**Abstract**

Two unstructured finite volume solvers for the Maxwell equations in 2D and 3D are introduced. The solvers are a generalization of FD-TD to unstructured grids and they use a third-order staggered Adams–Bashforth scheme for time discretization. Analysis and experiments of this time integrator reveal that we achieve a long term stable solution on general triangular grids. A Fourier analysis shows that the 2D solver has excellent dispersion characteristics on uniform triangular grids. In 3D a spatial filter of Laplace type is introduced to enable long simulations without suffering from late time instability.

The recursive convolution method proposed by Luebbers et al. to extend FD-TD to permit frequency dispersive materials is here generalized to the 3D solver. A better modelling of materials which have a strong frequency dependence in their constitutive parameters is obtained through the use of a general material model.

The finite volume solvers are not intended to be stand-alone solvers but one part in two hybrid solvers with FD-TD. The numerical examples in 2D and 3D demonstrate that the hybrid solvers are superior to stand-alone FD-TD in terms of accuracy and efficiency.

*Keywords*: Maxwell's equations, finite volumes, FD-TD, hybrid grids.

# Contents

# Chapter 1

# Introduction

The rapid increases in computing capabilities have facilitated the solutions of larger, more complex electromagnetic scattering and radiation problems. Numerical methods are today at a level where they can be used to accurately simulate the electromagnetic response from very complicated objects such as aircraft, satellites, cars and various electronic devices. A numerical simulation can often replace expensive and very time consuming measurements especially in the design and construction phase.

Computational Electromagnetics (CEM) can for instance be used to determine the radar cross section (RCS) for objects. The RCS is a far-field parameter, which characterizes the scattering properties of radar targets. For a target, there is monostatic and bistatic RCS. For monostatic RCS the transmitter and the receiver are at the same location, which is not the case for bistatic RCS. When designing a low-observable target, the RCS is the parameter you attempt to minimize.

Electronic systems are increasing rapidly in numbers and complexity as well as their sensitivity to electromagnetic disturbances. Such a disturbance could for example be direct or indirect effects of lightning. The high surface currents that are induced can damage safety critical electronic systems in the aircraft. Testing of Electromagnetic compatibility (EMC) properties for large and complex objects like aircraft is an area where CEM is used successfully.

Antenna design is another important field of application for CEM. For wireless communication systems the antenna is one of the most critical components and a good design of the antenna could improve the overall system performance. Software that can be used to simulate an entire antenna accurately is an important tool for engineers.

When solving the Maxwell equations the first choice is whether a time domain or a frequency domain method should be used. Since we are interested in applications where a broad range of frequencies are of interest we have chosen a time domain method. The most commonly used method for the time domain

Maxwell equations is the Finite Difference Time Domain method (FD-TD) [26]. It has several advantages compared to other time domain methods. The simple data structures make it straightforward to implement and facilitate vectorization and parallelization. With respect to time and memory consumption, the FD-TD method is an almost optimal second-order method for homogeneous materials.

The main drawback with the FD-TD method is its inability to accurately model curved objects and small geometrical features. The Cartesian FD-TD grid leads to a staircase approximation of the geometry and small details will not be resolved at all.

There are several suggested remedies to circumvent the effects of the errors introduced by the staircase approximations. These include Cartesian subgridding [19], conformal modelling [4], overlapping grids [27, 28, 29, 30], multiblock body fitted structured grids [2, 22] and unstructured grids [14]. Cartesian subgridding suffers from stability problems and the interpolations required at grid interfaces violate the divergence relations. The idea of the conformal modelling approach is to change the updating stencils in all cells that are intersected by the boundary. The main drawback with this method is its extreme complexity in 3D. Furthermore, the question of stability is not sufficiently investigated. The overlapping grids approach has serious stability problems due to the interpolations between the grids [30, 28]. Another disadvantage is the grid generation which often requires a high user intervention. The methods based on multiblock body fitted structured originate from CFD. They are designed to be able to capture shocks and are in general very dissipative, which is not a good property for a solver of Maxwell's equations. Moreover, the generation of the grid is a major hassle for complex objects and could take several man months even for a skilled engineer. Using an unstructured grid for the whole domain makes the grid generation considerably easier for complex objects. However, an unstructured solver generally implies a more complex data structure and more arithmetic operations per cell compared to a structured solver. Hence, none of the methods listed above combines stability, second-order accuracy and efficiency. By efficiency we refer to the time spent on implementation, grid generation and computation.

Another approach to avoid staircasing is to use unstructured grids near curved objects and around small geometrical details but revert to structured grids as quickly as possible for the rest of the computational domain. This combines the efficiency of structured grids with the flexibility of unstructured grids. Wu and Itoh [25] were first to present a combination of the FD-TD method and an implicit finite element (FE) method. A combination of explicit finite volumes (FV) and FD-TD showing impressive results has been proposed by Riley and Turner [20]. Madsen and Ziolkowski [17], and Madsen [16] have published work on two different methods allowing mixed polyhedral elements. None of the methods proposed by Madsen et al. is really a hybrid in the sense that the same algorithm is used throughout the computational domain, but they have the nice property of reducing to FD-TD on Cartesian grids. However, both methods have severe problems with late time instability [30]. Thus, based on the literature survey [5] we have chosen to follow a similar approach to Riley and Turner and implement an FV solver based on staggered unstructured grids and couple this solver to an FD-TD solver.

The main part of this thesis is devoted to the FV solvers. The next chapter focuses on the Maxwell equations. The solvers are thoroughly described in Chapters 3 and 4. The 2D solver is in space very similar to the solver used by Riley and Turner [20], whereas the solver in 3D differs in many ways. The main improvements of the 3D solver in space include the generation of the dual grid, the update of the node values and the treatment of material interfaces. The time integration method we use is a third-order staggered Adams–Bashforth scheme (ABS3) proposed by Ghrist et al. [9] for scalar wave equations. Stability analysis of this scheme shows that it works very well in our case. Furthermore, using ABS3 makes the time-coupling between FD-TD and the FV solver straightforward. A spatial filter of Laplace type is introduced in 3D to be able to run long simulations without suffering from late time instability. In Chapter 5 the coupling between the FV solver and FD-TD is described. Chapter 6 presents results on scattering from dielectric and diamagnetic cylinders in 2D that show that with the correct averaging of the permeability and permittivity second-order accuracy is obtained. The next chapter illustrates some of the 3D results, including a convergence study, scattering from a PEC sphere and scattering from the generic aircraft *RUND*. More 2D and 3D results are found in the appendices, which contain the papers *Hybrid Time Domain Solvers for the Maxwell Equations in 2D*, submitted for publication and *Explicit Hybrid Time Domain Solver for the Maxwell Equations in 3D*, accepted for publication in *Journal of Scientific Computing*.

The ability to treat frequency dispersive materials is an important feature for a general purpose electromagnetic code. They are for instance used as radar absorbing materials to minimize the RCS for an aircraft. Other examples include waveguides filled with dispersive materials, antenna structures and microstrip systems based on dispersive substrates. One of the most common approaches to extend FD-TD to permit these materials is to use the recursive convolution (RC) method [15]. In Chapter 8 the piecewise constant centered RC method presented in [21] is generalized to our FV solver. To enable better modelling of materials, which have a strong frequency dependence in their constitutive parameters a general material model is introduced. The last chapter summarizes the thesis and conclusions are drawn for future work. Part of the material in this thesis has been published in conference proceedings [6, 7, 13, 23] and most of the 2D work is included in the technical report [1].

# Chapter 2

# Governing equations

Electromagnetic fields are governed by the Maxwell equations. In general form they are given by

$$\nabla \cdot \boldsymbol{D} = \rho_f \,, \tag{2.1}$$

$$\frac{\partial \boldsymbol{B}}{\partial t} + \nabla \times \boldsymbol{E} = \boldsymbol{0} \,, \tag{2.2}$$

$$\nabla \cdot \boldsymbol{B} = 0 \,, \tag{2.3}$$

$$\frac{\partial \boldsymbol{D}}{\partial t} - \nabla \times \boldsymbol{H} + \boldsymbol{J}_f = \boldsymbol{0} \,, \tag{2.4}$$

where $\boldsymbol{D}$ is the electric flux density vector, $\boldsymbol{E}$ is the electric field, $\boldsymbol{B}$ is the magnetic flux density vector, $\boldsymbol{H}$ is the magnetic field, $\rho_f$ is the free charge, and $\boldsymbol{J}_f$ is the free current density. Equation (2.1) is Gauss' law for the electric field and it summarizes Coulomb's law of force between point charges plus the electrical effects of matter, while (2.2) represents Faraday's law of induction. The third equation reflects that free magnetic charges have not been observed. Finally, (2.4) includes Ampère's law of force between currents plus the magnetic effects of matter and conservation of free charge.

For linear, isotropic and non-dispersive media we have $\boldsymbol{D} = \epsilon \boldsymbol{E}$, $\boldsymbol{B} = \mu \boldsymbol{H}$ and $\boldsymbol{J}_f = \sigma \boldsymbol{E}$, where $\epsilon$ is the electric permittivity, $\mu$ is the magnetic permeability and $\sigma$ is the electric conductivity. Substituting these relations into (2.1)–(2.4) and assuming a source free medium implies

$$\nabla \cdot \epsilon \, \boldsymbol{E} = 0 \,, \tag{2.5}$$

$$\mu \, \frac{\partial \boldsymbol{H}}{\partial t} + \nabla \times \boldsymbol{E} = \boldsymbol{0} \,, \tag{2.6}$$

$$\nabla \cdot \mu \, \boldsymbol{H} = 0 \,, \tag{2.7}$$

$$\epsilon \, \frac{\partial \boldsymbol{E}}{\partial t} - \nabla \times \boldsymbol{H} + \sigma \, \boldsymbol{E} = \boldsymbol{0} \,. \tag{2.8}$$

Hence, the Maxwell equations in $\boldsymbol{E}$ and $\boldsymbol{H}$ is a system of eight scalar equations in six variables. At a first glance it may seem that the system is overdetermined, but as is readily shown if the two divergence relations are satisfied initially they are always satisfied. Together with boundary conditions and initial values the Maxwell equations constitute a well posed system of hyperbolic partial differential equations. Taking the curl of Faraday's and Ampère's laws and using the two Gauss' laws it is easily shown that both fields satisfy a homogeneous wave equation. Hence, the solutions to Maxwell's equations are given by electromagnetic waves.

Using Gauss' divergence theorem and Stokes law the integral formulation of Maxwell's equations for a source free medium is obtained

$$\oint_A \epsilon\, \boldsymbol{E} \cdot \boldsymbol{n}\, dA = 0\,, \tag{2.9}$$

$$\frac{\partial}{\partial t} \int_A \mu\, \boldsymbol{H} \cdot \boldsymbol{n}\, dA + \oint_\Gamma \boldsymbol{E} \cdot \boldsymbol{dl} = \boldsymbol{0}\,, \tag{2.10}$$

$$\oint_A \mu\, \boldsymbol{H} \cdot \boldsymbol{n}\, dA = 0\,, \tag{2.11}$$

$$\frac{\partial}{\partial t} \int_A \epsilon\, \boldsymbol{E} \cdot \boldsymbol{n}\, dA - \oint_\Gamma \boldsymbol{H} \cdot \boldsymbol{dl} + \int_A \sigma\, \boldsymbol{E} \cdot \boldsymbol{n}\, dA = \boldsymbol{0}\,, \tag{2.12}$$

where $A$ is an arbitrary area with unit normal $\boldsymbol{n}$ and $\Gamma$ is the contour that encloses $A$.

If we have no variation in the $z$-direction all derivatives with respect to $z$ equal zero. In that case the Maxwell curl equations in Cartesian coordinates reduce to

$$\mu\, \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y}\,, \tag{2.13}$$

$$\mu\, \frac{\partial H_y}{\partial t} = \frac{\partial E_z}{\partial x}\,, \tag{2.14}$$

$$\epsilon\, \frac{\partial E_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - \sigma\, E_z\,, \tag{2.15}$$

$$\epsilon\, \frac{\partial E_x}{\partial t} = \frac{\partial H_z}{\partial y} - \sigma E_x\,, \tag{2.16}$$

$$\epsilon\, \frac{\partial E_y}{\partial t} = -\frac{\partial H_z}{\partial x} - \sigma E_y\,, \tag{2.17}$$

$$\mu\, \frac{\partial H_z}{\partial t} = \frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x}\,, \tag{2.18}$$

where the first three equations constitute the 2D transverse magnetic (TM) Maxwell's equations and the last three constitute the 2D transverse electric (TE)

Maxwell's equations. The TM and TE modes are decoupled, i.e. they contain no common field component. The two modes are completely independent for isotropic materials and they can exist simultaneously with no mutual interaction.

# Chapter 3

# Finite Volume method in 2D

The Finite Volume (FV) solver is based on the following integral formulation of Ampère's and Faraday's laws:

$$\frac{\partial}{\partial t}\int_A \mu\, \boldsymbol{H}\, dA = -\oint_\Gamma \boldsymbol{n} \times \boldsymbol{E}\, dl\,, \tag{3.1}$$

$$\frac{\partial}{\partial t}\int_A \epsilon\, \boldsymbol{E}\, dA = \oint_\Gamma \boldsymbol{n} \times \boldsymbol{H}\, dl - \int_A \sigma \boldsymbol{E}\, dA\,, \tag{3.2}$$

where $A$ is an arbitrary area, $\Gamma$ is the path that encloses $A$ and $\boldsymbol{n}$ is a unit normal. We have chosen to focus on the TM Maxwell's equations (2.13)–(2.15). Due to the duality of the Maxwell equations a solver for the TE equations (2.16)–(2.18) could be obtained through a change of variables.

## 3.1 Spatial discretization

The integral formulations in (3.1) and (3.2) are discretized on a staggered grid by introducing a dual grid to the primary triangular grid. The magnetic components are situated at the nodes of the primary grid and the electric components, in the 2D TM case only $E_z$, are situated at the nodes of the dual grid. The dual grid is created at the preprocessing stage by defining dual nodes at the barycenters of the primary cells; see Section 3.3 for a detailed description.

In the 2D TM case integrating the magnetic and electric fields over each dual and primary cell gives the following integral form:

$$\frac{\partial}{\partial t}\int_{A_j^d} \tilde{\mu}_j^d\, \boldsymbol{H}\, dA = -\sum_k \int_{\Gamma_{j,k}^d} \boldsymbol{n}_{j,k}^d \times E_z\, dl\,, \tag{3.3}$$

$$\frac{\partial}{\partial t}\int_{A_i^p} \epsilon_i^p\, E_z\, dA = \sum_m \int_{\Gamma_{i,m}^p} \boldsymbol{n}_{i,m}^p \times \boldsymbol{H}\, dl - \int_{A_i^p} \sigma_i^p\, E_z\, dA\,, \tag{3.4}$$

where $A_j^d$ is the area of dual cell $j$, $\Gamma_j^d$ is the path that encloses $A_j^d$ and $\boldsymbol{n}_{j,k}^d$ are the unit edge normals for the dual edges $k$ in dual cell $j$. The variables belonging to the primary cell $i$ are defined similarly.

All materials are defined relative to primary grid cells. For dielectric materials that is a natural definition. However, the magnetic permeability, $\mu$, is associated with the dual grid cells. Therefore, $\mu$ requires averaging. The average permeability of dual cell $j$ is computed as

$$\tilde{\mu}_j^d = \sum_q \mu_q^p A_{j,q}^d \,, \tag{3.5}$$

where $A_{j,q}^d$ is the part of the area $A_j^d$ that is inside primary cell $q$. Performing the material averaging in this manner preserves the second-order accuracy of the solver for non-homogeneous materials [3].

The area integrals in (3.3) and (3.4) are evaluated by taking the average values of the fields multiplied by the areas of the respective cells. Simplifying the two integrands in the line integrals implies

$$\tilde{\mu}_j^d \frac{\partial}{\partial t} \boldsymbol{H}_j = \frac{1}{A_j^d} \sum_k \int_{\Gamma_{j,k}^d} \tilde{E}_z|_{j,k}\, \boldsymbol{t}_{j,k}^d \, dl \,, \tag{3.6}$$

$$\epsilon_i^p \frac{\partial}{\partial t} E_z|_i + \sigma_i^p E_z|_i = \frac{1}{A_i^p} \sum_m \int_{\Gamma_{i,m}^p} \tilde{\boldsymbol{H}} \cdot \boldsymbol{t}_{i,m}^p \, dl \,, \tag{3.7}$$

where $\boldsymbol{t}_{j,k}^d$ are unit vectors for the dual edges $k$ in dual cell $j$ and $\boldsymbol{t}_{i,m}^p$ are unit vectors for the edges $m$ in primary cell $i$. The line integral in (3.6) is evaluated by assuming that the electric field is piecewise linear along the dual edges. Hence, $\tilde{E}_z|_{j,k}$ is computed by taking the arithmetic mean value of the electric field at the two nodes defining the dual edge, $\boldsymbol{t}_{j,k}^d$. However, we cannot use the same approach when calculating the integral in (3.7) since that does not guarantee that the divergence is preserved on a local cell level. This has been found to be very critical if spurious modes in the numerical solution are to be suppressed. The divergence is preserved if we incorporate an "FD-TD"-correction along the edges in the primary grid (see Section 3.4 for a proof). Following Riley et al. [20], the magnetic field projected along the primary edge $\boldsymbol{t}_{i,m}^p$ is evaluated as

$$\begin{aligned} \tilde{\boldsymbol{H}} \cdot \boldsymbol{t}_{i,m}^p = {} & \boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d \left( \boldsymbol{n}_{j,k}^d \cdot \boldsymbol{t}_{i,m}^p \right) \\ & + \frac{1}{2} \left[ (\boldsymbol{H}_j + \boldsymbol{H}_r) - \left( (\boldsymbol{H}_j + \boldsymbol{H}_r) \cdot \boldsymbol{n}_{j,k}^d \right) \boldsymbol{n}_{j,k}^d \right] \cdot \boldsymbol{t}_{i,m}^p \,, \end{aligned} \tag{3.8}$$

where $\boldsymbol{H}_j$ and $\boldsymbol{H}_r$ are the magnetic field at the two nodes defining the primary edge and $\boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d$ is the FD-TD component in the direction $\boldsymbol{n}_{j,k}^d$ orthogonal to the dual edge $\boldsymbol{t}_{j,k}^d$, which crosses the primary edge $\boldsymbol{t}_{i,m}^p$, see Figure 3.1. The

FD-TD component is updated according to

$$\frac{\partial \boldsymbol{H} \cdot \boldsymbol{n}^d_{j,k}}{\partial t} = \frac{E_z|_q - E_z|_i}{\Delta^d_{j,k}\, \tilde{\mu}^d_{j,k}}\,, \tag{3.9}$$

where $\Delta^d_{j,k}$ is the length and $\tilde{\mu}^d_{j,k}$ is the average permeability of the dual edge $\boldsymbol{t}^d_{j,k}$. The average permeability, $\tilde{\mu}^d_{j,k}$, is approximated as

$$\tilde{\mu}^d_{j,k} = \mu^p_q\, \Delta_q + \mu^p_i \left(\Delta^d_{j,k} - \Delta_q\right)\,, \tag{3.10}$$

where $i$ and $q$ are the two primary cells sharing dual edge $\boldsymbol{t}^d_{j,k}$, $\Delta_q$ is the part of the length of the dual edge that is inside primary cell $q$, see Figure 3.1.

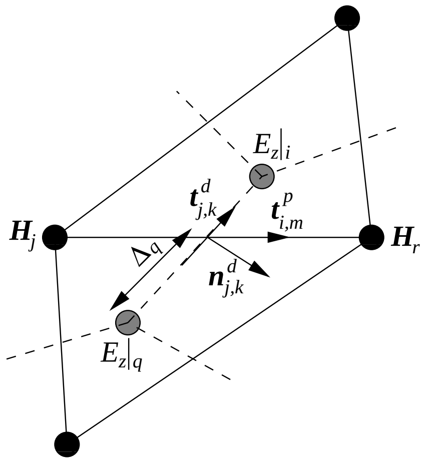


Figure 3.1: The magnetic field along $\boldsymbol{t}^p_{i,m}$ is corrected using the FD-TD value in direction $\boldsymbol{n}^d_{j,k}$.

Taking a closer look at (3.8) we see that if the primary and dual edges are orthogonal, the vectors $\boldsymbol{n}^d_{j,k}$ and $\boldsymbol{t}^p_{i,m}$ are parallel and the second part of (3.8) vanishes. Hence, the name "FD-TD"-correction is somewhat misleading since that term is actually the important one. The magnetic node values are only used to give a better approximation of the edge-projected field in the case when $\boldsymbol{n}^d_{j,k}$ and $\boldsymbol{t}^p_{i,m}$ do not align.

