ON A THIRD-ORDER FVTD SCHEME FOR THREE-DIMENSIONAL MAXWELL'S EQUATIONS

Marina Kotovshchikova

PhD thesis
submitted to the Faculty of Graduate Studies
of the University of Manitoba
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Department of Mathematics
University of Manitoba
Winnipeg, Manitoba, Canada

Abstract

This thesis considers the application of the type II third order WENO finite volume reconstruction for unstructured tetrahedral meshes proposed by Zhang and Shu in [135] and the third order multirate Runge-Kutta time-stepping to the solution of Maxwell's equations.

The dependance of accuracy of the third order WENO scheme on the small parameter ε in the definition of non-linear weights is studied in detail for one-dimensional uniform meshes and numerical results confirming the theoretical analysis are presented for the linear advection equation. This analysis is found to be crucial in the design of the efficient three-dimensional WENO scheme, full details of which are presented.

Several multirate Runge-Kutta (MRK) schemes which advance the solution with local time-steps assigned to different multirate groups are studied. Analysis of accuracy of three different MRK approaches for linear problems based on classic order-conditions is presented. The most flexible and efficient multirate schemes based on works by Tang and Warnecke [125] and Liu, Li and Hu [86] are implemented in three-dimensional finite volume time-domain (FVTD) method. The main characteristics of chosen MRK schemes are flexibility in defining the time-step ratios between multirate groups and consistency of the scheme. Various approaches to partition the three-dimensional computational domain into multirate groups to maximize the achievable speedup are discussed.

Numerical experiments with three-dimensional electromagnetic problems are presented to validate the performance of the proposed FVTD method. Three-dimensional results agree with theoretical and numerical accuracy analysis performed for the one-dimensional case. The proposed implementation of multirate schemes demonstrates greater speedup than previously reported in literature.

Acknowledgements

I would like to express my sincere gratitude to my advisor Dr. Shaun Lui for his guidance, patience and continuous support during the time of research and writing of this thesis. His guidance helped me to widen my research from various prospectives.

I would like to thank my husband and colleague Dr. Dmitry Firsov for sharing his expertise in FVTD implementation.

I would like to thank my thesis committee: Dr. Benqi Guo, Dr. Ronald D. Haynes, Dr. Parimala Thulasiraman for their time, and also for the valuable comments that helped me to further improve my thesis.

I also would like to acknowledge financial support given to me by the University of Manitoba, the Government of Manitoba and the National Science and Engineering Research Council of Canada.

And the last but not the least, I would like to express my thanks to all my family and friends for their support and encouragement.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Since Maxwell's equations are linear a lot of effort has been made to solve them analytically. However, only problems with very simple shapes such as a sphere or an infinite cylinder can be solved using analytic methods. For more complex problems one has to rely on experiments and numerical methods. Numerical methods for Maxwell's equations are referred to as computational electromagnetics (CEM), which have a wide range of applications including analysis and synthesis of antennas, radar cross section calculations, wireless communication devices, biomedical imaging and many more.

Maxwell's equations can be solved either in the time domain or in the frequency domain. Time-domain (TD) methods solve the time dependent Maxwell's equations and are suitable for broadband applications. The solution is obtained for all frequencies that can be resolved in one calculation. Frequency-domain (FD) methods solve the time-harmonic Maxwell's equations for one frequency at a time. They are suited for problems where only a few frequencies are needed. Many numerical methods are developed for both TD and FD formulations of Maxwell's equations, which include but not limited to finite differences (FD), discontinuous and continuous Galerkin finite elements (DG and FEM), and finite volumes (FV). Each approach has advantages and limitations and the choice must be made depending on the problem. The focus of this thesis is on the solution of time dependent Maxwell's equations, therefore the following review only concerns the numerical methods in the TD framework.

The finite difference time-domain (FDTD) method was introduced by Yee [132] in 1966 and further developed by Taflove in the 1970s. It is the most widespread simulation technique in CEM. It solves Maxwell's equations in partial differential form using second order accurate central differences on staggered grids in space and time for the electric and magnetic fields. For more details of the method we refer the reader to the book by Taflove and Hagness [124]. FDTD is conceptually very easy and efficient in terms of memory and computational time. Its main drawback is that it works well only on structured grids and so-called staircase approximations have to be used on curved boundaries.

To deal with more complex geometries computational domains are partitioned into unstructured grids (tetrahedral or mixed elements). Unstructured meshes allow better representation of objects with curved boundaries than unstructured meshes. They also permit higher resolution locally in order to better resolve fine geometrical structures and regions with fast variations of the solution. Most widely used methods for numerical approximations on unstructured grids include finite volumes, continuous and discontinuous Galerkin finite elements.

The finite volume time-domain (FVTD) algorithm for Maxwell's equations was adapted from computational fluid dynamics (CFD) in the late 1980's by Shankar et al. [114]. Since then several formulations of FVTD algorithms were developed including central [104, 102, 101] and upwind schemes [114, 18, 17]. Central formulation is the most straightforward with the main advantage of being non-dissipative, but is the least stable on highly distorted meshes even for linear problems. This formulation is coupled with the staggered leap-frog type time-stepping as a generalization of Yee's scheme [132] to unstructured meshes. The upwind formulation is usually based on so-called MUSCL (monotonic upwind scheme for conservation laws) developed by Van Leer [128]. It has better stability properties, but is dissipative. Upwind formulations are usually coupled with Lax-Wendroff or Runge-Kutta time integrations.

Higher order implementation of FVTD scheme can also be found in the literature [41]. The main limitation to extend the finite volume schemes to higher order is the use of an extended stencil to achieve better accuracy. Therefore, an alternative higher-order method based on polynomial approximation within each cell, namely the Discontinuous Galerkin (DG) method, have been extensively exploited in electromagnetic applications [27, 71, 95,

126]. DG methods provide higher accuracy, but they are also much more expensive and require smaller time steps. In addition to that, if a fine mesh is used to resolve small geometric features in a complex computational domain then the high accuracy of DG is useless. Therefore the less expensive and less accurate FVM method may be a better choice for complex geometries with very fine geometric structures. For more detailed comparison of strengths and limitations of the FVTD and the DGTD methods we refer to [33].

Geometrical flexibility of FVTD (and other schemes for unstructured meshes) is advantageous for problems where fine compared to the wavelength geometrical structures are embedded in large structures. In this case a high resolution mesh can be used locally near the fine features while cells of relatively large size can be used elsewhere. In CEM mesh nonuniformity and time-step restrictions also depend on dielectric contrasts. The main drawback of strong mesh and material inhomogeneities is that the maximum allowable time-step for explicit time evolution is limited by the finest cells with the smallest volume-area ratio. This leads to large CPU times as well as additional errors induced by non-optimal time-steps on large cells. To overcome these drawbacks a lot of effort has been put into development of local time-stepping (LTS) techniques. In the CEM literature there are LTS methods based on leap-frog type schemes [30, 95, 59, 10], multi-stage schemes (Runge-Kutta, predictor-corrector) [10, 44], multi-step schemes (Adams-Bashforth) [53, 59], as well as Cauchy-Kovalevskaja procedures [126].

1.1 Purpose

The purpose of this thesis is to study the application of a quadrature based WENO scheme for tetrahedral meshes and third order MRK schemes to solve the three-dimensional Maxwell's equations.

WENO schemes were designed to solve problems that encounter singularities in the solution. The idea of finite volume WENO scheme is to use an adaptive stencil to approximate the solution based on its smoothness. This avoids oscillations near singularities, but at the same time helps to preserve the accuracy of the smooth part of the solution. Although the scheme is designed with non-linear conservation laws in mind, its properties can also be useful for the solution of linear problems. Very few works implementing type II WENO

scheme on unstructured meshes can be found in the literature. The purpose of this thesis is to study the application of WENO scheme developed by Zhang and Shu [135] to solve linear hyperbolic system of equations, namely Maxwell's equations, in three-dimensional space. In this work we study the effect of non-oscillatory properties of the scheme [135] on the solution of Maxwell's equations containing singularities which may originate from the discontinuous source models as well as discontinuities in material properties.

The type II WENO scheme [135] was chosen because of a more compact stencil compared to the I WENO scheme [37, 38, 127, 88]. The design of type II WENO scheme follows the same idea as the classic one-dimensional WENO scheme [78]. Therefore convergence properties of the third order WENO scheme in the one-dimensional case are studied first to provide the insight for three-dimensional implementation. Following the idea in [11] the effect of various values of a small parameter ε in the definition of non-linear weights on the convergence of third order WENO scheme is studied. These results are then used in the implementation of the WENO scheme in the three-dimensional case. This accuracy analysis is a key factor for the efficiency of the three-dimensional scheme, because the numerical solution appears to be strongly affected by the choice of the value of ε.

In this work local time-stepping (LTS) schemes to improve the efficiency of the three-dimensional simulations are also studied. Since the third order spatial scheme is used in space, the time integration has to be performed by a third order scheme as well. Despite the abundance of LTS schemes, many of them lack accuracy analysis. The use of a higher order time scheme as a base method does not guarantee the same accuracy in the LTS approach. Even though the accuracy is lost only at interfaces between LTS domains it affects the entire solution. In order to achieve full accuracy of the time integration method the LTS procedure must preserve the accuracy of the base method. In this thesis the so-called classic order conditions up to the third order are used to analyze local time-stepping based on Runge-Kutta schemes. Such methods are often referred in literature as multirate Runge-Kutta (MRK) schemes [31, 75, 111]. The order conditions for MRK schemes applied to general nonlinear problems are derived using the Butcher form of Runge-Kutta schemes. They can be found in [63, 76]. The number of such conditions grows with the order of convergence. The total number of order conditions for third order accuracy is smaller for linear problems. In this thesis the third order conditions for linear problems are applied

to analyze several MRK schemes. Theoretical results are then confirmed by numerical experiments. In this thesis two multirate strategies based on the optimal second and third order strong stability-preserving Runge-Kutta (SSP RK) schemes were implemented for the solution of Maxwell's equations. The first one is based on the projection of the solution at intermediate time-step developed by Tang and Warnecke [125] (MRK-TW). This method is second order accurate with SSP RK2 but only first order accurate with SSP RK3 based on the order conditions for linear problems. The second one is based on the adjustment of stage values of the RK scheme near the LTS boundary developed by Liu, Li and Hu in [86] (MRK-LLH). This method is second order accurate with SSP RK2 and third order accurate with SSP RK3 as it satisfies all order conditions for linear problems.

Another aspect of implementation of multirate schemes on unstructured three-dimensional meshes is the partition of the domain into subdomains with different time-steps. Some of the methods are restricted to a specific time-step ratios and the order in which the multirate groups are distributed. The schemes implemented in this work allow an arbitrary ratio between the multirate time-steps, therefore a greater speedup can be achieved than previously reported in CEM literature.

1.2 Thesis outline

This work can be divided into two main parts: analysis and implementation of the third order finite volume WENO scheme, and an overview, analysis and implementation of MRK strategies.

In Chapter 2 the characteristic-based finite volume formulation of Maxwell's equations in time domain is reviewed. Temporal discretization based on strong stability-preserving Runge-Kutta (SSP RK) schemes is also discussed. In Chapter 3 the steps for construction of the third order WENO scheme in the one-dimensional case are outlined. The analysis of the effect of a small parameter ε in the definition of non-linear WENO coefficients on the overall accuracy is presented. Chapter 4 provides the details of the finite volume reconstruction based on the third order WENO scheme for tetrahedral meshes developed in [135].

Chapter 5 provides an overview of multirate schemes based on Runge-Kutta time stepping. The accuracy analysis of coupling strategies is presented, proving that only a few schemes satisfy the third order conditions even for linear problems. In Chapter 6 the steps for implementation of multirate strategies on three-dimensional meshes and various multirate partitions are outlined.

Chapter 7 finalizes this work presenting numerical experiments that validate the application of WENO reconstruction and multirate schemes to the solution of Maxwell's equations.

Chapter 2

FVTD for Maxwell's equations

This chapter serves as a review of the characteristic-based cell-centered finite volume scheme on a tetrahedral mesh for Maxwell's equations [18]. The procedure includes integration of the system of Maxwell's equations in conservative form over each tetrahedral cell and separation of the fluxes through the cell faces. The fluxes can be then approximated with a desired order of accuracy. In this thesis a third order weighted essentially non-oscillatory scheme [135] is implemented, this is the subject of Chapter 4. Temporal discretization based on a strong stability preserving Runge-Kutta schemes is discussed in this chapter. A multirate extension of the algorithm is presented in Chapter 6.

2.1 FV method in CEM overview

The finite volume method (FVM) was originally developed for the solution of hyperbolic conservation laws in fluid dynamics (see for example the LeVeque's book [85] for details) and is relatively new to computational electromagnetics (CEM). It has been successfully applied to time-domain electromagnetic problems for the last 20 years and is known as the finite volume time-domain (FVTD) method. Time-domain Maxwell's equations are hyperbolic in their nature and can be written in conservative form. Therefore it is natural to apply numerical approximations developed for hyperbolic conservation laws. Among different approaches there are FVTD solvers based on upwind or central flux approximations with predictor-corrector or leap-frog type methods in time.

The first implementations of FVTD methods were introduced in late 80's. Shankar et al. [114] applied a second order upwind scheme on a two-dimensional body-fitted structured mesh with predictor-corrector scheme in time. At the same time Madsen and Ziolkowski [90] proposed a staggered finite volume method similar to Yee's finite difference time domain (FDTD) method. The approach used by Shankar et al. was extended to three-dimensional tetrahedral meshes by Bonnet et al. in [18]. The scheme utilizes characteristic-based cell-centered finite volume formulation with all field components located at the same points and is second order accurate. This formulation has been applied with success to a variety of electromagnetic problems [17, 16, 18, 45, 44, 42, 77]. A comparative study of this FVTD scheme and the discontinuous Galerkin time-domain (DGTD) scheme was conducted by Deng et al. in [33]. A discussion on strengths and limitations of characteristic-bases FVTD method can be found in [43]. As an alternative, Remaki [104] proposed an FVTD method based on a centered flux formula and leap-frog approximation in time. This scheme was further studied in [102].

FVTD schemes mentioned above correspond to the class of cell-centered finite volume techniques. Although not new in computational fluid dynamics (CFD), very few works can be found exploiting cell-vertex FVM to solve Maxwell's equations. FVTD formulations based on the cell-vertex scheme originally proposed by Ni in [98] to solve Euler equations of gas dynamics can be found in [34]. The main advantage of this formulation is that the fields are located at the cell vertices which allows for a more accurate implementation of boundary conditions for perfectly conducting surfaces. On the other hand conditions at a dielectric boundary are not easy to implement in this case. For tetrahedral meshes the cell-vertex scheme is cumbersome to extend to higher order schemes because of the shape of control volumes.

In addition to pure FVTD methods a number of hybrid FVTD-FDTD methods have been developed [106, 133, 40]. In these works authors were motivated to combine geometrical flexibility of finite volumes with computational efficiency of finite differences. Another way to optimize computational cost of unstructured meshes is to use inhomogeneous meshes where the size of elements is adapted to geometry and physical parameters.

This chapter provides a background on the derivation of finite volume formulation of Maxwell's equations based on an upwind approach. The material presented here is not new

and is based on the work of Bonnet et al. [18] and other literature cited throughout the chapter.

2.2 Maxwell's equations in the time domain

The propagation of electromagnetic waves is described by the Maxwell's equations. In a region of space free from magnetic charges the time dependent Maxwell's equations consist of the following equations [48, 24]:
Ampère's law

$$\frac{\partial \mathbf{D}}{\partial \mathrm{t}} - \nabla \times \mathbf{H} = -\mathbf{J}, \tag{2.1}$$

Faraday's induction law

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

Gauss's law

$$\nabla \cdot \mathbf{D} = \rho, \tag{2.3}$$

and Gauss's law of Magnetism

$$\nabla \cdot \mathbf{B} = 0, \tag{2.4}$$

where $\mathbf{H}(\mathbf{x},\mathrm{t})$ and $\mathbf{E}(\mathbf{x},\mathrm{t})$ are magnetic and electric fields respectively, $\mathbf{D}(\mathbf{x},\mathrm{t})$ is the electric displacement, $\mathbf{B}(\mathbf{x},\mathrm{t})$ is the magnetic flux density, $\mathbf{J}$ is an electric current density, and $\rho(\mathbf{x},\mathrm{t})$ is the electric charge density. For isotropic, linear and time-invariant (non-dispersive) materials the following constitutive relationships hold

$$\mathbf{D} = \varepsilon \mathbf{E}, \tag{2.5}$$

$$\mathbf{B} = \mu \mathbf{H}, \tag{2.6}$$

where $\mu = \mu(\mathbf{x})$ is the magnetic permeability, and $\varepsilon = \varepsilon(\mathbf{x})$ is the electric permittivity. The current density consists of the conducting current density $\sigma\mathbf{E}$ (defined by the Ohm's law) and the imposed current density $\mathbf{J}_s$ representing a given source term, and can be written as

$$\mathbf{J} = \sigma\mathbf{E} + \mathbf{J}_s, \tag{2.7}$$

where $\sigma = \boldsymbol{\sigma}(\mathbf{x})$ is the electric conductivity. In a vacuum $\varepsilon = \varepsilon_0 \approx 8.854 \cdot 10^{-12} \frac{As}{Vm}$, $\mu = \mu_0 = 4\pi \cdot 10^{-7} \frac{Vs}{Am}$ and $\sigma = 0$.

Taking the divergence of Ampère's and Faraday's laws (2.1-2.2) and interchanging the spatial and temporal derivatives one can derive

$$\frac{\partial \nabla \cdot \mathbf{D}}{\partial \mathrm{t}} + \nabla \cdot \mathbf{J} = \mathbf{0}, \tag{2.8}$$

$$\frac{\partial \nabla \cdot \mathbf{B}}{\partial \mathrm{t}} = \mathbf{0}. \tag{2.9}$$

Using (2.3) in (2.8) the equation for electric charge conservation is derived as follows

$$\frac{\partial \rho}{\partial \mathrm{t}} + \nabla \cdot \mathbf{J} = \mathbf{0}. \tag{2.10}$$

From (2.8-2.9) one can see that the divergence of Ampère's and Faraday's laws are nothing but the time derivatives of Gauss's laws (2.3) and (2.4) respectively. Therefore, if the initial conditions satisfy (2.3), then Ampère's law together with electric charge conservation ensures that it holds at later times. Similarly, if (2.4) is satisfied at the initial time $\mathrm{t} = 0$, then Faraday's law enforces it for later times. Hence, we can view the Gauss's law (2.3) as the initial condition for charge density and Gauss's law of magnetism as a restriction for initial conditions for Faraday's law and solve the time-dependent part of Maxwell's equations.

Substituting the constitutive relations (2.5-2.6) together with (2.7) into Ampère's and Faraday's laws the following form of Maxwell's equations is derived

$$\varepsilon \frac{\partial \mathbf{E}}{\partial \mathrm{t}} - \nabla \times \mathbf{H} = -\mathbf{J}_s - \sigma\mathbf{E}, \tag{2.11}$$

$$\mu \frac{\partial \mathbf{H}}{\partial \mathrm{t}} + \nabla \times \mathbf{E} = \mathbf{0}. \tag{2.12}$$

The system (2.11-2.12) is used to advance **E** and **H** in numerical solutions of Maxwell's equations in the time-domain. The initial conditions are imposed on the fields **E** and **H** and must satisfy (2.3-2.4).

The system (2.11-2.12) can also be written in a normalized dimensionless form. To normalize the equations, the scaling parameters must be chosen in such a way, that all normalized variables have their peak values of order of unity. Consider the following normalized quantities

$$\boldsymbol{x} = \frac{\mathbf{x}}{l}, \quad t = \frac{\mathrm{t}}{l/c_0}, \tag{2.13}$$

where l is a reference length, $c_0 = (\mu_0\varepsilon_0)^{-\frac{1}{2}}$ is a dimensional speed of light in vacuum. Normalized permittivity and permeability are defined by

$$\varepsilon_r = \frac{\varepsilon}{\varepsilon_0}, \quad \mu_r = \frac{\mu}{\mu_0}.$$

The fields **E** and **H** can be normalized to a typical electric field intensity E by

$$\boldsymbol{E} = \frac{\mathbf{E}}{E}, \quad \boldsymbol{H} = \frac{Z_0}{E}\mathbf{H}, \quad \boldsymbol{J}_s = \frac{lZ_0}{E}\mathbf{J}_s, \quad \sigma_r = lZ_0\sigma, \tag{2.14}$$

where $Z_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}}$ is the dimensional free-space intrinsic impedance. Then the system (2.11-2.12) can be written in non-dimensional form as

$$\varepsilon_r\frac{\partial \boldsymbol{E}}{\partial t} - \nabla\times\boldsymbol{H} = -\boldsymbol{J}_s - \sigma_r\boldsymbol{E}, \tag{2.15}$$

$$\mu_r\frac{\partial \boldsymbol{H}}{\partial t} + \nabla\times\boldsymbol{E} = \mathbf{0}. \tag{2.16}$$

The same system of equations (2.15-2.16) can also be obtained by normalizing to a typical magnetic field intensity H [71] using

$$\boldsymbol{E} = \frac{Z_0^{-1}}{H}\mathbf{E}, \quad \boldsymbol{H} = \frac{\mathbf{H}}{H}, \quad \boldsymbol{J}_s = \frac{l}{H}\mathbf{J}_s, \quad \sigma_r = lZ_0\sigma. \tag{2.17}$$

Thus the computations can be performed with normalized non-dimensional quantities and the actual fields can then be recovered from either (2.14) or (2.17). The system (2.15-2.16) together with initial and boundary conditions form a first order linear hyperbolic system of equations.

The boundary conditions are imposed on interfaces between two different media as well as on artificial boundaries due to truncation of the unbounded domain. On the interface between the medium 1 characterized by ε_1 and μ_1, and the medium 2 characterized by ε_2 and μ_2 the electric and magnetic fields must satisfy continuity requirements for the tangential components [13]

$$\hat{\boldsymbol{n}} \times (\boldsymbol{E}_1 - \boldsymbol{E}_2) = \boldsymbol{0}, \tag{2.18}$$

$$\hat{\boldsymbol{n}} \times (\boldsymbol{H}_1 - \boldsymbol{H}_2) = \boldsymbol{0}. \tag{2.19}$$

If the medium 2 is a perfect electric conductor (PEC) the following condition is imposed on the boundary [13]

$$\hat{\boldsymbol{n}} \times \boldsymbol{E}_1 = 0, \tag{2.20}$$

and in case of perfect magnetic conductor (PMC) the boundary condition is [13]

$$\hat{\boldsymbol{n}} \times \boldsymbol{H}_1 = 0. \tag{2.21}$$

In the case of exterior problems, such as radiating antennas, the physical domain is unbounded. To solve these problems numerically the physical domain is truncated and boundary conditions are imposed on the artificial outer boundary of the computational domain. In FVTD the most natural condition for the artificial boundary is the Silver-Müller absorbing boundary condition [18]

$$\hat{\boldsymbol{n}} \times \boldsymbol{E}_1 + \sqrt{\frac{\mu_1}{\varepsilon_1}} \hat{\boldsymbol{n}} \times (\hat{\boldsymbol{n}} \times \boldsymbol{H}_1) = 0. \tag{2.22}$$

This ensures that waves coming in normal direction to the artificial boundary are not reflected. The Silver-Müller condition belongs to the family of local radiation boundary conditions and are the easiest and cheapest to implement. They require that the artificial

boundary is placed far enough from radiating objects to ensure that the waves coming to the boundary can be assumed to be locally plane waves. To improve the accuracy the global boundary conditions based on integral equation technique can be implemented [42]. These conditions allow to have the artificial boundary closer to the sources of radiation without compromising the physics. This method is not considered in this work and only mentioned for completeness.

There are two ways to model exterior scattering problems. One approach is to model a source (antenna) by $\boldsymbol{J}_s$ producing a field that scatters, for example, the bounded scatterer. In this case the function $\boldsymbol{J}_s$ is defined on a compact support away from the scatterer producing the fields $\boldsymbol{E}$ and $\boldsymbol{H}$. The initial conditions are given by

$$\boldsymbol{E}(\boldsymbol{x},0) = \boldsymbol{H}(\boldsymbol{x},0) = \mathbf{0}$$

on the computational domain.

Another approach is to use the linearity of Maxwell's equations and solve the problem for the scattered field $\left(\boldsymbol{E}^S, \boldsymbol{H}^S\right)$ only, rather than the total field. In this case the total field can be written as a combination of the incident field $\left(\boldsymbol{E}^I, \boldsymbol{H}^I\right)$ and the scattered field $\left(\boldsymbol{E}^S, \boldsymbol{H}^S\right)$ [13]

$$\boldsymbol{E} = \boldsymbol{E}^I + \boldsymbol{E}^S, \quad \boldsymbol{H} = \boldsymbol{H}^I + \boldsymbol{H}^S, \tag{2.23}$$

where the incident field is given and is the solution of Maxwell's equations in the free space ($\varepsilon_r = \mu_r = 1$, and $\sigma_r = 0$)

$$\frac{\partial \mathbf{E}^I}{\partial \mathrm{t}} - \nabla \times \mathbf{H}^I = \mathbf{0}, \tag{2.24}$$

$$\frac{\partial \mathbf{H}^I}{\partial \mathrm{t}} + \nabla \times \mathbf{E}^I = \mathbf{0}. \tag{2.25}$$

The scattered field formulation is recovered by inserting (2.23) into (2.15-2.16) with $\boldsymbol{J}_s = \mathbf{0}$. Using (2.24-2.25) we get Maxwell's system for the scattered field as

$$\varepsilon_r \frac{\partial \boldsymbol{E}^S}{\partial t} - \nabla \times \boldsymbol{H}^S = -\boldsymbol{J}_s - \sigma_r \boldsymbol{E}^S - \sigma_r \boldsymbol{E}^I + (1-\varepsilon_r)\frac{\partial \boldsymbol{E}^I}{\partial t}, \tag{2.26}$$

$$\mu_r \frac{\partial \boldsymbol{H}^S}{\partial t} + \nabla \times \boldsymbol{E}^S = (1-\mu_r)\frac{\partial \boldsymbol{H}^I}{\partial t}. \tag{2.27}$$

The initial conditions for the scattered formulation are given by

$$\boldsymbol{E}^S(\boldsymbol{x},0) = \boldsymbol{H}^S(\boldsymbol{x},0) = \boldsymbol{0},$$

and the boundary conditions (2.18-2.22) are reformulated for the scattered fields $\left(\boldsymbol{E}^S, \boldsymbol{H}^S\right)$.

2.3 Conservative form

Since the system of Maxwell's equations is hyperbolic in nature it can be written in conservative form, which is necessary for the application of the finite volume scheme. Consider propagation of electromagnetic waves in a bounded region $\Omega \subset \mathbb{R}^3$ governed by the system of Maxwell's equations in normalized form (2.15-2.16). Dropping the subscripts of ε_r, μ_r, σ_r and $\boldsymbol{J}_s$, we rewrite it as

$$\varepsilon \frac{\partial \boldsymbol{E}}{\partial \mathrm{t}} - \nabla \times \boldsymbol{H} = -\boldsymbol{J} - \sigma \boldsymbol{E}, \tag{2.28}$$

$$\mu \frac{\partial \boldsymbol{H}}{\partial \mathrm{t}} + \nabla \times \boldsymbol{E} = \boldsymbol{0}. \tag{2.29}$$

To apply FVM we need to rewrite the system (2.28-2.29) in the conservative form. As described in [18] define the following matrix operator

$$\boldsymbol{C}(\boldsymbol{\phi}) = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix}. \tag{2.30}$$

Then, if we take the divergence of the matrix $\boldsymbol{C}(\boldsymbol{\phi})$ we get

$$[\mathrm{Div}\,(\boldsymbol{C}(\boldsymbol{\phi}))]^T = \left(\begin{bmatrix} \partial_1 & \partial_2 & \partial_3 \end{bmatrix} \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix} \right)^T = \\ = \begin{bmatrix} \partial_2\phi_3 - \partial_3\phi_2 \\ \partial_3\phi_1 - \partial_1\phi_3 \\ \partial_1\phi_2 - \partial_2\phi_1 \end{bmatrix} = \nabla \times \boldsymbol{\phi}. \tag{2.31}$$

Using (2.31) Maxwell's equations (2.15-2.16) can be written as follows

$$\varepsilon \frac{\partial \boldsymbol{E}}{\partial t} - [\mathrm{Div}\,(\boldsymbol{C}(\boldsymbol{H}))]^T = -\boldsymbol{J} - \sigma \boldsymbol{E}, \tag{2.32}$$

$$\mu \frac{\partial \boldsymbol{H}}{\partial t} + [\mathrm{Div}\,(\boldsymbol{C}(\boldsymbol{E}))]^T = 0, \tag{2.33}$$

or in matrix form

$$\boldsymbol{\alpha} \frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{R}\boldsymbol{U} = -\boldsymbol{G} - \boldsymbol{\sigma}\boldsymbol{U}, \tag{2.34}$$

where $\boldsymbol{U} = [\boldsymbol{E}^T, \boldsymbol{H}^T]^T$ is the solution vector, and

$$\boldsymbol{\alpha} = \begin{bmatrix} \varepsilon \boldsymbol{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \mu \boldsymbol{I}_3 \end{bmatrix}, \quad \boldsymbol{R}\boldsymbol{U} = \begin{bmatrix} -[\mathrm{Div}\,(\boldsymbol{C}(\boldsymbol{H}))]^T \\ [\mathrm{Div}\,(\boldsymbol{C}(\boldsymbol{E}))]^T \end{bmatrix} \\ \boldsymbol{G} = \begin{bmatrix} \boldsymbol{J} \\ \boldsymbol{0} \end{bmatrix}, \qquad \boldsymbol{\sigma} = \begin{bmatrix} \sigma \boldsymbol{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \tag{2.35}$$

and $\boldsymbol{I}_3$ is a 3×3 identity matrix.

In applications such as simulations of waves scattered from an object the problem is solved using a scattered field formulation. In this case the electromagnetic field is represented as a sum of incident and scattered fields as in (2.23)

$$\boldsymbol{U} = \boldsymbol{U}^I + \boldsymbol{U}^S. \tag{2.36}$$

The incident field satisfies Maxwell's equations for free space (2.24-2.25) which can be rewritten as

$$\frac{\partial \boldsymbol{U}^I}{\partial t} + \boldsymbol{R}\boldsymbol{U}^I = \boldsymbol{0}. \tag{2.37}$$

Using (2.36) let us rewrite (2.34) as

$$\boldsymbol{\alpha}\frac{\partial}{\partial t}\left(\boldsymbol{U}^I + \boldsymbol{U}^S\right) + \boldsymbol{R}\left(\boldsymbol{U}^I + \boldsymbol{U}^S\right) = -\boldsymbol{\sigma}\left(\boldsymbol{U}^I + \boldsymbol{U}^S\right) - \boldsymbol{G}.$$

The components of incident field can be transferred to the right hand side of the above equation to form a source term

$$\boldsymbol{\alpha}\frac{\partial \boldsymbol{U}^S}{\partial t} + \boldsymbol{R}\boldsymbol{U}^S = -\boldsymbol{G} - \boldsymbol{\sigma}\left(\boldsymbol{U}^I + \boldsymbol{U}^S\right) - \left(\boldsymbol{\alpha}\frac{\partial \boldsymbol{U}^I}{\partial t} + \boldsymbol{R}\boldsymbol{U}^I\right). \tag{2.38}$$

Now using the equation for the incident field (2.37) we substitute $\frac{\partial \boldsymbol{U}^I}{\partial t} = -\boldsymbol{R}\boldsymbol{U}^I$ in (2.38) as follows

$$\boldsymbol{\alpha}\frac{\partial \boldsymbol{U}^S}{\partial t} + \boldsymbol{R}\boldsymbol{U}^S = -\boldsymbol{G} - \boldsymbol{\sigma}\boldsymbol{U}^S - \boldsymbol{\sigma}\boldsymbol{U}^I - (\boldsymbol{\alpha} - \boldsymbol{I}_6)\frac{\partial \boldsymbol{U}^I}{\partial t}. \tag{2.39}$$

Or, if we define the source term

$$\boldsymbol{G}^I = \boldsymbol{\sigma}\boldsymbol{U}^I + (\boldsymbol{\alpha} - \boldsymbol{I}_6)\frac{\partial \boldsymbol{U}^I}{\partial t} = \boldsymbol{\sigma}\boldsymbol{U}^I - (\boldsymbol{\alpha} - \boldsymbol{I}_6)\boldsymbol{R}\boldsymbol{U}^I, \tag{2.40}$$

where the time derivative can be computed numerically, the scattered field formulation is equivalent to Maxwell's equations with an added source term $\boldsymbol{G}^I$

$$\boldsymbol{\alpha}\frac{\partial \boldsymbol{U}^S}{\partial t} + \boldsymbol{R}\boldsymbol{U}^S = -\boldsymbol{G}^I - \boldsymbol{\sigma}\boldsymbol{U}^S - \boldsymbol{G}. \tag{2.41}$$

2.4 Finite volume method

The idea of finite volume (FV) method for hyperbolic problems is to decompose the domain into mesh cells and associate with each cell the solution average. The averages on each cell are then updated at each time-step by the flux through the cell's faces. Consider the partition

of some bounded domain $\Omega \subset \mathbb{R}^3$ into tetrahedral cells

$$\Omega_T = \cup_{i=1}^{N} T_i, \qquad T_i \cap T_{j,j\neq i} = \emptyset.$$

We denote by $\partial T_i = \cup_{j=1}^{4} S_{ij}$ the boundary of each cell consisting of 4 triangles S_{ij}, and by $|T_i| = \int_{T_i} dV$ the volume of T_i. We can define the cell-average $\overline{u}_i$ of the unknown function $u(\boldsymbol{x})$ over T_i as

$$\overline{u}_i = \frac{1}{|T_i|} \int_{T_i} u(\boldsymbol{x})\, dV = A(T_i)\, u, \tag{2.42}$$

here $A(T_i)$ is the cell averaging operator.

The design of finite volume schemes for hyperbolic problems is based on the approach often referred to as the REA (reconstruct-evolve-average) algorithm [85]. The idea of this algorithm was originally proposed by Godunov [52] for solving the non-linear Euler's equations of gas dynamics, and consists of the following steps for each time-step Δt:

1. Reconstruct the unknown function u from its cell averages $\overline{u} = \{\overline{u}_i\}_{i=1}^{N}$ by some reconstruction function $R(\boldsymbol{x}, \overline{u})$, such that

$$R(\boldsymbol{x}, \overline{u}) = u(\boldsymbol{x}) + O(h^r), \qquad \text{for } x \in T_i \text{ if } u \in C^r(T_i), \tag{2.43}$$

$$A(T_i)\, R(\boldsymbol{x}, \overline{u}) = \overline{u}_i, \tag{2.44}$$

   here h is an average linear cell size in the mesh. The reconstruction function is typically a piecewise polynomial of degree $r-1$ and is discontinuous across ∂T_i.

2. Evolve the solution (exactly or approximately) over the time interval Δt by evaluating the time derivative and fluxes across ∂T_i, this includes the boundary conditions on $\partial\Omega$.

3. Average the updated solution to obtain the new cell averages $\overline{u} = \{\overline{u}_i\}_{i=1}^{N}$.

The abstract form of the scheme can be given by

$$\overline{u}^{n+1} = A \cdot E_{\Delta t} \cdot R(\boldsymbol{x}, \overline{u}^n), \tag{2.45}$$

where A is the cell-averaging operator on Ω_T and $E_{\Delta t}$ is in our case an approximate evolution operator

$$u(t+\Delta t, \boldsymbol{x}) \approx E_{\Delta t} u(t, \boldsymbol{x}). \tag{2.46}$$

With a piecewise constant reconstruction

$$R(\boldsymbol{x}, \overline{u}^n) = \overline{u}_i, \qquad \boldsymbol{x} \in T_i, \tag{2.47}$$

in (2.45) one gets the Godunov's scheme [52], and with piecewise linear reconstruction

$$R(\boldsymbol{x}, \overline{u}^n) = \overline{u}_i + s_i(\boldsymbol{x} - \boldsymbol{x}_i), \qquad \boldsymbol{x} \in T_i, \tag{2.48}$$

where s_i is an approximation of the gradient $\nabla u(\boldsymbol{x}_i)$, it is a second order extension to Godunov's scheme [128]. Other reconstructions are possible to increase the spatial accuracy of the FV scheme. The third order WENO reconstructions for tetrahedral meshes are subjects of Chapter 4 and the explicit time integration in the evolution operator $E_{\Delta t}$ will be discussed later in Section 2.8. In this chapter the FV formulation of Maxwell's equations in conservative form is derived by averaging the system (2.32-2.33) on T_i. Then the approximate fluxes through ∂T_i are obtained by the upwind method.

