}_4$
$\hat{e}_5$	$\mathbf{v}_3, \mathbf{v}_4$
$\hat{e}_6$	$\mathbf{v}_1, \mathbf{v}_4$
$\hat{f}_1$	$\mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4$
$\hat{f}_2$	$\mathbf{v}_1, \mathbf{v}_3, \mathbf{v}_4$
$\hat{f}_3$	$\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_4$
$\hat{f}_4$	$\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$

Table B.1: Numbering of edges and faces of a tetrahedron, numbering indicates direction along edges

Bibliography

[1] Milton Abramowitz and Irene A Stegun, *Handbook of mathematical functions: With formulars, graphs, and mathematical tables*, vol. 55, Dover Publications, 1964.

[2] M. Ainsworth and J. Coyle, *Hierarchic finite element bases on unstructured tetrahedral meshes*, International journal for numerical methods in engineering **58** (2003), no. 14, 2103–2130.

[3] Douglas N Arnold, Franco Brezzi, Bernardo Cockburn, and L Donatella Marini, *Unified analysis of discontinuous galerkin methods for elliptic problems*, SIAM journal on numerical analysis **39** (2002), no. 5, 1749–1779.

[4] Peter Bermel, Chiyan Luo, Lirong Zeng, Lionel C Kimerling, and John D Joannopoulos, *Improving thin-film crystalline silicon solar cell efficiencies with photonic crystals*, Optics Express **15** (2007), no. 25, 16986–17000.

[5] Daniele Boffi, *Finite element approximation of eigenvalue problems*, Acta Numerica **19** (2010), no. 1, 1–120.

[6] Daniele Boffi, Matteo Conforti, and Lucia Gastaldi, *Modified edge finite elements for photonic crystals*, Numerische Mathematik **105** (2006), no. 2, 249–266.

[7] A. Buffa and I. Perugia, *Discontinuous Galerkin approximation of the maxwell eigenproblem*, SIAM Journal on Numerical Analysis **44** (2006), no. 5, 2198–2226.

[8] A. Bulovyatov, *A parallel multigrid method for band stucture computation of 3D photonic crystals with higher order finite elements*, Ph.D. thesis, Karlsruhe Institute of Technology, 2010.

[9] JAYADEEP Gopalakrishnan, LUIS E García-Castillo, and LESZEK F Demkowicz, *Nédélec spaces in affine coordinates*, Computers & Mathematics with Applications **49** (2005), no. 7, 1285–1294.

[10] David J. Griffiths, *Introduction to electrodynamics*, Pearson Education, Inc., Third edition, 2008.

[11] JS Hesthaven and T Warburton, *High–order nodal discontinuous galerkin methods for the maxwell eigenvalue problem*, Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences **362** (2004), no. 1816, 493–524.

[12] Paul Houston, Ilaria Perugia, Anna Schneebeli, and Dominik Schötzau, *Interior penalty method for the indefinite time-harmonic maxwell equations*, Numerische Mathematik **100** (2005), no. 3, 485–518.

[13] J.D. Joannopoulos et. al., *Photonic crystals: Molding the flow of light*, Princeton University Press, 2011.

[14] S.G. Johnson and J.D. Joannopoulos, *The MIT Photonic-Bands package, version 1.4.2.*

[15] ———, *Block-iterative frequency-domain methods for Maxwell's equations in a planewave basis*, Opt. Express **8** (2001), no. 3, 173–190.

[16] Armin Lechleiter, Michael Plum, et al., *Photonic crystals: Mathematical analysis and numerical approximation*, vol. 42, Springer, 2011.

[17] Marko Loncar, Axel Scherer, and Yueming Qiu, *Photonic crystal laser sources for chemical detection*, Applied Physics Letters **82** (2003), no. 26, 4648–4650.

[18] P. Monk, *Finite element methods for Maxwell's equations*, Numerical Analysis and Scientific Computation Series, Clarendon Press, 2003.

[19] J.C. Nédélec, *Mixed finite elements in* $\mathbb{R}^3$, Numerische Mathematik **35** (1980), no. 3, 315–341.

[20] D. Sármány, *High-order finite element approximations of the maxwell equations*, Ph.D. thesis, University of Twente, Enschede, February 2010.

[21] D. Sarmany, F. Izsák, and J.J.W. Vegt van der, *Optimal penalty parameters for symmetric discontinuous Galerkin discretisations of the time-harmonic Maxwell quations*, Journal of Scientific Computing **44** (2010), no. 3, 219–254.

[22] P. Šolín, K. Segeth, and I. Dolezel, *Higher-order finite element methods*, Studies in Advanced Mathematics, Chapman & Hall/CRC, 2004.

[23] J. Stewart, *Calculus: early transcendentals*, Brooks/Cole Publishing Company, 2003.

[24] Kane Yee, *Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media*, Antennas and Propagation, IEEE Transactions on **14** (1966), no. 3, 302–307.

[25] Chin-ping Yu and Hung-chun Chang, *Compact finite-difference frequency-domain method for the analysis of two-dimensional photonic crystals*, Optics Express **12** (2004), no. 7, 1397–1408.