The boundary condition for a perfect magnetic conductor (PMC) gives us that the tangential component of the magnetic field at the object is zero. A complication occurs whenever the computation of $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p_{i,m}$ does not reduce to the FD-TD term, $\boldsymbol{H} \cdot \boldsymbol{n}^d_{j,k}$, where $\boldsymbol{t}^p_{i,m}$ denotes a primary edge with one node on the boundary. When that is not the case the following alternative is used (see

Figure 3.2):

$$\begin{aligned}\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p_{i,m} &= \boldsymbol{H} \cdot \boldsymbol{n}^d_{j,k} \left(\boldsymbol{n}^d_{j,k} \cdot \boldsymbol{t}^p_{i,m}\right) \\ &+ \frac{1}{2}\left[\boldsymbol{H}_j + \left(\boldsymbol{H}_j \cdot \tilde{\boldsymbol{n}}_r\right)\tilde{\boldsymbol{n}}_r - \left(\boldsymbol{H}_j + \left(\boldsymbol{H}_j \cdot \tilde{\boldsymbol{n}}_r\right)\tilde{\boldsymbol{n}}_r\right) \cdot \boldsymbol{n}^d_{j,k}\right] \cdot \boldsymbol{t}^p_{i,m}\,,\end{aligned} \tag{3.11}$$

where $\tilde{\boldsymbol{n}}_r$ denotes an average normal at the boundary node. This normal is defined by taking the average direction of the two boundary edges including the boundary node and then taking the cross product with that direction and $\hat{z}$.

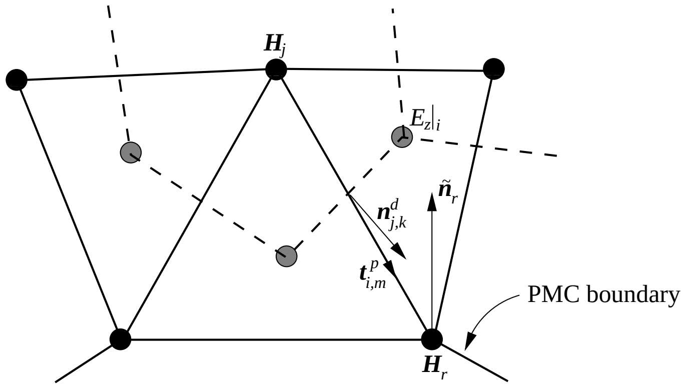


Figure 3.2: Primary and dual cells at a PMC boundary.

For a perfect electric conductor (PEC) the tangential electric components, in the TM case only $E_z$, should equal zero at the boundary. However, the $E_z$ components are not situated on the boundary. The boundary condition is enforced by setting the electric field below the conductor equal to the value of the electric field directly above the conductor with a change of sign, see Figure 3.3.

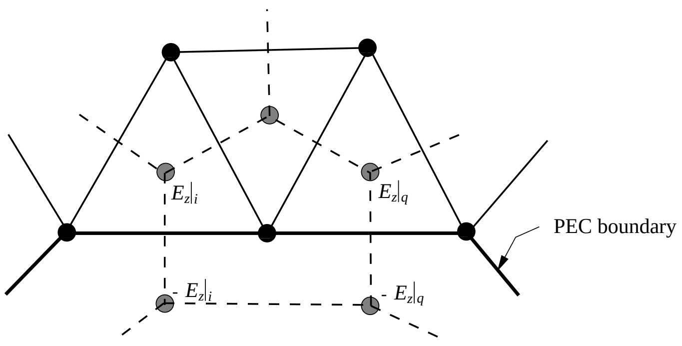


Figure 3.3: The PEC boundary condition is enforced using the method of images.

These uniquely determined virtual image nodes are identified when the dual grid is constructed in the preprocessing phase, see Section 3.3.

## 3.2 Time discretization

We use a third-order staggered Adams–Bashforth scheme (ABS3) [9],

$$
\begin{aligned}
\boldsymbol{H}_j^{n+\frac{1}{2}} &= \boldsymbol{H}_j^{n-\frac{1}{2}} && (3.12)\\
&\quad + \frac{\Delta t}{\tilde{\mu}_j^d\, A_j^d} \sum_k \left( \frac{25}{24} \tilde{E}_z|_{j,k}^n - \frac{1}{12} \tilde{E}_z|_{j,k}^{n-1} + \frac{1}{24} \tilde{E}_z|_{j,k}^{n-2} \right) \boldsymbol{t}_{j,k}^d \Delta_{j,k}^d\,, \\
E_z|_i^{n+1} &= E_z|_i^n + \frac{\Delta t}{\epsilon_i^p\, A_i^p} \sum_m \left( \frac{25}{24} \tilde{\boldsymbol{H}} \cdot \boldsymbol{t}_{i,m}^p|^{n+\frac{1}{2}} - \right. && (3.13)\\
&\qquad \left. \frac{1}{12} \tilde{\boldsymbol{H}} \cdot \boldsymbol{t}_{i,m}^p|^{n-\frac{1}{2}} + \frac{1}{24} \tilde{\boldsymbol{H}} \cdot \boldsymbol{t}_{i,m}^p|^{n-\frac{3}{2}} \right) \Delta_{i,m}^p \\
&\quad + \frac{\sigma_i^p\, \Delta t}{2\, \epsilon_i^p} \left( E_z|_i^{n+1} + E_z|_i^n \right)\,, \\
\boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d|^{n+\frac{1}{2}} &= \boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d|^{n-\frac{1}{2}} + \Delta t\, \mathcal{F} \left( \frac{25}{24} E_z^n - \frac{1}{12} E_z^{n-1} + \frac{1}{24} E_z^{n-2} \right)\,, && (3.14)
\end{aligned}
$$

where $\mathcal{F}$ is an operator taking care of the update of $\boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d$ according to (3.9). Since ABS3 is a staggered time integrator the time-coupling with FD-TD is straightforward, see Chapter 5. Furthermore, its stability properties are superior compared to the traditionally used leap-frog scheme, see Section 3.6.

## 3.3 Creating the dual grid

Perhaps the most demanding task in implementing the FV solver is the construction of the dual grid from the primary grid. This is done in the preprocessing phase by joining barycenters of the neighboring cells. Hence, the dual grid is uniquely determined by the primary grid.

Starting with the inner edges, i.e. edges not situated on a boundary, each inner edge in the primary grid is crossed by a dual edge, see Figure 3.4. These dual edges belong to the dual elements surrounding the start- and stop-nodes of the respective primary edges. The three different boundary conditions, PEC, PMC and Hybrid are treated separately. The PMC boundary is easiest because no dual element should surround a node lying on a PMC boundary (see Figure 3.2). The magnetic fields in these nodes are not updated.

At a PEC boundary, as already mentioned, the electric field nodes are mirrored in the boundary. The dual edges crossing the boundary edges are assigned to the dual elements surrounding the start- and stop-nodes of the boundary edges.

The dual edges lying inside the PEC object are constructed by joining the image nodes together and they are assigned to the dual elements that hold both the dual nodes of these edges.

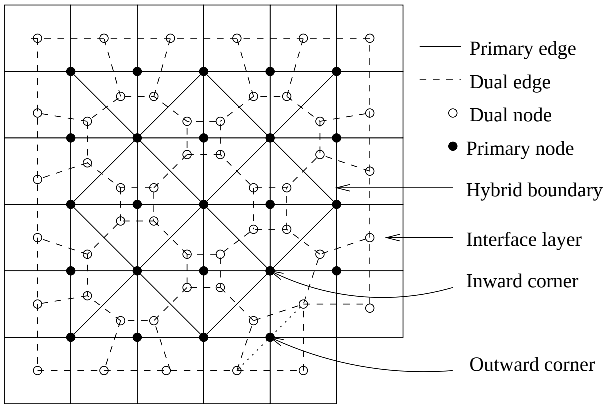


Figure 3.4: Construction of the dual grid from the primary grid.

Note that if we have a very thin PEC object it is possible that two dual elements lying on opposite sides of the boundary will overlap each other geometrically. This does not pose any problems because the image nodes are only virtual and the electric fields in these nodes are not updated, but assigned values from the field in their corresponding node.

The hybrid boundary is slightly more complicated. The nodes lying on a hybrid interface should be updated as usual, i.e. they have dual cells surrounding them, see Section 5.1. The dual edges that cross the edges lying on the interface boundary belong as usual to the dual elements surrounding the start- and stop-nodes of the boundary edges, see Figure 3.4. Generating the outermost dual edges in the interface layer is the trickiest part. Mainly because the interface is allowed to be staircased to minimize the number of unstructured cells. Hence, we cannot as in the PEC case create the outermost dual edges by simply joining the dual nodes lying in the interface layer. We have to take care of the outward corners, which are always four more than the inward corners. We know that we have an outward corner when the midpoint of the dual edge joining two consecutive dual nodes, moving counterclockwise in the interface layer, is the same point as the corner node in the primary grid; see the dotted line in bottom right corner of Figure 3.4. If that is the case we construct a new dual node and remove the dual edge going through the corner. Instead we generate the two dual edges going to and from the new node.

When all dual nodes and dual edges are constructed, the edges in each dual element are sorted such that they are traversed counterclockwise for each dual element. The area of a dual element is calculated by dividing it into triangles and summing the contributions from each triangle.

## 3.4 Preservation of divergence

The finite volume solver is based on an integral formulation of Ampère's and Faraday's laws. However, the Maxwell equations also include the Gauss' laws that have to be satisfied in order to ensure a physical solution. In vacuum the divergence of the magnetic displacement vector, $\boldsymbol{B}$, as well as that of the electric displacement vector, $\boldsymbol{D}$, should equal zero.

**Proposition 3.4.1** *The FV solver preserves* $\nabla \cdot \boldsymbol{D}$.

**Proof**

$$\nabla \cdot \boldsymbol{D} = \nabla \cdot \epsilon \boldsymbol{E} = \frac{\partial\, \epsilon E_x}{\partial x} + \frac{\partial\, \epsilon E_y}{\partial y} + \frac{\partial\, \epsilon E_z}{\partial z} = 0\,.$$

The first two terms equal zero since $\boldsymbol{E}$ only has a z-component in the TM case and the last term is zero because we do not have any variation in the z-direction in 2D. Hence, the divergence of the $\boldsymbol{D}$ field will always equal zero. ■

We now shift focus to the $\boldsymbol{B}$ field. As previously stated it is important that the divergence of the magnetic field is preserved on a local cell level in order to suppress spurious modes.

**Proposition 3.4.2** *The FV solver preserves* $\nabla \cdot \boldsymbol{B}$.

**Proof** For dual cell $j$ we have

$$\begin{aligned}
\frac{\partial}{\partial t} \int_{A_j^d} \nabla \cdot \boldsymbol{B}\, dA &= \frac{\partial}{\partial t} \int_{A_j^d} \nabla \cdot \mu\, \boldsymbol{H}\, dA = \frac{\partial}{\partial t} \oint_{\Gamma_j^d} \mu\, \boldsymbol{H} \cdot \boldsymbol{n}_j^d\, dl \\
&= \frac{\partial}{\partial t} \sum_k \int_{\Gamma_{j,k}^d} \tilde{\mu}_j^d\, \boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d\, dl = \frac{\partial}{\partial t} \sum_k \tilde{\mu}_j^d\, \boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d\, \Delta_{j,k}^d \\
&= \sum_k \tilde{\mu}_j^d\, \frac{\partial \boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d}{\partial t}\, \Delta_{j,k}^d = \sum_k (E_z|_q - E_z|_i) = 0\,,
\end{aligned}$$

where the last expression equals zero because in the sum over $k$ each electric node value occurs twice with opposite signs. Thus, the time derivative of the divergence is equal to zero and since the divergence is zero initially it will always equal zero. ■

Note that if $\boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^d$ instead had been computed as a projection of the arithmetic mean value of the two magnetic node values the divergence would in general have differed from zero.

## 3.5 Dispersion analysis on triangular grids

A frequently used technique to characterize the errors of a numerical scheme is Fourier analysis. Assuming that the fields are periodic in space and neglecting boundary conditions, we make the following ansatz for the three unknown field components:

$$\Psi|_{I,J}^{n} = \Psi_0\, \mathrm{e}^{j\left(\tilde{k}_x I\Delta + \tilde{k}_y J\Delta - \omega n\Delta t\right)}\,, \tag{3.15}$$

where $\tilde{k}_x$ and $\tilde{k}_y$ are the x- and y-components of the numerical wave-vector, respectively. The numerical wave-vector $\tilde{\boldsymbol{k}}$ will in general differ from the physical wave-vector $\boldsymbol{k}$ satisfying $|\boldsymbol{k}| = \omega/c$, where $\omega$ is the angular frequency and $c$ is the speed of light. That difference depends on how well the wave is resolved in the grid, the type of grid and the angle of propagation in the grid. The difference between $\tilde{\boldsymbol{k}}$ and $\boldsymbol{k}$ gives rise to numerical phase and group velocities that depart from the analytical values, which cause numerical errors that accumulate with distance. Hence, the dispersion characteristics of a numerical scheme used for wave propagation problems are crucial to obtain an accurate solution.

In Figure 3.5 the different uniform grids used in the analysis are shown. Note that the normals and edge vectors align in the equilateral and diamond grid. Thus, only $\boldsymbol{H} \cdot \boldsymbol{n}_{j,k}^{d}$ survives expression (3.8), so for these two grids we do not have to take the magnetic field in the nodes into account. For the one-directional grid, however, we have to consider both the magnetic field at the nodes and the FD-TD correction. Since the approximation of the dispersion relation is in the midpoint of the respective building blocks shown in Figure 3.5 the $E_z$ components are interpolated from the barycenters to the midpoint of the building blocks. Note that this interpolation is only performed in the analysis.

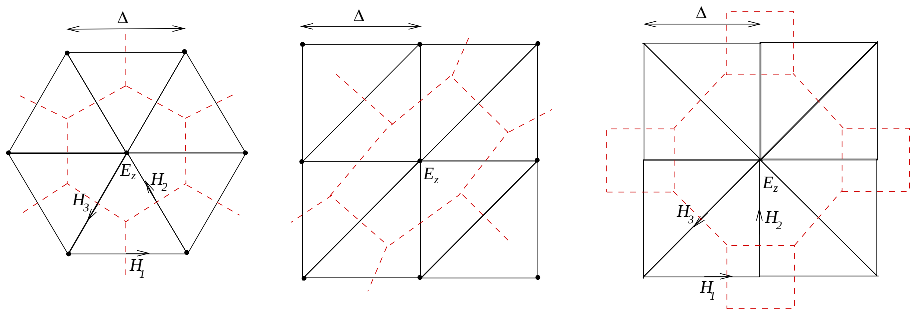


Figure 3.5: The different uniform triangular grids used in the analysis, the equilateral grid (left), the one-directional grid (middle) and the diamond grid (right). The respective dual grids are indicated by dashed lines.

Let us first look at the errors introduced by the spatial discretization for the equilateral grid. Substituting (3.15) into (3.13) and assuming that no error is introduced by the time discretization result in

$$
\begin{aligned}
j\omega E_z = \frac{4}{\sqrt{3}\,\epsilon\Delta} \Bigg[ & \sin\left(-\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right) H_1 + \sin\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) H_2 \\
& + \sin\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) H_3 \Bigg] .
\end{aligned}
\tag{3.16}
$$

Similarly, from (3.14) we have for the three edges (see Figure 3.5)

$$
j\omega H_1 = \frac{2\sqrt{3} j E_z}{\mu\Delta} \sin\left(-\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right) , \tag{3.17}
$$

$$
j\omega H_2 = \frac{2\sqrt{3} j E_z}{\mu\Delta} \sin\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) , \tag{3.18}
$$

$$
j\omega H_3 = \frac{2\sqrt{3} j E_z}{\mu\Delta} \sin\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) . \tag{3.19}
$$

Inserting (3.17)–(3.19) into (3.16) leads to the numerical dispersion relation

$$
\begin{aligned}
\left(\frac{\omega}{c}\right)^2 = \frac{8}{\Delta^2} \Bigg[ & \sin^2\left(\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right) + \sin^2\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) \\
& + \sin^2\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) \Bigg] .
\end{aligned}
\tag{3.20}
$$

Letting $\Delta \to 0$ we obtain

$$
\left(\frac{\omega}{c}\right)^2 = \left(\tilde{k}_x^2 + \tilde{k}_y^2\right) - \frac{1}{48} \left(\tilde{k}_x^2 + \tilde{k}_y^2\right)^2 \Delta^2 + O(\Delta^4) . \tag{3.21}
$$

Thus, the dispersion relation is approximated to second-order accuracy. The same analysis on a square grid with edge length $\Delta$, where the solver is identical to the FD-TD scheme, gives an error constant equal to $\frac{1}{12}$ [24]. The dispersion relations are also approximated to second-order accuracy for the grids with right triangles. For the one-directional grid the relation reads

$$
\begin{aligned}
\left(\frac{\omega}{c}\right)^2 = \left(\tilde{k}_x^2 + \tilde{k}_y^2\right) - \Bigg( & \frac{1}{108}\,\tilde{k}_x^4 - \frac{7}{90}\,\tilde{k}_x^2\tilde{k}_y^2 - \frac{1}{108}\,\tilde{k}_y^4 \\
& + \frac{2}{135}\,\tilde{k}_x^3\tilde{k}_y + \frac{2}{135}\,\tilde{k}_x\tilde{k}_y^3 \Bigg) \Delta^2 + O(\Delta^4) ,
\end{aligned}
\tag{3.22}
$$

and for the diamond grid it is given by

$$
\begin{aligned}
\left(\frac{\omega}{c}\right)^2 = \left(\tilde{k}_x^2 + \tilde{k}_y^2\right) - \frac{1}{36} \Bigg( & \tilde{k}_x^4 + \tilde{k}_y^4 + \frac{1}{2}\,\tilde{k}_x^3\tilde{k}_y \\
& + \frac{1}{2}\,\tilde{k}_x\tilde{k}_y^3 + \frac{4}{3}\,\tilde{k}_x^2\tilde{k}_y^2 \Bigg) \Delta^2 + O(\Delta^4) .
\end{aligned}
\tag{3.23}
$$

The dispersion error characteristics can be presented in terms of the error in the phase speed of waves of different grid resolution and propagation angle. Let $\tilde{k}_x = \tilde{k}\, cos(\alpha)$ and $\tilde{k}_y = \tilde{k}\, \sin(\alpha)$, where $\alpha$ is the angle of propagation relative to the x-axis and $\tilde{k} = |\tilde{\boldsymbol{k}}|$. Substituting these two expressions into the dispersion relation (3.20) and solving the nonlinear equation for $\tilde{k}$, results in a relation between the numerical wave speed, $v_p$, and $c$, since $v_p/c = k/\tilde{k}$. Figure 3.6 illustrates the numerical phase velocity as a function of propagation angle and grid resolution, where $\lambda$ is the wavelength. The results obtained by the FV solver are in excellent agreement with the true wave speed for all three grids even for a moderate grid resolution of ten points per wavelength. The dispersion characteristics are clearly better for triangular grids than for square grids. It is interesting to note that the numerical phase velocity is independent of the propagation direction for the uniform equilateral grid..

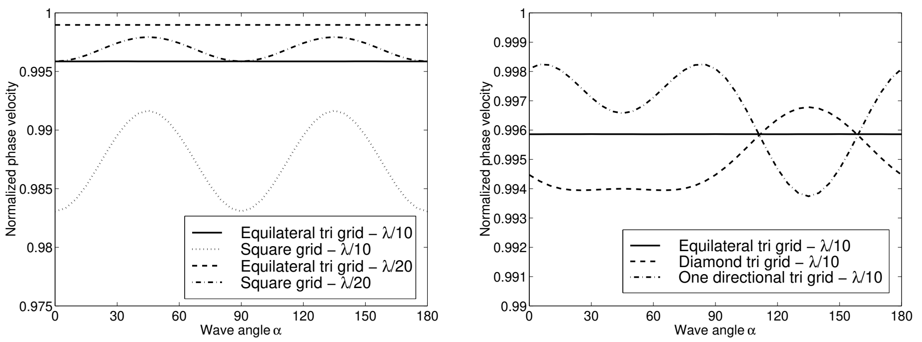


Figure 3.6: Variation of numerical phase velocity for different grids as a function of grid resolution and propagation angle.