2.5 FV formulation of Maxwell's equations

In this section the derivation of the finite volume formulation of Maxwell's equations based on the work of Bonnet et al. [18] is described. Consider the partition of the bounded domain $\Omega \subset \mathbb{R}^3$ into a tetrahedral mesh $\overline{\Omega}_T = \cup_{i=1}^{N} \overline{T}_i$. The boundary $\partial\Omega$ consists of the far-field boundary with absorbing conditions (2.22) and the boundary corresponding to the surface of a perfect electric conductor (PEC) or perfect magnetic conductor (PMC). We assume that the constitutive parameters ε, μ and σ are constant in each cell T_i, and, therefore, the matrices $\boldsymbol{\alpha}$ and $\boldsymbol{\sigma}$ defined by (2.35) are also constant in each T_i. To obtain a finite volume discretization, the system (2.32-2.33) is integrated over each element T_i as follows

$$\int_{T_i} \varepsilon \frac{\partial \boldsymbol{E}}{\partial t} dV - \int_{T_i} \left[\mathrm{Div}\left(\boldsymbol{C}\left(\boldsymbol{H}\right)\right)\right]^T dV = -\int_{T_i} \boldsymbol{J} dV - \int_{T_i} \sigma \boldsymbol{E} dV \tag{2.49}$$

$$\int_{T_i} \mu \frac{\partial \boldsymbol{H}}{\partial t} dV + \int_{T_i} \left[\mathrm{Div}\left(\boldsymbol{C}\left(\boldsymbol{E}\right)\right)\right]^T dV = \boldsymbol{0}. \tag{2.50}$$

Using the divergence theorem the following is obtained

$$\int_{T_i} \left[\mathrm{Div}\left(\boldsymbol{C}\left(\boldsymbol{\phi}\right)\right)\right]^T dV = \int_{\partial T_i} \left[\boldsymbol{C}\left(\boldsymbol{\phi}\right)\right]^T \hat{\boldsymbol{n}} dS,$$

where $\hat{\boldsymbol{n}} = [n_1, n_2, n_3]^T$ is the outward unit normal vector to ∂T_i. Now using vector product properties and the fact that $\boldsymbol{C}\left(\boldsymbol{\phi}\right)$ is a skew-symmetric matrix the following is derived

$$\left[\boldsymbol{C}\left(\boldsymbol{\phi}\right)\right]^T \hat{\boldsymbol{n}} = -\boldsymbol{C}\left(\boldsymbol{\phi}\right)\hat{\boldsymbol{n}} = -\boldsymbol{\phi} \times \hat{\boldsymbol{n}} = \hat{\boldsymbol{n}} \times \boldsymbol{\phi} = \boldsymbol{C}\left(\hat{\boldsymbol{n}}\right)\boldsymbol{\phi},$$

and hence

$$\int_{T_i} \left[\mathrm{Div}\left(\boldsymbol{C}\left(\boldsymbol{\phi}\right)\right)\right]^T dV = \int_{\partial T_i} \boldsymbol{C}\left(\hat{\boldsymbol{n}}\right)\boldsymbol{\phi}^* dS, \tag{2.51}$$

where

$$\boldsymbol{\phi}^* = \boldsymbol{\phi}|_{\partial T_i}. \tag{2.52}$$

Using the relation (2.51) the system (2.49-2.50) is transformed into

$$\varepsilon \frac{\partial \overline{\boldsymbol{E}}_i}{\partial t} - \frac{1}{|T_i|} \int_{\partial T_i} \boldsymbol{C}\left(\hat{\boldsymbol{n}}\right)\boldsymbol{H}^* dS = -\overline{\boldsymbol{J}}_i - \sigma \overline{\boldsymbol{E}}_i, \tag{2.53}$$

$$\mu \frac{\partial \overline{\boldsymbol{H}}_i}{\partial t} + \frac{1}{|T_i|} \int_{\partial T_i} \boldsymbol{C}\left(\hat{\boldsymbol{n}}\right)\boldsymbol{E}^* dS = \boldsymbol{0}. \tag{2.54}$$

Or in more compact form

$$\frac{\partial \overline{\boldsymbol{U}}_i}{\partial t} + \frac{1}{|T_i|} \int_{\partial T_i} \boldsymbol{\alpha}^{-1} \boldsymbol{R}\left(\hat{\boldsymbol{n}}\right)\boldsymbol{U}^* dS = -\boldsymbol{\alpha}^{-1}\left(\boldsymbol{G}_i + \boldsymbol{\sigma}\overline{\boldsymbol{U}}_i\right), \tag{2.55}$$

where $\boldsymbol{R}(\hat{\boldsymbol{n}})$ is given by

$$\boldsymbol{R}(\hat{\boldsymbol{n}}) = \begin{bmatrix} \mathbf{0} & -\boldsymbol{C}(\hat{\boldsymbol{n}}) \\ \boldsymbol{C}(\hat{\boldsymbol{n}}) & \mathbf{0} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & n_3 & -n_2 \\ 0 & 0 & 0 & -n_3 & 0 & n_1 \\ 0 & 0 & 0 & n_2 & -n_1 & 0 \\ 0 & -n_3 & n_2 & 0 & 0 & 0 \\ n_3 & 0 & -n_1 & 0 & 0 & 0 \\ -n_2 & n_1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.56}$$

Defining

$$\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}}) = \boldsymbol{\alpha}^{-1}\boldsymbol{R}(\hat{\boldsymbol{n}}), \quad \tilde{\boldsymbol{G}}_i = \boldsymbol{\alpha}^{-1}\boldsymbol{G}_i, \quad \tilde{\boldsymbol{\sigma}} = \boldsymbol{\alpha}^{-1}\boldsymbol{\sigma},$$

the system (2.55) can be rewritten as

$$\frac{\partial \overline{\boldsymbol{U}}_i}{\partial t} + \frac{1}{|T_i|}\int_{\partial T_i} \tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})\,\boldsymbol{U}^* dS = -\tilde{\boldsymbol{G}}_i - \tilde{\boldsymbol{\sigma}}\overline{\boldsymbol{U}}_i. \tag{2.57}$$

Then integrating over the surface of the tetrahedron one gets

$$\frac{\partial \overline{\boldsymbol{U}}_i}{\partial t} + \frac{1}{|T_i|}\sum_{j\in\mathscr{I}_i} \left|S_{ij}\right| \tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right) \boldsymbol{U}^*_{ij} = -\tilde{\boldsymbol{G}}_i - \tilde{\boldsymbol{\sigma}}\overline{\boldsymbol{U}}_i, \tag{2.58}$$

where $\mathscr{I}_i$ is the set of indexes of the neighbors of the element T_i, S_{ij} and $\left|S_{ij}\right|$ are the face shared by elements T_i and T_j and its area, and

$$\boldsymbol{U}^*_{ij} = \frac{1}{\left|S_{ij}\right|}\int_{S_{ij}} \boldsymbol{U}^* dS \tag{2.59}$$

is the surface-averaged value of $\boldsymbol{U}^*$ at the centroid of the face S_{ij}. The computation of $\boldsymbol{U}^*_{ij}$ is done using an approximation of specific order and will be discussed in Chapter 4.

For the scattered formulation we replace the total field $\boldsymbol{U}$ by its scattered component $\boldsymbol{U}^S$ in (2.58) and add the source term $-\tilde{\boldsymbol{G}}^I_i = -\boldsymbol{\alpha}^{-1}\boldsymbol{G}^I_i$ on the right-hand-side of the system

to get

$$\frac{\partial \overline{\boldsymbol{U}}_i^S}{\partial t} + \sum_{j \in \mathscr{I}_i} \frac{\left|S_{ij}\right|}{|T_i|} \tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right) \boldsymbol{U}_{ij}^{S*} = -\tilde{\boldsymbol{G}}_i - \tilde{\boldsymbol{\sigma}} \overline{\boldsymbol{U}}_i^S - \tilde{\boldsymbol{G}}_i^I. \tag{2.60}$$

2.6 Steger-Warming splitting

One of the common ways to define the flux function $\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})\boldsymbol{U}^*$ is to use a simple non-dissipative central flux. Due to the hyperbolic nature of Maxwell's equations, another natural approach is to use the upwind scheme based on the Steger-Warming flux vector splitting [123]. This splitting is based on the method of characteristics and separates the flux on a face into outgoing and incoming part. The flux $\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})\boldsymbol{U}^*$ is split into a positive and negative parts according to the sign of the eigenvalues of the matrix $\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})$. Consider the following decomposition of $\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})$ [18]

$$\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}}) = \boldsymbol{Q}\Lambda\boldsymbol{Q}^{-1}, \tag{2.61}$$

where $\boldsymbol{Q}$ is the matrix consisting of eigenvectors of $\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})$, and Λ is the diagonal matrix whose diagonal elements are the corresponding eigenvalues $\lambda = \{0,0,c,c,-c,-c\}$, and $c = (\varepsilon\mu)^{-1/2}$ is the speed of light in the medium. One of possible choices of matrix $\boldsymbol{Q}$ can be written as [18]

$$\boldsymbol{Q} = \begin{bmatrix} n_1 & 0 & \frac{n_1 n_3}{c\varepsilon} & -\frac{n_1 n_2}{c\varepsilon} & -\frac{n_1 n_3}{c\varepsilon} & \frac{n_1 n_2}{c\varepsilon} \\ n_2 & 0 & \frac{n_2 n_3}{c\varepsilon} & \frac{n_1^2+n_3^2}{c\varepsilon} & -\frac{n_2 n_3}{c\varepsilon} & -\frac{n_1^2+n_3^2}{c\varepsilon} \\ n_3 & 0 & -\frac{n_1^2+n_2^2}{c\varepsilon} & -\frac{n_2 n_3}{c\varepsilon} & \frac{n_1^2+n_2^2}{c\varepsilon} & \frac{n_2 n_3}{c\varepsilon} \\ 0 & n_1 & -n_2 & -n_3 & -n_2 & -n_3 \\ 0 & n_2 & n_1 & 0 & n_1 & 0 \\ 0 & n_3 & 0 & n_1 & 0 & n_1 \end{bmatrix}. \tag{2.62}$$

If we define by Λ^+ and Λ^- as the diagonal matrices with positive and negative eigenvalues of $\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})$ respectively, then the matrix $\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})$ can be split into a positive and negative part as

$$\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}}) = \boldsymbol{Q}\left(\Lambda^{+}+\Lambda^{-}\right)\boldsymbol{Q}^{-1} = \boldsymbol{Q}\Lambda^{+}\boldsymbol{Q}^{-1}+\boldsymbol{Q}\Lambda^{-}\boldsymbol{Q}^{-1} = \tilde{\boldsymbol{R}}^{+}(\hat{\boldsymbol{n}})+\tilde{\boldsymbol{R}}^{-}(\hat{\boldsymbol{n}}). \tag{2.63}$$

It can be shown, that matrices $\tilde{\boldsymbol{R}}^{+}(\hat{\boldsymbol{n}})$ and $\tilde{\boldsymbol{R}}^{-}(\hat{\boldsymbol{n}})$ can be written in the following convenient form

$$\tilde{\boldsymbol{R}}^{+}(\hat{\boldsymbol{n}}) = \frac{1}{2}\begin{bmatrix} -c\boldsymbol{C}(\hat{\boldsymbol{n}})^2 & -\frac{\boldsymbol{C}(\hat{\boldsymbol{n}})}{\varepsilon} \\ \frac{\boldsymbol{C}(\hat{\boldsymbol{n}})}{\mu} & -c\boldsymbol{C}(\hat{\boldsymbol{n}})^2 \end{bmatrix} = \frac{1}{2}\boldsymbol{\alpha}^{-1}\begin{bmatrix} -Y\boldsymbol{C}(\hat{\boldsymbol{n}})^2 & -\boldsymbol{C}(\hat{\boldsymbol{n}}) \\ \boldsymbol{C}(\hat{\boldsymbol{n}}) & -Z\boldsymbol{C}(\hat{\boldsymbol{n}})^2 \end{bmatrix}, \tag{2.64}$$

$$\tilde{\boldsymbol{R}}^{-}(\hat{\boldsymbol{n}}) = \frac{1}{2}\begin{bmatrix} c\boldsymbol{C}(\hat{\boldsymbol{n}})^2 & -\frac{\boldsymbol{C}(\hat{\boldsymbol{n}})}{\varepsilon} \\ \frac{\boldsymbol{C}(\hat{\boldsymbol{n}})}{\mu} & c\boldsymbol{C}(\hat{\boldsymbol{n}})^2 \end{bmatrix} = \frac{1}{2}\boldsymbol{\alpha}^{-1}\begin{bmatrix} Y\boldsymbol{C}(\hat{\boldsymbol{n}})^2 & -\boldsymbol{C}(\hat{\boldsymbol{n}}) \\ \boldsymbol{C}(\hat{\boldsymbol{n}}) & Z\boldsymbol{C}(\hat{\boldsymbol{n}})^2 \end{bmatrix}, \tag{2.65}$$

here $Y = \sqrt{\frac{\varepsilon}{\mu}}$ represents the conductance and $Z = Y^{-1}$ the impedance. Clearly, from (2.63) and (2.64-2.65) it can be deduced that matrices $\tilde{\boldsymbol{R}}^{+}(\hat{\boldsymbol{n}})$ and $\tilde{\boldsymbol{R}}^{-}(\hat{\boldsymbol{n}})$ have the following properties

$$\tilde{\boldsymbol{R}}^{-}(\hat{\boldsymbol{n}}) = -\tilde{\boldsymbol{R}}^{+}(-\hat{\boldsymbol{n}}), \tag{2.66}$$

$$\tilde{\boldsymbol{R}}^{+}(\hat{\boldsymbol{n}}) = \frac{1}{2}\left(\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})+\left|\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})\right|\right), \tag{2.67}$$

where

$$\left|\tilde{\boldsymbol{R}}(\hat{\boldsymbol{n}})\right| = \tilde{\boldsymbol{R}}^{+}(\hat{\boldsymbol{n}})+\tilde{\boldsymbol{R}}^{+}(-\hat{\boldsymbol{n}}) = \tilde{\boldsymbol{R}}^{+}(\hat{\boldsymbol{n}})-\tilde{\boldsymbol{R}}^{-}(\hat{\boldsymbol{n}}). \tag{2.68}$$

Now consider the numerical flux $\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^{*}_{ij}$ on the face S_{ij}. Using (2.63) it can be split into a positive (the outgoing part in the direction of $\hat{\boldsymbol{n}}_{ij}$) and negative part (the incoming part in the direction of -$\hat{\boldsymbol{n}}_{ij}$). Let $\boldsymbol{U}_{ij} = [\boldsymbol{E}^{T}_{ij}, \boldsymbol{H}^{T}_{ij}]^{T}$ be the boundary field on the left side of the face S_{ij} (the side of the element T_i), and $\boldsymbol{U}_{ji} = [\boldsymbol{E}^{T}_{ji}, \boldsymbol{H}^{T}_{ji}]^{T}$ be the right boundary field (the side of the element T_j). The numerical flux $\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^{*}_{ij}$ can be split using these left and right boundary fields. The computation of the fields $\boldsymbol{U}_{ij}$ and $\boldsymbol{U}_{ji}$ is the main challenge of the FVM. The numerical representation using the second and third order reconstructions will be discussed in Chapter 4. The fields $\boldsymbol{U}_{ij}$ and $\boldsymbol{U}_{ji}$ are computed using stencils of elements

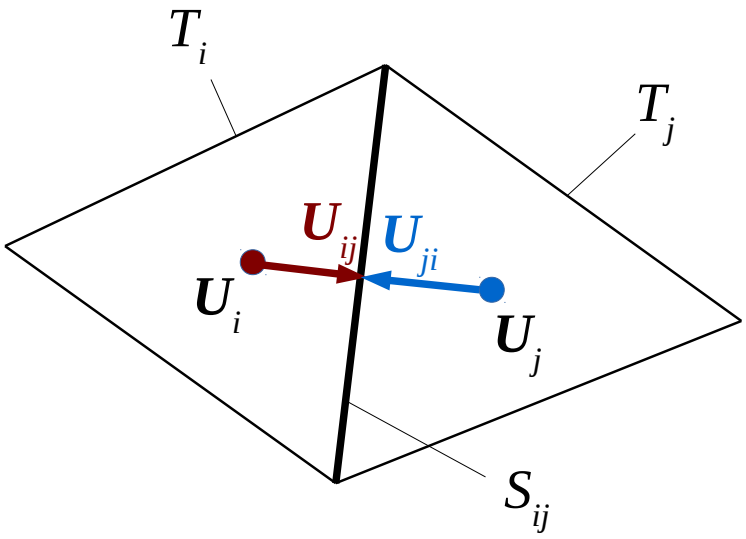


Figure 2.1: Boundary fields $\boldsymbol{U}_{ij}$ and $\boldsymbol{U}_{ji}$ at the face S_{ij} estimated using stencils for cells T_i and T_j respectively.

T_i and T_j respectively. As a result numerical approximations of the electromagnetic fields at two sides of S_{ij} are not equal. Theoretically the tangential components of the electric and magnetic fields are continuous across S_{ij}. Therefore the numerical approximation using the upwind scheme contradicts boundary conditions for the tangential components. The proper connection between $\boldsymbol{U}^*_{ij}$, $\boldsymbol{U}_{ij}$ and $\boldsymbol{U}_{ji}$ is discussed below.

2.7 Treatment of boundaries

In this section boundary conditions relevant to this work are reviewed. The discussion below is based on the work [18] with some details borrowed from [17, 77].

2.7.1 A face between two dielectrics

If we assume that the material properties in elements T_i and T_j on two sides of the face S_{ij} are given by (ε_i, μ_i) and (ε_j, μ_j) respectively, we can write the following expressions for

the left and right fluxes as

$$\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_i\right)\boldsymbol{U}_{ij}=\frac{1}{2}\boldsymbol{\alpha}_i^{-1}\begin{bmatrix}-Y_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}\right)-\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}\\-Z_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}\right)+\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}\end{bmatrix}, \tag{2.69}$$

and

$$\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_j\right)\boldsymbol{U}_{ji}=\frac{1}{2}\boldsymbol{\alpha}_j^{-1}\begin{bmatrix}Y_j\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ji}\right)-\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ji}\\Z_j\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ji}\right)+\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ji}\end{bmatrix}. \tag{2.70}$$

In the case when the dielectric properties of materials on two sides of a face are different, it is assumed that the tangential components of the electric and magnetic fields are continuous across the face [18]

$$\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}^{*}=\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}=\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ji}, \tag{2.71}$$

$$\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}^{*}=\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}=\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ji}. \tag{2.72}$$

From $(2.71-2.72)$ it follows that

$$\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}^{*}\right)=\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}\right)=\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ji}\right), \tag{2.73}$$

$$\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}^{*}\right)=\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}\right)=\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ji}\right). \tag{2.74}$$

Thus one can write

$$\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_i\right)\boldsymbol{U}_{ij}^{*}=\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_i\right)\boldsymbol{U}_{ij}, \tag{2.75}$$

$$\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_j\right)\boldsymbol{U}_{ij}^{*}=\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_j\right)\boldsymbol{U}_{ji}. \tag{2.76}$$

Using (2.69-2.70) the above can be rewritten as

$$\begin{bmatrix}-Y_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}^{*}\right)-\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}^{*}\\-Z_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}^{*}\right)+\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}^{*}\end{bmatrix}=\begin{bmatrix}-Y_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}\right)-\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}\\-Z_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}\right)+\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}\end{bmatrix},$$

and

$$\begin{bmatrix} Y_j\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}^*_{ij}\right)-\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}^*_{ij} \\ Z_j\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}^*_{ij}\right)+\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}^*_{ij} \end{bmatrix} = \begin{bmatrix} Y_j\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ji}\right)-\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ji} \\ Z_j\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ji}\right)+\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ji} \end{bmatrix},$$

and hence

$$\begin{aligned} \boldsymbol{R}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^*_{ij} &= \begin{bmatrix} -\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}^*_{ij} \\ \hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}^*_{ij} \end{bmatrix} \\ &= \begin{bmatrix} -\hat{\boldsymbol{n}}_{ij}\times\frac{\left[\hat{\boldsymbol{n}}_{ij}\times\left(\boldsymbol{E}_{ij}-\boldsymbol{E}_{ji}\right)+\left(Z_i\boldsymbol{H}_{ij}+Z_j\boldsymbol{H}_{ji}\right)\right]}{Z_i+Z_j} \\ \hat{\boldsymbol{n}}_{ij}\times\frac{\left[-\hat{\boldsymbol{n}}_{ij}\times\left(\boldsymbol{H}_{ij}-\boldsymbol{H}_{ji}\right)+\left(Y_i\boldsymbol{E}_{ij}+Y_j\boldsymbol{E}_{ji}\right)\right]}{Y_i+Y_j} \end{bmatrix}. \end{aligned} \tag{2.77}$$

If we define the transmission matrices by

$$\boldsymbol{T}_{ij} = \begin{bmatrix} \frac{2Z_i}{Z_i+Z_j}\boldsymbol{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \frac{2Y_i}{Y_i+Y_j}\boldsymbol{I}_3 \end{bmatrix}, \qquad \text{and } \boldsymbol{T}_{ji} = \begin{bmatrix} \frac{2Z_j}{Z_i+Z_j}\boldsymbol{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \frac{2Y_j}{Y_i+Y_j}\boldsymbol{I}_3 \end{bmatrix}, \tag{2.78}$$

then (2.77) can be written as

$$\boldsymbol{R}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^*_{ij} = \boldsymbol{\alpha}_i\boldsymbol{T}_{ij}\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_i\right)\boldsymbol{U}_{ij} + \boldsymbol{\alpha}_j\boldsymbol{T}_{ji}\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_j\right)\boldsymbol{U}_{ji}, \tag{2.79}$$

or

$$\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^*_{ij} = \boldsymbol{\alpha}_i^{-1}\left(\boldsymbol{\alpha}_i\boldsymbol{T}_{ij}\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_i\right)\boldsymbol{U}_{ij} + \boldsymbol{\alpha}_j\boldsymbol{T}_{ji}\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_j\right)\boldsymbol{U}_{ji}\right). \tag{2.80}$$

In the case when the physical properties on both side are the same, we have $\boldsymbol{T}_{ij} = \boldsymbol{T}_{ji} = \boldsymbol{I}_6$, and hence (2.80) gives

$$\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^*_{ij} = \tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij} + \tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ji}. \tag{2.81}$$

Using (2.80) in (2.58) we get the following FV scheme written as a method of lines ODE system

$$\frac{\partial \overline{\boldsymbol{U}}_i}{\partial t} + \sum_{j \in \mathscr{I}_i} \frac{|S_{ij}|}{|T_i|} \boldsymbol{\alpha}_i^{-1} \left(\boldsymbol{\alpha}_i \boldsymbol{T}_{ij} \tilde{\boldsymbol{R}}^{+} \left(\hat{\boldsymbol{n}}_{ij}, \boldsymbol{\alpha}_i \right) \boldsymbol{U}_{ij} + \boldsymbol{\alpha}_j \boldsymbol{T}_{ji} \tilde{\boldsymbol{R}}^{-} \left(\hat{\boldsymbol{n}}_{ij}, \boldsymbol{\alpha}_j \right) \boldsymbol{U}_{ji} \right) = -\tilde{\boldsymbol{G}}_i - \tilde{\boldsymbol{\sigma}} \overline{\boldsymbol{U}}_i. \tag{2.82}$$

In the scattered field formulation the boundary conditions (2.71-2.72) give

$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{E}_{ij}^{S} = \hat{\boldsymbol{n}}_{ij} \times \boldsymbol{E}_{ji}^{S},$$
$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{H}_{ij}^{S} = \hat{\boldsymbol{n}}_{ij} \times \boldsymbol{H}_{ji}^{S}.$$

2.7.2 A face at PEC

Now consider the case when the S_{ij} face of the element T_i is located on a perfect electric conductor (PEC). Then the tangential part of the electric field at the boundary vanishes [18]

$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{E}_{ij} = 0. \tag{2.83}$$

Using the image principle (see [14], p.167), which assumes that instead of the PEC boundary there is an incoming field with opposite orientation of electric components. Then the above condition can be implemented by

$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{E}_{ij} = -\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{E}_{ji}, \tag{2.84}$$
$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{H}_{ij} = \hat{\boldsymbol{n}}_{ij} \times \boldsymbol{H}_{ji}. \tag{2.85}$$

In the image principle, the PEC element T_j is assumed to have the same material properties as the element T_i. Therefore using the flux formula (2.81) together with (2.84-2.85) one can derive

$$\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij}^{*} = \frac{1}{2}\boldsymbol{\alpha}_i^{-1}\begin{bmatrix} -Y_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\left[\boldsymbol{E}_{ij}+\boldsymbol{E}_{ij}\right]\right)-\hat{\boldsymbol{n}}_{ij}\times\left[\boldsymbol{H}_{ij}+\boldsymbol{H}_{ij}\right] \\ -Z_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\left[\boldsymbol{H}_{ij}-\boldsymbol{H}_{ij}\right]\right)+\hat{\boldsymbol{n}}_{ij}\times\left[\boldsymbol{E}_{ij}-\boldsymbol{E}_{ij}\right] \end{bmatrix}$$
$$= \begin{bmatrix} 2\boldsymbol{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}\frac{1}{2}\boldsymbol{\alpha}_i^{-1}\begin{bmatrix} -Y_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}\right)-\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij} \\ -Z_i\hat{\boldsymbol{n}}_{ij}\times\left(\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{H}_{ij}\right)+\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij} \end{bmatrix},$$

and hence the flux on the boundary of the PEC can be expressed by

$$\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij}^{*} = \boldsymbol{T}_{PEC}\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij}, \tag{2.86}$$

with

$$\boldsymbol{T}_{PEC} = \begin{bmatrix} 2\boldsymbol{I}_3 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}.$$

2.7.3 Scattered field at PEC

Consider the scattered field solution on the face S_{ij} located on a PEC in the following form

$$\boldsymbol{U}_{ij} = \boldsymbol{U}^{I} + \boldsymbol{U}_{ij}^{S}, \tag{2.87}$$

here $\boldsymbol{U}^{I}$ is an incident field given in the computational domain and hence is an incident field on the face S_{ij}. The boundary condition on the face is imposed by

$$\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}_{ij}^{S} = -\hat{\boldsymbol{n}}_{ij}\times\boldsymbol{E}^{I}.$$

Using the above condition together with the image principle (2.84-2.85) at the PEC face S_{ij} the following can be obtained

$$\begin{aligned}\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij}^{*} =&\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij}+\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ji}=\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}\boldsymbol{E}_{ij}\\ \boldsymbol{H}_{ij}\end{bmatrix}+\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}-\boldsymbol{E}_{ij}\\ \boldsymbol{H}_{ij}\end{bmatrix}\\ =&\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}\boldsymbol{E}^{I}\\ \boldsymbol{H}^{I}\end{bmatrix}+\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}\boldsymbol{E}_{ij}^{S}\\ \boldsymbol{H}_{ij}^{S}\end{bmatrix}\\ &+\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}-2\boldsymbol{E}^{I}\\ \boldsymbol{0}\end{bmatrix}+\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}\boldsymbol{E}^{I}\\ \boldsymbol{H}^{I}\end{bmatrix}+\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}-\boldsymbol{E}_{ij}^{S}\\ \boldsymbol{H}_{ij}^{S}\end{bmatrix}.\end{aligned} \tag{2.88}$$

The second and last terms give the flux on a PEC boundary

$$\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}\boldsymbol{E}_{ij}^{S}\\ \boldsymbol{H}_{ij}^{S}\end{bmatrix}+\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}-\boldsymbol{E}_{ij}^{S}\\ \boldsymbol{H}_{ij}^{S}\end{bmatrix}=\boldsymbol{T}_{PEC}\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij}^{S}.$$

The third term is the PEC source term for the scattered field formulation

$$\tilde{\boldsymbol{G}}_{PEC}^{I}=\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}2\boldsymbol{E}^{I}\\ \boldsymbol{0}\end{bmatrix}.$$

The first and the fourth terms in (2.88) contribute to the source term $\tilde{\boldsymbol{G}}^{I}$ for dielectrics

$$\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}\boldsymbol{E}^{I}\\ \boldsymbol{H}^{I}\end{bmatrix}+\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij}\right)\begin{bmatrix}\boldsymbol{E}^{I}\\ \boldsymbol{H}^{I}\end{bmatrix}=\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^{I}.$$

The part $\boldsymbol{\alpha}\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}^{I}$ of the source term $\tilde{\boldsymbol{G}}^{I}$ appears from time integration of $\boldsymbol{U}^{I}+\boldsymbol{U}_{ij}^{S}$. Hence on the boundary of PEC, we have the following form of the FVM scheme for the scattered formulation

$$\frac{\partial\overline{\boldsymbol{U}}_{i}^{S}}{\partial t}+\sum_{j\in\mathscr{S}_{i}}\frac{\left|S_{ij}\right|}{\left|T_{i}\right|}\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right)\boldsymbol{U}_{ij}^{S*}=-\tilde{\boldsymbol{\sigma}}\overline{\boldsymbol{U}}_{i}^{S}-\tilde{\boldsymbol{G}}_{i}^{I}-\left[\tilde{\boldsymbol{G}}_{PEC}^{I}\right]_{i} \tag{2.89}$$

with the flux on the left hand side defined by (2.86).

2.7.4 A face at PMC

When the S_{ij} face of the element T_i is located on a perfect magnetic conductor (PMC) the boundary condition can be written as

$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{H}_{ij} = 0. \tag{2.90}$$

Using the image principle at the PMC boundary [14], which assumes that instead of the PMC boundary there is an incoming field with opposite orientation of magnetic components.

$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{E}_{ij} = \hat{\boldsymbol{n}}_{ij} \times \boldsymbol{E}_{ji}, \tag{2.91}$$

$$\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{H}_{ij} = -\hat{\boldsymbol{n}}_{ij} \times \boldsymbol{H}_{ji}, \tag{2.92}$$

the following formulation for the flux is obtained

$$\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right) \boldsymbol{U}_{ij}^{*} = \boldsymbol{T}_{PMC} \tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right) \boldsymbol{U}_{ij}, \tag{2.93}$$

with

$$\boldsymbol{T}_{PMC} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & 2\boldsymbol{I}_3 \end{bmatrix}.$$

2.7.5 Absorbing boundary conditions.

The benefit of using the flux-splitting formulation of the FVM is that it provides an easy handling of incoming and outgoing fluxes. Since the Silver-Müller boundary condition (2.22) is based on the assumption that the incoming flux is zero, its application to an upwind FV scheme is straight-forward. Consider the volume T_i with the j-th face belonging to the artificial far-field boundary. Then using the assumption of the Silver-Müller boundary condition the flux through the j-th face can be given by

$$\tilde{\boldsymbol{R}}\left(\hat{\boldsymbol{n}}_{ij}\right) \boldsymbol{U}_{ij}^{*} = \tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij}\right) \boldsymbol{U}_{ij}, \tag{2.94}$$

here $\boldsymbol{U}_{ij}$ is obtained by the reconstruction formula on a modified (boundary) stencil. It should be noted that the above condition ensures absorption of plane waves coming normal to the face. For waves that do not come in the normal direction, a partial reflection will be observed. Therefore, spherical outer boundaries with sufficient distance from the sources are usually chosen for the exterior radiation problems to ensure almost normal incidence of impinging waves and, thus, the accuracy of the Silver-Müller boundary conditions. As it is shown in [16] the sufficient distance from the radiating source is typically between one to two wavelengths at the lowest frequency. In this work for problems that require domain truncation only a time period before the signal reaches the outer boundary is considered. In this case the numerical solution is not polluted by partial reflection induced by boundary conditions.

2.8 Time integration

Once the spatial derivatives in the system of Maxwell's equations (2.34) are discretized we have a method of lines system of ODEs given by (2.82). It can be then integrated in time by either an explicit or implicit method. Both have advantages and disadvantages, which have to be evaluated based on the problem to be solved. Implicit methods require a much larger computational effort but can be designed to have unconditional linear stability. In this case the time-step size can be chosen based on characteristics of the problem only. They can also be optimized using parallel time integrators [25, 26]. Explicit time integrations require less effort per time-step, but are only conditionally stable. The maximum allowable time-step depends not only on the problem to be solved, but also on the characteristics of the mesh. In this work the focus is on explicit strong stability-preserving Runge-Kutta (SSP RK) schemes and their multirate (local time-stepping) versions. SSP RK schemes are most often used together with ENO and WENO approximations. The main argument for this is that non SSP RK scheme may produce spurious oscillations even when coupled with an essentially non-oscillatory scheme (see [56] for a numerical example). In this section a brief description of most popular SSP RK schemes is presented. Their extension to multirate time-stepping is the subject of Chapters 5 and 6.

2.8.1 SSP RK for Maxwell's system with zero source term

When solving hyperbolic problems linear stability is not sufficient for convergence when discontinuities are present in the solution and stronger non-linear stability such as TVD (total variation diminishing), TVB (total variation bounded) or ENO (essentially non-oscillatory) property is needed to avoid spurious oscillations. In the finite volume framework the total variation of the numerical solution is defined by

$$TV\left(\overline{u}\right) = \sum_{i=1}^{N-1} \left|\overline{u}_{i+1} - \overline{u}_i\right|. \tag{2.95}$$

The scheme is called total variation diminishing (TVD) if

$$TV\left(\overline{u}^{n+1}\right) \leq TV\left(\overline{u}^{n}\right), \tag{2.96}$$

total variation bounded (TVB) if

$$TV\left(\overline{u}^{n+1}\right) \leq M, \tag{2.97}$$

for some fixed $M > 0$ and for all n and Δt such that $n\Delta t \leq T$ [117]. For an essentially non-oscillatory (ENO) scheme the following holds (numerically) in the scalar case

$$TV\left(\overline{u}^{n+1}\right) \leq TV\left(\overline{u}^{n}\right) + O\left(h^{p+1}\right), \tag{2.98}$$

for some $p > 0$ [67]. The objective for the development of SSP Runge-Kutta methods is to preserve the non-linear stability properties (TVD, TVB, ENO etc.) of the approximation by higher order space scheme combined with the first order forward Euler (FE) scheme. In other words, if the higher order spatial discretization combined with the FE time discretization is strongly stable (TVD, TVB, ENO) under the CFL restriction

$$\Delta t \leq \Delta t_{FE},$$

then the higher order SSP RK scheme, which is a convex combination of the FE steps, preserves the same strong stability property under a CFL restriction

$$\Delta t \leq c\Delta t_{FE}, \tag{2.99}$$

where c is the SSP coefficient [56]. There is no theoretical proof that the use of SSP RK scheme preserves the non-oscillatory property of the WENO scheme. However, numerical results show that the SSP RK method combined with WENO scheme are non-oscillatory for the same Δt for which the non SSP RK method produces oscillations near discontinuity [56]. For extensive reviews of SSP RK schemes see [57, 58, 56]. Since the initial development for the non-linear scalar hyperbolic equation [120], the class of high order SSP RK(k,m) (k-stages, m-th order) has been successfully employed on a variety of applications including Maxwell's equations [23, 27].