We now turn our attention to the fully discrete case. Substituting (3.15) into (3.14), using (3.16) and discretize the time derivative with ABS3 we obtain

$$
\begin{aligned}
&-\frac{2j}{\Delta t}\sin\left(\frac{\omega\Delta t}{2}\right) E_z = \frac{4j}{\sqrt{3}\epsilon\Delta}\left(\frac{25}{24} - \frac{1}{12}\,\mathrm{e}^{-j\omega\Delta t} + \frac{1}{24}\,\mathrm{e}^{-2j\omega\Delta t}\right) \\
&\left[H_1 \sin\left(-\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right) + H_2 \sin\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) + H_3 \sin\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right)\right]
\end{aligned}
\tag{3.24}
$$

Similarly for the update of the magnetic fields we substitute (3.15) into (3.13), using (3.17)–(3.19)

$$
\begin{aligned}
&-\frac{2j}{\Delta t}\sin\left(\frac{\omega\Delta t}{2}\right)\begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix} = \\
&\frac{2j\sqrt{3}E_z}{\mu\Delta}\left(\frac{25}{24} - \frac{1}{12}\,\mathrm{e}^{-j\omega\Delta t} + \frac{1}{24}\,\mathrm{e}^{-2j\omega\Delta t}\right)\begin{bmatrix} \sin\left(-\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right) \\ \sin\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) \\ \sin\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) \end{bmatrix}
\end{aligned} \tag{3.25}
$$

Finally, substituting (3.25) into (3.24) we obtain the following fully discrete dispersion relation for the grid with equilateral triangles:

$$
\begin{aligned}
&\frac{4}{\Delta t^2}\sin^2\left(\frac{\omega\Delta t}{2}\right) = \frac{8c^2}{\Delta^2}\left(\frac{25}{24} - \frac{1}{12}\mathrm{e}^{-j\omega\Delta t} + \frac{1}{24}\mathrm{e}^{-2j\omega\Delta t}\right)^2 \\
&\left[\sin^2\left(\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right) + \sin^2\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) + \sin^2\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right)\right].
\end{aligned} \tag{3.26}
$$

The time step is chosen according to the stability limit for ABS3 (see next section) as $c\Delta t = \frac{6}{7}\frac{\Delta}{\sqrt{6}}$. As in the semi-discrete case we solve (3.26) for $\tilde{k}$. A plot of $v_p/c$ shows almost identical results to those in Figure 3.6 due to the fact that ABS3 is a third-order accurate scheme and the errors from the spatial discretization dominate. If we use the leap-frog scheme for the timestepping we get a small improvement in the dispersion characteristics since that scheme is only second-order accurate and the errors from the spatial and time discretization cancel to some extent. However, the errors are small in either case and, as will be discussed in the next section, the stability properties of the leap-frog scheme make it less suitable to use for general problems.

## 3.6 Stability analysis

An important issue for explicit solvers of the Maxwell equations is how to achieve long-term stability without adding too much artificial dissipation. The stability region for ABS3 for the scalar test equation $u' = \lambda u$ is given in Figure 3.7.

The scheme is stable for $\lambda$ between $\pm 12/7$ along the imaginary axis compared to the leap-frog scheme, which is stable between $\pm 2$. That implies that we have to use a shorter time step for ABS3. However, the main disadvantage with leap-frog is that it is only stable on the imaginary axis and becomes unstable as soon as we have eigenvalues with a nonzero real part, which we are likely to have on unstructured grids and when boundaries are taken into account.

By introducing operators $\mathcal{A}$ and $\mathcal{B}$ that take care of the spatial discretization

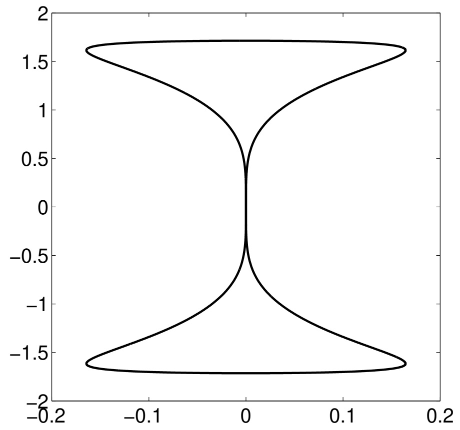


Figure 3.7: Stability region for ABS3.

we are able to write the semi-discrete problem on matrix form as

$$\frac{\partial}{\partial t}\begin{pmatrix}\hat{\boldsymbol{H}}\\ E_z\end{pmatrix} = \begin{pmatrix}\mathbf{0} & \mathcal{A}\\ \mathcal{B} & \mathbf{0}\end{pmatrix}\begin{pmatrix}\hat{\boldsymbol{H}}\\ E_z\end{pmatrix}, \tag{3.27}$$

where $\hat{\boldsymbol{H}} = \begin{pmatrix}\boldsymbol{H} & \boldsymbol{H}\cdot\boldsymbol{n}^d\end{pmatrix}$.

We begin by looking at the uniform triangular grids used in the analysis of the dispersion error. For the grid consisting of equilateral triangles the operator $\mathcal{A}$ is obtained from (3.25),

$$\mathcal{A} = \frac{2j\sqrt{3}E_z}{\mu\Delta}\begin{bmatrix}\sin\left(-\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right)\\ \sin\left(\frac{\tilde{k}_x\Delta}{4}+\frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right)\\ \sin\left(-\frac{\tilde{k}_x\Delta}{4}+\frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right)\end{bmatrix}, \tag{3.28}$$

whereas the operator $\mathcal{B}$ is obtained from (3.24),

$$\mathcal{B} = \frac{4j}{\sqrt{3}\epsilon\Delta}\begin{bmatrix}\sin\left(-\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right)\\ \sin\left(\frac{\tilde{k}_x\Delta}{4}+\frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right)\\ \sin\left(-\frac{\tilde{k}_x\Delta}{4}+\frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right)\end{bmatrix}^T. \tag{3.29}$$

The eigenvalues of the spatial operator are purely imaginary and given by

$$
\begin{aligned}
\sqrt{\mathcal{B}\mathcal{A}} = \pm j \left( \frac{8c^2}{\Delta^2} \left[ \sin^2\left(\frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right) + \sin^2\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right) \right.\right. \\
\left.\left. + \sin^2\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\tilde{k}_y\Delta}{4\sqrt{3}}\right)\right]\right)^{\frac{1}{2}},
\end{aligned}
\tag{3.30}
$$

The $\sin^2$ terms are between zero and one, therefore the eigenvalues are bounded by

$$
-j\frac{2\sqrt{6}\,c}{\Delta} \leq \sqrt{\mathcal{B}\mathcal{A}} \leq j\frac{2\sqrt{6}\,c}{\Delta}\,. \tag{3.31}
$$

To obtain stability a necessary condition is that all eigenvalues of the spatial operator are within the stability region for the time discretization operator. Hence, the stability condition for the leap-frog scheme on the equilateral triangular grid equals

$$
c\Delta t_{LF} \leq \frac{\Delta}{\sqrt{2}} \cdot \frac{\Delta_{min}}{\Delta}\,, \tag{3.32}
$$

where the first part in the right-hand side is the stability condition on Cartesian grids and $\Delta_{min}$ equals the shortest edge length in the primary and dual grids. On the equilateral grid $\Delta_{min} = \Delta/\sqrt{3}$.

We now analyze how ABS3 behaves in our case. Using the operators $\mathcal{A}$ and $\mathcal{B}$ we are after some straightforward algebra able to write (3.12)–(3.14) on matrix form as

$$
\begin{aligned}
\begin{pmatrix} \hat{\boldsymbol{H}}^{n+\frac{1}{2}} \\ E_z|^{n+1} \end{pmatrix} &= \begin{pmatrix} \mathcal{I} & \frac{25}{24}\Delta t\mathcal{A} \\ \frac{23}{24}\Delta t\mathcal{B} & \mathcal{I} + \frac{625}{576}\Delta t^2\mathcal{B}\mathcal{A} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{H}}^{n-\frac{1}{2}} \\ E_z|^{n} \end{pmatrix} \\
&+ \begin{pmatrix} \mathbf{0} & -\frac{1}{12}\Delta t\mathcal{A} \\ \frac{1}{24}\Delta t\mathcal{B} & -\frac{25}{288}\Delta t^2\mathcal{B}\mathcal{A} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{H}}^{n-\frac{3}{2}} \\ E_z|^{n-1} \end{pmatrix} \\
&+ \begin{pmatrix} \mathbf{0} & \frac{1}{24}\Delta t\mathcal{A} \\ \mathbf{0} & -\frac{25}{576}\Delta t^2\mathcal{B}\mathcal{A} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{H}}^{n-\frac{5}{2}} \\ E_z|^{n-2} \end{pmatrix}.
\end{aligned}
\tag{3.33}
$$

The plane wave ansatz

$$
\begin{pmatrix} \hat{\boldsymbol{H}}^{n+\frac{1}{2}} \\ E_z|^{n+1} \end{pmatrix} = \begin{pmatrix} \hat{\boldsymbol{H}}^{\frac{1}{2}} \\ E_z|^{1} \end{pmatrix} \rho^n\,, \tag{3.34}
$$

leads to the following generalized eigenvalue problem

$$\rho^{n-3}\left(\mathcal{I}\rho^3 - \tilde{\mathcal{A}}\rho^2 - \tilde{\mathcal{B}}\rho - \tilde{\mathcal{C}}\right)\begin{pmatrix}\hat{\boldsymbol{H}}^{\frac{1}{2}} \\ E_z|^1\end{pmatrix} = 0\,, \tag{3.35}$$

where $\tilde{\mathcal{A}}, \tilde{\mathcal{B}}$ and $\tilde{\mathcal{C}}$ are the matrices in (3.33), respectively. When we solve the generalized eigenvalue problem (3.35) we observe that stability is obtained if the time step suggested by (3.32) is decreased by a factor 6/7, which is exactly the relative difference in stability along the imaginary axis (see Figure 3.7) between the two schemes. Hence, the stability condition for ABS3 equals

$$c\Delta t_{ABS3} \leq \frac{6}{7}\frac{\Delta}{\sqrt{2}} \cdot \frac{\Delta_{min}}{\Delta}\,. \tag{3.36}$$

This stability condition holds true also for the other two triangular grids, which a similar analysis verifies.

To analyze the eigenvalues for a general unstructured grid including its boundaries we can no longer use Fourier analysis. Instead, let

$$\boldsymbol{z}^n = \begin{pmatrix} E_z|^n & \hat{\boldsymbol{H}}^{n-\frac{1}{2}} & E_z|^{n-1} & \hat{\boldsymbol{H}}^{n-\frac{3}{2}} & E_z|^{n-2}\end{pmatrix}^T, \tag{3.37}$$

and after some straightforward rearrangements we are able to write (3.12)–(3.14) on matrix form as $\boldsymbol{z}^{n+1} = \mathcal{P}(\mathcal{A},\mathcal{B})\boldsymbol{z}^n$, where

$$\mathcal{P} = \begin{pmatrix} \mathcal{I} + \frac{625}{576}\Delta t^2\mathcal{B}\mathcal{A} & \frac{23}{24}\Delta t\mathcal{B} & -\frac{25}{288}\Delta t^2\mathcal{B}\mathcal{A} & \frac{1}{24}\Delta t\mathcal{B} & \frac{25}{576}\Delta t^2\mathcal{B}\mathcal{A} \\ \frac{25}{24}\Delta t\mathcal{A} & \mathcal{I} & -\frac{1}{12}\Delta t\mathcal{A} & \mathbf{0} & \frac{1}{24}\Delta t\mathcal{A} \\ \mathcal{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{I} & \mathbf{0} & \mathbf{0} \end{pmatrix}. \tag{3.38}$$

Analyzing the eigenvalues of the companion matrix $\mathcal{P}$, for the grid shown in Figure 3.8, reveals that if we choose the time step for ABS3 in the same manner as above we get the eigenvalue spectrum shown in Figure 3.8, where all eigenvalues are within the unit circle. If we use the leap-frog scheme with the same time step, the largest eigenvalue is of the order 1.0003. Hence the leap-frog scheme is unstable even for a time step well within the stability limit along the imaginary axis. The eigenvalues close to the origin in Figure 3.8 have to do with the fact that ABS3 is a multi-step scheme, but only one of the roots is relevant and we get a number of parasitic roots. However, since these roots are all close or equal to zero they are quickly damped away.

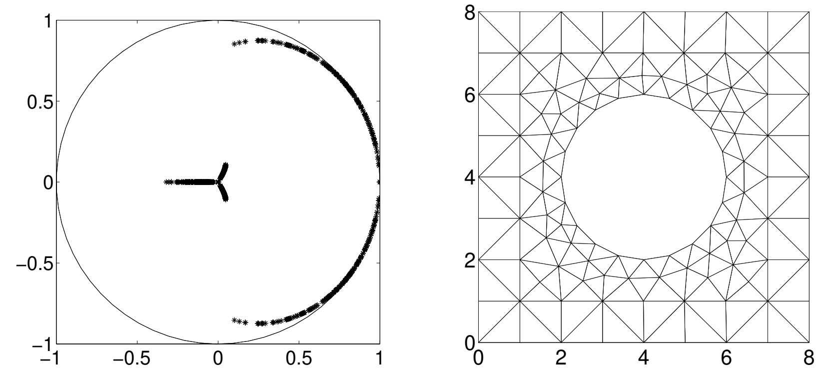


Figure 3.8: Eigenvalues of the companion matrix $P$ using ABS3 for a PMC cylinder scattering case. Primary grid around the cylinder is shown to the right.

## 3.7 Workload and memory requirements

To implement the solver efficiently we have chosen to compute as much as possible initially. Hence, the update of the fields is accomplished using matrices acting on the respective vectors. Due to the fact that the matrices will be sparse but without structure we store them in compress sparse row format. Hence, for each matrix we store three vectors. The first vector holds the number of nonzeros per row, the second vector keeps track of the columns of the nonzeros and the last vector contains their values. After the matrices have been created, all grid variables can be written to disk. Hence, this approach is much more efficient than recomputing the expressions needed to update the field variables at every time step or using indirect addressing in several levels.

To be able to obtain a theoretical estimate of the memory requirements and number of arithmetic operations of the solver we have to make a few assumptions. First of all we neglect boundary conditions. We put $\sigma$ equal to zero, but $\epsilon$ and $\mu$ may vary in space. The following is assumed about the grid:

- There are $n$ triangles.
- There are three edges in each triangle and each edge is shared by two triangles $\Rightarrow 1.5\,n$ edges.
- There are three nodes in each triangle and each node belongs to six triangles $\Rightarrow 0.5n$ nodes and six dual edges per dual cell.

The field variables that we need to store are $E_z$, $\boldsymbol{H} \cdot \boldsymbol{n}^d$, $\boldsymbol{H}$ and $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p$. Using ABS3, $E_z$ and $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p$ are stored at three time levels, whereas $\boldsymbol{H} \cdot \boldsymbol{n}^d$ and $\boldsymbol{H}$ only are stored at the latest time level.

| Variable | Floating point numbers |
|---|---|
| $E_z$ | $3 \cdot n$ |
| $\boldsymbol{H} \cdot \boldsymbol{n}^d$ | $1.5\,n$ |
| $\boldsymbol{H}$ | $2 \cdot 0.5\,n$ |
| $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p$ | $3 \cdot 1.5\,n$ |
| $\sum$ | $10\,n$ |

Table 3.1: Memory requirements to store the field variables.

The memory requirements for the matrix operators used to update the field variables can be calculated from (3.6)–(3.9).

| Variable | Floating point numbers | Integers |
|---|---|---|
| $E_z$ | $n \cdot 3 = 3\,n$ | $3\,n + n = 4\,n$ |
| $\boldsymbol{H} \cdot \boldsymbol{n}^d$ | $1.5\,n \cdot 2 = 3\,n$ | $3\,n + 1.5\,n = 4.5\,n$ |
| $\boldsymbol{H}$ | $n \cdot 6 = 6\,n$ | $6\,n + n = 7\,n$ |
| $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p$ | $1.5\,n \cdot 5 = 7.5\,n$ | $7.5\,n + 1.5\,n = 9\,n$ |
| $\sum$ | $19.5\,n$ | $24.5\,n$ |

Table 3.2: Memory requirements to store the matrices.

Hence, using the results in Tables 3.1 and 3.2 the total memory requirements for the FV solver is 29.5 floating point numbers and 24.5 integers per cell. Using 64-bit precision for the floating point numbers and 32-bit precision for the integers this means that approximately 334 bytes per cell are needed.

The number of arithmetic operations used by the solver are calculated from (3.6)–(3.9) and (3.12)–(3.14).

| Variable | Arithmetic operations |
|---|---|
| $E_z$ | $n \cdot 11 = 11\,n$ |
| $\boldsymbol{H} \cdot \boldsymbol{n}^d$ | $1.5\,n \cdot 4 = 6\,n$ |
| $\boldsymbol{H}$ | $n \cdot 12 = 12\,n$ |
| $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p$ | $1.5\,n \cdot 14 = 21\,n$ |
| $\tilde{E}_z$ | $n \cdot 5 = 5\,n$ |
| $\sum$ | $55\,n$ |

Table 3.3: Number of arithmetic operations per cell.

Hence, from Table 3.3 we conclude that the FV solver performs approximately 55 arithmetic operations per cell for each time step. This should be compared with FD-TD, which needs eleven arithmetic operations per cell and 36 bytes of memory. Hence, the FV solver is a factor 5 more expensive in terms of arithmetic operation per cell and a factor of nine in terms of memory per cell compared to FD-TD.

Note that in the above calculations we have not used the fact that when the primary and dual edges are orthogonal the second part of (3.8) vanishes and the $\boldsymbol{H}$ values are not needed. A case where this will happen is the equilateral grid. An implementation of the solver should of course take advantage of this and the memory requirements and workload will go down considerably. On an equilateral grid, or any other grid where orthogonality occurs, the memory requirements are 16.5 floating point numbers and 10 integers per cell. The total memory requirement is in this case 172 bytes using the same precision. The number of arithmetic operations goes down to 31 per cell for such a grid. Thus, it is possible to reduce the memory requirements and the number of arithmetic operations by approximately a factor of two for the special case when the grids are mutually orthogonal.

# Chapter 4

# Finite Volume method in 3D

The FV solver is based on the integral formulations (2.10) and (2.12) of Faraday's and Ampere's laws repeated here for convenience

$$\frac{\partial}{\partial t}\int_A \mu\, \boldsymbol{H}\cdot\boldsymbol{n}\, dA \;=\; -\oint_\Gamma \boldsymbol{E}\cdot\boldsymbol{dl}\,, \tag{4.1}$$

$$\frac{\partial}{\partial t}\int_A \epsilon\, \boldsymbol{E}\cdot\boldsymbol{n}\, dA \;=\; \oint_\Gamma \boldsymbol{H}\cdot\boldsymbol{dl} - \int_A \sigma\, \boldsymbol{E}\cdot\boldsymbol{n}\, dA\,, \tag{4.2}$$

where $A$ is an arbitrary area, $\Gamma$ is the path that encloses $A$ and $\boldsymbol{n}$ is a unit normal.

## 4.1 Space discretization

The integral formulations (4.1) and (4.2) are discretized on a staggered grid. The primary grid consists of tetrahedrons, which are generated by a Delaunay grid algorithm. Associated with the unstructured tetrahedral grid is a mutually orthogonal dual grid, the Dirichlet tessellation; see Section 4.3 for a detailed description on how the dual grid is created. The nodes in the dual grid are the centers of the circumscribed spheres of each tetrahedron. However, for a general tetrahedral grid there are circumcenters that lie outside the corresponding tetrahedrons. Such nodes deteriorate the accuracy of the solver and they are hence replaced by the barycenters of these tetrahedrons. Replacing a circumcenter with a barycenter implies that the mutual orthogonality is lost locally. Note that in 2D all dual nodes are situated at the barycenters of the triangles. However, since the approach taken here is better we will eventually change the dual grid generation in 2D.

The normal components of the magnetic field are stored at each triangular face, whereas the normal components of the electric field are stored at each dual face, see Figure 4.1.

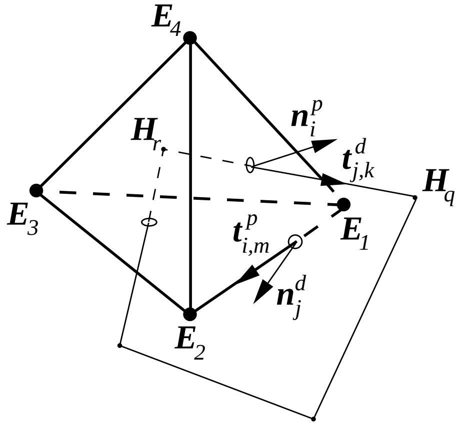


Figure 4.1: A cell in the primary grid and a dual face.