In this thesis the popular second and third order optimal SSP RK schemes are applied to the method of lines semi-discrete system of equations (2.82). Without the source term it can be written as the following system of ODEs

$$\boldsymbol{U}_t = L\boldsymbol{U}, \tag{2.100}$$

where the operator

$$[L\boldsymbol{U}]_i = -\frac{1}{|T_i|}\sum_{j\in\mathscr{I}_i}\left|S_{ij}\right|\boldsymbol{\alpha}_i^{-1}\left[\boldsymbol{\alpha}_i\boldsymbol{T}_{ij}\tilde{\boldsymbol{R}}^{+}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_i\right)\boldsymbol{U}_{ij}+\boldsymbol{\alpha}_j\boldsymbol{T}_{ji}\tilde{\boldsymbol{R}}^{-}\left(\hat{\boldsymbol{n}}_{ij},\boldsymbol{\alpha}_j\right)\boldsymbol{U}_{ji}\right].$$

Denoting the forward Euler step by

$$F\left(\boldsymbol{U},\Delta t\right) = \boldsymbol{U} + \Delta t L\boldsymbol{U}, \tag{2.101}$$

we can present the 2-stage second order optimal SSP Runge-Kutta scheme denoted by SSP RK2 (also known as Heun's method) as

$$\begin{aligned}\boldsymbol{U}^{(0)} =& \boldsymbol{U}^n,\\ \boldsymbol{U}^{(l)} =& F\left(\boldsymbol{U}^{(l-1)},\Delta t\right),\ l=1,2,\\ \boldsymbol{U}^{n+1} =& \frac{1}{2}\boldsymbol{U}^n+\frac{1}{2}\boldsymbol{U}^{(2)}.\end{aligned} \tag{2.102}$$

The 3-stage, third order optimal SSP Runge-Kutta scheme denoted by SSP RK3 can be written as

$$\begin{aligned}\boldsymbol{U}^{(1)} =& F\left(\boldsymbol{U}^n,\Delta t\right),\\ \boldsymbol{U}^{(2)} =& \frac{3}{4}\boldsymbol{U}^n+\frac{1}{4}F\left(\boldsymbol{U}^{(1)},\Delta t\right),\\ \boldsymbol{U}^{n+1} =& \frac{1}{3}\boldsymbol{U}^n+\frac{2}{3}F\left(\boldsymbol{U}^{(2)},\Delta t\right).\end{aligned} \tag{2.103}$$

These methods are optimal, that is they have non-negative coefficients and the optimal SSP coefficient $c=1$ in (2.99).

Since Maxwell's equations are linear, a special class of schemes developed for problems with linear constant coefficient operator L, namely SSP LRK, can also be employed [58]. SSP LRK(k,m) are not m-th order accurate for nonlinear L, and present a useful alternative for time integration of linear problems only if a method of order higher than 3 is needed. Nevertheless, these schemes are considered in the accuracy analysis of multirate schemes in Chapter 5 for completeness.

2.8.2 SSP RK for Maxwell's system with non-zero source term

Since the Maxwell's equations contain a source term, the time discretization schemes need to be slightly modified. Consider the system

$$\boldsymbol{U}_t = L\boldsymbol{U}-\boldsymbol{\sigma}\boldsymbol{U}-\boldsymbol{G}(t). \tag{2.104}$$

The source term $\boldsymbol{G}$ is a given function at any time and space, and the term $\boldsymbol{\sigma U}$ is treated implicitly for numerical stability giving the following Euler scheme

$$\boldsymbol{U}^{n+1} = \boldsymbol{U}^n + \Delta t L\boldsymbol{U}^n - \Delta t \boldsymbol{\sigma U}^{n+1} - \Delta t \boldsymbol{G}(t^n + \Delta t). \tag{2.105}$$

Then the Euler step can be written as

$$F_S(\boldsymbol{U}, \Delta t, t) = (\boldsymbol{I}_6 + \Delta t \boldsymbol{\sigma})^{-1} (\boldsymbol{U} + \Delta t L\boldsymbol{U} - \Delta t \boldsymbol{G}(t + \Delta t)). \tag{2.106}$$

Since any explicit SSP RK scheme is constructed as a convex combination of forward Euler steps we can use the Euler step defined by (2.106) instead of (2.101) in (2.102) or (2.103) to derive the scheme for (2.104). It should be noted that problems with nonzero conductivity σ were not tested in this work.

2.9 Chapter summary

In this chapter a method of lines formulation of FV scheme for Maxwell's equations based on upwind flux splitting was presented. It includes consideration of electromagnetic boundary conditions as well as scattered form of Maxwell's equations. To complete the discretization the spatial reconstructions of fields $\boldsymbol{U}_{ij}$ is needed. This is done by employing the third order WENO scheme which is the subject of Chapter 4. In the next chapter a simple one-dimensional case of the third order WENO reconstruction with application to the linear advection problem is discussed.

Chapter 3

Third order WENO in 1D

In this chapter the third order classic WENO reconstruction for the one-dimensional scalar hyperbolic initial value problem (IVP)

$$u_t + f(u)_x = 0, \tag{3.1}$$

$$u(x,0) = u^0(x), \tag{3.2}$$

is reviewed. WENO is a spatial reconstruction scheme designed for solutions with singularities, in which case simple polynomial reconstructions produce nonphysical solutions. The analysis of accuracy of the WENO3 scheme is presented along with numerical experiments. This analysis is based on the same approach as in [11]. The novelty of this work is that different values of the small parameter ε in the non-linear weights are studied for the third order WENO scheme. The analysis presented in this chapter is crucial to produce an efficient implementation of the three-dimensional WENO scheme, which is discussed in Chapter 4.

3.1 Essentially non-oscillatory finite volume reconstructions overview

It is well known that numerical solutions of hyperbolic problems may develop $O(1)$ spurious oscillations near points of discontinuity (Gibbs-like phenomenon). To design a scheme that produces numerical solution free from nonphysical oscillations, the notion of the total variation of a function (see (2.95)) was developed. Defined in Chapter 2 the scheme is called total variation diminishing (TVD) if

$$TV\left(\overline{u}^{n+1}\right) \leq TV\left(\overline{u}^{n}\right), \tag{3.3}$$

and total variation bounded (TVB) if

$$TV\left(\overline{u}^{n+1}\right) \leq M, \qquad \forall t^{n} \leq T. \tag{3.4}$$

TVD and TVB schemes resolve discontinuities without spurious oscillations. Another advantage of TVD and TVB schemes is that for any sequence of approximations $h \to 0$, $\Delta t = O(h)$ there is a subsequence convergent in L_1^{loc} to a weak solution of the IVP (3.1-3.2). If in addition an entropy condition is satisfied then the scheme is convergent [65]. Examples of TVD schemes are the first order Godunov scheme and the second order MUSCL scheme.

The main disadvantage of high order TVD schemes is that they are at most first order accurate in L_∞-norm and second order accurate in L_1-norm near critical points regardless of the order of the scheme. In [116] Shu proposed a modification of existing TVD schemes that produced TVB schemes of uniformly high-order accuracy. In [67] Harten et al. constructed essentially non-oscillatory (ENO) schemes to overcome the problem of order degeneracy at critical points. Instead of using limiters to overcome possible growth of total variation ENO schemes use adaptive selection of stencils according to the smoothness of u. The essentially non-oscillatory property of the scheme given by [67]

$$TV\left(R\left(x,\overline{u}\right)\right) \leq TV\left(u\right) + O\left(h^{r}\right), \tag{3.5}$$

implies that the reconstruction $R(x,\overline{u})$ does not generate $O(1)$ spurious oscillations, but may occasionally produce $O(h^r)$ spurious oscillations which are on the level of truncation error. Adaptive stencils selection allows resolution of discontinuities with better accuracy than a fixed stencil scheme by selecting the stencil that does not contain the singularity. In this case the smooth part of the solution is not polluted by the error at the discontinuity.

It is not proven that ENO schemes are TVB, but it is uniformly high-order in smooth regions and seems to be very stable from extensive numerical experiments. As it was pointed out in [67] standard linear stability analysis is inappropriate for ENO schemes. This is because ENO schemes are highly nonlinear due to the adaptive selection of stencils for the reconstruction step. In linear analysis one must choose a fixed stencil which may lead to unstable results. In ENO schemes once oscillations begin to form the adaptive selection of stencils reacts by changing the orientation of the stencil and therefore avoids the buildup of instability. In numerical experiments, ENO schemes demonstrate better stability performance than a fixed stencil scheme of the same order, but this stability is not proven analytically.

Weighted essentially non-oscillatory (WENO) schemes were developed to improve the performance of ENO schemes. The pioneering paper on weighted essentially non-oscillatory WENO scheme was presented by Liu, Osher and Chan in [87], where the third order finite volume WENO scheme was designed. The key idea of the WENO scheme is to use weighted combination of all ENO stencils for the reconstruction. The WENO weights (also called non-linear weights) are determined according to smoothness of the solution on each ENO stencil. Instead of adapting the stencil the non-linear weights and therefore the reconstruction are adapted. Advantage of WENO schemes over ENO is that they have better accuracy on the same stencils for smooth solutions. For a thorough review of WENO schemes we refer the reader to Shu's paper [119].

As with ENO schemes, stability of WENO schemes is difficult to prove due to their non-linear nature. Linear stability analysis of the most popular fifth order finite difference WENO scheme can be found in [129, 96, 70]. In [129] Wang and Spiteri applied von Neumann analysis to show that WENO5 scheme coupled with forward Euler or SSP RK2 scheme is linearly unstable. Later in [96] Mohammad et al. using a modified von Neumann analysis showed that WENO5 is linearly stable with either forward Euler or SSP

RK2 method provided that very small (and unpractical) time-steps are taken. In practice, of course, higher order methods such as SSP RK3 are chosen to couple with the fifth order WENO scheme. In this case the scheme is linearly stable with reasonable time-step limits [96]. More results on the CFL number for different combinations of finite difference WENO schemes coupled with various explicit RK schemes of order three and higher can be found in [70]. Since WENO schemes adapt based on the solution smoothness the spurious oscillations expected from linear analysis may be stabilized by non-linear nature of the reconstruction. The ability to stabilize the numerical solution using non-linearity of the reconstruction presumably permits a higher CFL number than the linear stability limit. In this work the classic third order FV WENO scheme coupled with the second and third order SSP RK time integration is considered. These combinations appear to be stable with a reasonable CFL number according to numerical experiments.

3.2 Third order WENO reconstruction

Here we follow the steps outlined in [119] to introduce WENO3 reconstruction on a uniform one-dimensional mesh $a = x_{\frac{1}{2}} < ... < x_{i-\frac{1}{2}} < x_{i+\frac{1}{2}} < x_{i+\frac{3}{2}} < ... < x_{N+\frac{1}{2}}$, with cell centers defined by $x_i = a + \left(i - \frac{1}{2}\right)h$, where h is a given cell size. The averages of the function $u(x)$ over the intervals (cells) $I_i = \left(x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right)$ defined by

$$\overline{u}_i = \frac{1}{h}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} u(x)dx \tag{3.6}$$

are given. The focus of this section is restricted to left-biased reconstructions, but the other direction follows the same framework. In this case the cell averages of $u(x)$ are used to approximate the values $u(x_{i+\frac{1}{2}})$ by third order WENO procedure based on the primitive function. If we define the primitive function of the solution $u(x)$ by

$$U(x) = \int_{x_{j-\frac{1}{2}}}^{x} u(x)dx, \tag{3.7}$$

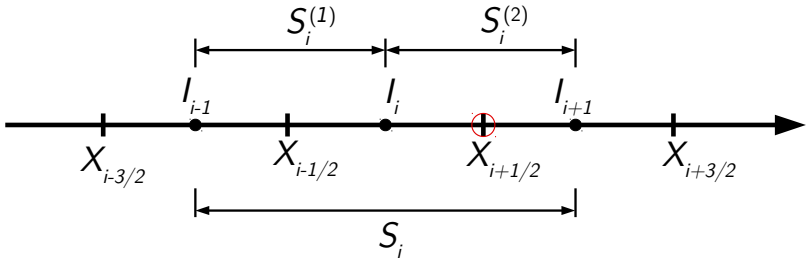


Figure 3.1: Big and small stencils for third order WENO reconstruction at the point $x_{i+\frac{1}{2}}$.

where $j \leq i$ can be arbitrary, then

$$U\left(x_{i+\frac{1}{2}}\right) = \int_{x_{j-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} u(x)dx = \sum_{k=j}^{i} h\overline{u}_k. \tag{3.8}$$

That is, the values of the primitive function $U(x)$ at the points $x_{i+\frac{1}{2}}$ can be found using the cell averages of $u(x)$. Therefore, one can interpolate the primitive function $U(x)$ by a polynomial $P(x)$, and use the derivative of that polynomial $p(x) = \frac{d}{dx}P(x)$ as an approximation of $u(x) = \frac{d}{dx}U(x)$. Note, that this procedure can be carried out on non-uniform meshes as well if we use h_k in place of h.

Let $S_i = \{I_{i-1}, I_i, I_{i+1}\}$ denote the big stencil which is a union of two small stencils $S_i^{(1)} = \{I_{i-1}, I_i\}$ and $S_i^{(2)} = \{I_i, I_{i+1}\}$ (see Figure 3.1). We will use two linear interpolating polynomials $\left\{p_{1,i}^{(l)}(x)\right\}_{l=1}^{2}$ on the small stencils $\left\{S_i^{(l)}\right\}_{l=1}^{2}$ to obtain the third order WENO reconstruction as a convex combination of $p_{1,i}^{(1)}(x)$ and $p_{1,i}^{(2)}(x)$.

We first construct the interpolating polynomial on the big stencil S_i. We will use it to derive the coefficients of the linear reconstruction. Let $P_{3,i}(x)$ be the polynomial of degree at most three which interpolates the function $U(x)$ at the four points $x_{j+\frac{1}{2}}$, $j = i-2, ..., i+1$, that is

$$U\left(x_{j+\frac{1}{2}}\right) = P_{3,i}\left(x_{j+\frac{1}{2}}\right), \quad j = i-2, ..., i+1.$$

Then $p_{2,i}(x) = \frac{d}{dx}P_{3,i}(x)$ is the unique polynomial of degree at most two that approximates the function $u(x) = \frac{d}{dx}U(x)$ on S_i in the sense that

$$\frac{1}{h}\int_{I_j} p_{2,i}(x)dx = \overline{u}_j, \quad j = i-1, ..., i+1. \tag{3.9}$$

The approximations $u_{i+\frac{1}{2}} \equiv p_{2,i}\left(x_{i+\frac{1}{2}}\right)$ are third order accurate,

$$u_{i+\frac{1}{2}} = u\left(x_{i+\frac{1}{2}}\right) + O\left(h^3\right), \tag{3.10}$$

if the function $u(x)$ is smooth in the stencil S_i, i.e. on the interval $\left[x_{i-\frac{3}{2}}, x_{i+\frac{3}{2}}\right]$. To derive the explicit formula for $p_{2,i}(x)$ consider the local coordinate $\xi = \xi(x) = \frac{x-x_i}{h}$, then the polynomial $p_{2,i}(x)$ can be written as

$$p_{2,i}(x) = a_{0,i} + a_{1,i}\xi + a_{2,i}\xi^2, \tag{3.11}$$

where

$$a_{k,i} = a_{k,i}\left(\overline{u}\right), \; k = 0, 1, 2. \tag{3.12}$$

Using (3.9) with $j = i$ one can get

$$\overline{u}_i = \frac{1}{h}\int_{I_i} p_{2,i}(x)dx = a_{0,i} + a_{1,i}\overline{[\xi]}_i + a_{2,i}\overline{[\xi^2]}_i, \tag{3.13}$$

where

$$\overline{[\xi]}_i = \frac{1}{h}\int_{I_i} \xi(x)dx = 0, \qquad \overline{[\xi^2]}_i = \frac{1}{h}\int_{I_i} \xi^2(x)dx = \frac{1}{12}.$$

Substituting (3.13) in (3.11)

$$p_{2,i}(x) = \overline{u}_i + a_{1,i}\left(\xi - \overline{[\xi]}_i\right) + a_{2,i}\left(\xi^2 - \overline{[\xi^2]}_i\right). \tag{3.14}$$

Now using (3.9) with $j = i-1, i+1$ the following 2×2 system is derived

$$\boldsymbol{A}\boldsymbol{a} = \boldsymbol{u},$$

where $\boldsymbol{a} = [a_{1,i}, a_{2,i}]^T$ and $\boldsymbol{u} = [\overline{u}_{i-1} - \overline{u}_i, \overline{u}_{i+1} - \overline{u}_i]^T$, and

$$\boldsymbol{A} = \begin{bmatrix} \overline{[\xi]}_{i-1} - \overline{[\xi]}_i & \overline{[\xi^2]}_{i-1} - \overline{[\xi^2]}_i \\ \overline{[\xi]}_{i+1} - \overline{[\xi]}_i & \overline{[\xi^2]}_{i+1} - \overline{[\xi^2]}_i \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}.$$

Then $\boldsymbol{a} = \boldsymbol{A}^{-1}\boldsymbol{u}$ gives

$$a_{1,i}(\overline{u}) = -\frac{1}{2}\overline{u}_{i-1} + \frac{1}{2}\overline{u}_{i+1}, \tag{3.15}$$

$$a_{2,i}(\overline{u}) = \frac{1}{2}\overline{u}_{i-1} - \overline{u}_i + \frac{1}{2}\overline{u}_{i+1}. \tag{3.16}$$

Using (3.14) and (3.15-3.16) the following approximation of u at $x_{i+\frac{1}{2}}$ ($\xi = \frac{1}{2}$) is obtained

$$u_{i+\frac{1}{2}} \equiv p_{2,i}\left(x_{i+\frac{1}{2}}\right) = -\frac{1}{6}\overline{u}_{i-1} + \frac{5}{6}\overline{u}_i + \frac{1}{3}\overline{u}_{i+1}. \tag{3.17}$$

It should be noted that the procedure above does not require a uniform mesh and can also be used to derive the reconstruction polynomials in 2 and 3 space dimensions [73, 135] on unstructured meshes.

Reconstructions on two small stencils $S_i^{(l)} = \{I_{i-2+l}, I_{i-1+l}\}$, $l = 1,2$, are derived in the same way as above. Let $P_{2,i}^{(l)}(x)$ be polynomials of degree at most two which interpolate the function $U(x)$ at three points $x_{j+\frac{1}{2}}$, $j = i-3+l, ..., i-1+l$, that is

$$U\left(x_{j+\frac{1}{2}}\right) = P_{2,i}^{(l)}\left(x_{j+\frac{1}{2}}\right), \quad j = i-3+l, ..., i-1+l, \quad l = 1,2.$$

Then $p_{1,i}^{(l)}(x) = \frac{d}{dx}P_{2,i}^{(l)}(x)$ are unique polynomials of degree at most one that approximate the function $u(x) = \frac{d}{dx}U(x)$ on $S_i^{(l)}$ in the sense that

$$\frac{1}{h}\int_{I_j} p_{1,i}^{(l)}(x)dx = \overline{u}_j, \quad j = i-2+l, ..., i-1+l, \quad l = 1,2. \tag{3.18}$$

The linear polynomials $p_{1,i}^{(1)}(x)$ and $p_{1,i}^{(2)}(x)$ using local coordinate $\xi = \xi(x) = \frac{x-x_i}{h}$ can be written as

$$p_{1,i}^{(l)}(x) = a_{0,i}^{(l)} + a_{1,i}^{(l)}\xi, \qquad l = 1,2. \tag{3.19}$$

Using (3.18) the linear polynomials $p_{1,i}^{(l)}(x)$ can be written as

$$p_{1,i}^{(1)}(x) = \overline{u}_i + \left[-\overline{u}_{i-1} + \overline{u}_i\right]\xi, \tag{3.20}$$

$$p_{1,i}^{(2)}(x) = \overline{u}_i + \left[-\overline{u}_i + \overline{u}_{i+1}\right]\xi. \tag{3.21}$$

At the point $x_{i+\frac{1}{2}}$ the following second order approximations in terms of cell averages are obtained from (3.20-3.21)

$$u_{i+\frac{1}{2}}^{(1)} \equiv p_{1,i}^{(1)}\left(x_{i+\frac{1}{2}}\right) = -\frac{1}{2}\overline{u}_{i-1} + \frac{3}{2}\overline{u}_i, \tag{3.22}$$

$$u_{i+\frac{1}{2}}^{(2)} \equiv p_{1,i}^{(2)}\left(x_{i+\frac{1}{2}}\right) = \frac{1}{2}\overline{u}_i + \frac{1}{2}\overline{u}_{i+1}. \tag{3.23}$$

The approximations $u_{i+\frac{1}{2}}^{(l)}$, $l = 1,2$, are second order accurate for smooth functions $u(x)$,

$$u_{i+\frac{1}{2}}^{(l)} = u\left(x_{i+\frac{1}{2}}\right) + O\left(h^2\right), \quad l = 1,2. \tag{3.24}$$

The third order approximation $u_{i+\frac{1}{2}}$ defined by (3.17) can be written as a linear convex combination of $u_{i+\frac{1}{2}}^{(1)}$ and $u_{i+\frac{1}{2}}^{(2)}$ defined by (3.22) and (3.23), respectively,

$$u_{i+\frac{1}{2}} = \gamma_1 u_{i+\frac{1}{2}}^{(1)} + \gamma_2 u_{i+\frac{1}{2}}^{(2)}, \tag{3.25}$$

where the linear weights are given by $\gamma_1 = \frac{1}{3}$ and $\gamma_2 = \frac{2}{3}$. The reconstruction (3.25) is often called the linear reconstruction in the WENO literature, because the weights γ_l are constant linear weights that depend on the geometry only. The idea of WENO is to write the final approximation as a convex combination so-called non-linear weights $\omega_{l,i}$ that depend on the function $u(x)$

$$u_{i+\frac{1}{2}}^{WENO} = \omega_{1,i} u_{i+\frac{1}{2}}^{(1)} + \omega_{2,i} u_{i+\frac{1}{2}}^{(2)}. \tag{3.26}$$

In addition to the condition

$$\omega_{1,i} + \omega_{2,i} = 1, \tag{3.27}$$

the following properties of the non-linear weights $\omega_{l,i}$ are required [119]

1. $\omega_{l,i} \approx \gamma_l$ for all l if $u(x)$ is smooth on the big stencil S_i;
2. $\omega_{l,i} \approx 0$ if $u(x)$ has a discontinuity in the small stencil $S_i^{(l)}$, but is smooth in $S_i \setminus S_i^{(l)}$.

The first property gives the accuracy of linear approximation for smooth $u(x)$, this is formulated in the following theorem:

Theorem 1. (Jiang and Shu [78]) *Let $u(x)$ be a smooth function on the big stencil S_i, then the WENO3 reconstruction (3.26) is up to the third order accurate, that is*

$$u_{i+\frac{1}{2}}^{WENO} = u\left(x_{i+\frac{1}{2}}\right) + O\left(h^{2+k}\right) \tag{3.28}$$

provided that

$$\omega_{l,i} = \gamma_l + O\left(h^k\right), \quad l = 1,2, \tag{3.29}$$

where $0 \leq k \leq 1$.

Proof. When the function $u(x)$ is smooth on the big stencil S_i the accuracy results (3.24) and (3.10) hold. Therefore, comparing (3.25) and (3.26) we get

$$\begin{aligned} u_{i+\frac{1}{2}}^{WENO} &= u_{i+\frac{1}{2}} + \sum_{l=1}^{2} \left(\omega_{l,i} - \gamma_l\right) u_{i+\frac{1}{2}}^{(l)} \\ &= u\left(x_{i+\frac{1}{2}}\right) + O\left(h^3\right) + \sum_{l=1}^{2} \left(\omega_{l,i} - \gamma_l\right) \left(u\left(x_{i+\frac{1}{2}}\right) + O\left(h^2\right)\right) \\ &= u\left(x_{i+\frac{1}{2}}\right) + O\left(h^3\right) + \sum_{l=1}^{2} \left(\omega_{l,i} - \gamma_l\right) O\left(h^2\right). \end{aligned}$$

Hence, the reconstruction (3.26) is third order accurate whenever $\omega_{l,i} - \gamma_l = O(h)$. □

Now assuming that $u(x)$ is smooth in $S_i^{(l)}$ and is singular in $S_i \setminus S_i^{(l)}$, the requirement that $\omega_{i,l} \approx 0$ guarantees up to second order accuracy of the WENO reconstruction (3.26), as the contribution from the small stencil containing discontinuity of $u(x)$ has almost zero

weight. Note, that if the singular point lies inside the interval I_i, then both small stencils contain a discontinuity.

Now we turn to a more precise definition of non-linear weights. The weights $\omega_{l,i}$ are defined as a function of linear weights γ_l and the so-called smoothness indicators $SI_{l,i}$ which estimate how smooth the function $u(x)$ is in the small stencils $S_i^{(l)}$. The weights $\omega_{l,i}$ are normalized to satisfy $\sum_{l=1}^{2} \omega_{l,i} = 1$. The classic WENO weights are defined as follows [87, 78]

$$\omega_{l,i} = \frac{\tilde{\omega}_{l,i}}{\tilde{\omega}_{1,i} + \tilde{\omega}_{2,i}}, \quad \text{with} \quad \tilde{\omega}_{l,i} = \frac{\gamma_l}{\left(\varepsilon + SI_{l,i}\right)^p},\ l = 1,2, \tag{3.30}$$

where $\varepsilon > 0$ is used to avoid a situation with zero denominator and is usually chosen to be $\varepsilon = 10^{-6}$, and p is usually chosen to be equal to 2 which is employed in this work. Smoothness indicators play the key role in WENO schemes. In most cases the indicators proposed by Jiang and Shu in [78] are used with great success. In the case when $p_{k,i}^{(l)}(x)$ are linear functions ($k = 1$), their smoothness indicators have the form

$$SI_{l,i} = h \int_{I_i} \left(\frac{d}{dx} p_{1,i}^{(l)}(x) \right)^2 dx = h \int_{I_i} \left(\frac{a_{1,i}^{(l)}}{h} \right)^2 dx = \left(a_{1,i}^{(l)} \right)^2, \tag{3.31}$$

or

$$SI_1 = \left(-\overline{u}_{i-1} + \overline{u}_i\right)^2, \tag{3.32}$$

$$SI_2 = \left(-\overline{u}_i + \overline{u}_{i+1}\right)^2. \tag{3.33}$$

3.2.1 Accuracy analysis

The choice of ε in (3.30) affects the overall accuracy of the reconstruction. For one-dimensional scalar problems $\varepsilon = 10^{-2}$ gives good results for smooth solutions, since the nonlinear weights are closer to linear weights. However, for a piecewise smooth solution, the same choice will produce oscillations. Therefore a very small $\varepsilon \leq 10^{-6}$ is used in most cases to ensure a non-oscillatory solution at discontinuities, but this does not guarantee the accuracy for a given mesh size h. The effect of the choice of ε was studied in [11, 69]. In [11] it was shown that dependence on h is crucial for the performance of the classic WENO

reconstruction with the best results demonstrated for $\varepsilon \sim h^2$. This can be explained by the fact that on stencils $S_i^{(l)}$ with a smooth solution $SI_{l,i} \sim O\left(h^2\right)$ in most cases and on stencils with discontinuity $SI_{l,i} \sim O(1)$. Therefore, if the solution is continuous on the big stencil S_i one can benefit from a big ε since the dominance of ε over $SI_{l,i}$ will provide $\omega_{l,i} \approx \gamma_{l,i}$ in this case. On the other hand, on S_i with a discontinuity it is expected that $\omega_{l_1,i} \approx 0$ on $S_i^{(l_1)}$ containing singularity and $\omega_{l_2,i} \approx 1$ on the stencil with smooth solution. This is achieved when ε is not bigger than $SI_{l_2,i}$ on the stencil without singularity. The choice $\varepsilon \sim O\left(h^2\right)$, as it was suggested in [11], seems to be the best compromise for performance of the classic WENO3 in both smooth and non-smooth cases for an arbitrary h. Inspired by the analysis from [11] we study the influence of the choice of $\varepsilon = h^k$, $k = 1,2,...,4$, in the context of a classical WENO3 reconstruction. The following theorem is a special case of the result presented in [11] for smoothness indicators (3.32-3.33)

Theorem 2. *Let $u(x) \in C^3$ on the big stencil S_i. Then the smoothness indicators (3.32-3.33) have the following properties*

1. *If $u'(x) \neq 0$* for all $x \in S_i$, then

$$SI_{l,i} = \alpha(x_i) h^2 + O\left(h^3\right), \qquad l \in \{1,2\} \tag{3.34}$$

for some locally Lipschitz continuous $\alpha(x)$, and

$$SI_{l_1,i} - SI_{l_2,i} = \beta_{l_1,l_2}(x_i) h^3 + O\left(h^4\right), \qquad l_1 \neq l_2,\ l_1,l_2 \in \{1,2\} \tag{3.35}$$

for some locally Lipschitz continuous $\beta_{l_1,l_2}(x)$.

2. *If $u(x)$ has a point $x^* \in S_i \setminus \{x_i\}$ such that $u'(x^*) = 0$*, then

$$SI_{l,i} = \alpha_l(x_i) h^4 + O\left(h^5\right), \tag{3.36}$$

and

$$SI_{l_1,i} - SI_{l_2,i} = \beta_{l_1,l_2}(x_i) h^4 + O\left(h^5\right), \tag{3.37}$$

for some locally Lipschitz continuous $\alpha_l(x)$ and $\beta_{l_1,l_2}(x)$ with $l_1 \neq l_2$, $l_1,l_2 \in \{1,2\}$.

Proof. 1. We start by rewriting the smoothness indicators given by (3.32-3.33) in terms of the primitive function $U(x)$ as follows

$$\begin{aligned} SI_{l,i} &= \left(-\overline{u}_{i+l-2}+\overline{u}_{i+l-1}\right)^2 \\ &= \left(-\frac{U\left(x_i+\left(l-\frac{3}{2}\right)h\right)-U\left(x_i+\left(l-\frac{5}{2}\right)h\right)}{h}\right. \\ &\quad \left.+\frac{U\left(x_i+\left(l-\frac{1}{2}\right)h\right)-U\left(x_i+\left(l-\frac{3}{2}\right)h\right)}{h}\right)^2. \end{aligned}$$

Since $u(x)$ is smooth in S_i the Taylor series of $U(x)$ can be used to expand it about x_i. As a result we get

$$SI_{l,i} = \left(U''(x_i)\,h+\left(l-\frac{3}{2}\right)U'''(x_i)\,h^2+O\left(h^3\right)\right)^2$$

or

$$SI_{l,i} = \left(u'(x_i)\,h+\left(l-\frac{3}{2}\right)u''(x_i)\,h^2+O\left(h^3\right)\right)^2. \tag{3.38}$$

Using the notation $\alpha(x_i) = \left[u'(x_i)\right]^2$ the statement (3.34) is derived.

Now using (3.38) for $S_i^{(l_1)}$ and $S_i^{(l_2)}$ one can obtain

$$SI_{l_1,i} - SI_{l_2,i} = 2\left(l_1-l_2\right)u'(x_i)\,u''(x_i)\,h^3+O\left(h^4\right). \tag{3.39}$$

With the notation $\beta_{l_1,l_2}(x_i) = 2\left(l_1-l_2\right)u'(x_i)\,u''(x_i)$ the above result gives the estimate (3.35).

2. Consider the case when $u'(x^*)=0$, for some $x^*\in S_i\setminus\{x_i\}$. Let $x_i-x^*=\kappa h$ with $0<|\kappa|<\frac{3}{2}$. Then using the Taylor series of $u'(x)$ about x_i at x^* one can get

$$u'(x^*) = u'(x_i)-u''(x_i)\,\kappa h+O\left(h^2\right),$$

and hence

$$u'(x_i) = u''(x_i)\,\kappa h+O\left(h^2\right). \tag{3.40}$$

Inserting (3.40) to (3.38) the following is obtained

$$SI_{l,i} = \left(\left(\kappa + l - \frac{3}{2} \right) u''(x_i) h^2 + O\left(h^3\right) \right)^2. \tag{3.41}$$

Therefore with $\alpha_l(x_i) = \left[\left(\kappa + l - \frac{3}{2} \right) u''(x_i) \right]^2$ the estimate (3.36) is proven.

Now using (3.40) in (3.39) one can get

$$SI_{l_1,i} - SI_{l_2,i} = 2\kappa (l_1 - l_2) \left[u''(x_i) \right]^2 h^4 + O\left(h^5\right). \tag{3.42}$$

Therefore with $\beta_{l_1,l_2}(x_i) = 2\kappa (l_1 - l_2) \left[u''(x_i) \right]^2$ the last result (3.37) is obtained. □

Remark 1. If $x^* = x_i$ then $SI_{l_1,i} - SI_{l_2,i} = O\left(h^5\right)$.

It should be noted that results similar to the ones in Theorem 2 can be proved for non-uniform meshes as well (see Appendix A for details).

Theorem 3. *Let $u(x) \in C^3$ on the big stencil S_i, and $\varepsilon = Mh^k$, for some $M > 0$ and $k \geq 0$. Then*

1. *If $u'(x) \neq 0$* for all $x \in S_i$, then

$$u_{i+\frac{1}{2}}^{WENO} - u\left(x_{i+\frac{1}{2}}\right) = O\left(h^3\right). \tag{3.43}$$

2. *If there is a point $x^* \in S_i \setminus \{x_i\}$ such that $u'(x^*) = 0$*, then

$$u_{i+\frac{1}{2}}^{WENO} - u\left(x_{i+\frac{1}{2}}\right) = \begin{cases} O\left(h^3\right), & k < 4, \\ O\left(h^2\right), & k \geq 4. \end{cases} \tag{3.44}$$

Proof. As in [11] we start by writing

$$\frac{1}{\left(\varepsilon + SI_{l_1,i}\right)^2} = \frac{1}{\left(\varepsilon + SI_{l_2,i}\right)^2} \left(1 + \frac{SI_{l_2,i} - SI_{l_1,i}}{\varepsilon + SI_{l_1,i}} \right)^2. \tag{3.45}$$

1. Consider the case when $u'(x) \neq 0$ for all $x \in S_i$. Then using (3.34) and (3.35)

$$\frac{SI_{l_2,i}-SI_{l_1,i}}{\varepsilon+SI_{l_1,i}}=\frac{\beta_{l_2,l_1}(x_i)\,h^{3-\min(2,k)}}{\eta_1 M+\eta_2\alpha(x_i)}+O\left(h^{4-\min(2,k)}\right), \tag{3.46}$$

where

$$\eta_1=\begin{cases}1, & k\le 2,\\ 0, & k>2,\end{cases}\qquad\text{and}\qquad \eta_2=\begin{cases}0, & k<2,\\ 1, & k\ge 2.\end{cases}$$

Using (3.46) in (3.45) one can derive

$$\frac{1}{\left(\varepsilon+SI_{l_1,i}\right)^2}=\frac{1}{\left(\varepsilon+SI_{l_2,i}\right)^2}\left(1+\nu_{l_1}(x_i)\,h^r+O\left(h^{r+1}\right)\right),$$

where $r=\max(1,3-k)\ge 1$ and $\nu_{l_1}(x_i)=\frac{2\beta_{l_2,l_1}(x_i)}{\eta_1 M+\eta_2\alpha(x_i)}$ is a locally Lipschitz continuous function. Then

$$\tilde{\omega}_{l_1,i}=\frac{\gamma_{l_1}}{\left(\varepsilon+SI_{l_2,i}\right)^2}\left(1+\nu_{l_1}(x_i)\,h^r+O\left(h^{r+1}\right)\right),$$

and

$$\begin{aligned}\tilde{\omega}_{l_1,i}+\tilde{\omega}_{l_2,i}&=\frac{\gamma_{l_1}}{\left(\varepsilon+SI_{l_2,i}\right)^2}\left(1+\nu_{l_1}(x_i)\,h^r+O\left(h^{r+1}\right)\right)+\frac{\gamma_{l_2}}{\left(\varepsilon+SI_{l_2,i}\right)^2}\\&=\frac{1}{\left(\varepsilon+SI_{l_2,i}\right)^2}\left(1+\gamma_{l_1}\nu_{l_1}(x_i)\,h^r+O\left(h^{r+1}\right)\right).\end{aligned}$$

As a result the non-linear weight for the small stencil $S_i^{(l_2)}$ is

$$\omega_{l_2,i}=\frac{\gamma_{l_2}}{\left(1+\gamma_{l_1}\nu_{l_1}(x_i)\,h^r+O\left(h^{r+1}\right)\right)}=\gamma_{l_2}\left(1-\gamma_{l_1}\nu_{l_1}(x_i)\,h^r+O\left(h^{r+1}\right)\right).$$

Following the same steps for $\omega_{l_1,i}$ one can get that

$$\omega_{l,i} = \gamma_l \left(1 + O\left(h^r\right)\right), \quad l = 1,2, \quad r \geq 1. \tag{3.47}$$

Then from the Theorem 1 it follows that

$$u_{i+\frac{1}{2}}^{WENO} - u\left(x_{i+\frac{1}{2}}\right) = O\left(h^3\right)$$

on stencils S_i without critical points regardless of the value of ε.

2. Now assume that at some point $x^* \in S_i \setminus \{x_i\}$, we have $u'(x^*) = 0$. Then

$$\frac{SI_{l_2,i} - SI_{l_1,i}}{\varepsilon + SI_{l_1,i}} = \frac{\beta_{l_2,l_1}(x_i)\, h^{4-\min(4,k)}}{\eta_1 M + \eta_2 \alpha_{l_1}(x_i)} + O\left(h^{5-\min(4,k)}\right),$$

where

$$\eta_1 = \begin{cases} 1, & k \leq 4, \\ 0, & k > 4, \end{cases} \qquad \text{and} \qquad \eta_2 = \begin{cases} 0, & k < 4, \\ 1, & k \geq 4. \end{cases}$$

Following the same steps for $k \leq 3$, we get (3.47) with $r = \max(0, 4-k) \geq 0$. For $k > 3$

$$\frac{SI_{l_2,i} - SI_{l_1,i}}{\varepsilon + SI_{l_1,i}} = \frac{\beta_{l_2,l_1}(x_i)}{\eta_1 M + \alpha_{l_1}(x_i)} + O(h).$$

Therefore on $S_i^{(l_2)}$ the non-linear weight is

$$\omega_{l_2,i} = \frac{\gamma_{l_2}}{(1 + \gamma_{l_1} \nu_{l_1}(x_i) + O(h))} = \frac{\gamma_{l_2}}{(1 + \gamma_{l_1} \nu_{l_1}(x_i))} + O(h).$$

Hence

$$\omega_{l_2,i} - \gamma_{l_2} = \frac{\gamma_{l_2}}{(1 + \gamma_{l_1} \nu_{l_1}(x_i))} - \gamma_{l_2} + O(h) = O(1).$$

The same result can be obtained for $S_i^{(l_1)}$. Thus for $k > 3$ WENO3 gives only second order reconstruction near the critical point x^*. □

Remark. The results of the Theorem 3 can also be extended to the arbitrary choice of p in the definition of non-linear weights (3.30) using the same proof. Therefore, the convergence of the solution by WENO scheme is affected by k in $\varepsilon = h^k$ but not by the choice of p in (3.30).

The results of Theorem 3 show that the accuracy of WENO3 scheme for smooth solutions diminishes near critical points if ε is very small. It also demonstrates that the smaller the value k in $\varepsilon = h^k$ the closer the non-linear weights are to linear weights. On the other hand, very small ε is needed to avoid spurious oscillations near discontinuities. The next theorem shows that WENO3 scheme is second order accurate near discontinuity for $\varepsilon = h^k$, $k \geq 1$.

Theorem 4. *Let $u(x)$ be a piecewise smooth function with a jump discontinuity $[u^*] = [u(x^*)]$ in $S_i \setminus S_i^{(l)}$, $l \in \{1,2\}$, at the point x^*. If ε in (3.30) is defined as $\varepsilon = Mh^k$, for some $k \geq 1$, the reconstruction (3.26) gives*

$$u_{i+\frac{1}{2}}^{WENO} = u\left(x_{i+\frac{1}{2}}\right) + O\left(h^2\right). \tag{3.48}$$

Proof. If $x^* \in S_i \setminus S_i^{(l_2)}$, then *(3.24) holds for l_2, while for l_1 we have

$$u_{i+\frac{1}{2}}^{(l_1)} = u\left(x_{i+\frac{1}{2}}\right) + O\left([u^*]\right). \tag{3.49}$$

Then (3.26) gives

$$\begin{aligned} u_{i+\frac{1}{2}}^{WENO} =& \omega_{l_1,i} u_{i+\frac{1}{2}}^{(l_1)} + \omega_{l_2,i} u_{i+\frac{1}{2}}^{(l_2)} \\ =& u\left(x_{i+\frac{1}{2}}\right) + \omega_{l_1,i}\left(O\left([u^*]\right)\right) + O\left(h^2\right). \end{aligned} \tag{3.50}$$

Since $SI_{l_1,i} = O\left([u^*]^2\right)$ and $SI_{l_2,i} = O\left(h^m\right)$, $m \in \{2,4\}$, for $\varepsilon = O\left(h^k\right)$ we get

$$\omega_{l,i} = \begin{cases} O\left(h^{2\min(k,m)}\left[u^*\right]^{-4}\right), & l = l_1, \\ O(1), & l = l_2. \end{cases} \tag{3.51}$$

Therefore, from (3.50) it follows that

$$u_{i+\frac{1}{2}}^{WENO} = u\left(x_{i+\frac{1}{2}}\right) + O\left(h^{2\min(k,m)}\left[u^*\right]^{-3}\right) + O\left(h^2\right),$$

which for $k > 0$ gives (3.48). □

Remark 2. Note that while WENO3 is second order accurate near discontinuity with $k = 1$ in $\varepsilon = h^k$, it still produces noticeable oscillations. Therefore values $k \leq 2$ are needed to control oscillations near discontinuities.

3.2.2 Mapping technique

The accuracy of classic WENO scheme was addressed in the literature with different weight designs and constraints on ε [69, 19, 131, 12]. Most of these solutions can not be easily adapted to WENO schemes on multidimensional unstructured meshes. Here we would like to discuss the mapping technique by Henrick et al. in [69] (referred to as WENO3M), since it is straightforward to implement for any WENO scheme on unstructured meshes. The mapping technique was proposed to improve the accuracy of the fifth order classic WENO scheme at critical points. Using the classic WENO weight as an initial guess, a more accurate weight can be obtained using the functions

$$g_l(\omega) = \frac{\omega\left(\gamma_l + \gamma_l^2 - 3\gamma_l\omega + \omega^2\right)}{\gamma^2 + \omega\left(1 - 2\gamma_l\right)}, \tag{3.52}$$

which are monotonically increasing with the properties

1. $0 \leq g_l(\omega) \leq 1$.
2. $g_l(0) = 0$, and $g_l(1) = 1$.
3. $g_l(\gamma_l) = \gamma_l$, $g_l'(\gamma_l) = g_l''(\gamma_l) = 0$.

The new non-linear weights are defined by

$$\omega_l^M = \frac{\tilde{\omega}_l^M}{\sum_{m=1}^{s}\tilde{\omega}_m^M}, \ \tilde{\omega}_l^M = g_l(\omega_l). \tag{3.53}$$

Using the evaluation at ω_l of the Taylor series of $g_l(\omega)$ about γ_l given by

$$\tilde{\omega}_l^M = g_l(\gamma_l) + g_l'(\gamma_l)(\omega_l - \gamma_l) + \frac{g_l''(\gamma_l)}{2}(\omega_l - \gamma_l)^2 + \frac{g_l'''(\gamma_l)(\omega_l - \gamma_l)^3}{6} + ... = \gamma_l + \frac{(\omega_l - \gamma_l)^3}{\gamma_l - \gamma_l^3} + ..., \tag{3.54}$$

it is easy to show that if $\omega_l = \gamma_l + O(h)$, then $\omega_l^M = \gamma_l + O\left(h^3\right)$, where h is defined by (4.13). Therefore this technique works well for the fifth order WENO, because $\omega_l = \gamma_l + O(h)$ at critical points in this case regardless of the value of ε. For WENO3 the estimate is $\omega_l = \gamma_l + O(1)$ for very small ε and $\omega_l = \gamma_l + O(h)$ otherwise. Therefore, the above mapping does not improve convergence of the third order scheme.

3.3 Numerical experiments

In this section the results of numerical solution of one-dimensional scalar advection equation

$$u_t + u_x = 0, \quad x \in (-1,1),\ t > 0, \tag{3.55}$$

with initial conditions

$$u(x,0) = u^0(x), \quad x \in [-1,1], \tag{3.56}$$

and periodic boundary conditions $u(-1,t) = u(1,t)$ are presented. The exact solution to this problem is found by method of characteristics as

$$u(x,t) = u^0(x-t). \tag{3.57}$$

Consider the mesh $\Omega = \cup_{i=1}^N I_i$ of size $N = 2/h$, where $I_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ with $x_{i-\frac{1}{2}} = -1 + (i-1)h$, $x_{i+\frac{1}{2}} = -1 + ih$, and $x_i = -1 + \left(i - \frac{1}{2}\right)h$, $i = 1,...,N$. The FV scheme for (3.55) on Ω is

$$\frac{d\overline{u}_i}{dt} + \frac{1}{h}\left[\hat{u}_{i+\frac{1}{2}} - \hat{u}_{i-\frac{1}{2}}\right] = 0, \tag{3.58}$$

where $\hat{u}_{i+\frac{1}{2}}$ and $\hat{u}_{i-\frac{1}{2}}$ are obtained using WENO3 reconstruction (3.26) as

$$\hat{u}_{i+\frac{1}{2}} = u^{WENO}_{i+\frac{1}{2}} = \omega_{1,i}\left(-\frac{1}{2}\overline{u}_{i-1} + \frac{3}{2}\overline{u}_i\right) + \omega_{2,i}\left(\frac{1}{2}\overline{u}_i + \frac{1}{2}\overline{u}_{i+1}\right), \tag{3.59}$$

$$\hat{u}_{i-\frac{1}{2}} = u^{WENO}_{i-\frac{1}{2}} = \omega_{1,i-1}\left(-\frac{1}{2}\overline{u}_{i-2} + \frac{3}{2}\overline{u}_{i-1}\right) + \omega_{2,i-1}\left(\frac{1}{2}\overline{u}_{i-1} + \frac{1}{2}\overline{u}_i\right), \tag{3.60}$$

with non-linear coefficients defined by

$$\omega_{l,i} = \frac{\tilde{\omega}_{l,i}}{\tilde{\omega}_{1,i} + \tilde{\omega}_{2,i}}, \quad \text{with} \quad \tilde{\omega}_{l,i} = \begin{cases} \frac{1}{3\left(\varepsilon + (-\overline{u}_{i-1} + \overline{u}_i)^2\right)^2}, & l = 1, \\ \frac{2}{3\left(\varepsilon + (-\overline{u}_i + \overline{u}_{i+1})^2\right)^2}, & l = 2. \end{cases}$$

and

$$\omega_{l,i-1} = \frac{\tilde{\omega}_{l,i-1}}{\tilde{\omega}_{1,i-1} + \tilde{\omega}_{2,i-1}}, \quad \text{with} \quad \tilde{\omega}_{l,i-1} = \begin{cases} \frac{1}{3\left(\varepsilon + (-\overline{u}_{i-2} + \overline{u}_{i-1})^2\right)^2}, & l = 1, \\ \frac{2}{3\left(\varepsilon + (-\overline{u}_{i-1} + \overline{u}_i)^2\right)^2}, & l = 2. \end{cases}$$

We also consider WENO3M by transforming the non-linear weights above using (3.53). The method of lines ODE (3.58) is solved using the optimal SSP RK3 method discussed in Chapter 2.

First consider the initial function

$$u^0(x) = \sin(\pi x), \tag{3.61}$$

which has two extrema in $[-1,1]$. The purpose of this test is to confirm the theoretical results on the accuracy of WENO3 scheme for various values of ε. We compare the numerical errors in L^1, L^2 and L^∞ at a given time $t = T$ which can be obtained by

$$l_q(u) = \begin{cases} \left(\sum_i h \left| u^{WENO}_{i+\frac{1}{2}} - u^{exact}\left(x_{i+\frac{1}{2}}\right)\right|^q\right)^{\frac{1}{q}}, & q = 1,2 \\ \max_i \left| u^{WENO}_{i+\frac{1}{2}} - u^{exact}\left(x_{i+\frac{1}{2}}\right)\right|, & q = \infty, \end{cases}, \tag{3.62}$$

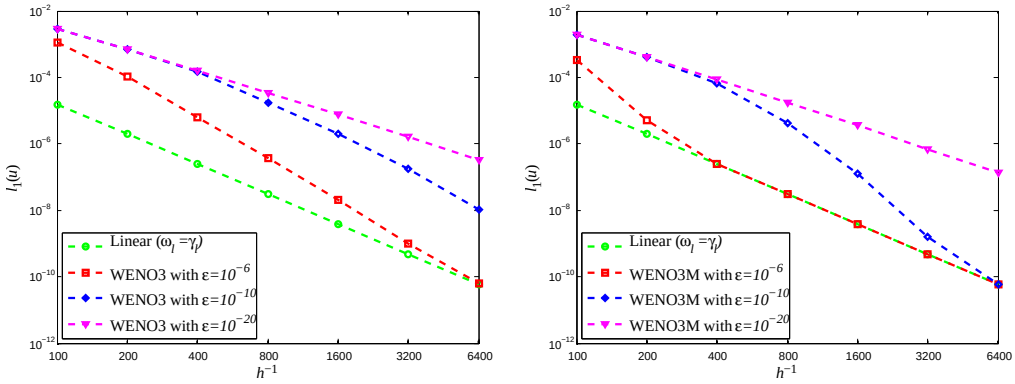


Figure 3.2: L^1 error for the solution of the linear advection equation with initial data $u^0(x) = \sin(\pi x)$ at $T = 1, CFL = 1$ using WENO3 and WENOM3 schemes with various fixed ε.

where $u^{exact}(x) = u^0(x-T)$. The order of the error in L^q is estimated from the two numerical solutions $u^{[h/2]}$ and $u^{[h]}$ with different resolutions by

$$r_q(u) = \frac{\log\left(l_q\left(u^{[h]}\right)/l_q\left(u^{[h/2]}\right)\right)}{\log(2)} = \log_2\left(l_q\left(u^{[h]}\right)/l_q\left(u^{[h/2]}\right)\right). \tag{3.63}$$

Figure 3.2 shows the error for both WENO3 and WENO3M scheme with different fixed values of ε. Only the L^1 errors are shown here, but the results for L^2 and L^∞ are similar. This experiment demonstrates that the rate of convergence of the error is not uniform when ε is fixed. Therefore it is hard to predict the error order for a given grid resolution. As expected, the mapped WENO scheme, which showed a significant improvement in accuracy of the fifth order scheme with an arbitrary fixed ε [69], does not provide uniform order of convergence for the third order scheme.

Now consider the case when $\varepsilon = h^k$ and compare the rate of convergence with the linear scheme. Results presented in Figures 3.3 and 3.4 show that the error decreases with uniform order regardless of the choices of k for both WENO3 and WENO3M. While third order convergence is achieved for $\varepsilon = h$ and h^2 regardless of the norm, for $\varepsilon = h^4$ the convergence rate varies with norm and does not exceed second order. This agrees with the analysis presented in this chapter. Comparing errors in Figures 3.3 and 3.4 we find that

mapping helps to reduce the errors but does not improve the order of convergence of third order WENO scheme.

To test how the classic WENO3 and WENO3M scheme performs for large time periods when the solution is smooth we ran the experiment with the initial condition given by (3.61) until $T = 50.1$ and monitored the total variation of the numerical solution. According to the analysis of this chapter the main contribution to the numerical error for smooth solutions is from the approximation near critical points. This error accumulates with time causing noticeable dissipation at the peak values of the solution. From the results presented in Figures 3.5 and 3.6 one can see that the smaller the ε the more dissipative the scheme becomes. This effect is even more pronounced for rapidly changing functions such as a Gaussian pulse (see the next example).

Now consider the initial condition given by the Gaussian pulse

$$u(x,0) = e^{-\beta(x-t_0)^2}, \tag{3.64}$$

with $\beta = \frac{\log 2}{36\delta^2}$, $\delta = 0.03$, and $t_0 = -0.25$. This initial condition is of particular interest for this work since Gaussian pulse and its first derivative are used to model incident fields and sources in Maxwell's equations. The results of convergence tests shown in Figure 3.7 agree with the convergence analysis of this chapter. Evolving the solution for a long period of time we find that the use of $\varepsilon = h^4$ noticeably changes the shape of the solution near its peak. We also see that the peak value is best resolved with $\varepsilon = h$.

While dissipation negatively affects smooth solutions, it helps to avoid oscillations, when solution is discontinuous. In the next example consider the linear advection problem with discontinuous initial data given by

$$u(x,0) = \begin{cases} 1, & x \in [0,0.5], \\ 0, & \text{otherwise.} \end{cases} \tag{3.65}$$

and periodic boundary conditions. The results of computations with both WENO3 and WENO3M at time $T = 50$ are presented in Figures 3.9 and 3.10. Numerical results show that the choice of $\varepsilon = h$ which gives the best results for smooth functions is not suitable in

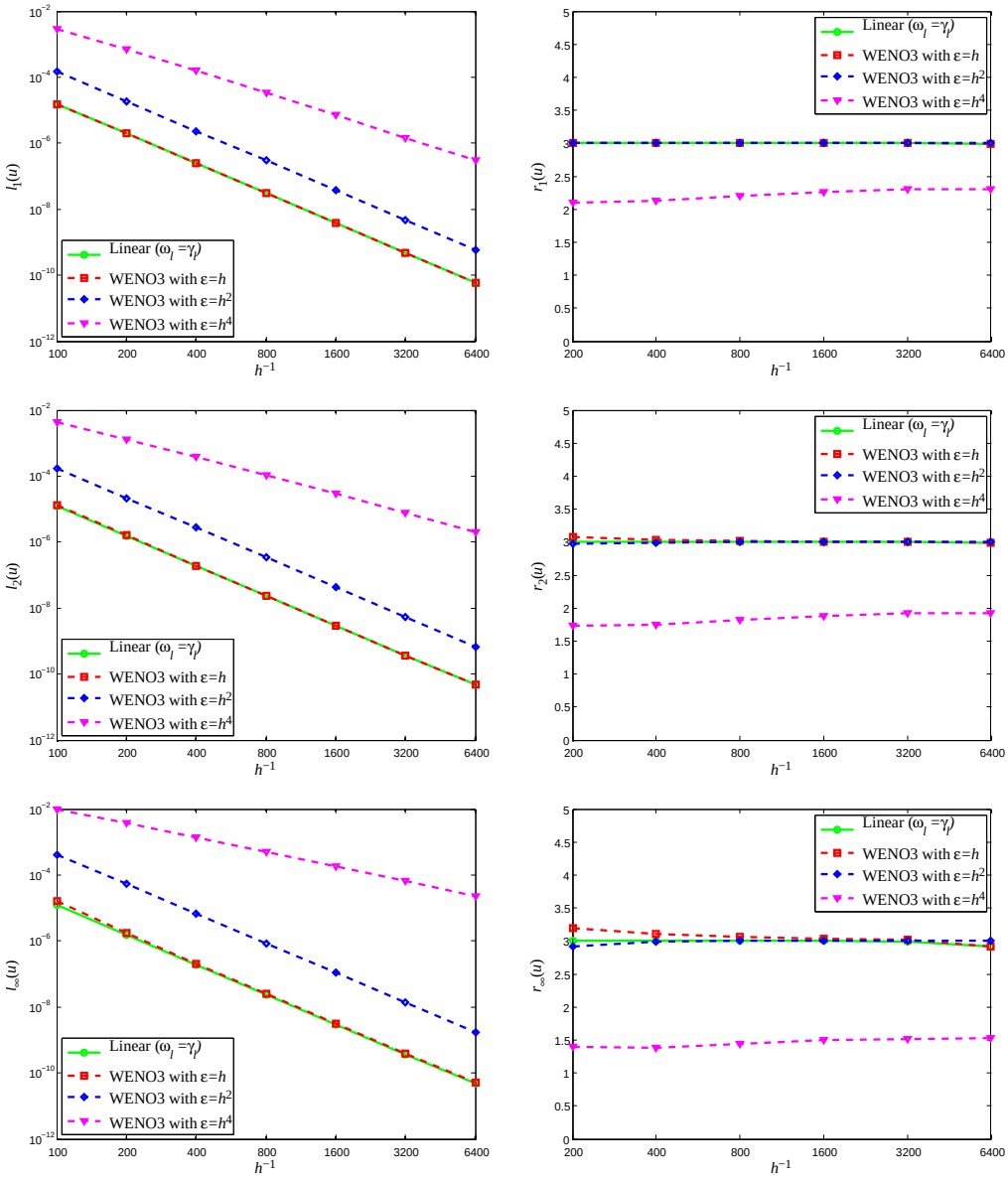


Figure 3.3: Errors and order of convergence of WENO3 schemes with $\varepsilon = h^k$, $k = 1, 2, 4$ for the solution of the linear advection equation with initial data $u(x,0) = \sin(\pi x)$ at $T = 1$.

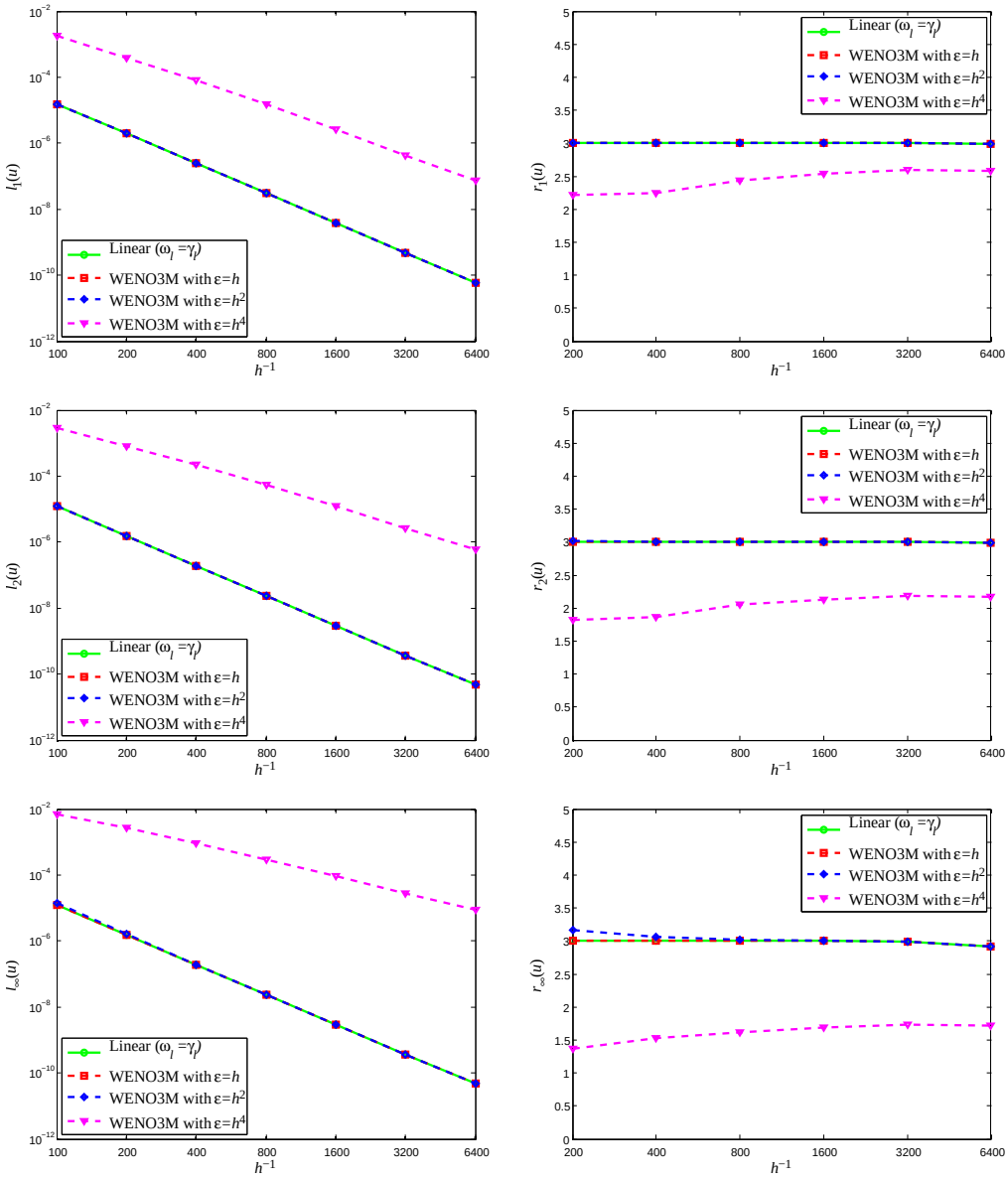


Figure 3.4: Error and order of convergence of WENO3M schemes with $\varepsilon = h^k$, $k = 1, 2, 4$, for the solution of the linear advection equation with initial data $u(x,0) = \sin(\pi x)$ at $T = 1$.

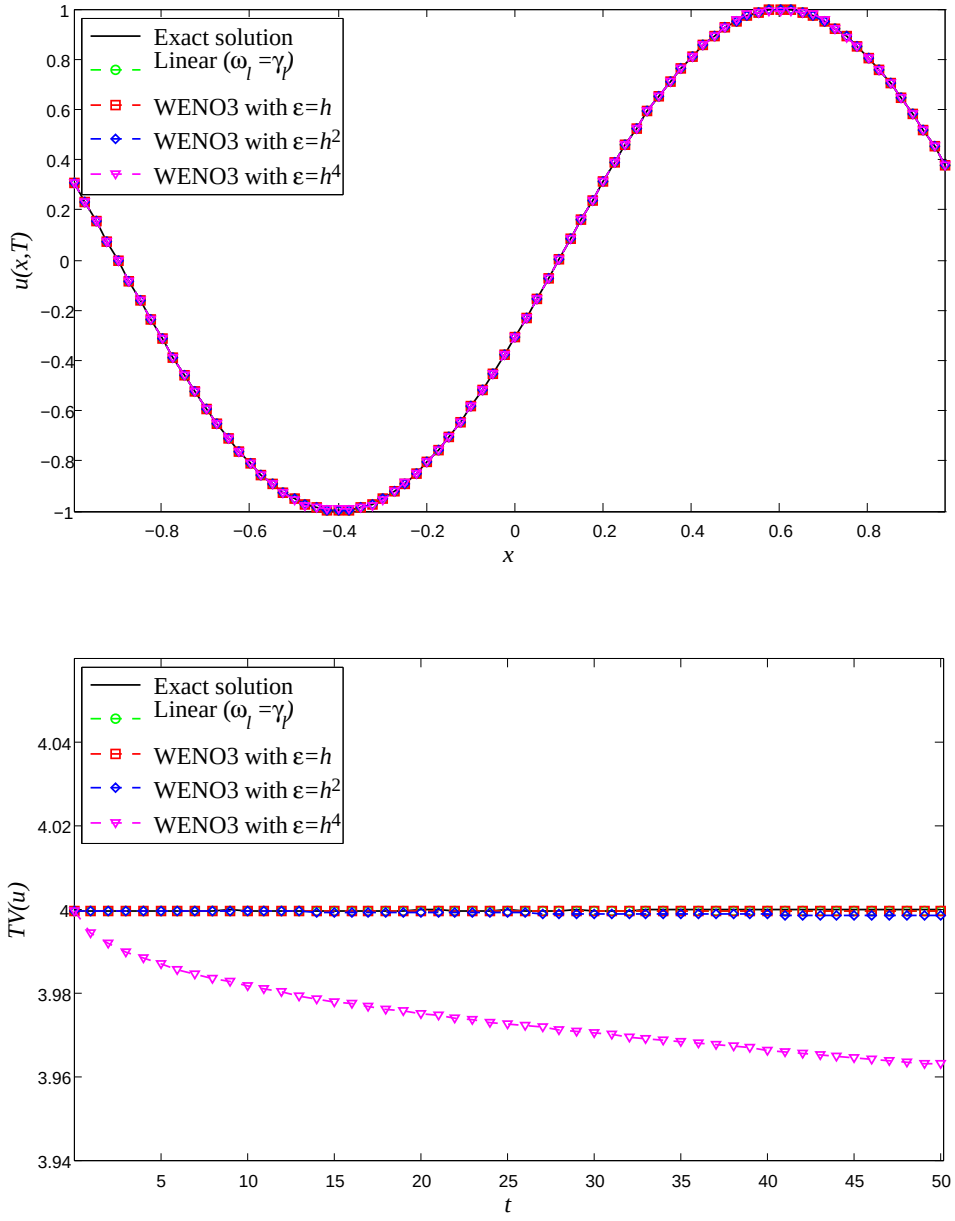


Figure 3.5: Numerical solutions and total variations of the solution of the linear advection equation with initial data $u^0 = \sin(\pi x)$ obtained by linear and WENO3 schemes for $T = 50.1$.

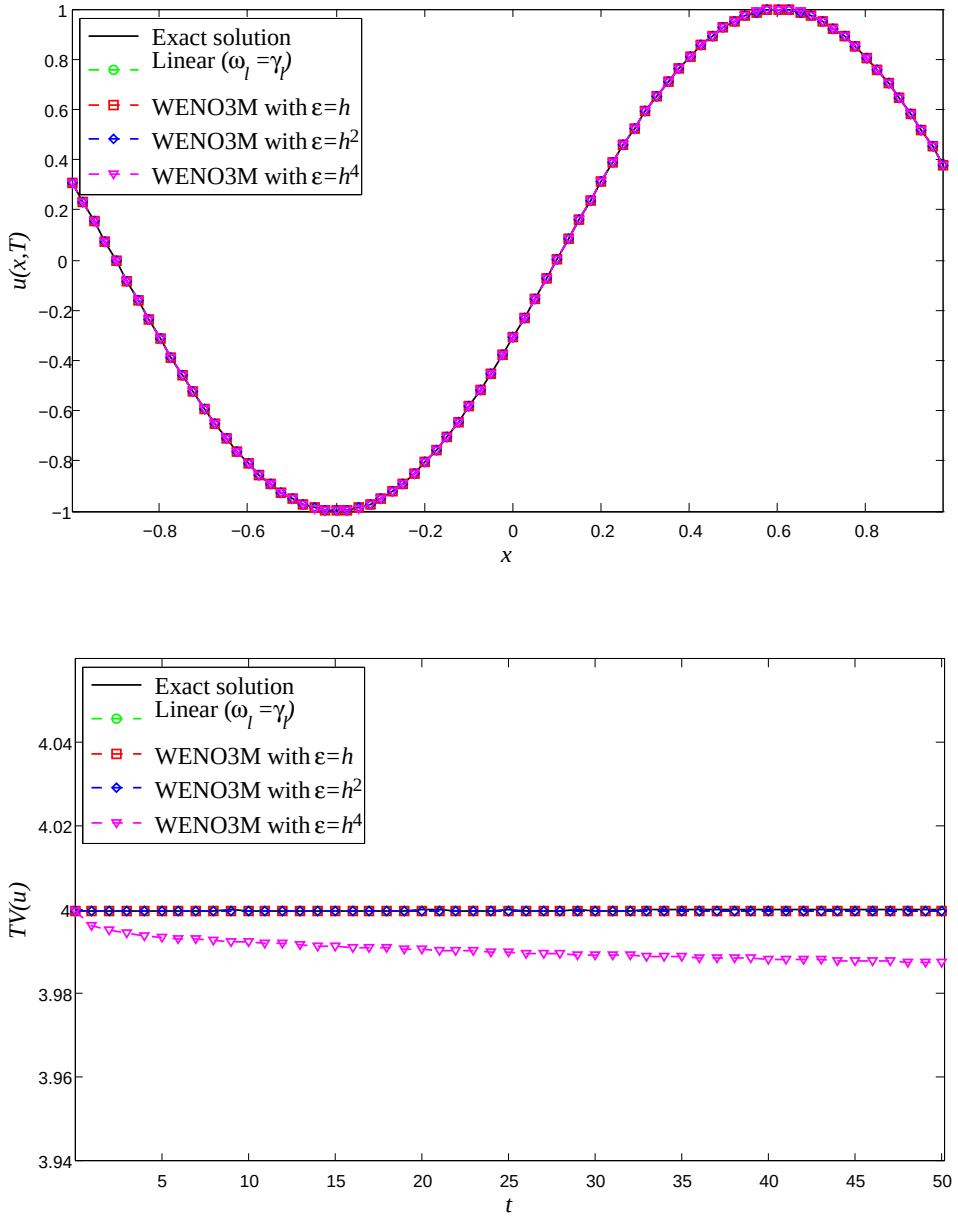


Figure 3.6: Numerical solutions and total variations of the solution of the linear advection equation with initial data $u^0 = \sin(\pi x)$ obtained by linear and WENO3 schemes for $T = 50.1$.

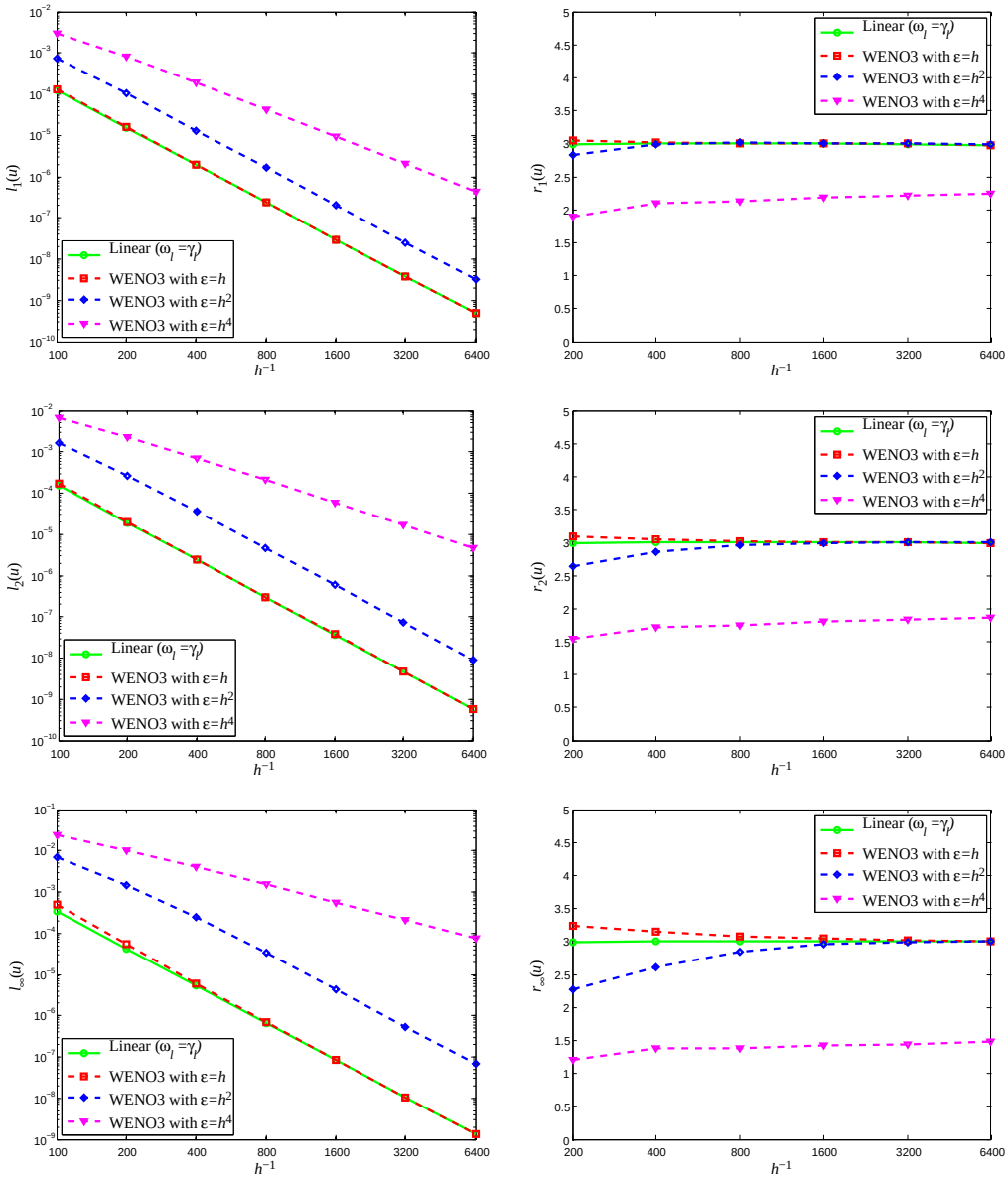


Figure 3.7: Errors and order of convergence of WENO3 scheme with $\varepsilon = h^k$, $k = 1, 2, 4$, for the solution of the linear advection equation with initial data given by the Gaussian (3.64) at $T = 0.5$.