Integrating (4.1) and (4.2) over a primary and a dual face, respectively, results in

$$\tilde{\mu}_i^p \frac{\partial}{\partial t} \int_{A_i^p} \boldsymbol{H} \cdot \boldsymbol{n}_i^p \, dA = -\sum_m \int_{\Gamma_{i,m}^p} \tilde{\boldsymbol{E}} \cdot \boldsymbol{t}_{i,m}^p \, dl \,, \tag{4.3}$$

$$\tilde{\epsilon}_j^d \frac{\partial}{\partial t} \int_{A_j^d} \boldsymbol{E} \cdot \boldsymbol{n}_j^d \, dA = \sum_k \int_{\Gamma_{j,k}^d} \tilde{\boldsymbol{H}} \cdot \boldsymbol{t}_{j,k}^d \, dl - \tilde{\sigma}_j^d \int_{A_j^d} \boldsymbol{E} \cdot \boldsymbol{n}_j^d \, dA \,, \tag{4.4}$$

where $A_i^p$ is the area of the primary face $i$, $\Gamma_i^p$ is the path that encloses $A_i^p$, $\boldsymbol{n}_i^p$ is the unit face normal for face $i$ and $\boldsymbol{t}_{i,m}^p$ are the unit vectors for the edges in face $i$. The variables belonging to the dual face $j$ are defined in the same way.

In the case when the primary and dual grids are mutually orthogonal the normals and edge vectors of the respective grids are parallel and the solver is basically a generalization of FD-TD to unstructured grids. The normal component of the electric field, $\boldsymbol{E} \cdot \boldsymbol{n}_j^d$, is updated by circulating the magnetic field along the edges of the dual face. The normal component of the magnetic field, $\boldsymbol{H} \cdot \boldsymbol{n}_i^p$, at each triangular face is updated in the dual way by circulating the electric field along the edges of the triangular face.

However, for non-orthogonal grids the normals and edge vectors do not align and we need an approximation for the electric and magnetic fields projected along the respective edges. This is accomplished by first calculating an approximate value of the fields at the primary and dual nodes, respectively. The magnetic fields at the dual nodes, e.g. $\boldsymbol{H}_r$ in Figure 4.1, are calculated as a least square fit of the magnetic fields normal to the four faces of each tetrahedron. Electric node values are defined by doing a similar fit for the electric fields normal to the dual faces of each dual cell. Note that calculating the node values in this way differs from what is used in 2D. In 2D the node values are updated using the two area integral formulations (3.1) and (3.2). However, using the corresponding volume integral formulations in 3D leads to significant node depolarization. For a plane wave travelling in a grid along one of the Cartesian directions only one of the components in each vector field is supposed to be different from zero for perfect propagation. If any of the other components is different from zero depolarization of the wave has occured. Looking at the vector fields at the nodes alone, we have observed depolarization as high as on the order of ten percent. The amount of depolarization depends on the grid and is also present using the least squares approach but to much less extent. However, this is not the whole story, because it is the edge projected fields that are important. The same situation occurs when using edge-based finite element methods in the unstructured grid. The edge projections tend to be accurate, but the node values are not.

Following Riley et al. [20], the magnetic field projected along the dual edge $\boldsymbol{t}^d_{j,k}$, see Figure 4.1, is evaluated as

$$\begin{aligned}\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^d_{j,k} &= (\boldsymbol{H} \cdot \boldsymbol{n}^p_i)\left(\boldsymbol{n}^p_i \cdot \boldsymbol{t}^d_{j,k}\right) \\ &\quad + \frac{1}{2}\left[(\boldsymbol{H}_r + \boldsymbol{H}_q) - ((\boldsymbol{H}_r + \boldsymbol{H}_q) \cdot \boldsymbol{n}^p_i)\, \boldsymbol{n}^p_i\right] \cdot \boldsymbol{t}^d_{j,k} \,.\end{aligned} \tag{4.5}$$

The electric field projected along the primary edge $\boldsymbol{t}^p_{i,m}$ is evaluated accordingly as

$$\begin{aligned}\tilde{\boldsymbol{E}} \cdot \boldsymbol{t}^p_{i,m} &= \left(\boldsymbol{E} \cdot \boldsymbol{n}^d_j\right)\left(\boldsymbol{n}^d_j \cdot \boldsymbol{t}^p_{i,m}\right) \\ &\quad + \frac{1}{2}\left[(\boldsymbol{E}_1 + \boldsymbol{E}_2) - \left((\boldsymbol{E}_1 + \boldsymbol{E}_2) \cdot \boldsymbol{n}^d_j\right) \boldsymbol{n}^d_j\right] \cdot \boldsymbol{t}^p_{i,m} \,.\end{aligned} \tag{4.6}$$

Taking a closer look at (4.5) and (4.6) we note that if the normal vectors, $\boldsymbol{n}^p_i$ and $\boldsymbol{n}^d_j$, are parallel to the edge vectors, $\boldsymbol{t}^d_{j,k}$ and $\boldsymbol{t}^p_{i,m}$, the second parts of (4.5) and (4.6) vanish. Evaluating the fields projected along the edges in this way guarantees that the divergence of the electric and magnetic fields are preserved on a local cell level up to machine precision; see Section 4.4 for a proof. This has been found crucial in order to suppress spurious modes in the solution.

All materials are defined relative to primary grid cells. Hence, the material parameters require averaging to get approximate values at the faces. The average permeability on primary face $i$, $\tilde{\mu}^p_i$, is defined by

$$\tilde{\mu}^p_i = \frac{2}{\frac{1}{\mu^p_r} + \frac{1}{\mu^p_q}} \,, \tag{4.7}$$

where $r$ and $q$ are the tetrahedrons sharing face $i$. At a boundary face, $\tilde{\mu}^p_i$ equals the permeability of the only tetrahedron associated with the face. The average electric material parameters on dual face $j$, $\tilde{\epsilon}^d_j$ and $\tilde{\sigma}^d_j$, are constructed as

$$\tilde{\epsilon}^d_j = \frac{\sum\limits_q \epsilon^p_q A^d_{j,q}}{A^d_j} \,, \quad \tilde{\sigma}^d_j = \frac{\sum\limits_q \sigma^p_q A^d_{j,q}}{A^d_j} \,, \tag{4.8}$$

where $A^d_{j,q}$ is the part of the area of the dual face which is inside tetrahedron $q$ and the sums are taken over the tetrahedrons associated with the dual face. Performing the material averaging in this manner preserves the second-order accuracy of the solver for non-homogeneous materials. The reason for the different averaging of magnetic and electric materials is that the normal magnetic field is discontinuous at a material interface, whereas the tangential electric field is continuous. Therefore, harmonic averaging is used for $\mu$ and area weighted arithmetic averaging is used for $\epsilon$ and $\sigma$ [3].

The boundary condition for a PEC is implemented in the same way as the PMC boundary condition in 2D, see (3.11). For a PMC the tangential magnetic components, should equal zero at the boundary. However, the $\boldsymbol{H}$components are not situated on the boundary. The boundary condition is enforced by setting the magnetic field below the conductor equal to the value of the magnetic field directly above the conductor with a change of sign in the tangential component, cf. the PEC boundary condition in 2D for the TM Maxwell's equations. The PMC boundary condition is non-physical, but useful as a symmetry boundary condition.

## 4.2 Time discretization

The equations (4.3) and (4.4) are discretized in time using the ABS3 scheme

$$\begin{aligned}
\boldsymbol{H}\cdot\boldsymbol{n}^p_i|^{n+\frac{1}{2}} &= \boldsymbol{H}\cdot\boldsymbol{n}^p_i|^{n-\frac{1}{2}} \qquad (4.9)\\
&+\frac{\Delta t}{\tilde{\mu}^p_i\,A^p_i}\sum_m\left(\frac{25}{24}\tilde{\boldsymbol{E}}\cdot\boldsymbol{t}^p_{i,m}|^n-\frac{1}{12}\tilde{\boldsymbol{E}}\cdot\boldsymbol{t}^p_{i,m}|^{n-1}+\frac{1}{24}\tilde{\boldsymbol{E}}\cdot\boldsymbol{t}^p_{i,m}|^{n-2}\right)\Delta^p_{i,m}\,,\\
\boldsymbol{E}\cdot\boldsymbol{n}^d_j|^{n+1} &= \boldsymbol{E}\cdot\boldsymbol{n}^d_j|^n \qquad (4.10)\\
&+\frac{\Delta t}{\tilde{\epsilon}^d_j\,A^d_j}\sum_n\left(\frac{25}{24}\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}^d_{j,k}|^{n+\frac{1}{2}}-\frac{1}{12}\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}^d_{j,k}|^{n-\frac{1}{2}}+\frac{1}{24}\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}^d_{j,k}|^{n-\frac{3}{2}}\right)\Delta^d_{j,k}\\
&+\Delta t\,\frac{\tilde{\sigma}^d_j}{2\,\tilde{\epsilon}^d_j}\left(\boldsymbol{E}\cdot\boldsymbol{n}^d_j|^{n+1}+\boldsymbol{E}\cdot\boldsymbol{n}^d_j|^n\right)\,,
\end{aligned}$$

where $\Delta^p_{i,m}$ are the lengths of the primary edges $\boldsymbol{t}^p_{i,m}$ and $\Delta^d_{j,k}$ are the lengths of the dual edges $\boldsymbol{t}^d_{j,k}$. Note that the notation $\boldsymbol{E}\cdot\boldsymbol{n}^d_j|^n$ means $\boldsymbol{E}\cdot\boldsymbol{n}^d_j$ at time step $n$ and accordingly for the other components. The stability properties of the ABS3 scheme are analyzed in Section 4.5.

## 4.3 Creating the dual grid

The creation of the dual grid from the primary grid is performed in the preprocessing phase. To simplify the description the dual grid at a PMC, used as symmetry boundary condition, is not treated here. In 3D the dual grid does not overlap the hybrid boundary, cf. Figure 3.4, which makes the generation considerably easier. Each tetrahedron has a corresponding dual node. The dual node is situated at the center of the circumscribed sphere if the center is inside the tetrahedron. As discussed previously the circumcenters are preferred due to orthogonality properties, however since dual nodes outside the tetrahedrons will destroy the accuracy of the solver such nodes are replaced by the barycenters.

A dual edge passes through every inner triangular face, e.g. $\boldsymbol{t}^d_{j,n}$ in Figure 4.1. Similarly, each inner primary edge, e.g. $\boldsymbol{t}^p_{i,m}$ in Figure 4.1, has a corresponding dual face. The dual edge $\boldsymbol{t}^d_{j,n}$ belongs to at most three dual faces, namely the faces that are associated with the three edges of the triangular face $\boldsymbol{n}^p_i$ in Figure 4.1. The dual edges in each dual face are sorted to specify a certain circulation around the perimeter of the face, which in turn defines a normal direction.

Finally, each inner node has a surrounding dual element. The dual faces belong to two dual elements, e.g. the dual face with normal $\boldsymbol{n}^d_j$ in Figure 4.1 belongs to the dual elements which surround the nodes 1 and 2. The face is positive for one of these elements, i.e. the normal of the face is defined outwards, and negative for the other, i.e. the normal is defined inwards. Hence, there is a one-to-one correspondence between the nodes, edges, faces and cells of the primary grid and the nodes, edges, faces and cells of the dual grid.

The definition of dual grid parameters such as length and direction of dual edges is trivial. The areas and normals of the dual faces are slightly more complicated due to the fact that the dual faces are not necessarily planar. The face area normal, $\boldsymbol{N}^d_j$, is defined as $\boldsymbol{N}^d_j = \int_{A^d_j} \boldsymbol{n}^d_j \, dA$. It is calculated by splitting the face into triangles using the barycenter of the face. The area of the face is then calculated as a dot product between the unit normal, $\boldsymbol{n}^d_j$, and the face area normal, $\boldsymbol{N}^d_j$. Hence, the areas and normals are calculated by a piecewise planar approximation of the faces.

## 4.4 Preservation of divergence

Here we prove that the FV solver preserves the divergence of the magnetic flux density vector, $\boldsymbol{B}$. The proof for the divergence of the electric flux density vector, $\boldsymbol{D}$, in a source free medium is analogous.

**Proposition 4.4.1** *The FV solver preserves* $\nabla \cdot \boldsymbol{B}$.

**Proof** Consider,

$$\begin{aligned}
\frac{\partial}{\partial t} \int_{V^p_q} \nabla \cdot \boldsymbol{B} \, dV &= \frac{\partial}{\partial t} \int_{V^p_q} \nabla \cdot (\mu \boldsymbol{H}) \, dV = \frac{\partial}{\partial t} \oint_{A^p} \mu \, \boldsymbol{H} \cdot \boldsymbol{n}^p \, dA \\
&= \sum_i \tilde{\mu}^p_i \frac{\partial}{\partial t} \int_{A^p_i} \boldsymbol{H} \cdot \boldsymbol{n}^p_i \, dA = \sum_i \sum_m \int_{\Gamma^p_{i,m}} \tilde{\boldsymbol{E}} \cdot \boldsymbol{t}^p_{i,m} \, dl = 0 \,,
\end{aligned}$$

where Gauss' divergence theorem and (4.4) have been used. The last expression is zero because in the double sum the electric field projected along the edges belonging to the faces $i$ appear twice with opposite signs. Hence, the divergence of the magnetic flux density vector is preserved up to machine precision. ■

Note that the proof does not assume that the faces have to be planar, which is not necessarily the case for the dual faces.

## 4.5 Stability analysis

In Section 3.6 we analyzed the ABS3 scheme on uniform triangular grids and on a general triangular grid. We have also shown in experiments, see Appendix A, that if the time step is chosen according to (3.36) ABS3 is stable for general triangular grids. In this section we will analyze ABS3 in 3D. The great advantage of using ABS3 compared to the commonly used leap-frog scheme is that we obtain a stable and accurate solution even if the highest frequency eigenvalues of the space operator have a small real part. Using ABS3 will suppress the amplitude of the highest frequency components, but they are not accurately resolved by the grid and are therefore considered parasitic.

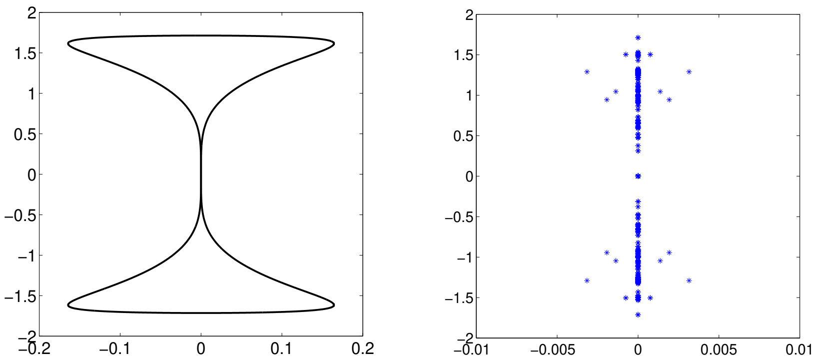


Figure 4.2: The stability region for ABS3 is shown to the left. To the right the eigenvalues for the perturbed $4 \times 4 \times 4$ grid. Note the different x-scales in the figures.

To analyze the eigenvalues in vacuum for the semi-discrete case, we write the solver on matrix form as

$$\frac{\partial}{\partial t}\begin{pmatrix} \boldsymbol{H}\cdot\boldsymbol{n}^p \\ \boldsymbol{E}\cdot\boldsymbol{n}^d \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathcal{A} \\ \mathcal{B} & \mathbf{0} \end{pmatrix}\begin{pmatrix} \boldsymbol{H}\cdot\boldsymbol{n}^p \\ \boldsymbol{E}\cdot\boldsymbol{n}^d \end{pmatrix}, \tag{4.11}$$

where the operators $\mathcal{A}$ and $\mathcal{B}$ take care of the space discretization of the respective fields. Consider a grid consisting of $4 \times 4 \times 4$ cubes, where each cube is split into five tetrahedrons in the same way as the cells in the transition layer, see Figure 5.3. Calculating the eigenvalues numerically to the block-matrix above reveals that they are all on the imaginary axis. In this case, the primary and dual grids are mutually orthogonal. Let us perturb the grid such that the orthogonality is lost by moving the midpoint along one of the Cartesian directions. The eigenvalues, scaled with the CFL number, are shown in the right part of Figure 4.2. Hence, the loss of orthogonality locally is enough to get eigenvalues with a nonzero real part.

For the fully-discrete case, let

$$\boldsymbol{z}^n = \left( \boldsymbol{E} \cdot \boldsymbol{n}^d |^n \quad \boldsymbol{H} \cdot \boldsymbol{n}^p |^{n-\frac{1}{2}} \quad \boldsymbol{E} \cdot \boldsymbol{n}^d |^{n-1} \quad \boldsymbol{H} \cdot \boldsymbol{n}^p |^{n-\frac{3}{2}} \quad \boldsymbol{E} \cdot \boldsymbol{n}^d |^{n-2} \right)^T . \quad (4.12)$$

After some straightforward rearrangements we are able to write (4.9) and (4.10) on matrix form as $\boldsymbol{z}^{n+1} = \mathcal{P}(\mathcal{A}, \mathcal{B})\boldsymbol{z}^n$, where

$$\mathcal{P} = \begin{pmatrix} \mathcal{I} + \frac{625}{576}\Delta t^2 \mathcal{B}\mathcal{A} & \frac{23}{24}\Delta t \mathcal{B} & -\frac{25}{288}\Delta t^2 \mathcal{B}\mathcal{A} & \frac{1}{24}\Delta t \mathcal{B} & \frac{25}{576}\Delta t^2 \mathcal{B}\mathcal{A} \\ \frac{25}{24}\Delta t \mathcal{A} & \mathcal{I} & -\frac{1}{12}\Delta t \mathcal{A} & \mathbf{0} & \frac{1}{24}\Delta t \mathcal{A} \\ \mathcal{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathcal{I} & \mathbf{0} & \mathbf{0} \end{pmatrix} . \quad (4.13)$$

Analyzing the eigenvalues of the companion matrix $\mathcal{P}$, for the perturbed $4 \times 4 \times 4$ grid, results in the eigenvalue spectrum shown in Figure 4.3, where all eigenvalues are within the unit circle. The time step is chosen on the stability limit for the FV solver and is given by the largest eigenvalue of the semi-discrete problem. The eigenvalues to the left in Figure 4.3 are the first to move outside the unit circle if the time step is increased. For a general unstructured grid an eigenvalue analysis is not possible and hence the CFL condition is not known. A rule of thumb, which works well in practice, is to decrease the time step given by the stability condition for stand-alone FD-TD with a factor given by the quotient between the shortest primary edge and the shortest edge in the structured grid. Note that this type of stability condition is very similar to the one derived in 2D, see (3.36). The only difference is that here $\Delta_{min}$ is the shortest among the primary edges, whereas in 2D it is the shortest among all unstructured edges. Building the dual grid using the circumcenters of the tetrahedrons often creates some very short dual edges. However, in experiments we have seen that they do not affect the CFL condition.

If ABS3 is replaced by the leap-frog scheme and the same time step is used, the largest eigenvalue is of the order 1.004, despite the fact that this time step is well within the stability limit for leap-frog. Hence, not surprisingly the leap-frog scheme is unstable due to the fact that we have eigenvalues with a nonzero real part.

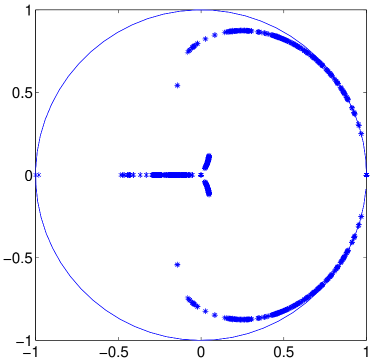


Figure 4.3: Eigenvalues of the companion matrix $\mathcal{P}$ using ABS3 for the perturbed $4 \times 4 \times 4$ grid. The time step is chosen on the stability limit.

However, using ABS3 does not guarantee long-term stability on general tetrahedral grids in 3D. Eigenvalues to the semi-discrete problem might have a real part that is too large for ABS3 to handle. One way of enhancing the numerical stability is to suppress the amplitude of the highest frequency components. A low-pass filter is ideal for this purpose. The undesirable highest frequency components are eliminated, whereas the rest of the solution remains unchanged. We use a spatial filter based on the Laplace operator. The filter is only applied to the node values and therefore it does not act on edges, which are orthogonal to their corresponding faces. For edges which are not orthogonal the larger the deviation from orthogonality the more the filter affects the edge values. A filtered value $\hat{\boldsymbol{H}}_r$ of $\boldsymbol{H}_r$ is given by

$$\hat{\boldsymbol{H}}_r = \boldsymbol{H}_r + \frac{1}{2N} \sum_{q=1}^{N} \left( \boldsymbol{H}_q - \boldsymbol{H}_r \right) , \tag{4.14}$$

where the sum is taken over the nodes $q$, which share an edge with node $r$. The filtering for the electric node values is analogous. Note that this filter is not needed in 2D, where the use of ABS3 is sufficient to obtain long term stability. However, the grids in 3D are in general much more complex for realistic applications and the deviation from orthogonality could be large, which we believe is the main source of the instability.