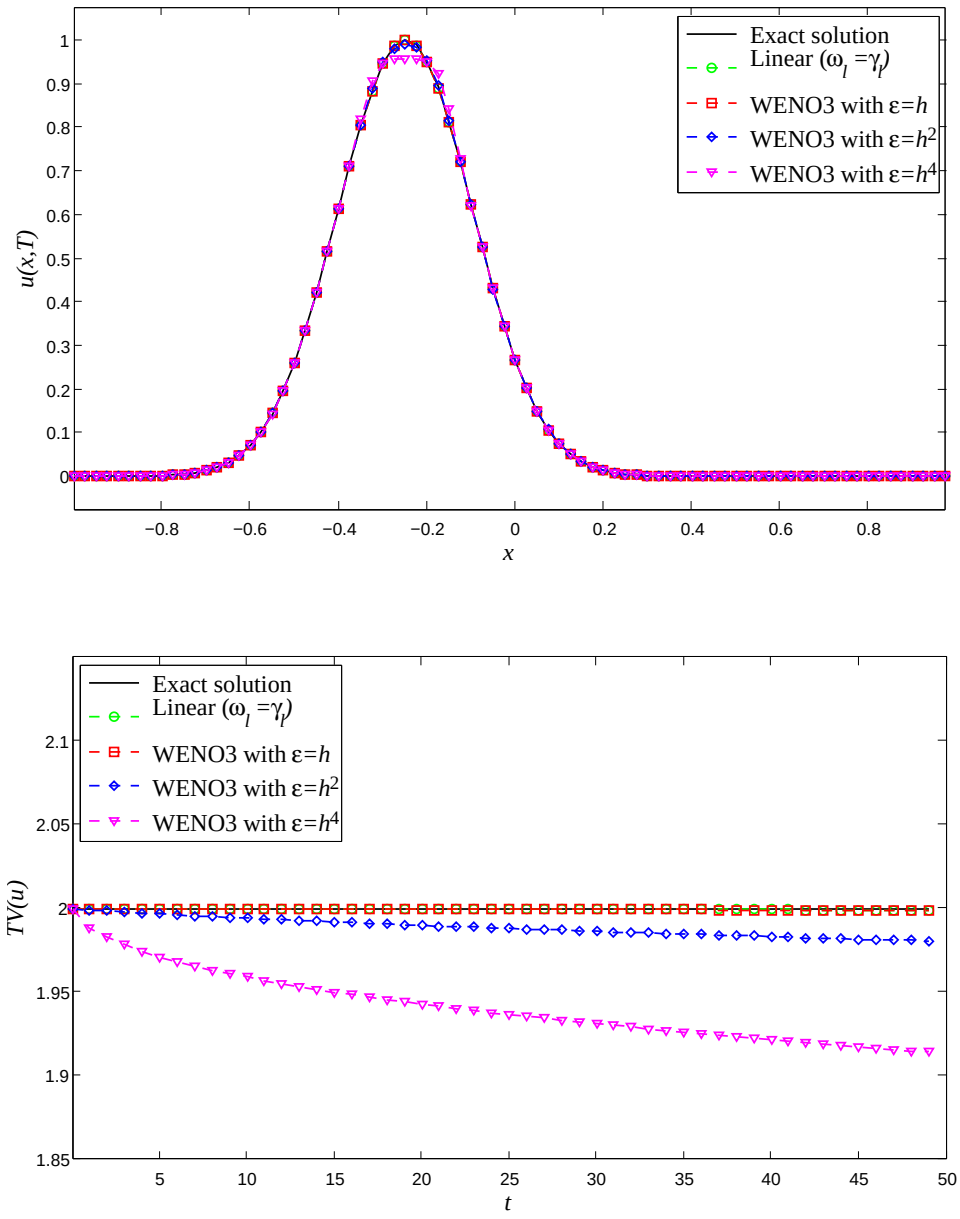


Figure 3.8: Numerical solutions and total variations of the solution of the linear advection equation with initial data given by the Gaussian (3.64) obtained by linear and WENO3 schemes for $T = 50$.

the presence of discontinuity. Large values of ε produce oscillatory results near discontinuity and pollute the solution in smooth regions. The amplitude of oscillations grows with time increasing the total variation. For $\varepsilon \leq h^2$ we get essentially non-oscillatory results with classical WENO3, and the least oscillatory solution is obtained with, $\varepsilon = h^4$. The use of mapping increases oscillations, and as a result the total variation grows significantly for $\varepsilon \geq h^2$.

In the following example numerical validation of the result of Theorem 4 is presented. Consider the function

$$u^{exact}(x) = \begin{cases} -cos(\pi x), & x \leq 0, \\ cos(\pi x), & x > 0, \end{cases}$$

and a uniform grid on $[-1,1]$ with $N_j = 50 \cdot 2^j + 1$, $j = 0, ..., 7$. In this case a discontinuity is inside the interval $\left(-\frac{h_j}{2}, \frac{h_j}{2}\right)$, where $h_j = \frac{2}{N_j}$. The errors of the approximations $u\left(x_{-\frac{h_j}{2}}\right)$ and $u\left(x_{\frac{h_j}{2}}\right)$ by linear and WENO reconstructions are computed using

$$e_{\pm\frac{h_j}{2}} = u\left(x_{\pm\frac{h_j}{2}}\right) - u^{exact}\left(x_{\pm\frac{h_j}{2}}\right), \; j = 0, ..., 7,$$

and order of convergence at $x_{-\frac{h_j}{2}}$ and $x_{\frac{h_j}{2}}$ is obtained using

$$r_{\pm\frac{h_j}{2}} = \frac{\log\left(e\left(x_{\pm\frac{h_{j-1}}{2}}\right)/e\left(x_{\pm\frac{h_j}{2}}\right)\right)}{\log\left(h_{j-1}/h_j\right)}, \; j = 1, ..., 7.$$

For WENO approximations $\varepsilon = 10^{-2}$, h, h^2 and h^4 is used. The results of approximation in the region of discontinuity for $j = 3$ are shown in Figure 3.11. Accuracy of approximations by linear and WENO schemes are shown in Figure 3.12. One can see that for WENO scheme with $\varepsilon = h^k$, $k \geq 1$, the approximation is second order accurate which agrees with the statement of the Theorem 4.

Summarizing the results from numerical experiments and the accuracy analysis, it can be deduced that performance of the third order WENO scheme strongly depends on the

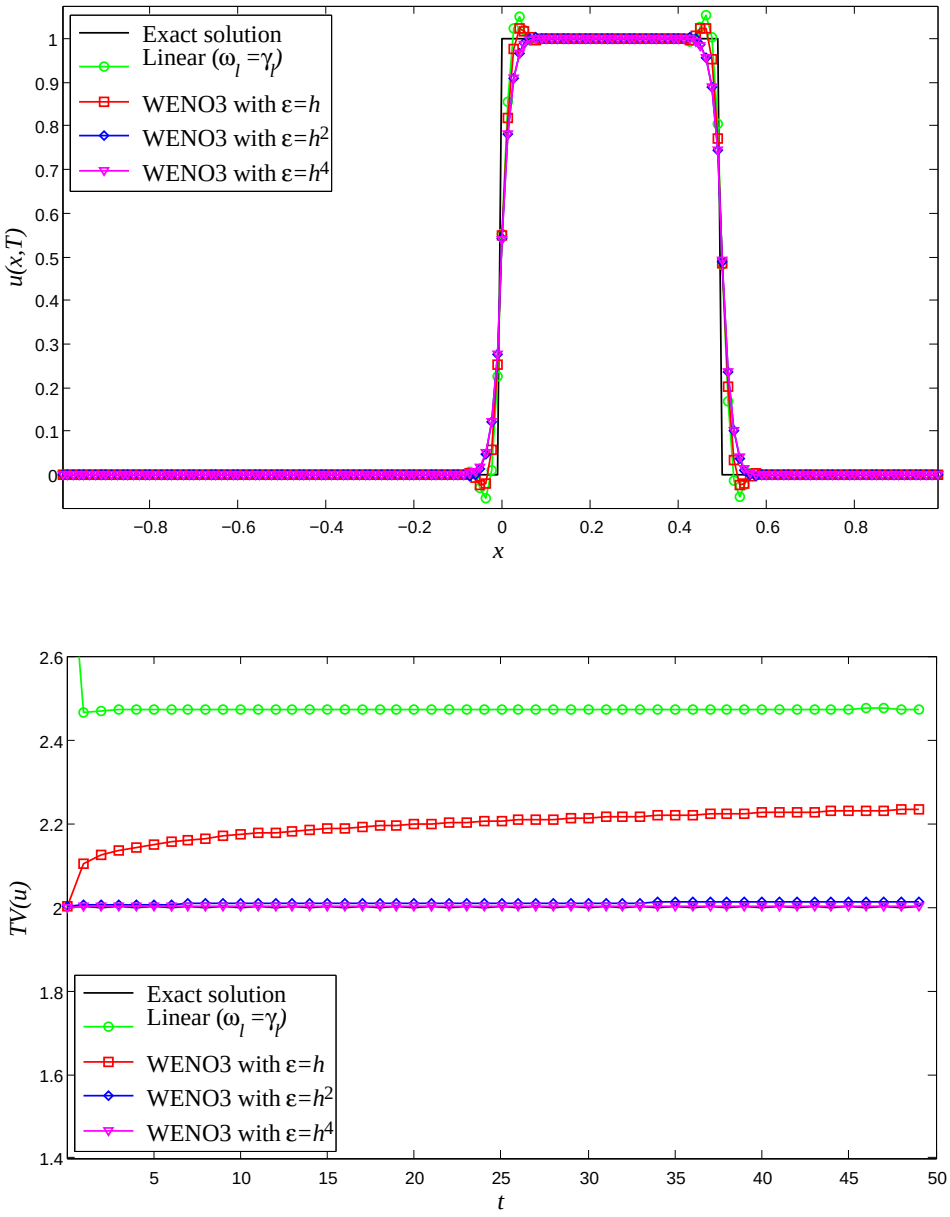


Figure 3.9: Numerical solutions and total variations of the solution of the linear advection equation with initial data $u(x,0) = \begin{cases} 1, & x \in [0,0.5], \\ 0, & \text{otherwise}, \end{cases}$ obtained by linear and WENO3 schemes for $T = 50$.

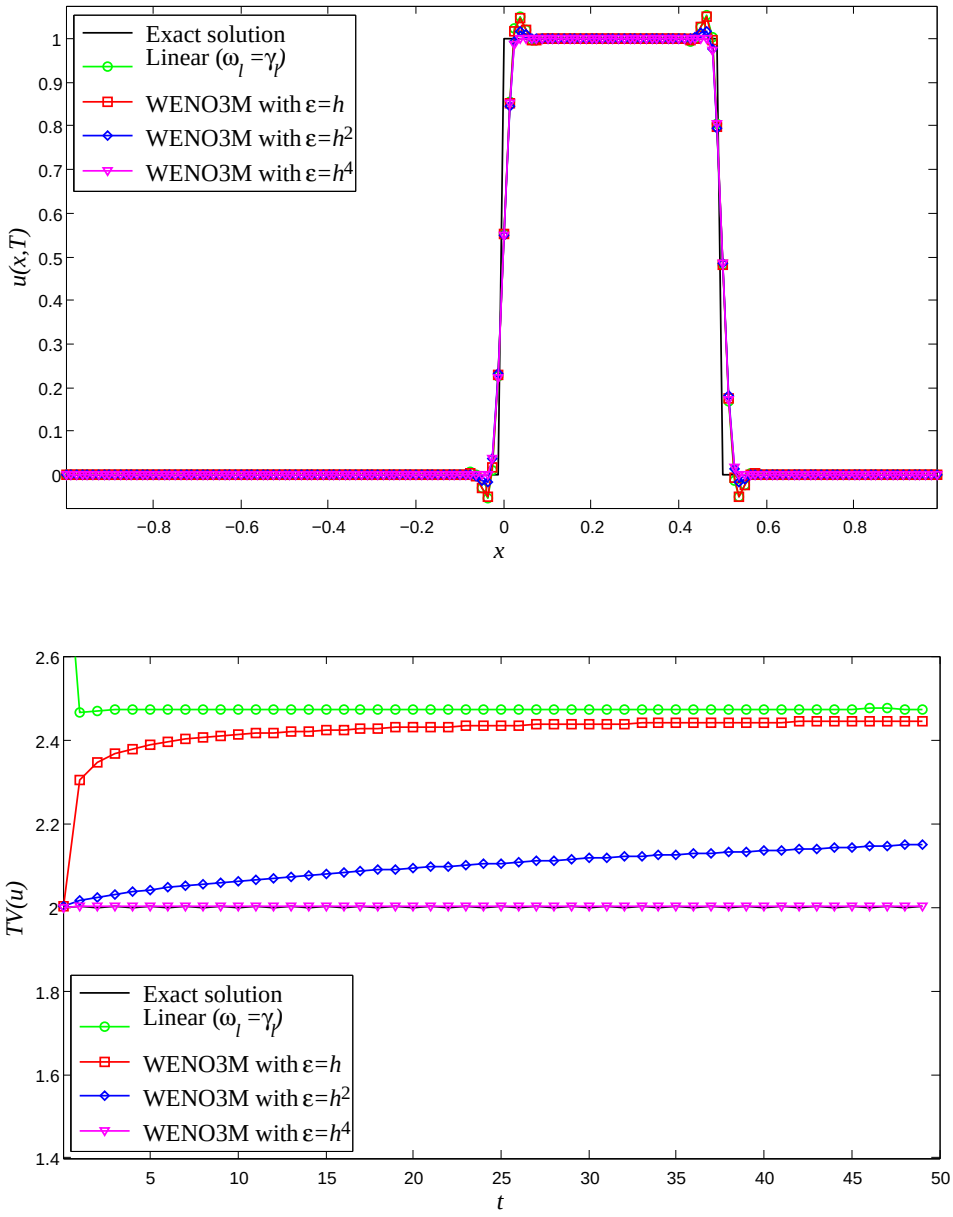


Figure 3.10: Numerical solutions and total variations of the solution of the linear advection equation with initial data $u(x,0) = \begin{cases} 1, & x \in [0,0.5], \\ 0, & \text{otherwise}, \end{cases}$ obtained by linear and WENO3M schemes for $T = 50$.

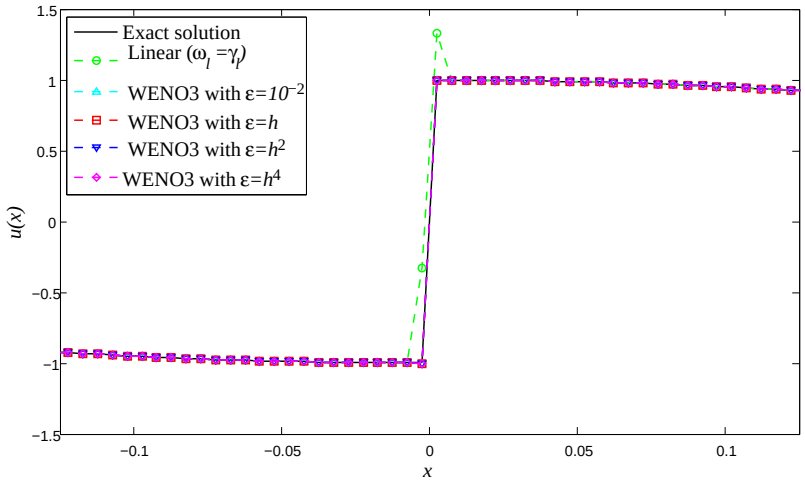


Figure 3.11: Approximation of $u^{exact}(x) = \begin{cases} -cos(\pi x), & x \leq 0, \\ cos(\pi x), & x > 0, \end{cases}$ near discontinuity by third order linear and WENO schemes.

choice of ε. While $\varepsilon = h$ allows better resolution for smooth solutions near critical points (this is especially noticeable for Gaussian pulse), the choice $\varepsilon = h^4$ better controls oscillations near discontinuities. The choice $\varepsilon = h^2$ as suggested in [11] seems to be the best compromise for the classic WENO3 scheme when a discontinuity is present in the solution. The use of mapping technique does not improve the formal order of convergence of WENO3 and also diminishes non-oscillatory properties of WENO scheme.

3.4 Chapter summary

In this chapter the third order WENO scheme for a one-dimensional uniform grid is studied. Following [11] the effect of the choice of ε in non-linear weights on the order of convergence for solutions containing critical points is studied. With a fixed value of ε, the accuracy of the final solution is hard to predict as it changes with grid size. On the other hand the rate of convergence is uniform for different meshes when ε depends on h.

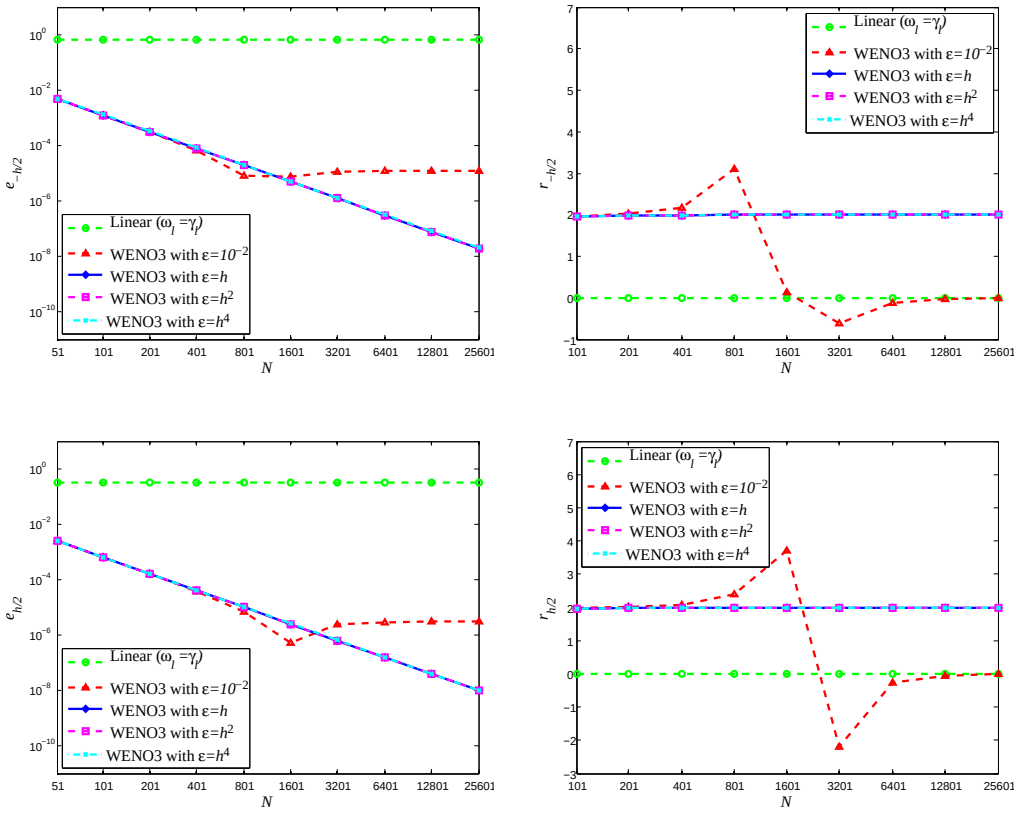


Figure 3.12: Errors and order of convergence of the approximation of $u^{exact}(x) = \begin{cases} -cos(\pi x), & x \leq 0, \\ cos(\pi x), & x > 0, \end{cases}$ at points near discontinuity by third order linear and WENO schemes.

Problems with smooth solutions benefit from large values of ε, while very small ε is required to suppress oscillations near discontinuities. The choice $\varepsilon = h^2$ appears to be the best compromise between the two cases, while it is not ideal for either smooth or discontinuous case. We also find that the mapping technique, which is very successful for the fifth order WENO scheme, did not improve the order of accuracy of the third order scheme. It slightly improves the error for problems with smooth solutions and reduces dissipation for long time computations. At the same time the WENO3M is not suitable for problems with discontinuities as it increases the size of oscillations.

It is also important to mention that solution of the linear advection equation by WENO3 coupled with the Euler time-integration is unstable unless an impractically small CFL number is used. This result is similar to the one found for the fifth order WENO [96]. The WENO3 scheme is stable when coupled with higher order time-integration such as the second or third order Runge-Kutta.

In the next chapter we discuss the same scheme for three-dimensional tetrahedral meshes. Since the size of elements can vary dramatically on strongly inhomogeneous unstructured meshes, it is even more important to adapt the value of ε to the size of each element to ensure an accurate solution. Hence the results obtained in this chapter are crucial in the development of a three-dimensional solver on unstructured meshes.

Chapter 4

Finite volume reconstructions in 3D

In this chapter the approximation of the fields $\boldsymbol{U}_{ij}$ and $\boldsymbol{U}_{ji}$ on the face between the elements T_i and T_j with third order weighted essentially non-oscillatory (WENO) scheme is discussed. The application of WENO reconstruction to systems of equations can be done componentwise, or using local characteristic directions. According to Shu [119], one can get good results with componentwise reconstructions for up to third order accurate schemes. Higher order reconstructions have better non-oscillatory properties if performed in characteristic directions [118], but have a higher computational cost. Since reconstructions of order higher than three are out of the scope of the present work, only componentwise reconstructions are considered.

4.1 FV schemes on unstructured meshes overview

Most Finite Volume approximations of Maxwell's equations use the so-called MUSCL (monotonic upwind scheme for conservation laws) developed by Van Leer [128]. MUSCL schemes have better stability properties compared to central schemes and are more suited for Runge-Kutta type time integration. MUSCL-type cell-centered finite volume schemes on unstructured meshes can be found in [39, 15, 74]. Details of the application of a MUSCL scheme to Maxwell's equations can be found in [18, 97].

In the presence of large gradients in the piecewise linear reconstruction used in MUSCL schemes one has to apply a slope limiter to suppress oscillations and maintain monotonicity. Limiting techniques for unstructured meshes in two space dimensions can be found in [15, 39]. The use of limiters unavoidably decreases the accuracy to first order at critical points [54], therefore should only be used if necessary for stability. Unlike MUSCL scheme, ENO and WENO schemes do not require limiters since they are based on finding the essentially non-oscillatory reconstructions. Several choices of essentially non-oscillatory schemes for multidimensional unstructured meshes can be found in literature. In [66] Harten and Chakravarthy extended their original ENO scheme [68] to multidimensional unstructured meshes. Alternative ENO schemes for unstructured meshes were presented by Abgrall in [4] and Sonar in [122]. ENO schemes were applied to Maxwell's equations in [97, 22, 130].

Weighted ENO schemes were designed by Liu et al. [87] to improve the performance of ENO schemes. While having the ENO property, WENO schemes have better accuracy on the same stencil in the one-dimensional case. For unstructured meshes there are two types of WENO schemes. The type I WENO schemes are easier to construct because the linear coefficients can be chosen as arbitrary positive numbers (usually a larger linear weight is given to the central small stencil). The accuracy of the resulting type I WENO is not higher than that on each small stencil, similar to ENO schemes. The type I WENO schemes on three-dimensional unstructured meshes were developed in [37, 38, 127, 88].

The reconstructions formed by the type II WENO schemes have an order of accuracy higher than that of the reconstructions on small stencils. These methods are more difficult to construct for unstructured meshes and there is no freedom in selecting the linear weights. Linear weights depend solely on the mesh quality and in most cases there are some negative weights, which create stability issues. The type II WENO scheme for tetrahedral meshes was proposed by Zhang and Shu in [135] and is tested in this thesis.

In the beginning of this chapter a brief review of the second order reconstruction by MUSCL is presented. This scheme is used in numerical experiments to test second order multirate Runge-Kutta schemes. Then following the procedure outlined in Chapter 3, the details of WENO reconstruction for three-dimensional tetrahedral meshes is presented. As

it will be seen, there are some differences between the reconstructions in one and three-dimensional cases. First, in the three-dimensional case the coefficients for the polynomials on big stencils and coefficients for linear reconstructions are defined in a least square sense. Second, the system that defines the linear weights has additional equations in the three-dimensional case. Therefore, the linear reconstruction does not replicate the polynomial reconstruction the way it occurs in the one-dimensional case.

4.2 MUSCL reconstruction

In this section we review the MUSCL reconstruction scheme for unstructured three-dimensional meshes. Let $\boldsymbol{x}_0 = \frac{1}{|T_0|}\int_{T_0} \boldsymbol{x} dV$ be the barycenter of some element T_0 and let $\overline{u}_0$ be the cell average of u on that element. Consider the Taylor series

$$u(\boldsymbol{x}) = \overline{u}_0 + \nabla u(\boldsymbol{x}_0) \cdot (\boldsymbol{x} - \boldsymbol{x}_0) + O\left(|\boldsymbol{x} - \boldsymbol{x}_0|^2\right), \tag{4.1}$$

where the gradient $\nabla u(\boldsymbol{x}_0)$ is approximated by the average of $\nabla u(\boldsymbol{x})$ on the element T_0 as follows

$$\nabla u(\boldsymbol{x}_0) \approx \overline{\nabla u}_0 = \frac{1}{|T_0|}\int_{T_0} \nabla u dV. \tag{4.2}$$

Now define the surface average of u on the face S_j by

$$u_{0j} = \frac{1}{|S_j|}\int_{S_j} u dS, \tag{4.3}$$

which using (4.1) and (4.2) is approximated by

$$u_{0j} \approx \overline{u}_0 + \overline{\nabla u}_0 \cdot \left(\boldsymbol{x}_{0j} - \boldsymbol{x}_0\right), \tag{4.4}$$

here $\boldsymbol{x}_{0j} = \frac{1}{|S_j|}\int_{S_j} \boldsymbol{x} dS$ is the center point of the face S_j. The are different approaches to define the reconstruction of the gradient. The most straightforward way is to employ the divergence theorem. Then from (4.2) the following formula can be obtained for the gradient

$$\overline{\nabla u}_0 \approx \frac{1}{|T_0|}\int_{T_0} \nabla u dV = \frac{1}{|T_0|}\int_{\partial T_0} u\hat{\boldsymbol{n}} dS. \tag{4.5}$$

Then the second order central approximation of $\overline{\nabla u}_0$ can be given by

$$\overline{\nabla u}_0 = \overline{\nabla_0 u}_0 = \frac{1}{T_0} \sum_{j=1}^{4} \left| S_j \right| \hat{\boldsymbol{n}}_j \left[\overline{u}_0 + \beta_j \left(\overline{u}_j - \overline{u}_0 \right) \right], \tag{4.6}$$

where $\hat{\boldsymbol{n}}_j$ is the unit normal vector on the face S_j and β_j is the coefficient of the linear interpolation to the point $\boldsymbol{x}_{0j}$ defined by

$$\beta_j = \frac{\left| \left(\boldsymbol{x}_0 - \boldsymbol{x}_{0j} \right) \cdot \hat{\boldsymbol{n}}_j \right|}{\left| \left(\boldsymbol{x}_j - \boldsymbol{x}_{0j} \right) \cdot \hat{\boldsymbol{n}}_j \right| + \left| \left(\boldsymbol{x}_0 - \boldsymbol{x}_{0j} \right) \cdot \hat{\boldsymbol{n}}_j \right|}. \tag{4.7}$$

Other reconstructions can be found in literature, they have been developed for CFD problems on triangular meshes [15, 39] which use neighboring cell centers to construct the control volume for gradient evaluation. These approaches are expensive for three-dimensional problems and present an obstacle when evaluating the gradient at a boundary element.

By excluding one of the neighboring elements T_j (assigning $\beta_j = 0$ in (4.6)) we can obtain several first order approximations $\overline{\nabla_j u}_0$ of $\overline{\nabla u}_0$. The central approximation $\overline{\nabla_0 u}_0$ of the gradient in (4.4) is usually sufficient in approximations of Maxwell's equations. However, in regions with large gradients of the solution, a slope limiting technique can be used to avoid oscillations. The slope limiting techniques for unstructured meshes discussed in literature include the limiter of Durlofsky et al. [39], the limited central difference (LCD) and the maximum limited gradient (MLG) [15] approaches.

The cheapest and easiest limiter is given by the LCD approach. It uses only the central gradient approximation $\overline{\nabla_0 u}_0$ to obtain the final limited gradient approximation

$$\overline{\nabla u}_0 \approx \left(\min_{j \leq 4} \alpha_j \right) \overline{\nabla_0 u}_0, \tag{4.8}$$

where

$$\alpha_j = \begin{cases} \frac{\max\left(\overline{u}_j - \overline{u}_0, 0\right)}{\left(\boldsymbol{x}_j - \boldsymbol{x}_0\right) \cdot \overline{\nabla u}_0}, & \left(\boldsymbol{x}_j - \boldsymbol{x}_0\right) \cdot \overline{\nabla u}_0 > \max\left(\overline{u}_j - \overline{u}_0, 0\right), \\ \frac{\min\left(\overline{u}_j - \overline{u}_0, 0\right)}{\left(\boldsymbol{x}_j - \boldsymbol{x}_0\right) \cdot \overline{\nabla u}_0}, & \left(\boldsymbol{x}_j - \boldsymbol{x}_0\right) \cdot \overline{\nabla u}_0 < \min\left(\overline{u}_j - \overline{u}_0, 0\right), \\ 1, & \text{otherwise.} \end{cases} \tag{4.9}$$

The limiter of Durlofsky et al. [39] takes four first order gradient approximations $\overline{\nabla_j u_0}$, $j = 1, ..., 4$, and assigns zero value for the central gradient approximation ($\overline{\nabla_0 u_0} = \mathbf{0}$). Then it chooses the gradient approximation with the maximum norm. The MLG limiter combines the first two approaches. The above described limiters provide enforcement of the maximum principle for scalar problems

$$\min\left(\overline{u}_j, \overline{u}_0\right) \leq u_{ij} \leq \max\left(\overline{u}_j, \overline{u}_0\right), \tag{4.10}$$

which is necessary for stability of the scheme. The slope limiter should be calculated only if the central approximation of the gradient given by (4.6) produces a reconstruction value u_{0j} that does not satisfy the maximum principle (4.10).

4.3 Third order WENO reconstruction

It is well known that WENO schemes perform better than a central scheme of the same order for problems with singularities. At the same time they can preserve the order of the central scheme for the smooth parts of the solution. Computational electromagnetics problems often involve materials with different dielectric properties as well as the propagation of smooth signals. Therefore, these features of WENO schemes become attractive for solving linear Maxwell's equations. In this section the type II WENO scheme developed by Zhang and Shu in [135] is presented in detail.

In the scheme (2.82) the integral over a triangular face S_j for boundary fields $\boldsymbol{U}_{0j}$ has to be approximated with third order accuracy. This can be done by applying Gaussian quadrature formula, which for the third order case is given by the 4-point rule [72, 135]

$$\boldsymbol{U}_{0j} = \sum_{k=1}^{4} g_k \boldsymbol{U}\left(\boldsymbol{x}_k^{(j)}\right), \tag{4.11}$$

where g_k and $\boldsymbol{x}_k^{(j)}$ are the Gaussian quadrature weights and points respectively. For the triangle with vertices $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$ the Gaussian quadrature points are given by [72]

$$\begin{aligned}
\boldsymbol{x}_1^{(j)} &= \lambda_1 \boldsymbol{x}_1 + \lambda_2 \boldsymbol{x}_2 + \lambda_3 \boldsymbol{x}_3, \\
\boldsymbol{x}_2^{(j)} &= \lambda_2 \boldsymbol{x}_1 + \lambda_1 \boldsymbol{x}_2 + \lambda_3 \boldsymbol{x}_3, \\
\boldsymbol{x}_3^{(j)} &= \lambda_4 \boldsymbol{x}_1 + \lambda_5 \boldsymbol{x}_2 + \lambda_6 \boldsymbol{x}_3, \\
\boldsymbol{x}_4^{(j)} &= \lambda_5 \boldsymbol{x}_1 + \lambda_4 \boldsymbol{x}_2 + \lambda_6 \boldsymbol{x}_3,
\end{aligned}$$

where

$$\lambda_1 = \frac{6-\sqrt{6}}{10}\left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right), \qquad \lambda_2 = \frac{6-\sqrt{6}}{10}\left(\frac{1}{2} + \frac{\sqrt{3}}{6}\right), \qquad \lambda_3 = \frac{4+\sqrt{6}}{10},$$
$$\lambda_4 = \frac{6+\sqrt{6}}{10}\left(\frac{1}{2} - \frac{\sqrt{3}}{6}\right), \qquad \lambda_5 = \frac{6+\sqrt{6}}{10}\left(\frac{1}{2} + \frac{\sqrt{3}}{6}\right), \qquad \lambda_6 = \frac{4-\sqrt{6}}{10},$$

and the weights are

$$g_1 = g_2 = \frac{9-\sqrt{6}}{36}, \quad g_3 = g_4 = \frac{9+\sqrt{6}}{36}.$$

At each Gaussian quadrature point $\boldsymbol{x}_k^{(j)}$ the third order WENO reconstruction is implemented to approximate the field value $\boldsymbol{U}\left(\boldsymbol{x}_k^{(j)}\right)$ using field averages. As was mentioned earlier, for schemes up to the third order, it is more efficient to perform the reconstruction componentwise. Therefore the rest of this chapter will consider WENO reconstructions for a scalar function u.

To find the coefficients of type II WENO reconstruction for each Gaussian point, the following steps need to be performed:

- find the third order central reconstruction $p_2\left(\boldsymbol{x}_k^{(j)}\right)$ on the big stencil S_0 in terms of cell averages on elements of the big stencil;
- find the second order reconstructions $p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right)$ for each small stencil $S_l \subset S_0$, $l = 1, ..., m$ in terms of cell averages on elements of the small stencil;

- find optimal linear weights γ_l that give the combination of the second order reconstructions $p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right)$ "closest" to the third order reconstruction $p_2\left(\boldsymbol{x}_k^{(j)}\right)$ in the least square sense;

- find non-linear weights ω_l by modifying the linear weights γ_l according to the smoothness of the function on each small stencil S_l.

The resulting combination of the second order reconstructions $p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right)$ with non-linear coefficients ω_l will give the third order type II WENO reconstruction of u at the Gaussian point $\boldsymbol{x}_k^{(j)}$.

4.3.1 Big stencil

The first step is to build the third order reconstruction of u on a face S_j of the element T_0 using the big stencil. To make reconstruction coefficients independent of mesh sizes local variables are introduced by

$$\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3) = \boldsymbol{\xi}(\boldsymbol{x}) = \frac{\boldsymbol{x} - \boldsymbol{x}_0}{h}, \tag{4.12}$$

where $\boldsymbol{x}_0 = \left((x_1)_0, (x_2)_0, (x_3)_0\right)$ is the barycenter of T_0, and

$$h = |T_0|^{1/3}. \tag{4.13}$$

The third order reconstruction is given by a quadratic polynomial $p_2(\boldsymbol{x})$ which has the same cell average as u on T_0. In the local coordinates (4.12) the polynomial $p_2(\boldsymbol{x})$ can be written as

$$p_2(\boldsymbol{x}) = \sum_{0 \le i_1+i_2+i_3 \le 2} a_{i_1 i_2 i_3} \xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3}. \tag{4.14}$$

Using the requirement that

$$\overline{u}_0 = \frac{1}{|T_0|} \int_{T_0} p_2(\boldsymbol{x})\, dV, \tag{4.15}$$

where $\overline{u}_0 = \frac{1}{|T_0|}\int_{T_0} u dV$ is the volume average on T_0, we get

$$a_{000} = \overline{u}_0 - \sum_{1\le i_1+i_2+i_3\le 2} a_{i_1 i_2 i_3} \overline{\left[\xi_1^{i_1}\xi_2^{i_2}\xi_3^{i_3}\right]_0}. \tag{4.16}$$

Therefore polynomial (4.14) can be written as

$$p_2(\boldsymbol{x}) = \overline{u}_0 + \sum_{1\le i_1+i_2+i_3\le 2} a_{i_1 i_2 i_3}\left(\xi_1^{i_1}\xi_2^{i_2}\xi_3^{i_3} - \overline{\left[\xi_1^{i_1}\xi_2^{i_2}\xi_3^{i_3}\right]_0}\right). \tag{4.17}$$

The terms $\overline{\left[\xi_1^{i_1}\xi_2^{i_2}\xi_3^{i_3}\right]_0}$ can be integrated analytically by transforming the volume integral into a sum of surface integrals, and then into a sum of line integrals [103].