To analyze the filter we use the same technique as in Section 3.5. For simplicity we start with a uniform Cartesian grid in 2D with edge length $\Delta$. We make the following ansatz for the field components at the nodes:

$$\Psi|_{I,J}^{n} = \Psi_0 \mathrm{e}^{j\left(\tilde{k}_x I\Delta + \tilde{k}_y J\Delta\right)} , \tag{4.15}$$

Hence, a filtered value $\hat{\Psi}|_{I,J}^n$ is obtained through

$$\begin{aligned}\hat{\Psi}|_{I,J}^n &= \left\{1 + \frac{1}{8}\left[e^{j\tilde{k}_x\Delta} + e^{-j\tilde{k}_x\Delta} + e^{j\tilde{k}_y\Delta} + e^{-j\tilde{k}_y\Delta} - 4\right]\right\}\Psi|_{I,J}^n \\ &= \left\{1 - \frac{1}{2}\left[\sin^2\left(\frac{\tilde{k}_x\Delta}{2}\right) + \sin^2\left(\frac{\tilde{k}_y\Delta}{2}\right)\right]\right\}\Psi|_{I,J}^n .\end{aligned} \tag{4.16}$$

Thus, the amplification factor, $g(\tilde{k}_x\Delta, \tilde{k}_y\Delta)$, is given by

$$g(\tilde{k}_x\Delta, \tilde{k}_y\Delta) = 1 - \frac{1}{2}\left[\sin^2\left(\frac{\tilde{k}_x\Delta}{2}\right) + \sin^2\left(\frac{\tilde{k}_y\Delta}{2}\right)\right] . \tag{4.17}$$

For a uniform grid in 1D the amplification factor equals

$$g(\tilde{k}_x\Delta) = 1 - \sin^2\left(\frac{\tilde{k}_x\Delta}{2}\right) . \tag{4.18}$$

Note that on a uniform Cartesian grid the filter equals the discrete Laplace operator obtained using standard centered differences. In Figure 4.4 the amplification factor in 1D and a contour plot of the amplification factor in 2D are illustrated.

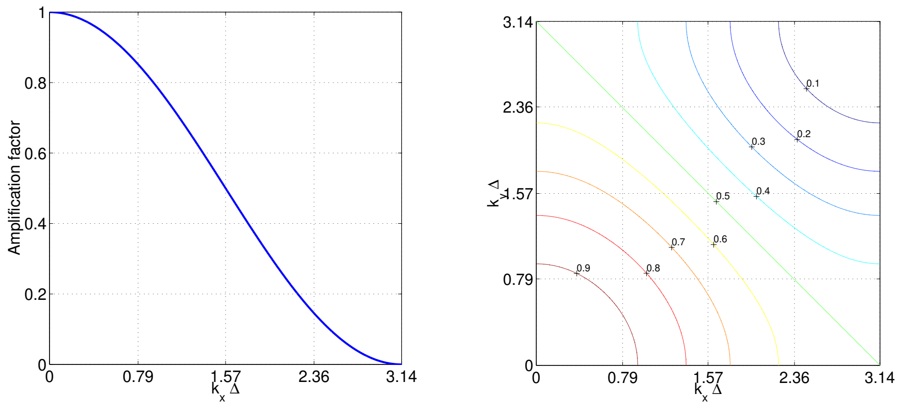


Figure 4.4: The amplification factor for the spatial filter as a function of the normalized wavenumber in 1D (left). Contour surfaces of the amplification factor on a structured grid in 2D (right).

Performing the same analysis on a uniform grid of equilateral triangles, see

Figure 3.5, results in

$$g(\tilde{k}_x\Delta, \tilde{k}_y\Delta) = 1 - \frac{1}{3}\left[\sin^2\left(\frac{\tilde{k}_x\Delta}{2}\right) + \sin^2\left(-\frac{\tilde{k}_x\Delta}{4} + \frac{\sqrt{3}\,\tilde{k}_y\Delta}{4}\right) + \sin^2\left(\frac{\tilde{k}_x\Delta}{4} + \frac{\sqrt{3}\,\tilde{k}_y\Delta}{4}\right)\right], \tag{4.19}$$

for the field in the primary nodes and

$$g(\tilde{k}_x\Delta, \tilde{k}_y\Delta) = \left|1 + \frac{1}{6}\left[\mathrm{e}^{j\left(-\frac{\tilde{k}_x\Delta}{2} + \frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right)} - 3 + \mathrm{e}^{j\left(\frac{\tilde{k}_x\Delta}{2} + \frac{\tilde{k}_y\Delta}{2\sqrt{3}}\right)} + \mathrm{e}^{j\left(\frac{\tilde{k}_y\Delta}{\sqrt{3}}\right)}\right]\right|, \tag{4.20}$$

for the field in the dual nodes. The contour plots of the amplification factors are shown in Figure 4.5.

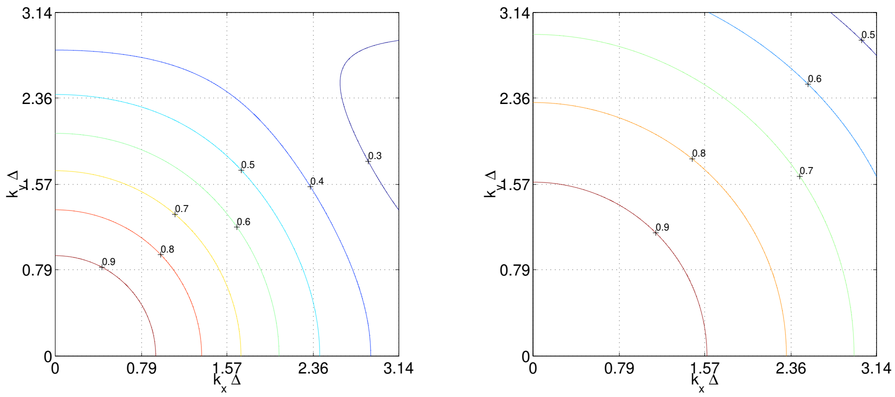


Figure 4.5: The amplification factor for the field in the primary nodes (left) and in the dual nodes (right) for a grid consisting of equilateral triangles.

The Figures 4.4 and 4.5 display that the filter suppresses the amplitude of the highest frequencies such that they are almost eliminated from the solution. It may seem that the damping of the low frequencies is also significant, but here we once again point out that the filter only acts on node values and what is important is the field projected along the edges. For most edges on a general grid the node values have a small contribution to the field along the edges. Analyzing the filter on uniform grids in 2D yields some insight in how the filter works, but what we are really interested in is general grids in 3D. However, on a general tetrahedral grid it is impossible to perform a rigorous analysis of the filter. In practice this filter has improved the stability considerably without loosing accuracy in the solution, see Chapter 7.

## 4.6 Workload and memory requirements

The implementation in 3D follows the same strategy as in 2D, i.e. as much as possible is computed initially and the update of the field variables is performed by matrix-vector multiplications. The matrices are stored in compress sparse row format, see Section 3.7.

To obtain a theoretical estimate of the memory requirements and number of arithmetic operations required by the FV solver we have to make a few assumptions. First of all we neglect boundary conditions. Note that in a real problem only a few layers of unstructured cells are used to get a better geometrical representation of the object and hence to neglect boundaries should be considered a major simplification. We put $\sigma$ equal to zero, $\epsilon$ and $\mu$ may vary in space but are not dependent on frequency. The following is assumed about the grid (based on the grids used in Section 8.2):

- There are $n$ tetrahedrons.
- There are four triangles in each tetrahedron and each triangle is shared by two tetrahedrons $\Rightarrow 2\,n$ triangles.
- There are six edges in each tetrahedron and each edge belongs to five tetrahedrons $\Rightarrow \frac{6\,n}{5}$ edges and five dual edges per dual face.
- There are four nodes in each tetrahedron and each node is shared by 24 tetrahedrons $\Rightarrow \frac{n}{6}$ nodes.
- Each node belongs to fourteen edges $\Rightarrow$ fourteen faces per dual cell.
- Half of the primary and dual edges are orthogonal to their respective faces.

The variables that have to be stored are $\boldsymbol{H}$, $\boldsymbol{E}$, $\boldsymbol{H} \cdot \boldsymbol{n}^p$, $\boldsymbol{E} \cdot \boldsymbol{n}^d$, $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^d$ and $\tilde{\boldsymbol{E}} \cdot \boldsymbol{t}^p$, where the last two are stored on three levels and the other four on one level.

The numbers in the Tables 4.1 and 4.2 are taken from (4.3)–(4.6) and (4.14), as well as from the grid assumptions and the way in which the node values are calculated. Using the results in Tables 4.1 and 4.2 we are able to sum up the total memory requirements for the FV solver, which is approximately 82.6 floating point numbers and 79.7 integers per tetrahedron. Using 64-bit precision for the floating point numbers and 32-bit precision for the integers this means that approximately 980 bytes per cell are needed. It is interesting to note that the cost in memory we have to pay for using ABS3 instead of leap-frog is only

| Variable | Floating point numbers |
|---|---|
| $\boldsymbol{H}$ | $3 \cdot n$ |
| $\boldsymbol{E}$ | $3 \cdot \frac{n}{6}$ |
| $\boldsymbol{H} \cdot \boldsymbol{n}^p$ | $2\,n$ |
| $\boldsymbol{E} \cdot \boldsymbol{n}^d$ | $\frac{6\,n}{5}$ |
| $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^d$ | $3 \cdot 2\,n$ |
| $\tilde{\boldsymbol{E}} \cdot \boldsymbol{t}^p$ | $3 \cdot \frac{6\,n}{5}$ |
| $\sum$ | $16.3\,n$ |

Table 4.1: Memory requirements to store the field variables.

| Variable | Floating point numbers | Integers |
|---|---|---|
| $\boldsymbol{H}$ | $3\,n \cdot 4 = 12\,n$ | $12\,n + 3\,n = 15\,n$ |
| $\boldsymbol{E}$ | $\frac{n}{2} \cdot 14 = 7\,n$ | $7\,n + \frac{n}{2} = \frac{15\,n}{2}$ |
| $\boldsymbol{H} \cdot \boldsymbol{n}^p$ | $2\,n \cdot 3 = 6\,n$ | $6\,n + 2\,n = 8\,n$ |
| $\boldsymbol{E} \cdot \boldsymbol{n}^d$ | $\frac{6\,n}{5} \cdot 5 = 6\,n$ | $6\,n + \frac{6\,n}{5} = \frac{36\,n}{5}$ |
| $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^d$ | $2\,n \cdot 4 = 8\,n$ | $8\,n + 2\,n = 10\,n$ |
| $\tilde{\boldsymbol{E}} \cdot \boldsymbol{t}^p$ | $\frac{6\,n}{5} \cdot 4 = \frac{24\,n}{5}$ | $\frac{24\,n}{5} + \frac{6\,n}{5} = 6\,n$ |
| $\hat{\boldsymbol{H}}$ | $3\,n \cdot 5 = 15\,n$ | $15\,n + 3\,n = 18\,n$ |
| $\hat{\boldsymbol{E}}$ | $\frac{n}{2} \cdot 15 = \frac{15\,n}{2}$ | $\frac{15\,n}{2} + \frac{n}{2} = 8\,n$ |
| $\sum$ | $66.3\,n$ | $79.7\,n$ |

Table 4.2: Memory requirements to store the matrices.

about 50 bytes per cell, i.e. only about five percent of the total cost. The filter, however, is responsible for a significant 30 percent of the total memory. Compare with FD-TD, whose memory requirements are six floating point numbers and six integers per cell, which results in 72 bytes per cell. Thus, the FV solver needs around a factor of fourteen more memory per cell than FD-TD.

| Variable | Arithmetic operations |
|---|---|
| $\boldsymbol{H}$ | $3\,n \cdot 7 = 21\,n$ |
| $\boldsymbol{E}$ | $\frac{n}{2} \cdot 27 = \frac{27\,n}{2}$ |
| $\boldsymbol{H} \cdot \boldsymbol{n}^p$ | $2\,n \cdot 6 = 12\,n$ |
| $\boldsymbol{E} \cdot \boldsymbol{n}^d$ | $\frac{6\,n}{5} \cdot 10 = 12\,n$ |
| $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^d$ | $2\,n \cdot 12 = 24\,n$ |
| $\tilde{\boldsymbol{E}} \cdot \boldsymbol{t}^p$ | $\frac{6\,n}{5} \cdot 12 = \frac{72\,n}{5}$ |
| $\hat{\boldsymbol{H}}$ | $3\,n \cdot 10 = 30\,n$ |
| $\hat{\boldsymbol{E}}$ | $\frac{n}{2} \cdot 30 = 15\,n$ |
| $\sum$ | $141.9\,n$ |

Table 4.3: Number of arithmetic operations per cell.

The number of arithmetic operations per cell used by the FV solver to perform one time step is given in Table 4.3. In addition to the equations used to obtain the memory requirements we also use (4.9)–(4.10). Due to the fact that the update of the field variables are matrix-vector multiplications the number of additions and multiplications is either equal or there is one more multiplication.

Hence, from Table 4.3 we conclude that the FV solver performs approximately 142 arithmetic operations per cell for each time step. Also in terms of arithmetic operations using ABS3 introduces only a small over-head of about ten percent, whereas the filter adds about 30 percent extra arithmetic operations. FD-TD makes 36 arithmetic operations per cell. Thus, FD-TD is about a factor of four cheaper per cell compared to the FV solver.

# Chapter 5

# Hybridization

It is very important that an electromagnetic solver for realistic industrial applications can handle complex geometries without compromising the efficiency more than necessary. We believe that the best way to achieve this is to combine structured grids with unstructured grids thereby getting a hybrid method. Our strategy of using hybrid grids is based on the following observations:

- The FD-TD method is an almost optimal second-order method for homogeneous materials, with respect to time and memory consumption.
- The Cartesian grid handles general geometries poorly, and fine details cannot be taken into account without special tricks.
- An unstructured grid can be fitted to general geometries and small details can be resolved.
- Unstructured grids lead to more elaborate memory accessing, and the number of operations per cell or wave length is higher than in the classical FD-TD method, see Sections 3.7 and 4.6.

Coupling of structured grids and unstructured grids strives to utilize their advantages without suffering too much from their disadvantages. The coupling is described in greater detail in [12].

## 5.1 Coupling in 2D

The unstructured grids have a semi-structured transition layer of split rectangles as outermost cells. This gives a total collocation of hybrid components and therefore interpolation between the grids is fully avoided. Coupling between the FD-TD solver for structured grids and the FV solver for unstructured grids is shown in Figures 5.1 and 5.2.

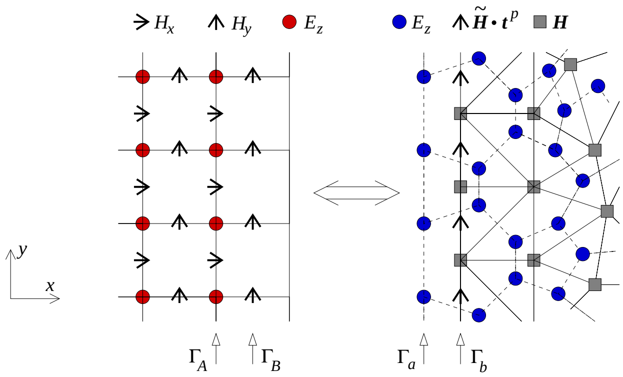


Figure 5.1: Coupling between structured FD-TD and unstructured FV. The variables along $\Gamma_A$-$\Gamma_a$ and $\Gamma_B$-$\Gamma_b$ are shared by the two solvers.

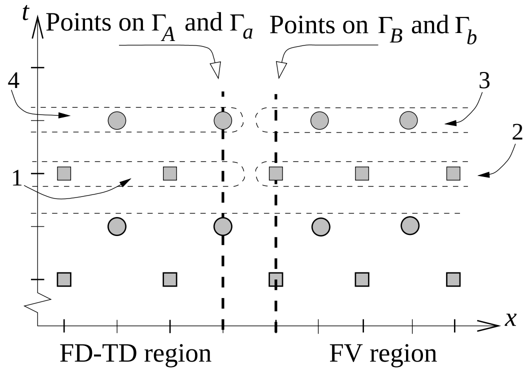


Figure 5.2: The timestepping mechanism for the FD-FV hybrid.

The timestepping algorithm for the hybrid solver is as follows:

1. The $H_x$- and $H_y$-components in the structured grid (to the left in Figure 5.1) are updated using the standard FD-TD method.
2. The $\boldsymbol{H}$- and $\boldsymbol{H} \cdot \boldsymbol{n}^d$-components in the unstructured grid are updated by the FV solver using the $E_z$ components along $\Gamma_a$ as boundary condition. The $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p$ components along $\Gamma_b$ are sent to $\Gamma_B$ in the structured grid.
3. The FV solver updates the $E_z$ components in the unstructured domain.
4. Finally, the time step is completed when FD-TD updates the $E_z$ components in the structured domain using the $\tilde{\boldsymbol{H}} \cdot \boldsymbol{t}^p$ components along $\Gamma_B$ as boundary condition. The $E_z$ components along $\Gamma_A$ are sent to $\Gamma_a$ to be used in the next time step by the FV solver.

## 5.2 Coupling in 3D

The hybridization technique presented here requires the unstructured grid to have a transition layer consisting of semi-structured cells. This means that the outermost unstructured cells are obtained by splitting a brick into five tetrahedrons (the brick is of the same size as the FD-TD cell). This outermost layer of cells is called the transition layer since the unstructured grid and the structured grid overlap here, see Figure 5.3. Hence, a number of components coincide in space and time for the solvers in the transition layer and it is these components that are involved in the actual communication between the solvers.

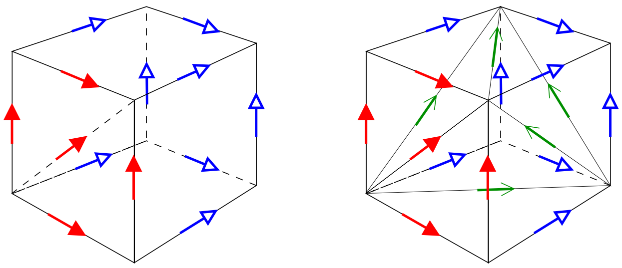

Figure 5.3: The transition layer is based on coincidental structured and unstructured cells. To the left an FD-TD cell and to the right a cluster of five tetrahedrons forming a brick is shown.

Based on Figure 5.3 the hybridization technique can be described as follows:

| `Left part of Figure 5.3,`<br>`the structured cell` | | `Right part of Figure 5.3,`<br>`the unstructured cells` |
|---|---|---|
| The four Cartesian components to the left (filled) are sent to the corresponding components in the unstructured region. The diagonal component (also filled) is calculated by interpolation of the four filled components and is also sent to the unstructured region. | $\Longrightarrow$ | The five filled components are received from the structured region and are treated by the unstructured solver as an inhomogeneous Dirichlet boundary condition. |
| $\Downarrow$ | | $\Downarrow$ |
| *(A full time step is taken)* | | *(A full time step is taken)* |
| $\Downarrow$ | | $\Downarrow$ |
| The remaining eight Cartesian components (hollow) are received from the corresponding components in the unstructured region. | $\Longleftarrow$ | When the components are updated in the unstructured region the remaining eight Cartesian components (hollow) are sent to the structured solver. |

The remaining five diagonal components (ordinary arrows) in the right part of Figure 5.3 are not involved in the hybridization. Note that sending the eight Cartesian components from the FV solver is equivalent in space to sending the four tangential $\boldsymbol{H}$ fields in the transition layer to FD-TD.

It is worthwhile to point out that the primary and dual grids are mutually orthogonal in the transition layer. Without this orthogonality also node values would have to be sent from the structured region, to update the electric edge values in the transition layer according to (4.6). In that case a careful interpolation would have to be performed since there are no nodal values in the FD-TD solver. That could possibly jeopardize the accuracy and stability of the hybrid solver.

The transition layer is one of the keys to the success of the hybrid scheme since it guarantees a good and controlled communication between the structured and unstructured solvers. One could think of methods without a transition layer but then the components used for communication are not (in general) collocated and hence interpolation is needed which will deteriorate the hybridization. The one cell overlap also makes it possible to communicate only $\boldsymbol{E}$ fields between the solvers and hence fit better into a multi block concept since the FD-TD code is only affected once per time step.

# Chapter 6

# Results in 2D

As discussed in Chapter 3 all materials are defined relative to primary grid cells and the magnetic permeability, $\mu$, which is associated with the spatially staggered dual cells requires averaging. To verify that averaging of $\mu$ according to (3.5) and (3.10) preserves second-order accuracy we perform a convergence analysis for a circular cylinder with relative permeability $\mu_r = 4$. The same analysis for a dielectric cylinder with relative permittivity $\epsilon_r = 4$ is also presented. Note that more 2D results are found in Appendix A.

## 6.1 Scattering from material cylinders

The incoming wave is Gaussian shaped and given by

$$f(t) = e^{-(t-t_0)^2/t_w^2}, \qquad t_0 = \frac{20}{c_0}, \quad t_w = \frac{t_0}{6}. \tag{6.1}$$

A layer of twelve U-PML cells is used as outer boundary condition. Figure 6.1 displays the coarsest primary- and dual-grid for the circular cylinder with a radius of two meters. This $10 \times 10$ meter grid is inserted in the center of a structured FD-TD grid with $40 \times 40$ cells of the size $\Delta = 1$ meter. According to (3.36) the stability limit for the FV solver is dictated by the shortest edge in the unstructured region, which in this case is a dual edge of length 0.235 meter located inside the cylinder. By choosing the time step such that CFL$= 1/\sqrt{32}$, where CFL equal to one is the stability limit for FD-TD, we are well within this limit.

The error has been measured in the $E_z$ components in the FD-TD grid that lie closest to the unstructured grid. By measuring the error in these points we avoid interpolation if we refine the grids with a factor of two. The error is defined as the difference between the computational solution and a reference solution. The reference solutions have been obtained by a highly resolved FD-TD calculation where we used $N_x = N_y = 5120$.

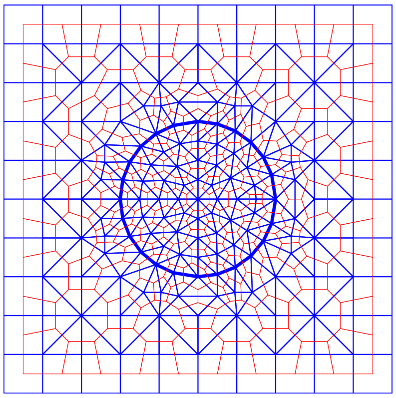

Figure 6.1: The coarsest unstructured grid.

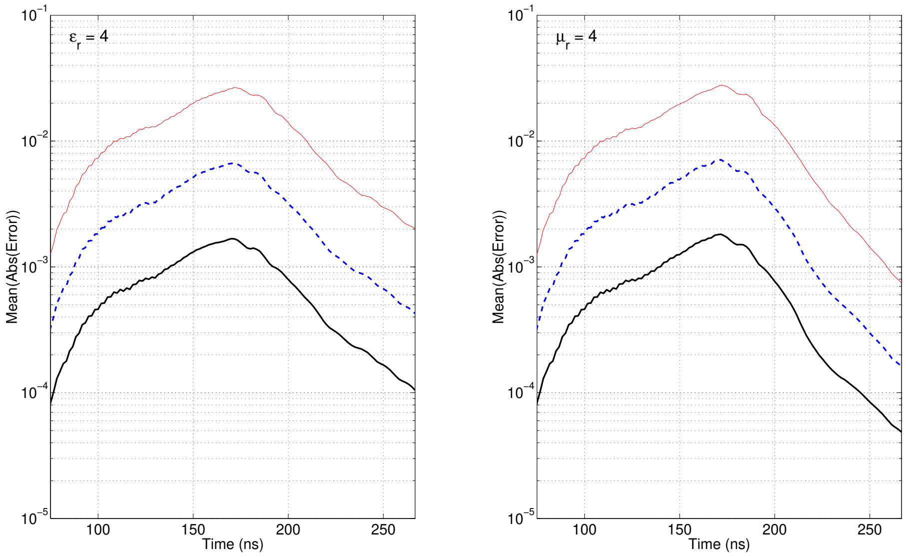


Figure 6.2: Errors for the three refinement levels.

The plots in Figure 6.2 display the mean value of the absolute error in the sample points as a function of time. The order of accuarcy has been estimated by calculating the $L_1$ norm of the errors in Figure 6.2. The dielectric cylinder gives order of accuracy 2.03 and 2.00, whereas the diamagnetic cylinder gives 2.04 and 1.97. Hence, the hybrid solver shows second-order accuracy in both cases.

# Chapter 7

# Results in 3D

The 3D results include three different scattering cases, a PEC sphere, a dielectric sphere and the generic aircraft *RUND*. Note that more 3D results are found in Appendix B. In the first section a convergence study is performed to determine the order of accuracy for the stand-alone FV solver. The hybrid solver is also validated in two different scattering cases, a PEC sphere and the generic aircraft *RUND*. For the PEC sphere analytical solutions, Mie series, for the RCS are known in frequency domain. For the generic aircraft a converged Method of Moments solutions is used as a reference solution. Since the estimation of RCS requires the scattered field intensity at infinity, the solution at the far-field is obtained using a near-to-far-field transformation [18]. The computational domains are truncated by eight layers of U-PML cells [8]. The CFL number is put to 0.2 in all cases, where CFL equal to one is the stability limit for FD-TD. The hybrid grids used in this section were generated by Jonas Gustafsson at Ericsson Saab Avionics using the commercial software *CADFIX*. The grids used for stand-alone FD-TD are generated such that all edges, which belong to a cell whose centroid is inside the object are considered to be PEC edges. The center of the sphere coincides with the center of an FD-TD cell.

## 7.1 Convergence study

To determine the order of accuracy for the FV solver we perform a grid refinement study. Since the FV solver is not a stand-alone solver this is accomplished by setting the FD-TD values analytically prior to communication with the FV solver. The analytical solution is given by

$$E_z(x,t) = e^{-(t-x/c-t_0)^2/t_w^2} ,$$

where $c$ is the speed of light, $t_0 = 40/c$ and $t_w = t_0/6$. Hence, the analytical solution is a plane wave travelling in the positive x-direction. Four of the $E_z$

components, which are sent from the FV solver to FD-TD are sampled. The space location of all four components are at the far end of the domain, i.e. the wave has passed through almost the whole domain. The grids are generated by splitting cubes in the same way as in the transition layer, see Figure 5.3. The coarsest grid has $6 \times 6 \times 6$ cubes and edge lengths $\Delta = 1m$. This grid is refined to obtain an intermediate and a fine grid consisting of $12 \times 12 \times 12$ and $24 \times 24 \times 24$ cubes, respectively.

To estimate the order of accuracy the $L_2$ norm of the errors in the right part of Figure 7.1 is calculated. Comparing the errors on the course and intermediate grids and on the intermediate and fine grids gives order of accuracy 2.4667 and 2.2307, respectively. Thus, the FV solver is second-order accurate on grids of this type. In Appendix B a convergence study for scattering from PEC spheres shows that the hybrid solver is second-order accurate in that case.

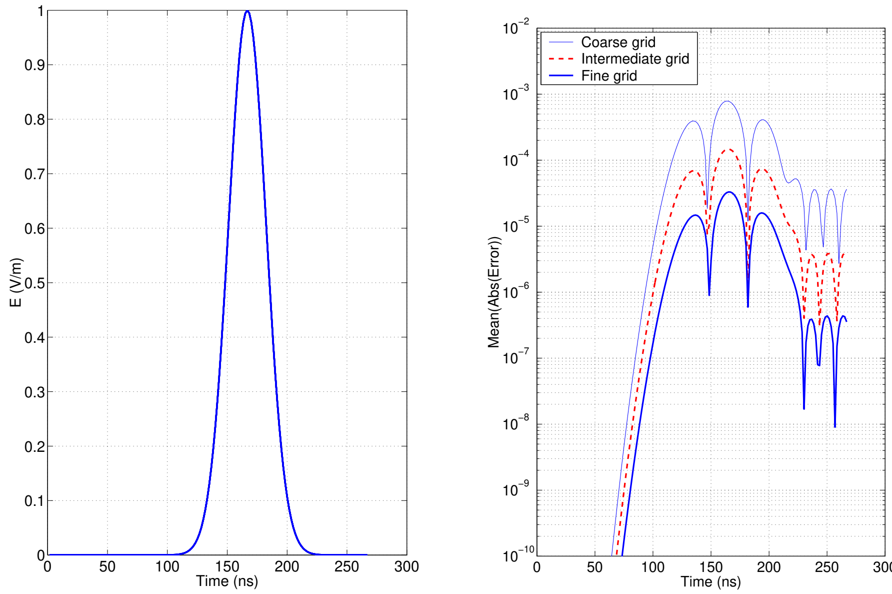


Figure 7.1: Electric field at one of the sample points as a function of time(left). Mean value of the errors at the four sample points as a function of time (right).

.

## 7.2 Scattering from a PEC sphere

A standard test case to validate electromagnetic codes in 3D is scattering from PEC spheres. In this case, the sphere has a radius of three meters and a Gaussian pulse with horizontal polarization impinges on the sphere at an angle of $\theta = 0$, using spherical polar coordinates. A few layers of unstructured cells are used in the near vicinity of the sphere to obtain a good geometrical representation of the sphere. A structured grid is used for the rest of the domain. The edges in the unstructured region are approximately of the same length as the ones in the structured region.

To investigate how much the nodal filter affects the accuracy of the solution we compare bistatic RCS from a PEC sphere obtained with and without the filter in Figure 7.2. The frequencies shown are 27 MHz and 37.5 MHz, which yields a resolution of eleven and eight points per wavelength with an edge length of one meter. The curves are almost on top of each other for both frequencies and the maximum difference is less than 0.05 dB. Thus, the far-field results are not affected by the filter and the accuracy is very good considering the very moderate resolution.

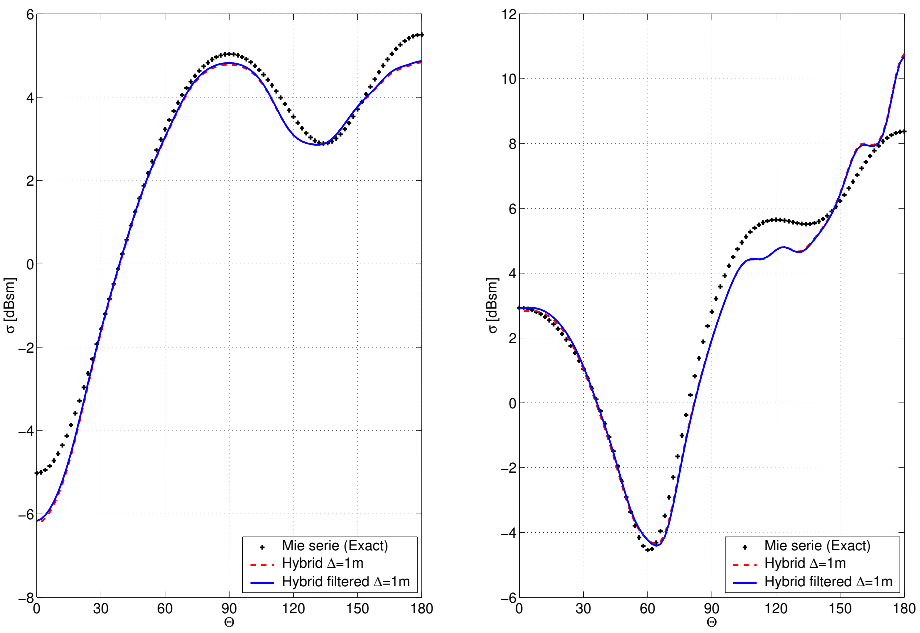


Figure 7.2: Bistatic RCS for a PEC sphere with and without the nodal filter at 27 MHz (left) and 37.5 MHz (right).

## 7.3 Scattering from the generic aircraft *RUND*

The RCS from canonical objects like spheres are often used for validation since analytical solutions are known. However, from an industrial perspective spheres are not targets of practical interest. A more realistic case is scattering from the generic aircraft *RUND*. This aircraft is approximately one meter long, one meter between the wing tips and half a meter high, see Figure 7.3.

The computational domain is discretized using roughly 150.000 tetrahedrons and 500.000 FD-TD cells. The FD-TD cells have one centimeter long edges, which corresponds to a resolution of 20 points per wavelength at 1.5 GHz. A Gaussian pulse illuminates the aircraft head on with horizontal polarization. Due to the fact that the unstructured cells are of the same size as the FD-TD cells, this case is very well suited for our explicit hybrid solver.

The results are presented in Figure 7.4. If we compare the results with the ones found in Appendix B for vertical polarization we observe that the dynamic range is much larger for the horizontal polarization. The wings have a much more significant contribution to the RCS for horizontal polarization. The horizontal polarization also generates surface currents that travel between the wings, see Figures 7.5 and 7.6 for snapshots of the surface currents with vertical and horizontal polarization, respectively. The snapshots are taken from two mpeg movies available at `www.tdb.uu.se/~fredrik/RUND/`.

The accuracy of the hybrid solution is very good except close to the monostatic angle. However, FD-TD is also far from the MoM solution in this region even for the highly resolved case with two millimeter long edges, corresponding to 100 points per wavelength at 1.5 GHz. It is not clear why the correspondence is poor in this region, it was certainly not the case for vertical polarization, see Appendix B. To get a better understanding of the problem we will try different frequencies and we would also like use a finer grid to get a better geometrical representation of the object. However, the latter is not possible without a parallel code.

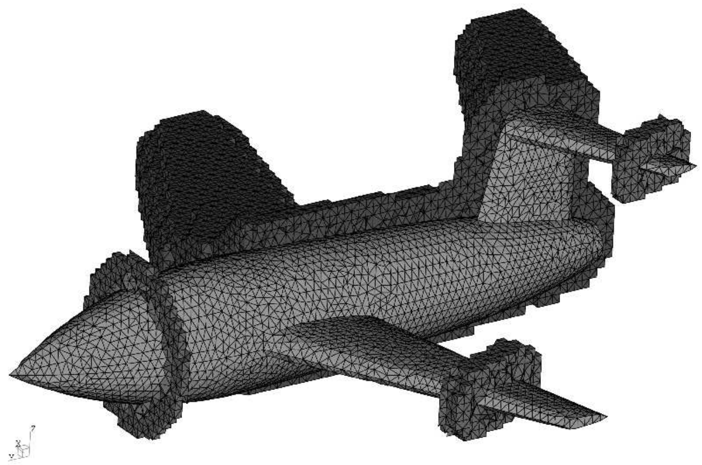

Figure 7.3: Part of unstructured mesh around generic aircraft *RUND*. The structured mesh continues outside the shown unstructured cells. Note that the interface between the grids is staircased to minimize the number of unstructured cells.

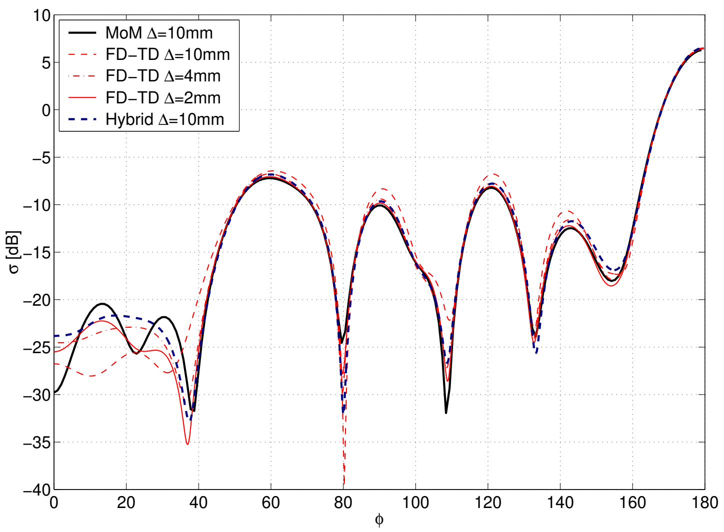


Figure 7.4: Comparison of bistatic RCS between the hybrid solver, FD-TD and a MoM solver at 1.5 GHz.

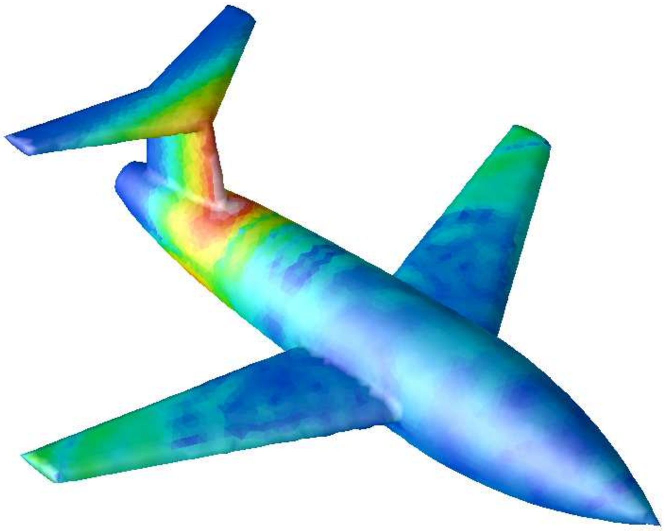

Figure 7.5: Surface currents on *RUND for vertical polarization when the main pulse hits the tail of the aircraft.*

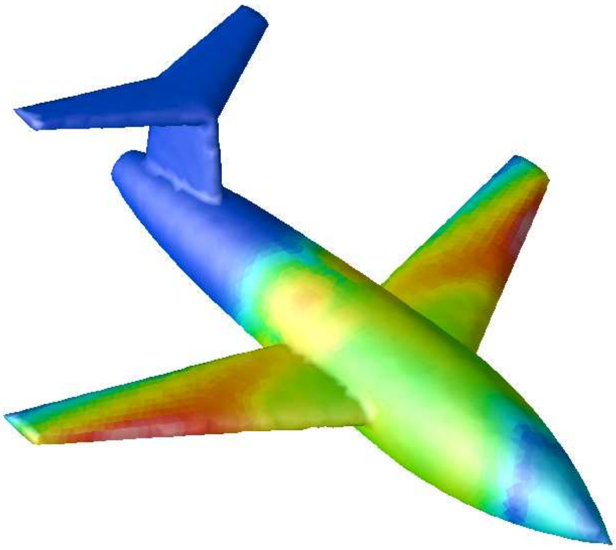

Figure 7.6: Surface currents on *RUND for horizontal polarization when the main pulse has reached the trailing edge of the wings.*

# Chapter 8

# Frequency dispersive materials

The ability to treat frequency dispersive materials is an important feature for a general purpose electromagnetic code. They are for instance used as radar absorbing materials to minimize the RCS for an aircraft. Other examples include waveguides filled with dispersive materials, antenna structures and microstrip systems based on dispersive substrates. One of the most common approaches to extend FD-TD to permit these materials is to use the recursive convolution (RC) method [15]. This method is computationally efficient and can be used for general linear materials. Another approach is to use an auxiliary differential equation (ADE) to relate the electric flux density vector to the electric field [10, 31]. The ADE method can also be used to treat nonlinear material in addition to linear materials at the cost of an increase in computational complexity.

The original RC formulation is only first order accurate. A modified approach was presented in [11], where the convolution integral is evaluated using a piecewise linear approximation of the electric field. It was shown by Schuster and Luebbers [21] that using a piecewise constant centered approximation of the electric field, second-order accuracy is obtained and the efficiency of the original method is retained. The technique could easily be extended to include also magnetically dispersive materials by using the duality of the Maxwell equations. We will here generalize the RC method presented in [21] to our unstructured FV solver. To enable better modeling of materials, which have a strong frequency dependence in their constitutive parameters a general model is introduced. The susceptibility function in frequency domain for materials described by the general model is in fractional form and consists of a sum of $G$ complex poles.