Let $S_0 = \{V_m\}_{m=0}^r$ denote the big stencil for WENO reconstruction, where r is the number of cells in the stencil excluding T_0. It is assumed that $V_0 = T_0$, and $\{V_m\}_{m=1}^r$ typically consists of two layers of elements around the element T_0:

$$S_0 = \{V_m\}_{m=0}^r = \left\{ T_0, \underbrace{T_{01},...,T_{04}}_{\text{neighbors of } T_0}, \underbrace{T_{11},T_{12},T_{13},...,T_{41},T_{42},T_{43}}_{\text{neighbors of } T_{0j}} \right\}.$$

The total number of elements in a typical big stencil is 17. Notice that some elements in the stencil may coincide, so the total number of cells in the stencil may be less than 17. We have to determine nine coefficients $a_{i_1 i_2 i_3}$ in the quadratic polynomial (4.17). Therefore at least 9 neighbors of T_0 are needed to form the big stencil. If $r < 9$ another layer of elements is added (this case is rare, because it means each neighbor T_{0j} has only one neighbor which does not coincide with another neighbor's neighbor, that can happen near the boundary).

To determine the coefficients $a_{i_1 i_2 i_3}$ in the quadratic polynomial (4.17), its cell averages on every element of $S_0 \setminus \{T_0\}$ are matched with the cell averages of u on these elements, that is,

$$\frac{1}{|V_m|}\int_{V_m} p_2(\boldsymbol{x})\, dV = \overline{u}_m, \quad m = 1,\ldots,r.$$

This gives an over-determined $r \times 9$ system of equations for 9 coefficients $a_{i_1 i_2 i_3}$

$$\sum_{1 \le i_1+i_2+i_3 \le 2} a_{i_1 i_2 i_3} \left(\overline{\left[\xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3} \right]}_m - \overline{\left[\xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3} \right]}_0 \right) = \overline{u}_m - \overline{u}_0, \quad m = 1, ...r, \tag{4.18}$$

which can be written in matrix form as

$$\boldsymbol{A}\boldsymbol{a} = \boldsymbol{u}, \tag{4.19}$$

where $\boldsymbol{a} = [a_{100}, a_{010}, a_{001}, a_{110}, a_{101}, a_{011}, a_{200}, a_{020}, a_{002}]^T$, $\boldsymbol{u} = [\overline{u}_1 - \overline{u}_0, ..., \overline{u}_r - \overline{u}_0]^T$, and each row m of the matrix $\boldsymbol{A}$ consists of the elements

$$\begin{aligned}
A_{1m} &= \overline{[\xi_1]}_m - \overline{[\xi_1]}_0, & A_{2m} &= \overline{[\xi_2]}_m - \overline{[\xi_2]}_0, & A_{3m} &= \overline{[\xi_3]}_m - \overline{[\xi_3]}_0, \\
A_{4m} &= \overline{[\xi_1 \xi_2]}_m - \overline{[\xi_1 \xi_2]}_0, & A_{5m} &= \overline{[\xi_1 \xi_3]}_m - \overline{[\xi_1 \xi_3]}_0, & A_{6m} &= \overline{[\xi_2 \xi_3]}_m - \overline{[\xi_2 \xi_3]}_0, \\
A_{7m} &= \overline{[\xi_1^2]}_m - \overline{[\xi_1^2]}_0, & A_{8m} &= \overline{[\xi_2^2]}_m - \overline{[\xi_2^2]}_0, & A_{9m} &= \overline{[\xi_3^2]}_m - \overline{[\xi_3^2]}_0.
\end{aligned}$$

To avoid computation of integrals $\overline{\left[\xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3} \right]}_m$ over each element V_m, the approach from [99] is used. Starting by replacing the components of $\boldsymbol{\xi}$ by

$$\xi_l = \frac{1}{h} \left[(x_l - (x_l)_m) + ((x_l)_m - (x_l)_0) \right],$$

then the integrals $\overline{\left[\xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3} \right]}_m$ can be rewritten as

$$\begin{aligned}
\overline{\left[\xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3} \right]}_m &= \frac{1}{|V_m|} \int_{V_m} \frac{1}{h^i} \prod_{l=1}^{3} \left((x_l - (x_l)_m) + ((x_l)_m - (x_l)_0) \right)^{i_l} dV \\
&= \frac{1}{h^i} \sum_{k_1=0}^{i_1} \sum_{k_2=0}^{i_2} \sum_{k_3=0}^{i_3} \left[\prod_{l=1}^{3} C_{k_l}^{i_l} \left((x_l)_m - (x_l)_0 \right)^{k_l} \right] \overline{\left[x_1^{i_1-k_1} x_2^{i_2-k_2} x_3^{i_3-k_3} \right]}_m
\end{aligned} \tag{4.20}$$

where $i = \sum_{l=1}^{3} i_l$, and $C_{k_l}^{i_l} = \frac{i_l!}{k_l!(i_l-k_l)!}$.

The coefficients of the third order polynomial reconstruction can now be found from

$$\boldsymbol{a} = \boldsymbol{a}(\boldsymbol{u}) = \boldsymbol{B}\boldsymbol{u}, \tag{4.21}$$

where $\boldsymbol{B} = \boldsymbol{A}^{\dagger}$ is the pseudo-inverse of the $9 \times r$ matrix $\boldsymbol{A}$, which can be obtained using the singular value decomposition (SVD). Matrix $\boldsymbol{B}$ is computed only at the initialization for each cell of the mesh. In this work the Armadillo C++ linear algebra library [107] was used for SVD computations. Then the quadratic polynomial (4.14) can be rewritten using the coefficients $a_{i_1 i_2 i_3}(\boldsymbol{u})$ defined in terms of cell averages on the big stencil as

$$p_2(\boldsymbol{x}) = \overline{u}_0 + \sum_{1 \le i_1+i_2+i_3 \le 2} a_{i_1 i_2 i_3}(\boldsymbol{u}) \left(\xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3} - \overline{\left[\xi_1^{i_1} \xi_2^{i_2} \xi_3^{i_3} \right]}_0 \right). \tag{4.22}$$

At the k-th quadrature point on the j-th face $\boldsymbol{x}_k^{(j)}$, the third order reconstruction polynomial is given by

$$p_2\left(\boldsymbol{x}_k^{(j)}\right) = \sum_{m=0}^{r} c_m \overline{u}_m = \left(1 - \sum_{m=1}^{r} c_m\right) \overline{u}_0 + \sum_{m=1}^{r} c_m \overline{u}_m, \tag{4.23}$$

where

$$[c_1, \ldots, c_r] = \left[\boldsymbol{v}_k^{(j)}\right]^T \boldsymbol{B}, \tag{4.24}$$

and

$$\boldsymbol{v}_k^{(j)} = \begin{bmatrix} \xi_1\big|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_1]}_0 \\ \xi_2\big|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_2]}_0 \\ \xi_3\big|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_3]}_0 \\ \xi_1\xi_2\big|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_1\xi_2]}_0 \\ \xi_1\xi_3\big|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_1\xi_3]}_0 \\ \xi_2\xi_3\big|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_2\xi_3]}_0 \\ \xi_1^2\big|_{\boldsymbol{x}_k^{(j)}} - \overline{\left[\xi_1^2\right]}_0 \\ \xi_2^2\big|_{\boldsymbol{x}_k^{(j)}} - \overline{\left[\xi_2^2\right]}_0 \\ \xi_3^2\big|_{\boldsymbol{x}_k^{(j)}} - \overline{\left[\xi_3^2\right]}_0 \end{bmatrix}. \tag{4.25}$$

The coefficients c_m, $m = 1, \ldots, r$, in (4.24) depend on the geometry only and are precomputed for each quadrature point $\boldsymbol{x}_k^{(j)}$ at the initialization.

4.3.2 Small stencils

To find the third order WENO reconstruction several first order polynomials giving second order approximations at quadrature points need to be built. Then the linear combination will be determined to give the approximation closest to the second order polynomial p_2 at each quadrature point $\boldsymbol{x}_k^{(j)}$. These polynomials are constructed on small stencils whose union gives the big stencil S. Each small stencil consists of four elements from the big stencil, and includes the target element T_0. The first four stencils consist of the element T_0 and three of its neighbors. The rest of the stencils are constructed by including T_0 with one of its neighbors T_{0j} and two neighbors of that neighbor other than T_0. Typically there are up to $s = 16$ candidates for small stencils $\cup_{l=1}^{s} S_l = S_0$. While the first four stencils always exist, some of the other twelve stencils may coincide, this is checked to exclude duplication.

For each small stencil $S_l = \left\{V_m^{(l)}\right\}_{m=0}^{3}$, with $V_0 = T_0$, a linear polynomial

$$p_1^{(l)}(\boldsymbol{x}) = a_0^{(l)} + \sum_{i=1}^{3} a_i^{(l)} \xi_i, \, l = 1, ..., s, \tag{4.26}$$

is constructed, such that

$$\overline{u}_0 = \frac{1}{|T_0|} \int_{T_0} p_1^{(l)}(\boldsymbol{x}) \, dV. \tag{4.27}$$

This gives

$$a_0^{(l)} = \overline{u}_0 - \sum_{i=1}^{3} a_i^{(l)} \overline{[\xi_i]}_0, \tag{4.28}$$

and therefore

$$p_1^{(l)}(\boldsymbol{x}) = \overline{u}_0 + \sum_{i=1}^{3} a_i^{(l)} \left(\xi_i - \overline{[\xi_i]}_0\right). \tag{4.29}$$

The coefficients are found by agreeing with cell averages on elements $\left\{V_m^{(l)}\right\}_{m=1}^{3}$

$$\overline{u}_0 + \sum_{i=1}^{3} a_i^{(l)} \left(\overline{[\xi_i]}_m^{(l)} - \overline{[\xi_i]}_0\right) = \overline{u}_m^{(l)}, \quad m = 1, ..., 3, \, l = 1, ..., s.$$

This gives s systems of three equations with three unknowns

$$\boldsymbol{A}^{(l)}\boldsymbol{a}^{(l)} = \boldsymbol{u}^{(l)}, \tag{4.30}$$

where $\boldsymbol{a}^{(l)} = \left[a_1^{(l)}, a_2^{(l)}, a_3^{(l)}\right]^T$, $\left[\boldsymbol{u}^{(l)}\right]_m = \overline{u}_m^{(l)} - \overline{u}_0$, and

$$\boldsymbol{A}^{(l)} = \begin{bmatrix} \overline{[\xi_1]}_1^{(l)} - \overline{[\xi_1]}_0 & \overline{[\xi_2]}_1^{(l)} - \overline{[\xi_2]}_0 & \overline{[\xi_3]}_1^{(l)} - \overline{[\xi_3]}_0 \\ \overline{[\xi_1]}_2^{(l)} - \overline{[\xi_1]}_0 & \overline{[\xi_2]}_2^{(l)} - \overline{[\xi_2]}_0 & \overline{[\xi_3]}_2^{(l)} - \overline{[\xi_3]}_0 \\ \overline{[\xi_1]}_3^{(l)} - \overline{[\xi_1]}_0 & \overline{[\xi_2]}_3^{(l)} - \overline{[\xi_2]}_0 & \overline{[\xi_3]}_3^{(l)} - \overline{[\xi_3]}_0 \end{bmatrix}. \tag{4.31}$$

The coefficients $\left\{a_i^{(l)}\right\}_{i=1}^3$ are obtained in terms of cell averages by

$$\boldsymbol{a}^{(l)} = \boldsymbol{a}^{(l)}\left(\boldsymbol{u}^{(l)}\right) = \boldsymbol{B}^{(l)}\boldsymbol{u}^{(l)}, \tag{4.32}$$

where $\boldsymbol{B}^{(l)}$ is the inverse of $\boldsymbol{A}^{(l)}$. In the rare case that the inverse of $\boldsymbol{A}^{(l)}$ does not exist (centroids of elements in the small stencil lie on the same line or plane with $\boldsymbol{x}_0$), the small stencil S_l is excluded since there is always more than enough small stencils in the set. Now the polynomial (4.29) can be written as

$$p_1^{(l)}(\boldsymbol{x}) = \overline{u}_0 + \sum_{i=1}^{3} a_i^{(l)}\left(\boldsymbol{u}^{(l)}\right)\left(\xi_i - \overline{[\xi_i]}_0\right), \quad l = 1, ..., s. \tag{4.33}$$

We define the second order polynomial reconstruction of u at $\boldsymbol{x}_k^{(j)}$ using the stencil S_l as

$$p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right) = \sum_{m=0}^{3} c_m^{(l)}\overline{u}_m^{(l)} = \left(1 - \sum_{m=1}^{3} c_m^{(l)}\right)\overline{u}_0 + \sum_{m=1}^{3} c_m^{(l)}\overline{u}_m^{(l)}, \quad l = 1, ..., s, \tag{4.34}$$

where the coefficients $c_m^{(l)}$ are obtained from

$$\left[c_1^{(l)}, c_2^{(l)}, c_3^{(l)}\right] = \left[\xi_1|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_1]}_0, \xi_2|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_2]}_0, \xi_3|_{\boldsymbol{x}_k^{(j)}} - \overline{[\xi_3]}_0\right]\boldsymbol{B}^{(l)}. \tag{4.35}$$

Just as with the big stencil, the coefficients $c_m^{(l)}$, $m = 1,2,3$, $l = 1,...,s$, depend on the local geometry only and are precomputed at the initialization.

4.3.3 Linear weights

For each quadrature point $\boldsymbol{x}_k^{(j)}$ the linear weights $\{\gamma_l\}_{l=1}^s$ such that the polynomial obtained by the combination of linear polynomials $p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right)$ has the closest value to $p_2\left(\boldsymbol{x}_k^{(j)}\right)$ need to be found, i.e.,

$$p_2\left(\boldsymbol{x}_k^{(j)}\right) = \sum_{l=1}^{s} \gamma_l p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right). \tag{4.36}$$

The linear weights $\{\gamma_l\}_{l=1}^s$ are constants depending only on the geometry, and can be precomputed for each quadrature point $\boldsymbol{x}_k^{(j)}$ at the initialization. Since polynomials on both sides of (4.36) are expressed in terms of cell averages, the ideal case would be to have (4.36) satisfied for arbitrary u. But, unlike the one-dimensional case, in two or three-dimensional reconstructions the equality in (4.36) is considered in the least square sense. To build the linear system for $\{\gamma_l\}_{l=1}^s$ we follow the procedure outlined in [135]. It consists of two parts: the first part is constructed from the assumption that (4.36) holds exactly for polynomials 1, ξ_1, ξ_2, ξ_3, ξ_1^2, ξ_2^2, ξ_3^2, $\xi_1\xi_2$, $\xi_1\xi_3$, $\xi_2\xi_3$, the second part is obtained from the equality (4.36) for an arbitrary u in the least square sense.

To build the first part of the system for linear weights we take $u = 1$, ξ_1, ξ_2, ξ_3, ξ_1^2, ξ_2^2, ξ_3^2, $\xi_1\xi_2$, $\xi_1\xi_3$, $\xi_2\xi_3$ in (4.36). Both p_2 and all $p_1^{(l)}$ reproduce functions 1, ξ_1, ξ_2, ξ_3 exactly. Therefore the equality (4.36) with these functions gives the same constraint

$$\sum_{l=1}^{s} \gamma_l = 1. \tag{4.37}$$

Using the constraint (4.37) together with six equations

$$u\left(\boldsymbol{x}_k^{(j)}\right) = \sum_{l=1}^{s} \gamma_l \left(\overline{u}_0 + \sum_{m=1}^{3} c_m^{(l)}\left(\overline{u}_m^{(l)} - \overline{u}_0\right)\right), \tag{4.38}$$

where $u = \xi_1^2, \xi_2^2, \xi_3^2, \xi_1\xi_2, \xi_1\xi_3, \xi_2\xi_3$, we form a $7 \times s$ linear system

$$\boldsymbol{C\gamma} = \boldsymbol{b}. \tag{4.39}$$

Since in general the number of small stencils is greater than 7, (4.39) is an under-determined system. To define the optimal linear weights it is required that (4.36) holds for an arbitrary u in the least square sense. This means that the linear combination of the second order reconstructions on small stencils $\left\{S^{(l)}\right\}_{l=1}^{s}$ is the closest to the third order reconstruction on the big stencil S_0. Using (4.23) and (4.34) the equation (4.36) is rewritten as

$$\sum_{l=1}^{s} \gamma_l \sum_{m=0}^{3} c_m^{(l)} \overline{u}_m^{(l)} = \sum_{q=0}^{r} c_q \overline{u}_q. \tag{4.40}$$

This gives a $r+1 \times s$ linear system

$$\boldsymbol{D\gamma} = \boldsymbol{c}, \tag{4.41}$$

where $\boldsymbol{c} = [c_0, c_1, ...c_r]^T$ and each row q ($q = 0, 1, \ldots, r$) of the matrix $\boldsymbol{D}$ consists of coefficients $c_m^{(l)}$ corresponding to the averages on elements $\left\{V_m^{(l)}\right\} \in S^{(l)}$ such that $V_m^{(l)} = V_q \in S_0$. Then the systems (4.39) and (4.41) are solved together to find the linear weights $\{\gamma_l\}_{l=1}^{s}$ for each quadrature point. Then the linear reconstruction of u at the quadrature point $\boldsymbol{x}_k^{(j)}$ is obtained by

$$u^{Lin}\left(\boldsymbol{x}_k^{(j)}\right) = \sum_{l=1}^{s} \gamma_l p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right). \tag{4.42}$$

Since the type II WENO scheme uses smaller stencils than type I WENO, it does not have the same flexibility in determining the linear weights. Linear weights are completely dependent on the geometry. As it turns out, in almost all cases, some of the linear weights are negative. This is due to the fact that for three-dimensional problems with complex geometry, the mesh quality is hard to control, and that affects the quality of the least square solution for linear weights. For mild negative weights the splitting technique from [115] can be implemented. If $\min\limits_{1 \le l \le s} \gamma_l < 0$, linear weights are split into positive and negative parts by

$$\tilde{\gamma}_l^{+} = \frac{1}{2}\left(\gamma_l + 3\left|\gamma_l\right|\right), \quad \tilde{\gamma}_l^{-} = \tilde{\gamma}_l^{+} - \gamma_l, \quad l = 1, \ldots, s. \tag{4.43}$$

Then the new linear weights are scaled by

$$\sigma^{\pm} = \sum_{l=1}^{s} \tilde{\gamma}_l^{\pm}, \quad \gamma_l^{\pm} = \tilde{\gamma}_l^{\pm} / \sigma^{\pm}, \quad l = 1, \ldots, s, \tag{4.44}$$

and the new third order linear reconstruction of u at the quadrature point $\boldsymbol{x}_k^{(j)}$ is

$$u^{Lin}\left(\boldsymbol{x}_k^{(j)}\right) = \sigma^{+} \sum_{l=1}^{s} \gamma_l^{+} p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right) - \sigma^{-} \sum_{l=1}^{s} \gamma_l^{-} p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right). \tag{4.45}$$

As it was pointed out in [135], the key idea of splitting is to ensure that every stencil has a significant representation in both positive and negative parts.

From the numerical experiments it was found that implementation of type II WENO scheme with the above splitting technique on the entire mesh creates instability due to the presence of some reconstructions with very negative linear weights. In the case of very negative linear weights, $\max_l(|\gamma_l|) > \zeta$, or when linear weights do not exist, the type II WENO reconstruction can be replaced by a more expensive type I WENO reconstruction. This was done by Liu and Zhang in [89] and tested on a second order hyperbolic conservation law on two-dimensional meshes. They found via numerical experiments that the choice for optimal value of the threshold ζ is between 1 and 10. For larger ζ an increase in accuracy errors is observed. In the present work, WENO reconstructions are replaced with the third order polynomial reconstruction in cases of very negative linear weights. In numerical experiments with discontinuous solutions such substitution did not cause any problems. This can be explained by the fact that very negative linear weights appear only for some quadrature points of a given face. As a result the surface integral (4.11) is a combination of different types of reconstructions, where WENO partially compensates oscillations produced by polynomial scheme. It is found that for arbitrary three-dimensional meshes, the value of the threshold ζ should be less than or equal to 3 to ensure stability. In case when $2 \leq \zeta \leq 3$ the replacement by polynomial reconstructions occurs in 1-3 percent of the reconstructions. These numbers are different from the two-dimensional case presented in [89]. Some numerical results on comparison of the hybrid WENO/polynomial scheme with different values of the threshold ζ will be presented in Chapter 7.

4.3.4 Smoothness indicator and non-linear weights

While the reconstruction based on linear weights works well for smooth solutions and relatively good unstructured meshes, our goal is to adapt it to the case where the solution is not smooth and the mesh quality is arbitrary. We still take a linear combination of the reconstructions using small stencils, but now, so-called non-linear weights $\{\omega_l\}_{l=1}^{s}$ are employed. Those are designed so that $\omega_l \approx \gamma_l$ in cells where the solution is smooth (so third order accuracy is maintained) and $\omega_l \approx 0$ otherwise to suppress oscillations. To obtain non-linear weights, first we compute the smoothness indicator SI_l for each small stencil $S^{(l)}$. The indicator shows how smooth the polynomial $p_1^{(l)}(\boldsymbol{x})$ is on T_0. That is, the smaller the smoothness indicator, the smoother the function $p_1^{(l)}(\boldsymbol{x})$ on T_0. For the linear reconstruction polynomial $p_1^{(l)}(\boldsymbol{x})$ defined on the target element T_0, the formula for the smoothness indicator SI_l can be obtained from the general form in [135] as

$$SI_l = \sum_{i=1}^{3} \int_{T_0} |T_0|^{-\frac{1}{3}} \left(\frac{\partial p_1^{(l)}(\boldsymbol{x})}{\partial x_i} \right)^2 dV. \tag{4.46}$$

Substituting (4.33) in (4.46) we get

$$SI_l = \sum_{i=1}^{3} \int_{T_0} |T_0|^{-\frac{1}{3}} \left(\frac{a_i^{(l)}\left(\boldsymbol{u}^{(l)}\right)}{|T_0|^{1/3}} \right)^2 dV = \sum_{i=1}^{3} \left(a_i^{(l)}\left(\boldsymbol{u}^{(l)}\right) \right)^2,$$

and finally

$$SI_l = \sum_{i=1}^{3} \left(\sum_{m=1}^{3} \boldsymbol{B}_{im}^{(l)} \left(\overline{u}_m^{(l)} - \overline{u}_0 \right) \right)^2. \tag{4.47}$$

Therefore the smoothness indicator is a quadratic function of the cell averages on the stencil. The coefficients of the matrix $\boldsymbol{B}_{im}^{(l)}$ are precomputed and stored at the initialization. For smooth parts of the solution the smoothness indicators should be all small and about the same size for all small stencils. On the other hand, if the sub-stencil S_l contains a discontinuity, then $SI_l \gg SI_m$ when S_m has no discontinuities.

Using the linear weights γ_l and smoothness indicators SI_l the non-linear weights are defined as

$$\omega_l = \frac{\tilde{\omega}_l}{\sum_{m=1}^{s} \tilde{\omega}_m}, \ \tilde{\omega}_l = \frac{\gamma_l}{(\varepsilon + SI_l)^2}, \tag{4.48}$$

where ε is a small number traditionally chosen to be between 10^{-6} and 10^{-40} to avoid division by zero. Following earlier works [11, 69], in the previous chapter the effect of the choice of ε on accuracy of classic WENO3 reconstruction for one-dimensional reconstructions on uniform meshes was studied. Even for this simple case the dependence of ε on the size of mesh was shown to be crucial for accuracy as $h \to 0$. Therefore one can assume that this dependance is even more important for reconstructions on three-dimensional meshes with high inhomogeneity in the element size. In our computations we test the choices

$$\varepsilon = h_i^k, \ k = 1, 2, 4. \tag{4.49}$$

It should be noted that if the problem is solved in non-normalized form, an appropriate scaling parameter f needs to be applied in the definition of ε to ensure accuracy of the WENO scheme [11].

Now to form the type II WENO reconstruction at the point $\boldsymbol{x}_k^{(j)}$, the linear weights γ_l in (4.42) are replaced by the non-linear ω_l defined in (4.48)

$$u^{WENO}\left(\boldsymbol{x}_k^{(j)}\right) = \sum_{l=1}^{s} \omega_l p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right). \tag{4.50}$$

If the splitting technique was applied to the linear weights, then WENO reconstruction will have the form

$$u^{WENO}\left(\boldsymbol{x}_k^{(j)}\right) = \sigma^+ \sum_{l=1}^{s} \omega_l^+ p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right) - \sigma^- \sum_{l=1}^{s} \omega_l^- p_1^{(l)}\left(\boldsymbol{x}_k^{(j)}\right), \tag{4.51}$$

where $\omega_l^{\pm}$ are computed from (4.48) using $\gamma_l^{\pm}$ in place of γ_l.

Some modifications to the computation of non-linear weights are suggested in literature to improve the quality of WENO reconstructions. For unstructured meshes the mapping technique introduced by Henrick et al. in [69]) is often suggested [135, 89]. In Chapter 3 we

discussed the mapped WENO scheme defined by weights (3.53) and found no improvement of order of accuracy for the third order case. It was found that mapping improves the resolution of smooth solutions but at the same time it increases the size of oscillations at discontinuities. This is due to the fact that mapping makes non-linear weights closer to linear, therefore, reducing the non-oscillatory effect of the scheme. The same effect can be obtained by taking large ε for smooth solutions. Moreover mapping also adds 20-30% to the CPU time, which make three-dimensional experiments even more expensive.

4.4 Chapter summary

In this chapter the details of implementation of the third order type II WENO reconstructions on unstructured meshes are presented. It follows the same framework as the one-dimensional case discussed in Chapter 3, except that the weights are defined in a least square sense. The main advantage of type II WENO over type I WENO is that it has higher accuracy for the same big stencil. The third order type I WENO scheme would require more storage for coefficients and CPU time. On the other hand type II WENO scheme requires the solution of a least square system to determine linear weights. In most cases the solution to that system contains negative components which create unstable results. Even the application of weight splitting technique does not guarantee a stable scheme on general unstructured meshes due to the presence of very negative weights in some reconstructions. It can be treated by using a hybrid scheme, where in case of very negative weights the WENO reconstruction is replaced by something else (type I WENO, polynomial reconstruction on the big stencil). The percentage of very negative weights usually varies between 1 and 3 depending on the mesh quality and the threshold. Another aspect that impacts the accuracy of the solution is the choice of ε in the definition of non-linear weights. Traditional choice of very small quantity that prevents division by zero will produce a highly inaccurate result, especially if the problem is not in its normalized form. Following earlier works and the analysis presented in Chapter 3, the choice of $\varepsilon = h^2$ is recommended for general problems with discontinuities and significant mesh irregularities. The choices discussed in this chapter are based on the analysis in Chapter 3 and extensive numerical experiments, parts of which are presented in Chapter 7.

Chapter 5

Multirate Runge-Kutta schemes

Many real life simulations require dealing with complicated geometries and highly non-uniform meshes. When explicit methods are used, the maximum allowed time-step is defined by the smallest elements in the mesh. When a fine mesh is required only in a small region of a computational domain, it is not a desirable expense. In addition to that, when small time step is used on a coarse grid, it often generates dissipation in the solution. To overcome the need for a restrictive time step requirement, local time-stepping (LTS) or multirate methods are very useful. In this case local stability conditions (CFL) are imposed on sub-domains of the computational domain in place of a global more restrictive stability condition. The idea of multirate time-stepping is to evolve different components of the solution with different time-steps and sometimes with different schemes using special coupling between the components. Constructing proper interface conditions so that the multirate method retains the properties of the base method is the main challenge in the development of such schemes. In this chapter we will study some of these properties for multirate methods based on Runge-Kutta schemes.

5.1 Multirate schemes overview

The earliest works on multirate methods include multirate Runge-Kutta schemes by Rice [105] and Andrus [8, 9], multirate linear multi-step by Gear and Wells [49], and local time-stepping with forward Euler by Osher and Sanders [100]. Over the last three decades

multirate versions of many traditional temporal schemes, such as explicit Runge-Kutta [31, 32, 61, 86, 125], Adams-Bashforth [108], as well as implicit-explicit (IMEX) methods [111] were designed.

In the pioneer work [100] Osher and Sanders proposed the local time-stepping scheme with forward Euler as a base method, while finite volume method on non-uniform grid was used for the spatial discretization. The solution is evolved in time by a full time step on the coarsest grid and a number of smaller step on finer ones. They proved convergence of the solution obtained by their local forward Euler scheme to the entropy solution of conservation laws. Dawson and Kirby extended the work by Osher and Sanders in [32]. They showed a maximum principle for a local forward Euler method when limited slopes are present. Then they also showed that the same idea can be extended to second order time discretization using SSP Runge-Kutta method. They proved L^∞ stability for the second order scheme for a constant coefficient case. Later Tang and Warnecke [125] proposed an alternative to the Osher and Sanders's scheme. Their approach is based on a simple projection of the solution increments at each local time-step. The scheme is internally consistent, but not mass-conserving at the interface between two multirate subdomains. The second order extension of their approach is based on the SSP RK2 method and is also internally consistent and second order accurate.

In [31] Constantinescu and Sandu proposed another multirate scheme based on the second order SSP RK2. They presented rigorous analysis of their scheme, proving that it is conservative and second order accurate. Their method was further tested in the framework of large-scale marine flows using DG method for space approximation by Seny et al. in [113]. They tried up to fourth order Runge-Kutta base methods and achieved second order accuracy in all cases. But it was also pointed out, that the third order Runge-Kutta appeared to be slightly more efficient than others.

Searching for the third order multirate approach we find the scheme developed by Schlegel et al. in [111]. Their method is based on the implicit/explicit (IMEX) method introduced by Knoth and Wolke in [81], where the stiff term is treated implicitly. The multirate approach is developed for the explicit term. They showed that it is third order accurate with a specific third order Runge-Kutta methods, which satisfy an additional condition.

Methods discussed so far were designed for non-linear conservation laws. In [86] Liu, Li and Hu presented another multirate method which, according to authors, preserves the order of accuracy of the base Runge-Kutta method for linear problems.

5.2 MPRK methods in Butcher form

Following the method of lines (MOL) framework, we introduce the semi-discrete problem defined by the ODE

$$u_t = Lu \tag{5.1}$$

on some bounded region $\Omega \subset \mathbb{R}$ with a given initial value $u(0) = u^0$. Here the operator L represents the spatial approximation of the linear operator in the conservation law with some given order. Consider the partition of the computational domain into two subdomains $\Omega = D_1 \cup D_2 \cup \Gamma_{12}$, where D_1 represents the fine mesh with size $h/2$ and D_1 is the coarse mesh with size h. Here $\Gamma_{12} = \partial D_1 \cap \partial D_2$ is the boundary between subdomains D_1 and D_2. Assuming that the local time-step satisfying CFL condition on D_2 is Δt, then the local time-step on D_1 is $\Delta t/2$. Define by P_1 and P_2 two projection matrices onto domains D_1 and D_2, respectively, then we can split the right-hand side in (5.1) as

$$u_t = P_1 Lu + P_2 Lu. \tag{5.2}$$

For the analysis of multirate Runge-Kutta schemes as done in [31, 75, 111], it is convenient to consider their partitioned form, referred to as multirate partitioned Runge-Kutta (MPRK) methods. An s-stage explicit Runge-Kutta method for (5.1) can be written in the Butcher form [20] as

$$u^{(i)} = u^n + \Delta t \sum_{j=1}^{i-1} a_{ij} L u^{(j)},\ i = 1, \ldots, s, \tag{5.3}$$

$$u^{n+1} = u^n + \Delta t \sum_{i=1}^{s} b_i L u^{(i)}. \tag{5.4}$$

Let us define the matrix $\boldsymbol{A} \in \mathbb{R}^{s \times s}$ and vectors $\boldsymbol{b}, \boldsymbol{c} \in \mathbb{R}^s$ as

0	0	0
1	1	0
	1/2	1/2

(a) SSP RK22

0	0	0	0
1	1	0	0
1/2	1/4	1/4	0
	1/6	1/6	2/3

(b) SSP RK3

0	0	0	0
1	1	0	0
2	1	1	0
	2/3	1/6	1/6

(c) SSP LRK3

0	0	0	0
1/2	1/2	0	0
1	1/2	1/2	0
	1/3	1/3	1/3

(d) SSP LRK32

0	0	0	0	0
1/2	1/2	0	0	0
1	1/2	1/2	0	0
3/2	1/2	1/2	1/2	0
	1/2	1/6	1/6	1/6

(e) SSP LRK43

Table 5.1: Butcher tableau for the second and third order SSP Runge-Kutta schemes.

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ a_{21} & 0 & \cdots & \cdots & 0 \\ a_{31} & a_{32} & \ddots & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & a_{s,s-1} & 0 \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_s \end{bmatrix}, \boldsymbol{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_s \end{bmatrix}, \tag{5.5}$$

where

$$c_i = \sum_{j=1}^{s} a_{ij}.$$

Then the scheme (5.3-5.4) can be presented in a tableau form [20]

$$\begin{array}{c|c} \boldsymbol{c} & \boldsymbol{A} \\ \hline & \boldsymbol{b}^T \end{array}.$$

Butcher tableau of some of the Runge-Kutta methods are presented in Table 5.1 .

The partitioned Runge-Kutta form of an s-stage multirate method for (5.2) with two levels of refinement (local time-steps) in our notations can be written as

$$u^{(i)} = u^n + \Delta t \sum_{j=1}^{i-1} \left(a_{ij}^{(1)} P_1 L u^{(j)} + a_{ij}^{(2)} P_2 L u^{(j)} \right),\ i = 1, \ldots, s, \tag{5.6}$$

$$u^{n+1} = u^n + \Delta t \sum_{i=1}^{s} \left(b_i^{(1)} P_1 L u^{(i)} + b_i^{(2)} P_2 L u^{(i)} \right). \tag{5.7}$$

It should be noted that the time-step factor is taken into account in the coefficients $a_{ij}^{(1)}$ and $a_{ij}^{(2)}$ and s is the number of MPRK stages. Using the above form of MPRK method, we can determine the accuracy of the scheme using the classic order conditions.

5.3 Order conditions

In this subsection the order conditions for multirate schemes based on Runge-Kutta methods are reviewed. It is assumed that the order of the spatial approximation is the same as the order of the base Runge-Kutta method. Therefore, the spatial truncation error can be ignored, assuming that it is of the same order as the temporal one.

For our order analysis of multirate schemes based on the Runge-Kutta methods, we need to derive their partitioned form. Then the order conditions derived for PRK methods can be used. The order conditions consist of the order conditions for each $\left(\boldsymbol{A}^{(1)}, \boldsymbol{b}^{(1)}, \boldsymbol{c}^{(1)}\right)$ and $\left(\boldsymbol{A}^{(2)}, \boldsymbol{b}^{(2)}, \boldsymbol{c}^{(2)}\right)$ and additional coupling conditions. For the explicit Runge-Kutta methods in Butcher form (5.3-5.4) the formal order conditions can be derived [63]. These conditions for order up to three are listed below

$$\text{1st order :} \quad \boldsymbol{b}^T \mathbf{1} = 1 \tag{5.8}$$

$$\text{2nd order :} \quad \boldsymbol{b}^T \boldsymbol{c} = \frac{1}{2}, \tag{5.9}$$

$$\text{3rd order :} \quad \boldsymbol{b}^T \boldsymbol{C} \boldsymbol{c} = \frac{1}{3}, \tag{5.10}$$

$$\boldsymbol{b}^T \boldsymbol{A} \boldsymbol{c} = \frac{1}{6}, \tag{5.11}$$

where

$$\mathbf{1} = \underbrace{[1,\ldots,1]}_{s}^{T}, \text{ and } \boldsymbol{C} = diag\{\boldsymbol{c}\}. \tag{5.12}$$

The above conditions are for the Runge-Kutta methods applied to general non-linear conservation laws. Runge-Kutta methods designed specifically for linear problems like (c)-(d) in Table 5.1 do not satisfy the third order condition (5.10), therefore are only second order accurate for non-linear problems. It can be shown that only (5.11) is needed for the third order accuracy in the linear case.