## 8.1 Recursive Convolution

For a linear isotropic material the polarization, $\boldsymbol{P}(\boldsymbol{r},\omega)$, is proportional to the electric field, $\boldsymbol{E}(\boldsymbol{r},\omega)$,

$$\boldsymbol{P}(\boldsymbol{r},\omega) = \epsilon_0\, \chi(\boldsymbol{r},\omega)\boldsymbol{E}(\boldsymbol{r},\omega)\,, \tag{8.1}$$

where $\epsilon_0$ is the permittivity in vacuum and $\chi(\boldsymbol{r},\omega)$ is the electric susceptibility function. The electric flux density vector, $\boldsymbol{D}(\boldsymbol{r},\omega)$, is related to the electric field as

$$\boldsymbol{D}(\boldsymbol{r},\omega) = \boldsymbol{P}(\boldsymbol{r},\omega) + \epsilon_0\,\epsilon_\infty\,\boldsymbol{E}(\boldsymbol{r},\omega) = \epsilon_0\left(\epsilon_\infty + \chi(\boldsymbol{r},\omega)\right)\boldsymbol{E}(\boldsymbol{r},\omega)\,, \tag{8.2}$$

where $\epsilon_\infty$ is the limiting permittivity at infinite frequency. Applying the inverse Fourier transform to (8.2) we obtain the corresponding time domain relation

$$\boldsymbol{D}(\boldsymbol{r},t) = \epsilon_0\,\epsilon_\infty \boldsymbol{E}(\boldsymbol{r},t) + \epsilon_0 \int_0^t \boldsymbol{E}(\boldsymbol{r},t-\tau)\,\chi(\tau)\;d\tau\,. \tag{8.3}$$

Discretizing (8.3) in time results in

$$\boldsymbol{D}^n = \epsilon_0\,\epsilon_\infty \boldsymbol{E}^n + \epsilon_0 \int_0^{n\Delta t} \boldsymbol{E}(n\Delta t-\tau)\,\chi(\tau)\;d\tau\,, \tag{8.4}$$

where $\boldsymbol{D}^n = \boldsymbol{D}(n\Delta t)$ and the explicit dependence on $\boldsymbol{r}$ has been omitted to simplify the notation. Following Luebbers et al. $\boldsymbol{E}^m$ is approximated as constant over the interval $[(m-1/2)\Delta t, (m+1/2)\Delta t]$. Hence,

$$\boldsymbol{D}^n = \epsilon_0\epsilon_\infty \boldsymbol{E}^n + \epsilon_0 \boldsymbol{E}^n \int_0^{\frac{\Delta t}{2}} \chi(\tau)\,d\tau + \epsilon_0 \sum_{m=0}^{n-1} \boldsymbol{E}^{n-m-1} \int_{(m+\frac{1}{2})\Delta t}^{(m+\frac{3}{2})\Delta t} \chi(\tau)\,d\tau, \tag{8.5}$$

where it is assumed that $\boldsymbol{E}^0 = 0$. Discretizing the area integral formulation of Ampère's law on dual face $j$ according to (4.10) yields,

$$\begin{aligned} &\frac{\boldsymbol{D}\cdot\boldsymbol{n}_j^d|^{n+1} - \boldsymbol{D}\cdot\boldsymbol{n}_j^d|^n}{\Delta t}\cdot A_j^d + \sigma\,\frac{\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n+1} + \boldsymbol{E}\cdot\boldsymbol{n}_j^d|^n}{2}\cdot A_j^d = \\ &\sum_k \left(\frac{25}{24}\,\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_{j,k}^d|^{n+\frac{1}{2}} - \frac{1}{12}\,\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_{j,k}^d|^{n-\frac{1}{2}} + \frac{1}{24}\,\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_{j,k}^d|^{n-\frac{3}{2}}\right)\Delta_{j,k}^d\,. \end{aligned} \tag{8.6}$$

To simplify the notation in the update equation define,

$$\chi^0 = \int_0^{\frac{\Delta t}{2}} \chi(\tau)d\tau\,, \tag{8.7}$$

$$\chi^{m+\frac{1}{2}} = \int_{(m+\frac{1}{2})\Delta t}^{(m+\frac{3}{2})\Delta t} \chi(\tau)d\tau\,, \tag{8.8}$$

$$\Delta\chi^{m+1} = \chi^{m+\frac{1}{2}} - \chi^{m+\frac{3}{2}}\,, \tag{8.9}$$

and

$$\mathcal{H}\left(\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_j^d|^n\right)=\sum_k\left(\frac{25}{24}\,\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_{j,k}^d|^{n+\frac{1}{2}}-\frac{1}{12}\,\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_{j,k}^d|^{n-\frac{1}{2}}+\frac{1}{24}\,\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_{j,k}^d|^{n-\frac{3}{2}}\right)\Delta_{j,k}^d\,. \tag{8.10}$$

Substituting (8.5) and (8.7)–(8.10) into (8.6) yields after some rearrangements

$$\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n+1}=\frac{1}{\epsilon_\infty+\chi^0+\frac{\sigma\Delta t}{2\epsilon_0}}\Bigg\{\left(\epsilon_\infty+\chi^0-\chi^{\frac{1}{2}}-\frac{\sigma\Delta t}{2\epsilon_0}\right)\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^n+\sum_{m=0}^{n-1}\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n-m-1}\Delta\chi^{m+1}+\frac{\Delta t}{A_j^d\epsilon_0}\,\mathcal{H}\left(\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_j^d|^n\right)\Bigg\}\,. \tag{8.11}$$

Assigning the summation over $m$ to the new variable $\Psi_j^d|^n$,

$$\Psi_j^d|^n=\sum_{m=0}^{n-1}\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n-m-1}\Delta\chi^{m+1}\,, \tag{8.12}$$

implies

$$\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n+1}=\frac{1}{\epsilon_\infty+\chi^0+\frac{\sigma\Delta t}{2\epsilon_0}}\Bigg\{\left(\epsilon_\infty+\chi^0-\chi^{\frac{1}{2}}-\frac{\sigma\Delta t}{2\epsilon_0}\right)\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^n+\Psi_j^d|^n+\frac{\Delta t}{A_j^d\epsilon_0}\,\mathcal{H}\left(\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_j^d|^n\right)\Bigg\}\,. \tag{8.13}$$

Luebbers et al. made the key observation that is is possible to update $\Psi_j^d|^n$ recursively as

$$\Psi_j^d|^n=\Delta\chi^1\boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n-1}+C_{rec}\Psi_j^d|^{n-1}\,, \tag{8.14}$$

where $C_{rec}$ is a constant, whose value depends on the material. Thus, the whole time history of $\boldsymbol{E}\cdot\boldsymbol{n}_j^d$ is not needed in order to update $\boldsymbol{E}\cdot\boldsymbol{n}_j^d$ on the new time level.

### 8.1.1 Debye material

First order dispersive materials are described by the Debye model, whose electric susceptibility function is given by

$$\chi(\omega) = \frac{\epsilon_s - \epsilon_\infty}{1 + j\omega t_0} . \tag{8.15}$$

Here, $\epsilon_s$ is the permittivity at zero frequency, $t_0$ is the relaxation time, $j = \sqrt{-1}$ and the $\exp(j\omega t)$ time convention is assumed. Applying the inverse Fourier-transform to (8.15) the time domain susceptibility function is

$$\chi(t) = \frac{\epsilon_s - \epsilon_\infty}{t_0} \mathrm{e}^{-\frac{t}{t_0}} u(t) , \tag{8.16}$$

where $u(t)$ is the unit step function. Inserting (8.16) into (8.7) and (8.8), and calculating the integrals imply

$$\chi^0 = (\epsilon_s - \epsilon_\infty) \left(1 - \mathrm{e}^{-\frac{\Delta t}{2t_0}}\right) , \tag{8.17}$$

$$\chi^{m+\frac{1}{2}} = (\epsilon_s - \epsilon_\infty) \left(1 - \mathrm{e}^{-\frac{\Delta t}{t_0}}\right) \mathrm{e}^{-(m+\frac{1}{2})\frac{\Delta t}{t_0}} . \tag{8.18}$$

Using (8.18) in (8.9), we find that $\Delta\chi^{m+1}$ for a Debye material equals

$$\begin{aligned}
\Delta\chi^{m+1} &= (\epsilon_s - \epsilon_\infty) \left(1 - \mathrm{e}^{-\frac{\Delta t}{t_0}}\right) \left(\mathrm{e}^{-(m+\frac{1}{2})\frac{\Delta t}{t_0}} - \mathrm{e}^{-(m+\frac{3}{2})\frac{\Delta t}{t_0}}\right) \\
&= (\epsilon_s - \epsilon_\infty) \left(\mathrm{e}^{-(m-\frac{1}{2})\frac{\Delta t}{t_0}} - \mathrm{e}^{-(m+\frac{1}{2})\frac{\Delta t}{t_0}}\right) \mathrm{e}^{-\frac{\Delta t}{t_0}} \\
&= \mathrm{e}^{-\frac{\Delta t}{t_0}} \Delta\chi^m .
\end{aligned} \tag{8.19}$$

We can now show that $\Delta\chi^{m+1}$ has the desired exponential time dependence to be able to update $\Psi_j^d|^n$ recursively according to (8.14),

$$\begin{aligned}
\Psi_j^d|^n &= \sum_{m=0}^{n-1} \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-m-1} \Delta\chi^{m+1} \\
&= \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-1} \Delta\chi^1 + \sum_{m=1}^{n-1} \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-m-1} \Delta\chi^{m+1} \\
&= \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-1} \Delta\chi^1 + \sum_{m=1}^{n-1} \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-m-1} \Delta\chi^m \mathrm{e}^{-\frac{\Delta t}{t_0}} \\
&= \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-1} \Delta\chi^1 + \sum_{m=0}^{n-2} \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-1-m-1} \Delta\chi^{m+1} \mathrm{e}^{-\frac{\Delta t}{t_0}} \\
&= \Delta\chi^1 \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-1} + \mathrm{e}^{-\frac{\Delta t}{t_0}} \Psi_j^d|^{n-1} .
\end{aligned} \tag{8.20}$$

Note that all the constants are real in the case of a Debye material.

### 8.1.2 Lorentz material

Lorentz materials are characterized by damped resonances and have susceptibility functions with one or more pairs of complex conjugate poles. The susceptibility function for a material described by a single pair pole is in frequency domain given by

$$\chi(\omega) = \frac{(\epsilon_s - \epsilon_\infty)\,\omega_0^2}{\omega_0^2 + j2\omega\delta - \omega^2}\,, \tag{8.21}$$

where $\epsilon_s$ and $\epsilon_\infty$ have the same meaning as for Debye materials, $\omega_0$ is the resonant frequency and $\delta$ is the damping coefficient. The time domain susceptibility function equals

$$\chi(t) = \frac{(\epsilon_s - \epsilon_\infty)\,\omega_0^2}{\sqrt{\omega_0^2 - \delta^2}} \mathrm{e}^{-\delta t} \sin\left(\sqrt{\omega_0^2 - \delta^2}\,t\right) u(t)\,. \tag{8.22}$$

This time domain susceptibility function does not have the exponential time dependence required by the recursive convolution approach. However, if we define

$$\hat{\chi}(t) = -j\gamma \mathrm{e}^{(-\alpha + j\beta)\,t} u(t)\,, \tag{8.23}$$

where $\chi(t) = \Re e\{\hat{\chi}(t)\}$ and

$$\alpha = \delta\,,\ \beta = \sqrt{\omega_0^2 - \delta^2}\ \text{ and }\ \gamma = \frac{(\epsilon_s - \epsilon_\infty)\,\omega_0^2}{\beta}\,, \tag{8.24}$$

the susceptibility function has the desired exponential time dependence. Inserting (8.23) into (8.7)–(8.9) implies

$$\hat{\chi}^0 = \frac{-j\gamma}{\alpha - j\beta}\left(1 - \mathrm{e}^{(-\alpha + j\beta)\frac{\Delta t}{2}}\right)\,, \tag{8.25}$$

$$\hat{\chi}^{m+\frac{1}{2}} = \frac{-j\gamma}{\alpha - j\beta}\left(1 - \mathrm{e}^{(-\alpha + j\beta)\Delta t}\right) \mathrm{e}^{(-\alpha + j\beta)\left(m + \frac{1}{2}\right)\Delta t}\,, \tag{8.26}$$

$$\Delta\hat{\chi}^{m+1} = \mathrm{e}^{(-\alpha + j\beta)\Delta t} \Delta\hat{\chi}^m\,. \tag{8.27}$$

Using (8.27) the complex recursive accumulator is derived in the same way as for Debye materials,

$$\hat{\mathbf{\Psi}}_j^d|^n = \Delta\hat{\chi}^1 \boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n-1} + \mathrm{e}^{(-\alpha + j\beta)\Delta t}\, \hat{\mathbf{\Psi}}_j^d|^{n-1}\,. \tag{8.28}$$

In the electric field update equation (8.13) the real part of the recursive accumulator and the other constants are inserted.

### 8.1.3 General material

In this section we introduce a general model to enable better modeling of materials, which have a strong frequency dependence in their constitutive parameters. Examples of such materials include among others biological and optical materials. A general material is described by multiple first- and second-order poles. The susceptibility function for a material characterized by $G$ poles is given by

$$\chi(\omega) = \sum_{g=1}^{G} \frac{R_g}{j\omega - s_g} , \tag{8.29}$$

where $R_g$ and $s_g$ are the residues and poles of the function, respectively. In time domain the susceptibility function equals

$$\hat{\chi}(t) = \sum_{g=1}^{G} R_g \, \mathrm{e}^{s_g t} \, u(t) . \tag{8.30}$$

Hence, we obtain a time domain susceptibility function for each term in the summation. Inserting the $g$th term in (8.30) into (8.7)–(8.9) yields

$$\hat{\chi}_g^0 = -\frac{R_g}{s_g} \left(1 - \mathrm{e}^{s_g \frac{\Delta t}{2}}\right) , \tag{8.31}$$

$$\hat{\chi}_g^{m+\frac{1}{2}} = -\frac{R_g}{s_g} \left(1 - \mathrm{e}^{s_g \Delta t}\right) \mathrm{e}^{\left(m+\frac{1}{2}\right) s_g \Delta t} , \tag{8.32}$$

$$\Delta\hat{\chi}_g^{m+1} = \mathrm{e}^{s_g \Delta t} \Delta\hat{\chi}_g^m . \tag{8.33}$$

The $g$th complex recursive function is derived in the same way as for Debye materials and is given by

$$\hat{\Psi}_j^d|_g^n = \Delta\hat{\chi}_g^1 \, \boldsymbol{E} \cdot \boldsymbol{n}_j^d|^{n-1} + \mathrm{e}^{s_g \Delta t} \, \hat{\Psi}_j^d|_g^{n-1} . \tag{8.34}$$

The sum of the real parts of (8.31)–(8.34) is finally inserted in the update equation for the electric field (8.13).

Note that it is possible to treat both Debye and Lorentz type materials by the general model. If

$$R_1 = \frac{\epsilon_s - \epsilon_\infty}{t_0} , \; s_1 = -\frac{1}{t_0} , \tag{8.35}$$

we get the Debye susceptibility function. For Lorentz materials the two poles and residues equal

$$R_1 = -j\frac{(\epsilon_s - \epsilon_\infty)\,\omega_0^2}{2\sqrt{\omega_0^2 - \delta^2}}\,, \tag{8.36}$$

$$R_2 = j\frac{(\epsilon_s - \epsilon_\infty)\,\omega_0^2}{2\sqrt{\omega_0^2 - \delta^2}}\,, \tag{8.37}$$

$$s_1 = -\delta + j\sqrt{\omega_0^2 - \delta^2}\,, \tag{8.38}$$

$$s_2 = -\delta - j\sqrt{\omega_0^2 - \delta^2}\,. \tag{8.39}$$

However, since $R_1$ is the complex conjugate of $R_2$ and $s_1$ is the complex conjugate of $s_2$ we have

$$\hat{\chi}(t) = R_1 \mathrm{e}^{s_1 t} + R_2 \mathrm{e}^{s_2 t} = R_1 \mathrm{e}^{s_1 t} + R_1^* \mathrm{e}^{s_1^* t} = 2\,\Re e\left\{R_1 \mathrm{e}^{s_1 t}\right\}, \tag{8.40}$$

which equals expression (8.22). Thus, it is enough to use one of the poles and no efficiency is lost by treating a Lorentz material in the same way as a general material. For Debye materials a factor of two is lost due to the fact that we get complex variables.

### 8.1.4 Material interfaces

Our grid generator supplies us with information about the material parameters in each tetrahedron and in each cube in the structured domain. But to update the electric field we need these parameters on the dual faces. Hence, some kind of averaging has to be performed. Note that averaging is only needed at material interfaces. To preserve second-order accuracy this is accomplished by splitting the area integral of $\boldsymbol{D}\cdot\boldsymbol{n}_j^d$ over $A_j^d$ into $q$ sub-integrals, where $q$ is the number of tetrahedrons sharing dual face $j$, cf. (4.4) and (4.8). Each sub-integral is taken over $A_{j,q}^d$, which is the area of dual face $j$ that is inside tetrahedron $q$. The contributions from each sub-integral sum up to the total integral. Let,

$$C_{j,q}^1 = \epsilon_\infty|_q^p + \chi^0|_q^p + \frac{\sigma_q^p \Delta t}{2\epsilon_0}\,, \tag{8.41}$$

$$C_{j,q}^2 = \epsilon_\infty|_q^p + \chi^0|_q^p - \chi^{\frac{1}{2}}|_q^p - \frac{\sigma_q^p \Delta t}{2\epsilon_0}\,, \tag{8.42}$$

where $\epsilon_\infty|_q^p$ denotes $\epsilon_\infty$ in tetrahedron $q$ and accordingly for the other constants. Using (8.41) and (8.42) the update equation (8.13) is at material interfaces given by

$$\begin{aligned}\sum_q C_{j,q}^1\, A_{j,q}^d\, \boldsymbol{E}\cdot\boldsymbol{n}_j^d|^{n+1} &= \sum_q C_{j,q}^2\, A_{j,q}^d\, \boldsymbol{E}\cdot\boldsymbol{n}_j^d|^n \\ &\quad + \sum_q \Psi_j^d|^n\, A_{j,q}^d + \frac{\Delta t}{\epsilon_0}\,\mathcal{H}\left(\tilde{\boldsymbol{H}}\cdot\boldsymbol{t}_j^d|^n\right)\,.\end{aligned} \tag{8.43}$$

## 8.2 Results

In this section we present scattering results for spheres described by the three different material models treated in the previous section.

### 8.2.1 Scattering from a Debye sphere

A typical Debye material is water. The parameters in the Debye model are set to

$$\begin{cases} \epsilon_\infty = 5.27\,, \\ \epsilon_s = 80.0\,, \\ t_0 = 1.0 \cdot 10^{-11}\,, \\ \sigma = 0\,. \end{cases} \tag{8.44}$$

In Figure 8.1 the real and imaginary parts of the relative permittivity as a function of frequency are shown.

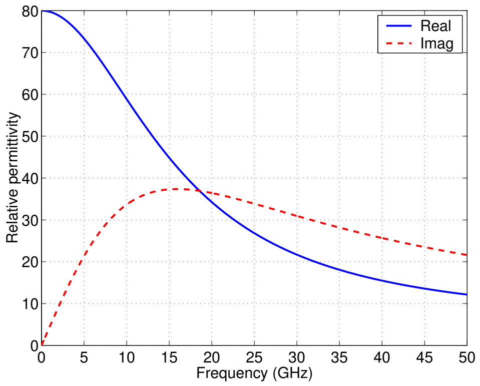


Figure 8.1: Relative permittivity for water as a function of frequency.

In order to validate the method we compute the scattering from a sphere of water with parameters given in (8.44) and vacuum outside. The radius of the sphere is 420 $\mu$m, and the edge lengths are 140 $\mu$m, 70 $\mu$m and 35 $\mu m$, respectively. This resolution corresponds to ten, twenty and forty points per wavelength in the sphere at 50 GHz. Results for monostatic and bistatic RCS are compared to the analytical Mie series solutions and to stand-alone FD-TD solutions in Figure 8.2.

The hybrid results agree very well with the analytical solutions with a moderate resolution of ten points per wavelength having in mind that the sphere is not particularly well resolved in that case. In the monostatic case with a resolution of twenty points per wavelength the numerical solution is hardly separable

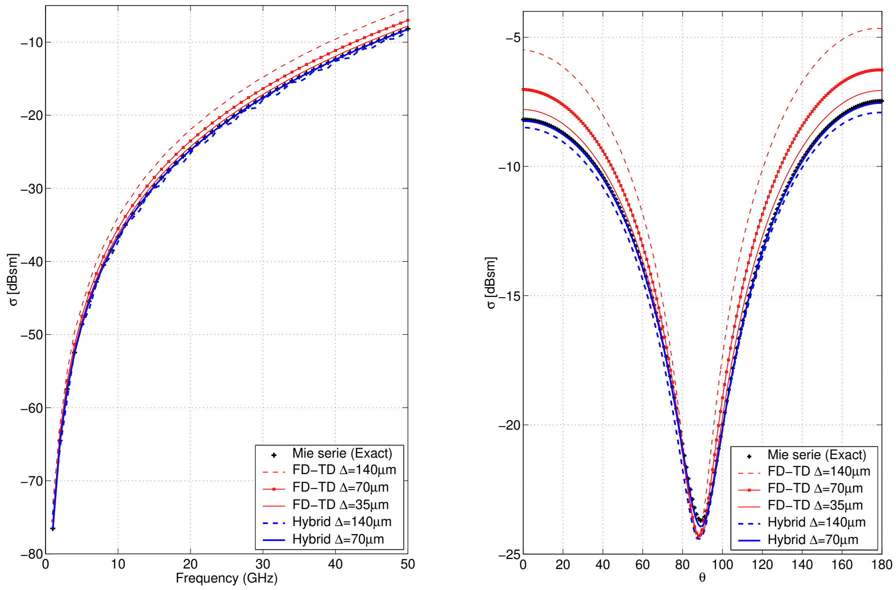


Figure 8.2: Backscattering from a water sphere (left) and bistatic RCS for a water sphere at 50 GHz (right).

from the analytical one. The coarsest hybrid solution shows approximately the same accuracy as the finest FD-TD solution.

### 8.2.2 Scattering from a Lorentz sphere

We perform the same numerical test here, i.e. scattering from a sphere. The sphere is in this case characterized by a single pair of complex conjugate poles. The parameters in (8.21) are given by

$$\begin{cases} \epsilon_\infty = 4.3\,, \\ \epsilon_s = 6.0\,, \\ \omega_0 = 40\,\pi \cdot 10^9\,, \\ \delta = 0.1\,\omega_0\,, \\ \sigma = 0\,. \end{cases} \tag{8.45}$$

The complex permittivity function is plotted in Figure 8.3, where the resonance is seen around 20 GHz.

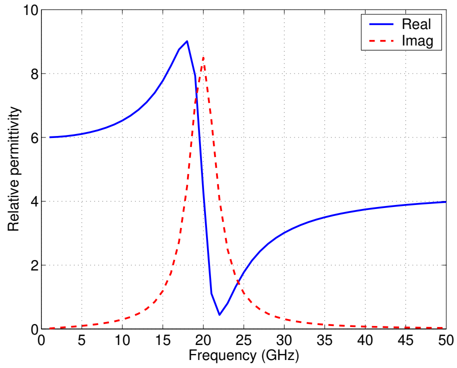


Figure 8.3: Relative permittivity for the second-order dispersive material as a function of frequency.

The radius of the sphere is 900 $\mu m$, and the edge lengths are 300 $\mu m$ and 150 $\mu m$. This resolution corresponds to ten and twenty points per wavelength in the sphere at 50 GHz. The results for monostatic RCS are presented in Figure 8.4. We get very good agreement with the analytical solution for the whole frequency

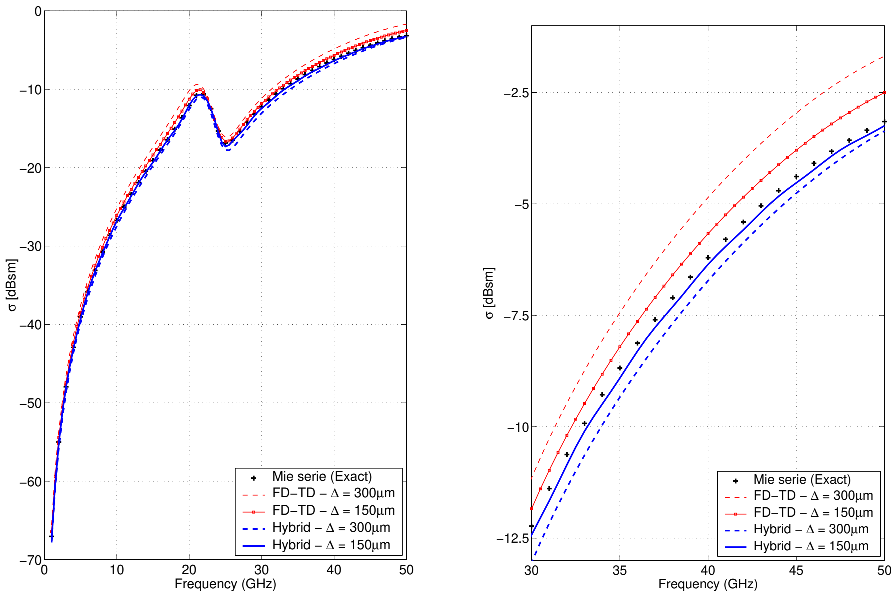


Figure 8.4: Backscattering from a sphere of Lorentz material.

spectrum. FD-TD produces surprisingly good results. In this case it is enough to decrease the edge length with a factor of two compared to the hybrid solver to produce results of similar accuracy. The errors in the hybrid solution are mainly due to the Huygens surfaces, used to generate the incoming plane wave, and to reflections at the hybrid interface.

### 8.2.3 Scattering from a General sphere

In order to demonstrate the accuracy we perform the same scattering case as before. The parameters characterizing the sphere are given by two second-order Lorentz poles, see Table 8.1.

| | |
|---|---|
| $\epsilon_s = 6.0$ | $\epsilon_\infty = 4.3$ |
| $\omega_1 = 40\,\pi \cdot 10^9$ | $\omega_2 = 100\,\pi \cdot 10^9$ |
| $\delta_1 = 0.1\,\omega_1$ | $\delta_2 = 0.1\,\omega_2$ |
| $G_1 = 0.4$ | $G_2 = 0.6$ |

Table 8.1: Parameters for the fourth-order dispersive material.

The complex permittivity function is illustrated in Figure 8.5. In this case we have two resonance frequencies one at 20 GHz and another one at 50 GHz.

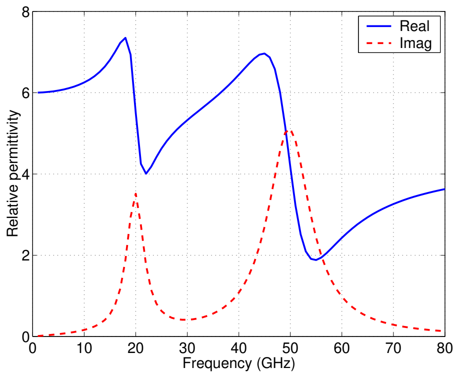


Figure 8.5: Relative permittivity for the fourth-order dispersive material as a function of frequency.

Using (8.36) and (8.38) the residues and poles are calculated from

$$\begin{cases} R_1 = -j\frac{G_1(\epsilon_s-\epsilon_\infty)\omega_1^2}{\sqrt{\omega_1^2-\delta_1^2}}\,, \\ s_1 = -\delta + j\sqrt{\omega_1^2-\delta_1^2}\,, \\ R_2 = -j\frac{G_2(\epsilon_s-\epsilon_\infty)\omega_2^2}{\sqrt{\omega_2^2-\delta_2^2}}\,, \\ s_2 = -\delta + j\sqrt{\omega_2^2-\delta_2^2}\,. \end{cases} \tag{8.46}$$

The radius of the sphere is 600 $\mu m$, and the edge lengths are 200 $\mu m$ and 100 $\mu m$, respectively. This resolution corresponds to ten and twenty points per wavelength in the sphere at 80 GHz. The solution in the monostatic case is presented in Figure 8.6. Clearly, for a resolution of ten points per wavelength the hybrid solution is accurate over the whole frequency spectrum. Also in this case FD-TD needs only twice as many points per wavelength to obtain results of similar accuracy.

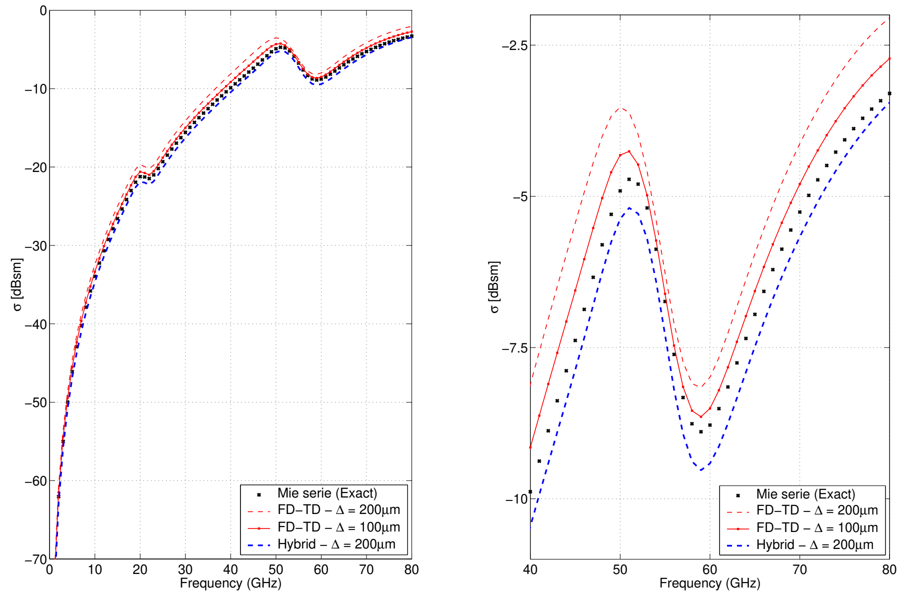


Figure 8.6: Backscattering from a sphere of the fourth-order dispersive material.

## 8.3 Conclusions

We have presented a generalization of the piecewise constant centered RC method proposed by Luebbers et al. to treat frequency dispersive materials in an unstructured finite volume solver, which is one part in a hybrid solver with FD-TD. The numerical examples for materials of Debye and Lorentz type as well as materials described by multiple second-order poles show excellent correspondence with the analytical Mie series solution. The moderate resolution needed to obtain accurate results clearly shows that our hybrid concept is a good strategy to avoid the staircasing errors present in standard FD-TD. In the numerical examples presented FD-TD typically require at least twice as many points per wavelength to produce as accurate results as the hybrid solver. The severity of the staircasing errors in FD-TD depends strongly on the differences in phase speeds in the spheres compared to the outside medium. The differences are a factor of two for the spheres of Lorentz and General material, and a factor of four for the water sphere. The staircased FD-TD solutions are much worse for the water sphere than for the other two cases. However, for a very low signature object it might even be the case that standard FD-TD works better than the hybrid solver due to the fact that the reflections at the grid interface could destroy the accuracy of the hybrid solution.

# Chapter 9

# Conclusions

We have introduced two explicit hybrid solvers for the 2D and 3D Maxwell equations in time domain. By combining the efficiency of the classical FD-TD method with the flexibility of an FV solver for unstructured grids we obtain a very efficient and accurate solver. The FV solver is basically a generalization of FD-TD to unstructured grids. The solver preserves the divergence of the electric and magnetic flux density vectors up to machine precision. This is a critical property for solvers of Maxwell's equations to avoid spurious modes in the numerical solution. Due to the fact that the explicit solver must obey a stability limit where the maximum time step is related to the smallest unstructured cell, the explicit solver is intended for problems where the cell sizes in the unstructured region are of the same size as the FD-TD cells. The generic aircraft $RUND$ is a good example where it is advantageous to use an explicit solver.

The numerical examples, both in 2D and in 3D, clearly demonstrate that our hybrid solver is much more efficient than standard FD-TD on complex geometries. It is also interesting to note that the numerical results show no signs of large reflections at the grid interface. We believe that this is mainly due to the transition layer, which ensures a smooth transition from the structured to the unstructured grid. The number of unstructured cells is minimized by allowing a staircased interface between the structured and unstructured grids.

Using Fourier analysis a careful study of the dispersion characteristics in 2D revealed that the FV solver is much less affected by dispersion errors than the standard FD-TD scheme. An interesting results is that for a uniform equilateral grid the numerical wave speed is independent of the propagation direction in the grid. The time discretization method ABS3 enables an easy coupling in time between the FV solver and FD-TD and its stability characteristics makes it a very suitable time integrator for an unstructured grid solver. An eigenvalue analysis and numerical experiments in 2D show that we obtain a stable solution on general grids using ABS3. However, in 3D the grids are much more complex and the use of ABS3 is in general not enough to guarantee a long term stable solution. The severity of the instability depends strongly on the deviation from orthogonality in the grid. For a grid where the primary and dual grids are mutually orthogonal no problems with stability are observed and all eigenvalues of the semi-discrete problem are on the imaginary axis. For general grids a spatial Laplace filter is introduced. The filter has been shown to work very well in numerical simulations and enables us to run all the scattering cases in Chapters 7 and 8 without any signs of instability. Since the filter only affects edges that are not orthogonal to their corresponding surfaces the accuracy of the solution is not deteriorated. The main drawback with the filter is that it is expensive in terms of memory requirements and number of arithmetic operations. The filter is responsible for roughly 30% of the total memory requirements as well as 30% of the arithmetic operations in the FV solver.

Future work includes to optimize the filter such that the over-head introduced will be much less significant. Furthermore, parallelizing the FV solver is clearly needed to be able to solve electrically larger problems. Subcell models for arbitrarily oriented thin wires as well as thin material sheets will also be added to the FV solver. Moreover, a local timestepping algorithm will be implemented, which implies that the time step used by the FV solver will be an integral subdivision of the FD-TD time step. Hence, the FV solver is advanced several sub-time-steps for each FD-TD time step and second-order time interpolation will be used to couple the two solvers in time. Since the errors in FD-TD decreases when the CFL number approaches one this is advantageous and also more efficient because fewer time steps have to performed by FD-TD.

## Acknowledgments

I would like to express my sincere gratitude to my two colleagues Gunnar Ledfelt and Ulf Andersson for many valuable discussions regarding the material in this thesis.

I am grateful to my advisor Professor Per Lötstedt for sharing his expertize in numerical analysis as well as leading me into this subject.

I would also like to thank Dr. Bo Strand for fruitful discussions regarding frequency dispersive materials and Jonas Gustafsson at Ericsson Saab Avionics for generating the hybrid grids used in this thesis.

This work was supported in part by the National Aeronautical Research Program and by the CEM Program at the Parallel and Scientific Computing Institute (PSCI). I have been able to participate in a number of conferences thanks to: PSCI, Bernt Järmarks stiftelse för vetenskaplig forskning and the European Space Agency.

# Bibliography


[1] E. Abenius, U. Andersson, F. Edelvik, L. Eriksson, and G. Ledfelt. Hybrid time domain solvers for the Maxwell equations in 2D. Technical Report 00:01, PSCI, Parallel and Scientific Computing Institute, KTH, SE-100 44 Stockholm, Sweden, February 2000. Available at `http://www.psci.kth.se/Activities/Reports/List.html`.

[2] V. Ahuja and L. N. Long. A parallel Finite-Volume Runge-Kutta algorithm for electromagnetic scattering. *J. of Computational Physics*, 137:299–320, 1997.

[3] U. Andersson. *Time Domain Methods for the Maxwell Equations.* PhD thesis, Royal Institute of Technology Stockholm (Sweden), to appear December 2000.

[4] S. Dey and R. Mittra. A locally conformal finite-difference time-domain (FDTD) algorithm for modeling three-dimensional perfectly conducting objects. *IEEE Microwave Guided Wave Lett.*, 7(9):273–275, September 1997.

[5] F. Edelvik. Survey of finite volume methods for solving Maxwell's equations. Technical report, Department of Scientific Computing, Uppsala University, Sweden, May 1998.

[6] F. Edelvik. Analysis of a finite volume solver for Maxwell's equations. In Vilsemeier R., editor, *Finite Volumes for Complex Applications II*, pages 141–148, Paris, France, July 1999. IVG, University of Duisburg, Hermes.

[7] F. Edelvik, U. Andersson, and G. Ledfelt. Hybrid finite volume-finite difference solver for the Maxwell equations. In *Millenium Conference on Antennas and Propagation*, volume 1, Davos, Switzerland, April 2000.

[8] S. D. Gedney. An anisotropic PML absorbing media for the FDTD simulation of fields in lossy and dispersive media. *Electromagnetics*, 16(4):399–415, 1996.

[9] M. Ghrist, T. A. Driscoll, and B. Fornberg. Staggered time integrators for the wave equation. *to appear in SIAM J. Sci. Comput.*, 2000.

[10] R. M. Joseph, S. C. Hagness, and A. Taflove. Direct time integration of Maxwell's equations in linear dispersive media with absorption for scattering and propagation of femtosecond electromagnetic pulses. *Optics Lett.*, 16(18):1412–1414, 1991.

[11] D. F. Kelley and R. J. Luebbers. Piecewise linear recursive convolution for dispersive media using FDTD. *IEEE Trans. Antennas Propagat.*, 44(6):792–797, June 1996.

[12] G. Ledfelt. *Time Domain Methods for the Maxwell Equations*. PhD thesis, Royal Institute of Technology Stockholm (Sweden), to appear December 2000.

[13] G. Ledfelt, F. Edelvik, L. Eriksson, and U. Andersson. Hybrid time domain solver for the Maxwell equations. In J. O. Gustafsson, editor, *RadioVetenskap och Kommunikation 99*, pages 70–74, Karlskrona, Sweden, June 1999. SNRV, NUTEK, Högskolan i Karlskrona/Ronneby.

[14] J.-F. Lee, R. Lee, and A. Cangellaris. Time-Domain Finite-Element methods. *IEEE Trans. Antennas Propagat.*, 45(3):430–442, March 1997.

[15] R. J. Luebbers and F. Hunsberger. FDTD for $N$th-order dispersive media. *IEEE Trans. Antennas Propagat.*, 40(11):1297–1301, November 1992.

[16] N. K. Madsen. Divergence preserving discrete surface integral methods for Maxwell's curl equations using non-orthogonal unstructured grids. *J. of Computational Physics*, 119:34–45, 1995.

[17] N. K. Madsen and R. W. Ziolkowski. A three-dimensional modified Finite Volume technique for Maxwell's equations. *Electromagnetics*, 10(1/2):147–161, 1990.

[18] T. Martin. An improved near- to far-zone transformation for the Finite-Difference Time-Domain method. *IEEE Trans. Antennas Propagat.*, 46(9):1263–1271, 1998.

[19] M. Okoniewski, E. Okoniewska, and M. A. Stuchly. Three-dimensional subgridding algorithm for FDTD. *IEEE Trans. Antennas Propagat.*, 45(3):422–429, March 1997.

[20] D. J. Riley and C. D. Turner. VOLMAX: A solid-model-based, transient volumetric Maxwell solver using hybrid grids. *IEEE Antennas Propagat. Magazine*, 39(1):20–33, 1997.

[21] J. W. Schuster and R. J. Luebbers. An accurate FDTD algorithm for dispersive media using a piecewise constant recursive convolution technique. In *IEEE Antennas and Propagat. Soc. Int. Symposium*, volume 4, pages 2018–2021, Atlanta, GA, June 1998.

[22] V. Shankar, W. F. Hall, and A. Mohammadian. A Time-Domain differential solver for electromagnetic scattering problems. *Proc. IEEE*, 77(5):709–721, 1989.

[23] B. Strand, U. Andersson, F. Edelvik, J. Edlund, L. Eriksson, S. Hagdahl, and G. Ledfelt. GEMS-A Swedish electromagnetic suite of hybrid solvers-Technical aspects. In *Millenium Conference on Antennas and Propagation*, volume 1, Davos, Switzerland, April 2000.

[24] A. Taflove. *Computational Electrodynamics: The Finite-Difference Time-Domain Method*. Artech House, Boston, MA, 1995.

[25] R.-B. Wu and T. Itoh. Hybridizing FD-TD analysis with unconditionally stable FEM for objects of curved boundary. In *IEEE MTT-S Int. Microwave Symp.*, volume 2, pages 833–836, Orlando, FL, 1995.

[26] K. S. Yee. Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media. *IEEE Trans. Antennas Propagat.*, 14(3):302–307, March 1966.

[27] K. S. Yee and J. S. Chen. Conformal hybrid Finite Difference Time Domain and Finite Volume Time Domain. *IEEE Trans. Antennas Propagat.*, 42(10):1450–1455, October 1994.

[28] K. S. Yee and J. S. Chen. The Finite-Difference Time-Domain (FDTD) and the Finite-Volume Time-Domain (FVTD) methods in solving Maxwell's equations. *IEEE Trans. Antennas Propagat.*, 45(3):354–363, March 1997.

[29] K. S. Yee, J. S. Chen, and A. H. Chang. Conformal Finite-Difference Time-Domain (FDTD) with overlapping grids. *IEEE Trans. Antennas Propagat.*, 40(9):1068–1075, September 1992.

[30] K. S. Yee, J. S. Chen, and A. H. Chang. Numerical experiments on PEC boundary condition and late time growth involving the FDTD/FDTD and FDTD/FVTD hybrid. In *IEEE Antennas and Propagat. Soc. Int. Symp.*, volume 1, pages 624–627, Newport Beach, CA, June 1995.

[31] J. L. Young, A. Kittichartphayak, Y. M. Kwok, and D. Sullivan. On the dispersion errors related to $(\mathrm{FD})^2$-TD type schemes. *IEEE Trans. Microwave Theory Tech.*, 43(8):1902–1910, August 1995.