The order conditions for partitioned Runge-Kutta methods were derived by several authors. Hairer [62] developed a theory of P-series to obtain the order conditions for (5.6-5.7). In [76] these conditions were derived using the order theory of Albrecht [7]. The order conditions for PRK method of order up to three are listed below

$$\text{2nd order} : \left(\boldsymbol{b}^{(d_1)}\right)^T \boldsymbol{c}^{(d_2)} = \frac{1}{2},\ d_1,d_2 = 1,2, \tag{5.13}$$

$$\text{3rd order} : \quad \left(\boldsymbol{b}^{(d_1)}\right)^T \boldsymbol{C}^{(d_2)}\boldsymbol{c}^{(d_3)} = \tfrac{1}{3}, \quad d_1,d_2,d_3 = 1,2, \tag{5.14}$$

$$\left(\boldsymbol{b}^{(d_1)}\right)^T \boldsymbol{A}^{(d_2)}\boldsymbol{c}^{(d_3)} = \tfrac{1}{6}, \quad d_1,d_2,d_3 = 1,2. \tag{5.15}$$

The second order conditions are satisfied by all multirate schemes based on SSP RK2. Since the number of order conditions quickly increases with order, it becomes challenging to generate a higher than 2nd order multirate method. As it will be seen later, extensions of second order multirate methods by using third order Runge-Kutta base methods does not automatically generate the third order MPRK method. The number of third order conditions can be reduced if only linear problems, as in this work, are considered. For simplicity of the notations we define

$$L_1 = P_1 L, \quad L_2 = P_2 L,$$

and then the operator L can be decomposed as

$$L = L_1 + L_2. \tag{5.16}$$

Note that the discrete operator L is approximated by a non-linear scheme. But one can still assume linearity of L since the same proofs are valid for the non-discretized problem. In the remainder of this subsection it will be shown that the order conditions given by (5.14) can be dropped if the operator L is assumed to be linear. To show this first the definition of the local truncation error is given:

Definition 1. The local truncation error, e^{n+1}, at the time-step $t^{n+1} = t^n + \Delta t$ is an estimate of the error after one time-step defined by

$$\tau^{n+1} = u^{n+1} - v\left(t^{n+1}\right), \tag{5.17}$$

where v is defined by

$$\begin{aligned} v_t &= Lv, \\ v\left(t^n\right) &= u^n. \end{aligned}$$

The temporal discretization method is consistent of order 1 if the local truncation error is $O(\Delta t^2)$, and is p-th order accurate if the local truncation error is $O\left(\Delta t^{p+1}\right)$. Now the order conditions for MPRK method of order three for linear problems are stated in the following theorem:

Theorem 5. *The multirate partitioned Runge-Kutta method of the form*

$$u^{(i)} = u^n + \Delta t \sum_{k=1,2} \sum_{j=1}^{i-1} a_{ij}^{(k)} L_k u^{(j)},\ i = 1, \ldots, s, \tag{5.18}$$

$$u^{n+1} = u^n + \Delta t \sum_{k=1,2} \sum_{i=1}^{s} b_i^{(k)} L_k u^{(i)}, \tag{5.19}$$

where L_1 and L_2 are linear constant-coefficient operators, is third order accurate if the following order conditions are satisfied

$$\textit{1st order :} \qquad \left(\boldsymbol{b}^{(k_1)}\right)^T \mathbf{1} = 1 \qquad k_1,k_2 = 1,2, \tag{5.20}$$

$$\textit{2nd order :} \qquad \left(\boldsymbol{b}^{(k_1)}\right)^T \boldsymbol{c}^{(k_2)} = \tfrac{1}{2}, \qquad k_1,k_2 = 1,2, \tag{5.21}$$

$$\textit{3rd order :} \quad \left(\boldsymbol{b}^{(k_1)}\right)^T \boldsymbol{A}^{(k_2)}\boldsymbol{c}^{(k_3)} = \tfrac{1}{6}, \quad k_1,k_2,k_3 = 1,2. \tag{5.22}$$

Proof. From (5.18) using linearity of L_1 and L_2 the stage values $u^{(i)}$ can be expressed in terms of u^n as follows

$$\begin{aligned} u^{(i)} = u^n + \Delta t \sum_{k_1=1,2} c_i^{(k_1)} L_{k_1} u^n + \Delta t^2 \sum_{k_1,k_2=1,2} \sum_{j=1}^{s} a_{ij}^{(k_1)} c_j^{(k_2)} L_{k_1} L_{k_2} u^n + \dots \\ + \Delta t^{i-1} \sum_{k_1,\dots,k_{i-1}=1,2} a_{i,i-1}^{(k_1)} \dots a_{21}^{(k_{i-1})} L_{k_1} \dots L_{k_{i-1}} u^n. \end{aligned} \tag{5.23}$$

Substituting (5.23) in (5.19), the numerical solution at time $t = t^{n+1}$ can be written as

$$\begin{aligned} u^{n+1} = u^n + \Delta t \sum_{k_1=1,2} \left(\boldsymbol{b}^{(k_1)}\right)^T \mathbf{1} L_{k_1} u^n + \\ \Delta t^2 \sum_{k_1,k_2=1,2} \left(\boldsymbol{b}^{(k_1)}\right)^T \boldsymbol{c}^{(k_2)} L_{k_1} L_{k_2} u^n + \\ \Delta t^3 \sum_{k_1,k_2,k_3=1,2} \left(\boldsymbol{b}^{(k_1)}\right)^T \boldsymbol{A}^{(k_1)} \boldsymbol{c}^{(k_2)} L_{k_1} L_{k_2} L_{k_3} u^n + \dots + \\ \Delta t^s \sum_{k_1,\dots,k_i=1,2} b_s^{(k_1)} a_{s,s-1}^{(k_2)} \dots a_{21}^{(k_s)} L_{k_1} \dots L_{k_s} u^n, \end{aligned}$$

where $\mathbf{1}$ is defined by (5.12) with s equal to the size of $\boldsymbol{b}^{(k_1)}$. To find the local truncation error using Definition 1, one can get

$$
\begin{aligned}
v\left(t^{n+1}\right) =& v\left(t^n\right) + \sum_{j=1}^{\infty} \frac{\Delta t^j}{j!} \frac{\partial^j v}{\partial t^j}\left(t^n\right) \\
=& u^n + \sum_{j=1}^{\infty} \frac{1}{j!} \Delta t^j \sum_{k_1,\ldots k_j=1,2} L_{k_1} \ldots L_{k_j} v\left(t^n\right) \\
=& u^n + \Delta t \sum_{k_1=1,2} L_{k_1} u^n + \frac{1}{2} \Delta t^2 \sum_{k_1,k_2=1,2} L_{k_1} L_{k_2} u^n + \\
& \frac{1}{6} \Delta t^3 \sum_{k_1,k_2,k_3=1,2} L_{k_1} L_{k_2} L_{k_3} u^n + O\left(t^4\right).
\end{aligned}
$$

Taking the difference between u^{n+1} and $v\left(t^{n+1}\right)$ the following expression for the local truncation error at t^{n+1} are derived

$$
\begin{aligned}
\tau^{n+1} =& \Delta t \left[\sum_{k_1=1,2} \left(1 - \left(\boldsymbol{b}^{(k_1)} \right)^T \mathbf{1} \right) L_{k_1} \right] v^n + \\
& \Delta t^2 \left[\sum_{k_1,k_2=1,2} \left(\frac{1}{2} - \left(\boldsymbol{b}^{(k_1)} \right)^T \boldsymbol{c}^{(k_2)} \right) L_{k_1} L_{k_2} \right] v^n + \\
& \Delta t^3 \left[\sum_{k_1,k_2,k_3=1,2} \left(\frac{1}{6} - \left(\boldsymbol{b}^{(k_1)} \right)^T \boldsymbol{A}^{(k_2)} \boldsymbol{c}^{(k_3)} \right) L_{k_1} L_{k_2} L_{k_3} \right] v^n + O\left(\Delta t^4\right).
\end{aligned}
\tag{5.24}
$$

Now the truncation error is $O\left(\Delta t^4\right)$ if the first three term in (5.24) are zero. This gives the order conditions (5.20-5.22). □

Note that the above order conditions are necessary for elements that are close to the interface between sub-domains with different time-steps. Away from Γ_{12} the base method $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$ with the desired order of accuracy is used. As it will be demonstrated later by numerical experiments for two time increments, the application of the multirate method with lower accuracy at the interface affects the solution considerably. Therefore, the overall accuracy of the solution will decrease with every added LTS subdomain.

5.4 Conservation and consistency incompatibility

The scheme (5.6-5.7) is called internally consistent [75], if

$$\boldsymbol{c}^{(1)} = \boldsymbol{c}^{(2)}. \tag{5.25}$$

This condition ensures that the stage values on adjacent subdomains are consistent approximations to $u(t^n + c_i\Delta t)$. Failure to satisfy the internal consistency condition may lead to lower accuracy at interface points.

Another very important property for schemes solving conservation laws is conservation. Consider, for example, the one-dimensional finite volume scheme for a linear advection equation (3.55) given by

$$\frac{d\overline{u}_i}{dt} + \frac{1}{h}F_i(\overline{u}) = 0,$$

where $F_i(\overline{u}) = \hat{u}_{i+\frac{1}{2}} - \hat{u}_{i-\frac{1}{2}}$. The finite volume discretization is conservative in the sense that

$$\sum_i \frac{d\overline{u}_i}{dt}h = \sum_i \left[\hat{u}_{i+\frac{1}{2}} - \hat{u}_{i-\frac{1}{2}}\right] = 0$$

and hence

$$\sum_i \overline{u}_i h = const.$$

Therefore the time discretization combined with the conservative space approximation gives a conservative fully discrete method if

$$\sum_i \overline{u}_i^{n+1} h = \sum_i \overline{u}_i^n h.$$

For the MPRK scheme we have

$$h\overline{u}_i^{n+1} = h\overline{u}_i^n + h\Delta t \sum_{j=1}^{s} \left(b_j^{(1)} P_1 F_i\left(\overline{u}^{(j)}\right) + b_j^{(2)} P_2 F_i\left(\overline{u}^{(j)}\right)\right).$$

Taking the summation and assuming that $i+1/2 = i_0+1/2$ is the interface between the multirate domains, we get

$$\begin{aligned} 0 =& h\Delta t \sum_i \sum_{j=1}^{s} \left(b_j^{(1)} P_1 F_i \left(\overline{u}^{(j)} \right) + b_j^{(2)} P_2 F_i \left(\overline{u}^{(j)} \right) \right) \\ =& h\Delta t \sum_{j=1}^{s} \left(b_j^{(1)} - b_j^{(2)} \right) \hat{u}_{i_0+\frac{1}{2}}^{(j)}. \end{aligned}$$

This result is summarized in the following theorem:

Theorem 6. (Constantinescu and Sandu [31]) *Any partitioned Runge-Kutta method (5.6-5.7) is conservative if*

$$\boldsymbol{b}^{(1)} = \boldsymbol{b}^{(2)}. \tag{5.26}$$

As was pointed out in [75], the incompatibility of consistency and conservation is typical for all MPRK schemes that are based on single base method. General partitioned Runge-Kutta methods can be modified to provide both consistency and conservation by adding extra stages, but this does not lead to an efficient multirate scheme.

5.5 Tang-Warnecke scheme

In [125] Tang and Warnecke developed the multirate scheme based on a simple projection of the solution at each local time-step. We will refer to their scheme as MPRK-TW. For a partition into two sub-domains D_1 and D_2 defined by local time-steps

$$\Delta t_1 = \frac{\Delta t}{2}, \qquad \Delta t_2 = \Delta t. \tag{5.27}$$

The MPRK2-TW scheme can be written as

$$D_1: \quad \begin{array}{c|c} \boldsymbol{c}^{(1)} & \boldsymbol{A}^{(1)} \\ \hline & \left[\boldsymbol{b}^{(1)}\right]^T \end{array} \quad = \quad \begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 \\ 1/2 & 1/4 & 1/4 & 0 & 0 \\ 1 & 1/4 & 1/4 & 1/2 & 0 \\ \hline & 1/4 & 1/4 & 1/4 & 1/4 \end{array}$$

$$D_2: \quad \begin{array}{c|c} \boldsymbol{c}^{(2)} & \boldsymbol{A}^{(2)} \\ \hline & \left[\boldsymbol{b}^{(2)}\right]^T \end{array} \quad = \quad \begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 \\ 1/2 & 1/4 & 1/4 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ \hline & 1/2 & 0 & 0 & 1/2 \end{array}$$

Table 5.2: Butcher tableau of MPRK-TW scheme (5.28-5.32).

$$u^{(1)} = u^n, \tag{5.28}$$

$$u^{(2)} = u^n + \frac{1}{2}\Delta t L u^{(1)}, \tag{5.29}$$

$$u^{(3)} = u^n + \frac{1}{4}\Delta t L\left(u^{(1)} + u^{(2)}\right), \tag{5.30}$$

$$u^{(4)} = u^n + \frac{1}{2}\Delta t P_1\left(\frac{1}{2}\left(Lu^{(1)} + Lu^{(2)}\right) + Lu^{(3)}\right) + \Delta t P_2 L u^{(1)}, \tag{5.31}$$

$$u^{n+1} = u^n + \frac{1}{4}\Delta t P_1 L\sum_{i=1}^{4} u^{(i)} + \frac{1}{2}\Delta t P_2 L\left(u^{(1)} + u^{(4)}\right), \tag{5.32}$$

or in the form of Butcher tableau in Table 5.2.

To generalize their scheme to an arbitrary base method $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$ the following decompositions are introduced

$$\boldsymbol{A} = \boldsymbol{A}_1 + \boldsymbol{A}_2 = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ a_{21} & 0 & \cdots & 0 \\ a_{31} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{s1} & 0 & \cdots & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & \cdots & 0 \\ 0 & a_{32} & \ddots & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & a_{s2} & \cdots & a_{s,s-1} & 0 \end{bmatrix}, \tag{5.33}$$

$$\boldsymbol{b} = \boldsymbol{b}_1 + \boldsymbol{b}_2 = b_1\hat{\boldsymbol{e}}_1 + (\boldsymbol{b} - b_1\hat{\boldsymbol{e}}_1), \tag{5.34}$$

$$D_1: \quad \begin{array}{c|c} \boldsymbol{c}^{(1)} & \boldsymbol{A}^{(1)} \\ \hline & \left[\boldsymbol{b}^{(1)}\right]^T \end{array} \quad = \quad \begin{array}{c|cc} \frac{1}{2}\boldsymbol{c} & \frac{1}{2}\boldsymbol{A} & \\ \frac{1}{2}\mathbf{1}+\frac{1}{2}\boldsymbol{c} & \frac{1}{2}\boldsymbol{b}^T\otimes\mathbf{1} & \frac{1}{2}\boldsymbol{A} \\ \hline & \frac{1}{2}\boldsymbol{b}^T & \frac{1}{2}\boldsymbol{b}^T \end{array}$$

$$D_2: \quad \begin{array}{c|c} \boldsymbol{c}^{(2)} & \boldsymbol{A}^{(2)} \\ \hline & \left[\boldsymbol{b}^{(2)}\right]^T \end{array} \quad = \quad \begin{array}{c|cc} \frac{1}{2}\boldsymbol{c} & \frac{1}{2}(\boldsymbol{A}_1+\boldsymbol{A}_2) & \\ \frac{1}{2}\hat{\boldsymbol{e}}_1+\boldsymbol{c} & \frac{1}{2}\boldsymbol{b}^T\otimes\hat{\boldsymbol{e}}_1+\boldsymbol{A}_1 & \boldsymbol{A}_2 \\ \hline & \boldsymbol{b}_1^T & \boldsymbol{b}_2^T \end{array}$$

Table 5.3: MPRK-TW scheme for arbitrary base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ and time-step ratio 2.

where $\hat{\boldsymbol{e}}_1 = [\underbrace{1,0,...,0}_{s}]^T$. Then the extension of MPRK-TW scheme to an arbitrary base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ can be given in the Butcher form by Table 5.3, where $\mathbf{1}$ is defined by (5.12).

From Tables 5.2 and 5.3 we notice that $\boldsymbol{b}^{(1)} \neq \boldsymbol{b}^{(2)}$. Therefore, the MPRK scheme by Tang and Warnecke is not conservative. To study the accuracy of MPRK-TW scheme we first look at the internal consistency of the method.

Theorem 7. *The partitioned Runge-Kutta scheme defined by the Butcher tableau in Table 5.3 is internally consistent if*

$$\boldsymbol{c} = \mathbf{1} - \hat{\boldsymbol{e}}_1, \tag{5.35}$$

Proof. The proof is a straight forward application of the condition (5.25). □

Corollary 1. *The partitioned Runge-Kutta scheme defined by the Butcher tableau in Table 5.3 is internally consistent with SSP RK2 as a base method.*

The application of the consistency condition (5.35) reveals that the MPRK-TW scheme is internally inconsistent with other base methods in Table 5.1.

Now the order conditions can be applied to the scheme in Table 5.3. The following results can be proven based on order conditions given in Subsection 5.3.

Theorem 8. *The partitioned Runge-Kutta scheme defined by the Butcher tableau in Table 5.3 is second order accurate if the base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ is at least second order accurate and satisfies*

$$b_1 = \frac{1}{2}. \tag{5.36}$$

The scheme has at most second order accurate coupling regardless of the base method.

Proof. Assume that the base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ satisfies the second order conditions (5.8-5.9). First we check the order conditions for the methods $\left(\boldsymbol{A}^{(1)},\boldsymbol{b}^{(1)},\boldsymbol{c}^{(1)}\right)$ and $\left(\boldsymbol{A}^{(2)},\boldsymbol{b}^{(2)},\boldsymbol{c}^{(2)}\right)$ separately. The first order condition (5.8) is satisfied by both methods since

$$\begin{aligned}\left(\boldsymbol{b}^{(1)}\right)^T\mathbf{1} &= \frac{1}{2}(\boldsymbol{b})^T\mathbf{1}+\frac{1}{2}(\boldsymbol{b})^T\mathbf{1}=1,\\ \left(\boldsymbol{b}^{(2)}\right)^T\mathbf{1} &= (\boldsymbol{b}_1)^T\mathbf{1}+(\boldsymbol{b}_2)^T\mathbf{1}=(\boldsymbol{b})^T\mathbf{1}=1.\end{aligned}$$

The second order condition (5.9) is also satisfied by both methods

$$\begin{aligned}\left(\boldsymbol{b}^{(1)}\right)^T\boldsymbol{c}^{(1)} &= \frac{1}{2}\boldsymbol{b}^T\left(\frac{1}{2}\boldsymbol{c}+\frac{1}{2}\mathbf{1}+\frac{1}{2}\boldsymbol{c}\right)=\frac{1}{4}\left(2(\boldsymbol{b})^T\boldsymbol{c}+(\boldsymbol{b})^T\mathbf{1}\right)=\frac{1}{2},\\ \left(\boldsymbol{b}^{(2)}\right)^T\boldsymbol{c}^{(2)} &= \left(b_1\hat{\boldsymbol{e}}_1^T\right)\frac{1}{2}\boldsymbol{c}+(\boldsymbol{b}-b_1\hat{\boldsymbol{e}}_1)^T\left(\frac{1}{2}\hat{\boldsymbol{e}}_1+\boldsymbol{c}\right)=(\boldsymbol{b})^T\boldsymbol{c}=\frac{1}{2}.\end{aligned}$$

The second order coupling conditions (5.13-5.15) applied to the scheme in Table 5.3 give the following

$$\begin{aligned}b_i^{(1)}c_i^{(2)} &= \frac{1}{2}\boldsymbol{b}^T\left(\frac{1}{2}\boldsymbol{c}+\frac{1}{2}\hat{\boldsymbol{e}}_1+\boldsymbol{c}\right)=\frac{3}{4}(\boldsymbol{b})^T\boldsymbol{c}+\frac{1}{4}b_1=\frac{1}{2} && \Leftrightarrow \quad b_1=\frac{1}{2},\\ b_i^{(2)}c_i^{(1)} &= \left(b_1\hat{\boldsymbol{e}}_1^T\right)\frac{1}{2}\boldsymbol{c}+(\boldsymbol{b}-b_1\hat{\boldsymbol{e}}_1)^T\left(\frac{1}{2}\mathbf{1}+\frac{1}{2}\boldsymbol{c}\right)\\ &= \frac{1}{2}(\boldsymbol{b})^T\mathbf{1}-\frac{1}{2}b_1+\frac{1}{2}(\boldsymbol{b})^T\boldsymbol{c}=\frac{1}{2} && \Leftrightarrow \quad b_1=\frac{1}{2}.\end{aligned}$$

Hence the method is second order accurate provided that $b_1=\frac{1}{2}$.

Assume that the base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ is second order accurate and also satisfies the third order conditions for linear problems (5.15). One of the linear coupling conditions gives

$$\left(\boldsymbol{b}^{(1)}\right)^T \boldsymbol{A}^{(2)}\boldsymbol{c}^{(2)} = \frac{1}{2}\boldsymbol{b}^T\frac{1}{2}\boldsymbol{A}\frac{1}{2}\boldsymbol{c} + \frac{1}{2}\boldsymbol{b}^T\left(\frac{1}{2}\boldsymbol{b}^T\otimes\hat{\boldsymbol{e}}_1 + \boldsymbol{A}_1\right)\frac{1}{2}\boldsymbol{c} + \frac{1}{2}\boldsymbol{b}^T\boldsymbol{A}_2\left(\frac{1}{2}\hat{\boldsymbol{e}}_1 + \boldsymbol{c}\right)$$
$$= \frac{1}{8}\boldsymbol{b}^T\boldsymbol{A}\boldsymbol{c} + \frac{1}{16}b_1 + \frac{1}{2}\boldsymbol{b}^T\boldsymbol{A}\boldsymbol{c} = \frac{5}{48} + \frac{1}{16}b_1,$$

where the fact that $\boldsymbol{A}_2\boldsymbol{c} = \boldsymbol{A}\boldsymbol{c}$ is used. Therefore, if the condition $b_1 = \frac{1}{2}$ is imposed to satisfy second order accuracy we get

$$\left(\boldsymbol{b}^{(1)}\right)^T \boldsymbol{A}^{(2)}\boldsymbol{c}^{(2)} = \frac{13}{96}.$$

Hence, the scheme can only achieve the second order accuracy even for linear problems. □

Corollary 2. *The partitioned Runge-Kutta scheme defined by the Butcher tableau in Table 5.3 is second order accurate with SSP RK2 as a base method.*

From Corollaries 1 and 2 we can conclude that SSP RK2 is the only good candidate for the MPRK-TW scheme. It will be confirmed numerically that application of the higher order method with $b_1 \neq \frac{1}{2}$ (e.g. SSP RK3) in Tang-Warnecke scheme produces solutions with significantly larger error since it is only first order accurate at the interface between two LTS subdomains.

The main advantage of the method presented in this subsection is that it is very flexible in terms of time-step ratios of LTS subdomains. It can be applied to the cases when the time-step ratio is an arbitrary $\kappa \in \mathbb{Z}$, or to multi-time increments $\kappa_1\Delta t, \kappa_2\Delta t, ..., \kappa_k\Delta t$, where $\sum_{i=1}^{k}\kappa_i = 1$. In Table 5.4 we present the extension of the MPRK-TW scheme to the case when $\kappa^{-1} = k \in \mathbb{N}$ and arbitrary base method $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$. A more arbitrary time-step ratio, when $\kappa \neq k^{-1}$, will be discussed in Chapter 6 for Maxwell's equations.

It can be verified that the same accuracy results as in Theorems 7 and 8 are valid for the scheme given by Table 5.4.

Theorem 9. *The partitioned Runge-Kutta scheme defined by the Butcher tableau in Table 5.4 is internally consistent if the base method $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$ satisfies (5.35), and is second order accurate if the base method is at least second order accurate and satisfies (5.36).*

D_1:

$$\begin{array}{c|cccc}
\frac{1}{k}\boldsymbol{c} & \frac{1}{k}\boldsymbol{A} & & & \\
\frac{1}{k}\mathbf{1}+\frac{1}{k}\boldsymbol{c} & \frac{1}{k}\boldsymbol{b}^T\otimes\mathbf{1} & \frac{1}{k}\boldsymbol{A} & & \\
\vdots & \vdots & \ddots & \ddots & \\
\frac{k-1}{k}\mathbf{1}+\frac{1}{k}\boldsymbol{c} & \frac{1}{k}\boldsymbol{b}^T\otimes\mathbf{1} & \cdots & \frac{1}{k}\boldsymbol{b}^T\otimes\mathbf{1} & \frac{1}{k}\boldsymbol{A} \\
\hline
 & \frac{1}{k}\boldsymbol{b}^T & \cdots & \frac{1}{k}\boldsymbol{b}^T & \frac{1}{k}\boldsymbol{b}^T
\end{array}$$

D_2:

$$\begin{array}{c|ccccccc}
\frac{1}{k}\boldsymbol{c} & \frac{1}{k}(\boldsymbol{A}_1+\boldsymbol{A}_2) & & & & & \\
\frac{1}{k}\hat{\boldsymbol{e}}_1+\frac{2}{k}\boldsymbol{c} & \frac{1}{k}\boldsymbol{b}^T\otimes\hat{\boldsymbol{e}}_1+\frac{2}{k}\boldsymbol{A}_1 & \frac{2}{k}\boldsymbol{A}_2 & & & & \\
\frac{2}{k}\hat{\boldsymbol{e}}_1+\frac{3}{k}\boldsymbol{c} & \frac{2}{k}\boldsymbol{b}_1^T\otimes\hat{\boldsymbol{e}}_1+\frac{3}{k}\boldsymbol{A}_1 & \frac{2}{k}\boldsymbol{b}_2^T\otimes\hat{\boldsymbol{e}}_1 & \frac{3}{k}\boldsymbol{A}_2 & & & \\
\vdots & \vdots & & \ddots & \ddots & & \\
\vdots & \vdots & & & \ddots & \ddots & \\
\frac{k-1}{k}\hat{\boldsymbol{e}}_1+\boldsymbol{c} & \frac{k-1}{k}\boldsymbol{b}_1^T\otimes\hat{\boldsymbol{e}}_1+\boldsymbol{A}_1 & \mathbf{0} & \mathbf{0} & \cdots & \frac{k-1}{k}\boldsymbol{b}_2^T\otimes\hat{\boldsymbol{e}}_1 & \boldsymbol{A}_2 \\
\hline
 & \boldsymbol{b}_1^T & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \boldsymbol{b}_2^T
\end{array}$$

Table 5.4: MPRK-TW scheme for arbitrary base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ and the time-step ratio $k\in\mathbb{N}$.

Proof. The consistency requirement for the scheme in Table 5.4 is

$$\frac{i}{k}\mathbf{1}+\frac{1}{k}\boldsymbol{c}=\frac{i}{k}\hat{\boldsymbol{e}}_1+\frac{i+1}{k}\boldsymbol{c},\qquad i=0,1,...,k-1,$$

which gives (5.35).

The order conditions for methods $\left(\boldsymbol{A}^{(1)},\boldsymbol{b}^{(1)},\boldsymbol{c}^{(1)}\right)$ and $\left(\boldsymbol{A}^{(2)},\boldsymbol{b}^{(2)},\boldsymbol{c}^{(2)}\right)$ are satisfied since

$$\begin{aligned}
\left(\boldsymbol{b}^{(1)}\right)^T\boldsymbol{c}^{(1)}&=\frac{1}{k}\boldsymbol{b}^T\left(\frac{\sum_{i=1}^{k-1}i}{k}\mathbf{1}+\boldsymbol{c}\right)=\frac{k-1}{2k}+\frac{1}{2k}=\frac{1}{2},\\
\left(\boldsymbol{b}^{(2)}\right)^T\boldsymbol{c}^{(2)}&=\boldsymbol{b}_2^T\boldsymbol{c}=\frac{1}{2}.
\end{aligned}$$

The second order coupling conditions applied to the generalized scheme give

$$\begin{aligned}
\left(\boldsymbol{b}^{(1)}\right)^T \boldsymbol{c}^{(2)} =& \frac{1}{k}\boldsymbol{b}^T\left(\frac{1}{k}\boldsymbol{c}+\frac{1}{k}\hat{\boldsymbol{e}}_1+\frac{2}{k}\boldsymbol{c}+\frac{2}{k}\hat{\boldsymbol{e}}_1+\ldots+\frac{k-1}{k}\hat{\boldsymbol{e}}_1+\boldsymbol{c}\right) \qquad \Leftrightarrow \quad b_1=\frac{1}{2},\\
=& \frac{k+1}{4k}+\frac{k-1}{2k}b_1=\frac{1}{2}\\
\left(\boldsymbol{b}^{(2)}\right)^T \boldsymbol{c}^{(1)} =& \boldsymbol{b}_1^T\frac{1}{k}\boldsymbol{c}+\boldsymbol{b}_2^T\left(\frac{k-1}{k}\mathbf{1}+\frac{1}{k}\boldsymbol{c}\right)=\frac{1}{2k}+\frac{k-1}{k}\left(1-b_1\right)=\frac{1}{2} \qquad \Leftrightarrow \quad b_1=\frac{1}{2}.
\end{aligned}$$

□

Thus, we showed that the scheme defined by Table 5.4 is again consistent and second order accurate with SSP RK2 as the base method.

5.6 Constantinescu-Sandu scheme

In [31] Constantinescu and Sandu developed a conservative multirate Runge-Kutta scheme, which we will denote by MPRK-CS. For two time-step increments (5.27) and SSP RK2 method as a base their scheme is given by the equations

$$u^{(1)} = u^n \tag{5.37}$$

$$u^{(2)} = u^n + \frac{\Delta t}{2}P_1 L u^{(1)} + \Delta t P_2 L u^{(1)}, \tag{5.38}$$

$$u^{(3)} = u^n + \frac{\Delta t}{4}P_1\left(L u^{(1)} + L u^{(2)}\right), \tag{5.39}$$

$$u^{(4)} = u^n + \frac{\Delta t}{2}P_1\left(\frac{1}{2}\left(L u^{(1)} + L u^{(2)}\right) + L u^{(3)}\right) + \Delta t P_2 L u^{(3)}, \tag{5.40}$$

$$u^{n+1} = u^n + \frac{1}{4}\Delta t\left(L u^{(1)} + L u^{(2)} + L u^{(3)} + L u^{(4)}\right). \tag{5.41}$$

In the form of Butcher tableau the same scheme is shown in Table 5.5, or for arbitrary base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ in Table 5.6. The method is applicable to the case when the time-step ratio is an arbitrary $k \in \mathbb{N}$, its Butcher tableau is given in Table 5.7 [31].

$$D_1: \quad \begin{array}{c|c} \boldsymbol{c}^{(1)} & \boldsymbol{A}^{(1)} \\ \hline & \left[\boldsymbol{b}^{(1)}\right]^T \end{array} \quad = \quad \begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 \\ 1/2 & 1/4 & 1/4 & 0 & 0 \\ 1 & 1/4 & 1/4 & 1/2 & 0 \\ \hline & 1/4 & 1/4 & 1/4 & 1/4 \end{array}$$

$$D_2: \quad \begin{array}{c|c} \boldsymbol{c}^{(2)} & \boldsymbol{A}^{(2)} \\ \hline & \left[\boldsymbol{b}^{(2)}\right]^T \end{array} \quad = \quad \begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & 1/4 & 1/4 & 1/4 & 1/4 \end{array}$$

Table 5.5: MPRK-CS scheme with base method SSP RK2.

$$D_1: \quad \begin{array}{c|c} \boldsymbol{c}^{(1)} & \boldsymbol{A}^{(1)} \\ \hline & \left[\boldsymbol{b}^{(1)}\right]^T \end{array} \quad = \quad \begin{array}{c|cc} \frac{1}{2}\boldsymbol{c} & \frac{1}{2}\boldsymbol{A} & \\ \frac{1}{2}\boldsymbol{1}+\frac{1}{2}\boldsymbol{c} & \frac{1}{2}\boldsymbol{b}^T\otimes\boldsymbol{1} & \frac{1}{2}\boldsymbol{A} \\ \hline & \frac{1}{2}\boldsymbol{b}^T & \frac{1}{2}\boldsymbol{b}^T \end{array}$$

$$D_2: \quad \begin{array}{c|c} \boldsymbol{c}^{(2)} & \boldsymbol{A}^{(2)} \\ \hline & \left[\boldsymbol{b}^{(2)}\right]^T \end{array} \quad = \quad \begin{array}{c|cc} \boldsymbol{c} & \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{c} & \boldsymbol{0} & \boldsymbol{A} \\ \hline & \frac{1}{2}\boldsymbol{b}^T & \frac{1}{2}\boldsymbol{b}^T \end{array}$$

Table 5.6: MPRK-CS scheme for arbitrary base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ and time-step ratio 2.

$$D_1: \quad \begin{array}{c|ccc} \frac{1}{k}\boldsymbol{c} & \frac{1}{k}\boldsymbol{A} & & \\ \frac{1}{k}\boldsymbol{1}+\frac{1}{k}\boldsymbol{c} & \frac{1}{k}\boldsymbol{b}^T\otimes\boldsymbol{1} & \frac{1}{k}\boldsymbol{A} & \\ \vdots & \vdots & \ddots & \ddots \\ \frac{k-1}{k}\boldsymbol{1}+\frac{1}{k}\boldsymbol{c} & \frac{1}{k}\boldsymbol{b}^T\otimes\boldsymbol{1} & \cdots & \frac{1}{k}\boldsymbol{b}^T\otimes\boldsymbol{1} \quad \frac{1}{k}\boldsymbol{A} \\ \hline & \frac{1}{k}\boldsymbol{b}^T & \cdots & \frac{1}{k}\boldsymbol{b}^T \quad \frac{1}{k}\boldsymbol{b}^T \end{array}$$

$$D_2: \quad \begin{array}{c|ccc} \boldsymbol{c} & \boldsymbol{A} & \cdots & \boldsymbol{0} \quad \boldsymbol{0} \\ \boldsymbol{c} & \boldsymbol{0} & \boldsymbol{A} & \ddots \quad \boldsymbol{0} \\ \vdots & \vdots & \ddots & \ddots \quad \vdots \\ \boldsymbol{c} & \boldsymbol{0} & \cdots & \boldsymbol{0} \quad \boldsymbol{A} \\ \hline & \frac{1}{k}\boldsymbol{b}^T & \cdots & \frac{1}{k}\boldsymbol{b}^T \quad \frac{1}{k}\boldsymbol{b}^T \end{array}$$

Table 5.7: MPRK-CS scheme for arbitrary base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ and time-step ratio k.

Constantinescu and Sandu presented an extensive analysis of their MPRK-CS scheme in [31]. They showed the conservation condition (Theorem 6), which proves that MPRK-CS scheme is conservative. The accuracy result for the scheme in Table 5.7 is given by the following theorem:

Theorem 10. (Constantinescu and Sandu [31]) *The partitioned Runge-Kutta methods defined by the Butcher tableau in Table 5.7 are second order accurate if the base method* $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ *is at least second order accurate, and have at most second order coupling regardless of the order of the base method.*

Note that the lack of third order coupling was shown for the general non-linear conservation law by using the coupling conditions (5.14). But, assuming the problem is linear and using the coupling order conditions (5.15), we still find that the method is only second order accurate. Since the scheme is conservative ($\boldsymbol{b}^{(1)} = \boldsymbol{b}^{(2)}$), the following third order linear coupling conditions have to be verified

$$\left(\boldsymbol{b}^{(1)}\right)^T \boldsymbol{A}^{(1)}\boldsymbol{c}^{(2)} = \left(\boldsymbol{b}^{(1)}\right)^T \boldsymbol{A}^{(2)}\boldsymbol{c}^{(1)} = \frac{1}{6}.$$

The second coupling condition gives

$$\begin{aligned}\left(\boldsymbol{b}^{(1)}\right)^T \boldsymbol{A}^{(2)}\boldsymbol{c}^{(1)} =& \frac{1}{k}\boldsymbol{b}^T\boldsymbol{A}\left(\frac{1}{k}\boldsymbol{c}+\frac{1}{k}\boldsymbol{1}+\frac{1}{k}\boldsymbol{c}+\ldots+\frac{k-1}{k}\boldsymbol{1}+\frac{1}{k}\boldsymbol{c}\right)\\ =& \frac{1}{k}\boldsymbol{b}^T\boldsymbol{A}\left(\frac{k-1}{2}\boldsymbol{1}+\boldsymbol{c}\right) = \frac{k-1}{4k}+\frac{1}{6k} = \frac{1}{6}, \qquad \Leftrightarrow \qquad k=1.\end{aligned}$$

Hence, the coupling is second order accurate for $k > 1$.

The main advantage of MPRK-CS is its conservation property. Also, unlike MPRK-TW, it is second order accurate for any base Runge-Kutta method given by $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$.

5.7 Liu-Li-Hu linear multirate scheme

To get internal consistency for schemes of order $r > 2$, we need to adjust the solution on both sides of the interface Γ_{12}. This can provide higher order multirate scheme for linear problems [86]. In this section it is shown that the scheme proposed in [86] is third order accurate for some third order Runge-Kutta base methods.

5.7.1 Formulation of the method

Introducing the notation

$$\boldsymbol{u} = \left[u^{(1)}, u^{(2)}, \ldots, u^{(s)}\right]^T, \; \boldsymbol{u}^n = [\underbrace{u^n, \ldots, u^n}_{s}]^T,$$

$$\boldsymbol{L}_s = diag\left\{I, L, L^2, \ldots, L^{s-1}\right\}, \; \boldsymbol{L} = diag\{\underbrace{L, L, \ldots, L}_{s}\},$$

for linear problems we can rewrite the Runge-Kutta method (5.3-5.4) as

$$\boldsymbol{u} = \boldsymbol{C}\boldsymbol{T}_{\Delta t}\boldsymbol{L}_s\boldsymbol{u}^n, \tag{5.42}$$

$$u^{n+1} = u^n + \boldsymbol{b}^T\boldsymbol{L}\boldsymbol{u}, \tag{5.43}$$

where

$$\boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & a_{21} & 0 & \cdots & 0 \\ 1 & \sum a_{3j} & a_{32}a_{21} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 1 & \sum a_{sj} & \sum a_{sj}a_{jk} & \cdots & a_{s,s-1}\ldots a_{21} \end{bmatrix}, \tag{5.44}$$

and

$$\boldsymbol{T}_{\Delta t} = diag\left\{1, \Delta t, \ldots, \Delta t^{s-1}\right\}. \tag{5.45}$$

Since for the linear case

$$L^i\, u|_{t=t^n} = \left.\frac{d^i u}{dt^i}\right|_{t=t^n}, \tag{5.46}$$

the RK stage values $\boldsymbol{u}$ can be written in terms of time derivatives of u^n.

Now consider the partition $\Omega = D_1 \cup D_2 \cup \Gamma_{12}$ defined by the local time-steps (5.27). First the solution is advanced on both subdomains from $t = t^n$ with their local time-steps defined by (5.27). The stage values at the time level t^n inside of each sub-domain are computed by

$$\boldsymbol{u}_d = \boldsymbol{C}\boldsymbol{T}_{\Delta t_d}\boldsymbol{d}\boldsymbol{u}^n,\ d = 1,2, \tag{5.47}$$

where

$$\boldsymbol{d}\boldsymbol{u}^n = \left[u, \frac{du}{dt}, \frac{d^2u}{dt^2}, \ldots, \frac{d^{s-1}u}{dt^{s-1}}\right]^T_{t=t^n} \tag{5.48}$$

is the vector of time derivatives of u at $t = t^n$. To calculate the fluxes on the interface Γ_{12} the stage values from the fine mesh D_1 have to be consistent with the stage values on D_2. Using (5.47) the stages $\tilde{\boldsymbol{u}}_1$ can be obtained from $\boldsymbol{u}_1$ for time advancing on the coarse mesh, and $\tilde{\boldsymbol{u}}_2$ can be obtained from $\boldsymbol{u}_2$ for time advancing on the fine mesh. The following coupling was suggested in [86]

$$[\tilde{\boldsymbol{u}}_1]_{t^n} = \boldsymbol{C}\boldsymbol{T}_{\Delta t_2}\boldsymbol{d}\boldsymbol{u}^n = \boldsymbol{C}\boldsymbol{T}_{\Delta t_2}\boldsymbol{T}^{-1}_{\Delta t_1}\boldsymbol{C}^{-1}\,[\boldsymbol{u}_1]_{t^n} = \boldsymbol{G}^{(1)}\,[\boldsymbol{u}_1]_{t^n}\,, \tag{5.49}$$

$$[\tilde{\boldsymbol{u}}_2]_{t^n} = \boldsymbol{C}\boldsymbol{T}_{\Delta t_1}\boldsymbol{d}\boldsymbol{u}^n = \boldsymbol{C}\boldsymbol{T}_{\Delta t_1}\boldsymbol{T}^{-1}_{\Delta t_2}\boldsymbol{C}^{-1}\,[\boldsymbol{u}_2]_{t^n} = \boldsymbol{G}^{(2)}\,[\boldsymbol{u}_2]_{t^n}\,. \tag{5.50}$$

The matrices $\boldsymbol{G}^{(1)}$ and $\boldsymbol{G}^{(2)}$ are lower triangular and have the following properties

$$\boldsymbol{G}^{(1)}\boldsymbol{G}^{(2)} = \boldsymbol{G}^{(2)}\boldsymbol{G}^{(1)} = \boldsymbol{I}_s, \tag{5.51}$$

$$\sum_{j=1}^{s} G^{(1)}_{ij} = \sum_{j=1}^{s} G^{(2)}_{ij} = 1, \tag{5.52}$$

$$\boldsymbol{G}^{(1)}\tilde{\boldsymbol{u}}_2 = \boldsymbol{u}_2, \tag{5.53}$$

$$\boldsymbol{G}^{(2)}\tilde{\boldsymbol{u}}_1 = \boldsymbol{u}_1. \tag{5.54}$$

At the second step only the solution on the fine mesh D_1 is advanced in time. Therefore, the coupling stage values $\tilde{\boldsymbol{u}}_2$ at time level $t = t^n + \Delta t_1$ are needed. Using the Taylor series for time derivative of u at $t = t^n + \Delta t_1$ the following relation for the second step coupling at $t = t^n + \Delta t_1$ can be derived [86]

$$[\tilde{\boldsymbol{u}}_2]_{t^n+\Delta t_1} = \boldsymbol{C}\boldsymbol{T}_{\Delta t_1}\boldsymbol{H}_{\Delta t_1}\boldsymbol{T}^{-1}_{\Delta t_2}\boldsymbol{C}^{-1}\,[\boldsymbol{u}_2]_{t^n} = \boldsymbol{K}\,[\boldsymbol{u}_2]_{t^n}\,, \tag{5.55}$$

where

$$\boldsymbol{H}_{\Delta t} = \begin{bmatrix} 1 & \Delta t & \frac{1}{2}\Delta t^2 & \cdots & \frac{1}{(s-1)!}\Delta t^{s-1} \\ 0 & 1 & \Delta t & \cdots & \frac{1}{(s-2)!}\Delta t^{s-2} \\ 0 & 0 & 1 & \cdots & \frac{1}{(s-3)!}\Delta t^{s-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}. \tag{5.56}$$

Now the scheme [86] can be written as follows

$$u^{(1)} = \tilde{u}^{(1)} = u^n, \tag{5.57}$$

$$u^{(i)} = u^n + \Delta t_1 \sum_{j=1}^{i-1} a_{ij} L_1\left(u_1^{(j)} + \tilde{u}_2^{(j)}\right) + \tag{5.58}$$

$$\Delta t_2 \sum_{j=1}^{i-1} a_{ij} L_2\left(\tilde{u}_1^{(j)} + u_2^{(j)}\right),\ i = 2,\dots,s,$$

$$\tilde{u}^{(i)} = \sum_{j=1}^{i}\left(G_{ij}^{(1)} u_1^{(j)} + G_{ij}^{(2)} u_2^{(j)}\right), \tag{5.59}$$

$$u^{(s+1)} = u^n + \Delta t_1 \sum_{i=1}^{s} b_i L_1\left(u_1^{(i)} + \tilde{u}_2^{(i)}\right) + \Delta t_2 \sum_{i=1}^{s} b_i L_2\left(\tilde{u}_1^{(i)} + u_2^{(i)}\right), \tag{5.60}$$

$$\tilde{u}_2^{(s+1)} = \sum_{j=1}^{s} K_{1j} u_2^{(j)}, \tag{5.61}$$

$$
\begin{aligned}
u_1^{(s+i)} =& u^{(s+1)} + \Delta t_1 \sum_{j=1}^{i-1} a_{ij} L_1 \left(u_1^{(s+j)} + \tilde{u}_2^{(s+j)} \right),\ i = 2, \ldots, s, \\
\tilde{u}_2^{(s+i)} =& \sum_{j=1}^{s} K_{ij} u_2^{(j)},\ i = 2, \ldots, s, \\
u^{n+1} =& u^{(s+1)} + \Delta t_1 \sum_{i=1}^{s} b_i L_1 \left(u_1^{(s+i)} + \tilde{u}_2^{(s+i)} \right).
\end{aligned} \tag{5.62}
$$

According to authors [86], the scheme given by (5.57-5.62) retains the order of the base method for linear problems. In the following subsection we will use order conditions (5.20-5.22) and show which third order RK base methods indeed give the third order accuracy for linear problems.

5.7.2 Partitioned form of the method

In this subsection the partitioned form of (5.57-5.62) is derived (we denote it MPRK-LLH) and the accuracy is investigated using order conditions (5.20-5.22).

Theorem 11. *The multirate scheme given by (5.57-5.62) is equivalent to the scheme given by*

$$
v^{(i)} = u^n + \frac{1}{2} \Delta t \sum_{j=1}^{i-1} a_{ij} L v^{(j)},\ i = 1, \ldots, s, \tag{5.63}
$$

$$
\begin{aligned}
v^{(s+i)} = u^n + \frac{1}{2} \Delta t & \left(\sum_{j=1}^{s} b_j L_1 v^{(j)} + \sum_{j=1}^{i-1} a_{ij} L_1 v^{(s+j)} \right) + \\
& \Delta t \sum_{j=1}^{s-1} \left(\sum_{k=j+1}^{s} \sum_{l=j}^{k-1} K_{ik} a_{kl} G_{lj}^{(1)} \right) L_2 v^{(j)},\ i = 1, \ldots, s,
\end{aligned} \tag{5.64}
$$

$$
u^{n+1} = u^n + \frac{1}{2} \Delta t \sum_{i=1}^{s} b_i L_1 \left(v^{(i)} + v^{(s+i)} \right) + \Delta t \sum_{i=1}^{s} \sum_{j=1}^{s} b_j G_{ji}^{(1)} L_2 v^{(i)}. \tag{5.65}
$$

Proof. Using (5.49-5.50) and (5.55) the scheme (5.57-5.62) can be rewritten as

$$u_1^{(i)} = u_1^n + \frac{1}{2}\Delta t \sum_{j=1}^{i-1} a_{ij} L_1 \left(u_1^{(j)} + \sum_{k=1}^{j} G_{jk}^{(2)} u_2^{(k)} \right),\ i = 1,\ldots,s, \tag{5.66}$$

$$u_2^{(i)} = u_2^n + \Delta t \sum_{j=1}^{i-1} a_{ij} L_2 \left(\sum_{k=1}^{j} G_{jk}^{(1)} u_1^{(k)} + u_2^{(j)} \right),\ i = 1,\ldots,s, \tag{5.67}$$

$$u^{(s+1)} = u^n + \frac{1}{2}\Delta t \sum_{i=1}^{s} b_i L_1 \left(u_1^{(i)} + \sum_{k=1}^{i} G_{ik}^{(2)} u_2^{(k)} \right), \tag{5.68}$$

$$u_1^{(s+i)} = u_1^{(s+1)} + \frac{1}{2}\Delta t \sum_{j=1}^{i-1} a_{ij} L_1 \left(u_1^{(s+j)} + \sum_{k=1}^{s} K_{jk} u_2^{(k)} \right),\ i = 2,\ldots,s, \tag{5.69}$$

$$\begin{aligned} u^{n+1} = u^{(s+1)} &+ \frac{1}{2}\Delta t \sum_{i=1}^{s} b_i L_1 \left(u_1^{(s+i)} + \sum_{j=1}^{s} K_{ij} u_2^{(j)} \right) + \\ &\Delta t \sum_{i=1}^{s} b_i L_2 \left(\sum_{j=1}^{i} G_{ij}^{(1)} u_1^{(j)} + u_2^{(i)} \right). \end{aligned} \tag{5.70}$$

Note, that time advancing on coarse mesh is moved from $s+1$ stage in (5.60) to the last step $n+1$ in (5.70). This can be done, because $u_2^{(s+1)}$ is not used in the computation of stage values $u_1^{(s+i)}$. For the analysis the following new stage values are introduced

$$v^{(i)} = \begin{cases} u_1^{(i)} + \sum_{j=1}^{i} G_{ij}^{(2)} u_2^{(j)}, & 1 \le i \le s, \\ u_1^{(i)} + \sum_{j=1}^{s} K_{i-s,j} u_2^{(j)}, & s+1 \le i \le 2s, \end{cases} \tag{5.71}$$

$$w^{(i)} = \begin{cases} \sum_{j=1}^{i} G_{ij}^{(1)} u_1^{(j)} + u_2^{(i)}, & 1 \le i \le s, \\ 0, & s+1 \le i \le 2s. \end{cases} \tag{5.72}$$

Then the scheme (5.66-5.70) can be written in terms of new stage values as

$$u_1^{(i)} = u_1^n + \frac{1}{2}\Delta t \sum_{j=1}^{i-1} a_{ij} L_1 v^{(j)},\ i = 1,\dots,s, \tag{5.73}$$

$$u_2^{(i)} = u_2^n + \Delta t \sum_{j=1}^{i-1} a_{ij} L_2 w^{(j)},\ i = 1,\dots,s, \tag{5.74}$$

$$u_1^{(s+i)} = u_1^n + \frac{1}{2}\Delta t \sum_{i=1}^{s} b_i L_1 v^{(i)} + \frac{1}{2}\Delta t \sum_{j=1}^{i-1} a_{ij} L_1 v^{(s+j)},\ i = 1,\dots,s, \tag{5.75}$$

$$u^{n+1} = u^n + \frac{1}{2}\Delta t \sum_{i=1}^{s} b_i L_1 \left(v^{(i)} + v^{(s+i)} \right) + \Delta t \sum_{i=1}^{s} b_i L_2 w^{(i)}. \tag{5.76}$$

For the first s stage values $v^{(i)}$ and $w^{(i)}$ defined by (5.71-5.72) the following properties hold

$$v^{(i)} = \sum_{j=1}^{i} G_{ij}^{(2)} w^{(j)},$$

$$w^{(i)} = \sum_{j=1}^{i} G_{ij}^{(1)} v^{(j)}.$$

Therefore the scheme (5.73-5.76) can be rewritten in terms of the stage values $v^{(i)}$. For the first part of the scheme it can be shown that

$$\sum_{j=1}^{i} G_{ij}^{(2)} u_2^{(j)} = \frac{1}{2}\Delta t \sum_{j=1}^{i-1} a_{ij} L_2 v^{(j)}, \tag{5.77}$$

$$\sum_{j=1}^{i} G_{ij}^{(1)} u_1^{(j)} = \Delta t \sum_{j=1}^{i-1} a_{ij} L_1 v^{(j)}. \tag{5.78}$$

For the second part of the scheme the following holds

$$\begin{aligned} \sum_{j=1}^{s} K_{ij} u_2^{(j)} &= \sum_{j=1}^{s} K_{ij} \left(u_2^n + \Delta t \sum_{k=1}^{j-1} a_{jk} L_2 w^{(k)} \right) \\ &= u_2^n + \Delta t \sum_{j=1}^{s} \sum_{k=1}^{j-1} \sum_{l=1}^{k} K_{ij} a_{jk} G_{kl}^{(1)} L_2 v^{(l)}. \end{aligned} \tag{5.79}$$

$$D_1: \quad \begin{array}{c|c} \boldsymbol{c}^{(1)} & \boldsymbol{A}^{(1)} \\ \hline & \left[\boldsymbol{b}^{(1)}\right]^T \end{array} \quad = \quad \begin{array}{c|cc} \frac{1}{2}\boldsymbol{c} & \frac{1}{2}\boldsymbol{A} & \boldsymbol{0} \\ \frac{1}{2}\boldsymbol{1}+\frac{1}{2}\boldsymbol{c} & \frac{1}{2}\boldsymbol{b}^T\otimes\boldsymbol{1} & \frac{1}{2}\boldsymbol{A} \\ \hline & \frac{1}{2}\boldsymbol{b}^T & \frac{1}{2}\boldsymbol{b}^T \end{array}$$

$$D_2: \quad \begin{array}{c|c} \boldsymbol{c}^{(2)} & \boldsymbol{A}^{(2)} \\ \hline & \left[\boldsymbol{b}^{(2)}\right]^T \end{array} \quad = \quad \begin{array}{c|cc} \frac{1}{2}\boldsymbol{c} & \frac{1}{2}\boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{q} & \boldsymbol{Q} & \boldsymbol{0} \\ \hline & \boldsymbol{b}^T\boldsymbol{G}^{(1)} & \boldsymbol{0} \end{array}$$

Table 5.8: MPRK-LLH scheme for arbitrary base method $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ and time-step ratio 2.

The sum in (5.79) can be rearranged as

$$\sum_{j=1}^{s} K_{ij}u_2^{(j)} = u_2^n + \Delta t \sum_{j=1}^{s-1}\left(\sum_{k=j+1}^{s}\sum_{l=j}^{k-1} K_{ik}a_{kl}G_{lj}^{(1)}\right) L_2 v^{(j)}. \tag{5.80}$$

Using (5.77) and (5.80) we derive (5.63-5.65). □

Now defining the matrix

$$\boldsymbol{Q} = \boldsymbol{K}\boldsymbol{A}\boldsymbol{G}^{(1)}, \tag{5.81}$$

and the vector $\boldsymbol{q} = [q_1,\ldots,q_s]^T$, where

$$q_i = \sum_{j=1}^{s} Q_{ij} = \sum_{j=1}^{s-1}\left(\sum_{k=j+1}^{s}\sum_{l=j}^{k-1} K_{ik}a_{kl}G_{lj}^{(1)}\right), \tag{5.82}$$

we can present the MPRK-LLH method (5.63-5.65) by the Butcher tableau shown in Table 5.8. In Table 5.9 we present the matrices $\boldsymbol{Q}$ as well as vectors $\boldsymbol{q}$ and $\boldsymbol{b}^T\boldsymbol{G}^{(1)}$ for base methods in Table 5.1.

Consistency and accuracy analysis for MPRK-LLH scheme can be summarized by the theorem:

Theorem 12. *The partitioned Runge-Kutta scheme defined by the Butcher tableau in Table 5.8 is internally consistent and is second order accurate for all base methods* $(\boldsymbol{A},\boldsymbol{b},\boldsymbol{c})$ *given in Table 5.1, and is third order accurate for the base methods SSP RK3 and SSP LRK3.*

Base method	$\boldsymbol{Q}$	$\boldsymbol{q}$	$\boldsymbol{b}^T\boldsymbol{G}^{(1)}$
SSP RK2	$\begin{bmatrix} \frac{1}{2} & 0 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix}$	$[0,1]$
SSP LRK32	$\begin{bmatrix} 0 & \frac{1}{2} & 0 \\ -\frac{1}{4} & 1 & 0 \\ -\frac{3}{4} & \frac{7}{4} & 0 \end{bmatrix}$	$\begin{bmatrix} \frac{1}{2} \\ \frac{3}{4} \\ 1 \end{bmatrix}$	$\left[\frac{1}{3}, -\frac{2}{3}, \frac{4}{3}\right]$
SSP LRK3	$\begin{bmatrix} \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{3}{4} & 0 \\ -\frac{1}{4} & \frac{7}{4} & 0 \end{bmatrix}$	$\begin{bmatrix} \frac{1}{2} \\ 1 \\ \frac{3}{2} \end{bmatrix}$	$\left[\frac{2}{3}, -\frac{1}{3}, \frac{2}{3}\right]$
SSP RK3	$\begin{bmatrix} \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{3}{4} & 0 \\ \frac{1}{8} & \frac{5}{8} & 0 \end{bmatrix}$	$\begin{bmatrix} \frac{1}{2} \\ 1 \\ \frac{3}{4} \end{bmatrix}$	$\left[-\frac{4}{3}, -\frac{1}{3}, \frac{8}{3}\right]$
SSP LRK43	$\begin{bmatrix} \frac{1}{6} & 0 & \frac{1}{3} & 0 \\ \frac{1}{6} & -\frac{1}{4} & \frac{5}{6} & 0 \\ \frac{1}{6} & -1 & \frac{11}{6} & 0 \\ \frac{7}{24} & -\frac{17}{8} & \frac{37}{12} & 0 \end{bmatrix}$	$\begin{bmatrix} \frac{1}{2} \\ \frac{3}{4} \\ 1 \\ \frac{5}{4} \end{bmatrix}$	$\left[1, -\frac{2}{3}, -\frac{2}{3}, \frac{4}{3}\right]$

Table 5.9: Matrix $\boldsymbol{Q}$ and vectors $\boldsymbol{q}$ and $\boldsymbol{b}^T\boldsymbol{G}^{(1)}$ for the Butcher form of the scheme (5.63-5.65) for Runge-Kutta methods in Table 5.1.

Proof. The scheme (5.57-5.62) is internally consistent by design. To confirm the internal consistency for the MPRK-LLH scheme (5.63-5.65), we need to check the condition

$$\boldsymbol{q} = \frac{1}{2}\mathbf{1} + \frac{1}{2}\boldsymbol{c}. \tag{5.83}$$

Using Table 5.9 it can be verified that all base methods of Table 5.1 satisfy (5.83).

It follows from the application of the order conditions (5.20-5.22) to the coefficient matrices in Table 5.9 that the scheme MPRK-LLH is at least second order accurate for all base methods from Table 5.1. Furthermore, it is third order accurate with the base methods SSP RK3 and SSP LRK3, but not SSP LRK43. □

Thus the above theorem shows that the scheme given by (5.63-5.65) is the only MPRK scheme considered in this chapter that satisfies third order conditions for linear problems (5.20-5.22). These results will be confirmed by numerics in Section 5.8.

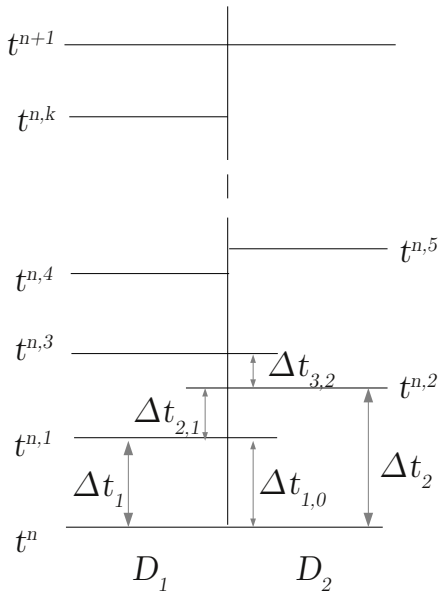


Figure 5.1: Local time steps for the multirate scheme by Liu et al. [86] with an arbitrary time-step ratio.

To conclude this subsection we present the second order MPRK-LLH scheme based on SSP RK2

$$u^{(1)} = u^n, \tag{5.84}$$

$$u^{(2)} = u^n + \frac{1}{2}\Delta t L u^{(1)}, \tag{5.85}$$

$$u^{(3)} = u^n + \frac{1}{4}\Delta t P_1 L\left(u^{(1)} + u^{(2)}\right) + \frac{1}{2}\Delta t P_2 L u^{(1)}, \tag{5.86}$$

$$u^{(4)} = u^n + \frac{1}{2}\Delta t P_1\left(\frac{1}{2}\left(L u^{(1)} + L u^{(2)}\right) + L u^{(3)}\right) + \Delta t P_2 L u^{(1)}, \tag{5.87}$$

$$u^{n+1} = u^n + \frac{1}{4}\Delta t P_1 L \sum_{i=1}^{4} u^{(i)} + \frac{1}{2}\Delta t P_2 L u^{(2)}. \tag{5.88}$$

5.7.3 Extension to arbitrary time-step ratio

The extension of the scheme to any arbitrary time-step ratio is straight forward [86]. Figure 5.1 illustrates the steps. Consider two domains D_1 with time-step Δt_1 and D_2 with Δt_2. To update u_1 and u_2 from the time level t^n the coupling stage values $\tilde{u}_2$ and $\tilde{u}_1$ at time t^n are

obtained by formulas (5.49-5.50). To update u_1 from the time level $t^{n,1}$ to $t^{n,3}$ the coupling stage values $\tilde{u}_2$ at time $t^{n,1}$ need to be computed from the known stage values u_2 at t^n. They are obtained by the following coupling formula

$$[\tilde{\boldsymbol{u}}_2]_{t^{n,1}} = \boldsymbol{C}\boldsymbol{T}_{\Delta t_1}\boldsymbol{H}_{\Delta t_{1,0}}\boldsymbol{T}_{\Delta t_2}^{-1}\boldsymbol{C}^{-1}[\boldsymbol{u}_2]_{t^n} = \boldsymbol{K}^{(2)}_{\Delta t_{1,0}}[\boldsymbol{u}_2]_{t^n}, \tag{5.89}$$

where $\Delta t_{1,0} = t^{n,1} - t^n$. Then u_1 at $t^{n,1}$ is obtained by

$$\tilde{\boldsymbol{u}}_2 = \boldsymbol{K}^{(2)}_{\Delta t_{1,0}}\boldsymbol{u}_2, \tag{5.90}$$

$$u_1^{(i)} = u_1^{(i-1)} + \Delta t_1 \sum_{j=1}^{i-1} a_{ij} L_1\left(u_1^{(j)} + \tilde{u}_2^{(j)}\right),\ i = 2,\ldots,s, \tag{5.91}$$

$$u_1 = u^{(s+1)} + \Delta t_1 \sum_{i=1}^{s} b_i L_1\left(u_1^{(i)} + \tilde{u}_2^{(i)}\right). \tag{5.92}$$

Similarly, to advance u_2 from $t^{n,2}$ to $t^{n,5}$ we obtain the stage value on D_1 at $t^{n,2}$ from the stage value available at $t^{n,1}$ by

$$[\tilde{\boldsymbol{u}}_1]_{t^{n,2}} = \boldsymbol{C}\boldsymbol{T}_{\Delta t_2}\boldsymbol{H}_{\Delta t_{2,1}}\boldsymbol{T}_{\Delta t_1}^{-1}\boldsymbol{C}^{-1}[\boldsymbol{u}_1]_{t^{n,1}} = \boldsymbol{K}^{(1)}_{\Delta t_{2,1}}[\boldsymbol{u}_1]_{t^{n,1}}, \tag{5.93}$$

where $\Delta t_{2,1} = t^{n,2} - t^{n,1}$. Then u_2 at $t^{n,5}$ is obtained by

$$\tilde{\boldsymbol{u}}_1 = \boldsymbol{K}^{(1)}_{\Delta t_{2,1}}\boldsymbol{u}_1, \tag{5.94}$$

$$u_2^{(i)} = u_2^{(i-1)} + \Delta t_1 \sum_{j=1}^{i-1} a_{ij} L_1\left(\tilde{u}_1^{(j)} + u_2^{(j)}\right),\ i = 2,\ldots,s, \tag{5.95}$$

$$u_2 = u^{(s+1)} + \Delta t_1 \sum_{i=1}^{s} b_i L_1\left(\tilde{u}_1^{(i)} + u_2^{(i)}\right). \tag{5.96}$$

The next step is to update u_1 to $t^{n,4}$, therefore we obtain the stage values $[\tilde{\boldsymbol{u}}_2]_{t^{n,3}}$ from the available ones at $t^{n,2}$ using $\boldsymbol{K}^{(2)}_{\Delta t_{3,2}}$.

The general formula for (5.50) and (5.93) can be derived as

$$[\tilde{\boldsymbol{u}}_1]_{t^{n,k_2}} = \boldsymbol{K}^{(1)}_{\Delta t_{k_2,k_1}} [\boldsymbol{u}_2]_{t^{n,k_1}}, \qquad \boldsymbol{K}^{(1)}_{\Delta t_{k_2,k_1}} = \boldsymbol{C}\boldsymbol{T}_{\Delta t_2}\boldsymbol{H}_{\Delta t_{k_2,k_1}}\boldsymbol{T}^{-1}_{\Delta t_1}\boldsymbol{C}^{-1}, \tag{5.97}$$

$$[\tilde{\boldsymbol{u}}_2]_{t^{n,k_2}} = \boldsymbol{K}^{(2)}_{\Delta t_{k_2,k_1}} [\boldsymbol{u}_1]_{t^{n,k_1}}, \qquad \boldsymbol{K}^{(2)}_{\Delta t_{k_2,k_1}} = \boldsymbol{C}\boldsymbol{T}_{\Delta t_1}\boldsymbol{H}_{\Delta t_{k_2,k_1}}\boldsymbol{T}^{-1}_{\Delta t_2}\boldsymbol{C}^{-1}, \tag{5.98}$$

with $\Delta t_{k_2,k_1} = t^{n,k_2} - t^{n,k_1}$. It should be noted that if $t^{n,k_1} = t^{n,k_2}$ then $\boldsymbol{H}_{\Delta t_{k_2,k_1}} = \boldsymbol{I}$ and $\boldsymbol{K}^{(d)}_{\Delta t_{k_2,k_1}} = \boldsymbol{G}^{(d)}$.

5.8 Numerical experiments

In this section MPRK schemes discussed in this chapter are applied to the linear advection problem

$$u_t + u_x = 0, \qquad x \in \Omega = (-1,1), \tag{5.99}$$

$$u(x,0) = \sin(\pi x) \tag{5.100}$$

with periodic boundary conditions. The goal of these numerical experiments is to verify the theoretical accuracy results obtained in the previous section for MPRK-TW, MPRK-CS and MPRK-LLH schemes. The computational domain is decomposed as $\Omega = D_1 \cup D_2 \cup \Gamma_{12}$, where $D_1 = (-1,0)$, $D_2 = (0,1)$, and $\Gamma_{12} = \{0\}$. Subdomain D_1 is discretized using a fine grid with $h_1 = h/2$, and D_2 is discretized using a coarse grid with spacing $h_2 = h$. The time-steps Δt_1 and Δt_2 are proportional with the grid sizes h_1 and h_2 to satisfy the CFL restriction

$$\Delta t_i = CFL h_i, \quad i = 1,2. \tag{5.101}$$

Numerical error and order of convergence in L^1, L^2 and L^∞ defined by (3.62) and (3.63) respectively are compared for various MPRK schemes.

For the space approximation the classic WENO3 scheme with $\varepsilon_i = h_i$, $i = 1,2$ is used. It was found by numerical experiments that MPRK2-CS requires $CFL = 0.5$, while other second order multirate methods converge for $CFL = 1$, therefore in our comparison of second order MPRK schemes we use $CFL = 0.5$.

First the second order schemes with SSP RK2 base method are considered. They are refereed to as MPRK2-TW, MPRK2-CS and MPRK2-LLH. The results for the problem (5.99-5.100) are given in Figure 5.2. All schemes appear to be second order convergent in L^1, while in L^2 and L^∞ errors for MPRK2-CS converge slower than other for other schemes.

Next the third order accuracy results of multirate schemes are validated. SSP RK3 scheme is used a base method for all multirate discretizations which are denoted as MPRK3-TW, MPRK3-CS and MPRK3-LLH. The results for the problem (5.99-5.100) are given in Figure 5.3. The results confirm the third order accuracy for MPRK3-LLH and only first order accuracy for MPRK3-TW.

To confirms the theoretical results presented in subsection 5.7.2, we also demonstrate numerically, that not all third order base methods yield third order coupling in MPRK-LLH. Numerical results are shown in Figure 5.4. As expected from the accuracy analysis, the base method SSP LRK43 produce only second order convergent MPRK scheme.

5.9 Chapter summary

In this chapter several multirate Runge-Kutta methods were reviewed. The order conditions for the partitioned form of multirate Runge-Kutta schemes for linear problems were derived. All of the methods considered show second order accuracy when SSP RK2 is used as a base method and the time-step ratio is an arbitrary integer. The use of higher order base methods in MPRK schemes does not lead to higher order accuracy at the interface between the two LTS subdomains. Moreover, as it was shown that the accuracy of MPRK-TW scheme even decreases to first order with a higher order base method. It was also shown, that MPRK-LLH method is third order accurate for linear problems with specific third order Runge-Kutta methods. Another MPRK method that satisfies all third order conditions (including (5.14)) for a particular base method, called Recursive Flux Splitting Multirate (RFSMR), was given in [111]. The choice of the base method in RFSMR strongly depends on the desired time-step ratio. The base method that gives the third order MPRK satisfies some additional condition, found in [81] and is not SSP. The scheme can only be used if the time-step ratio on the neighboring subdomains is equal to 2.

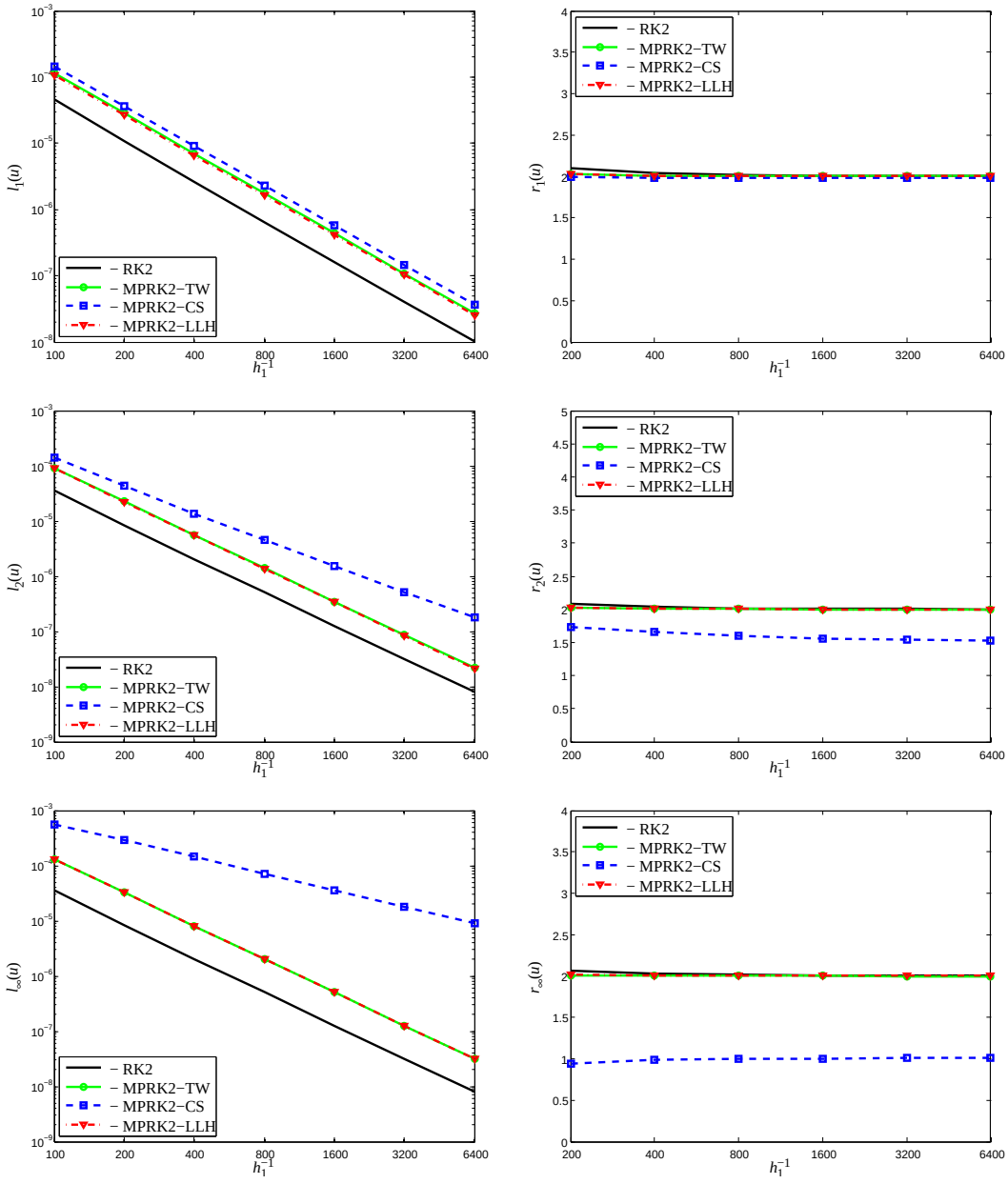


Figure 5.2: Comparison of second order MPRK schemes for the linear advection equation with initial data $u(x,0) = \sin(\pi x)$ at $T = 1$, CFL=0.5.

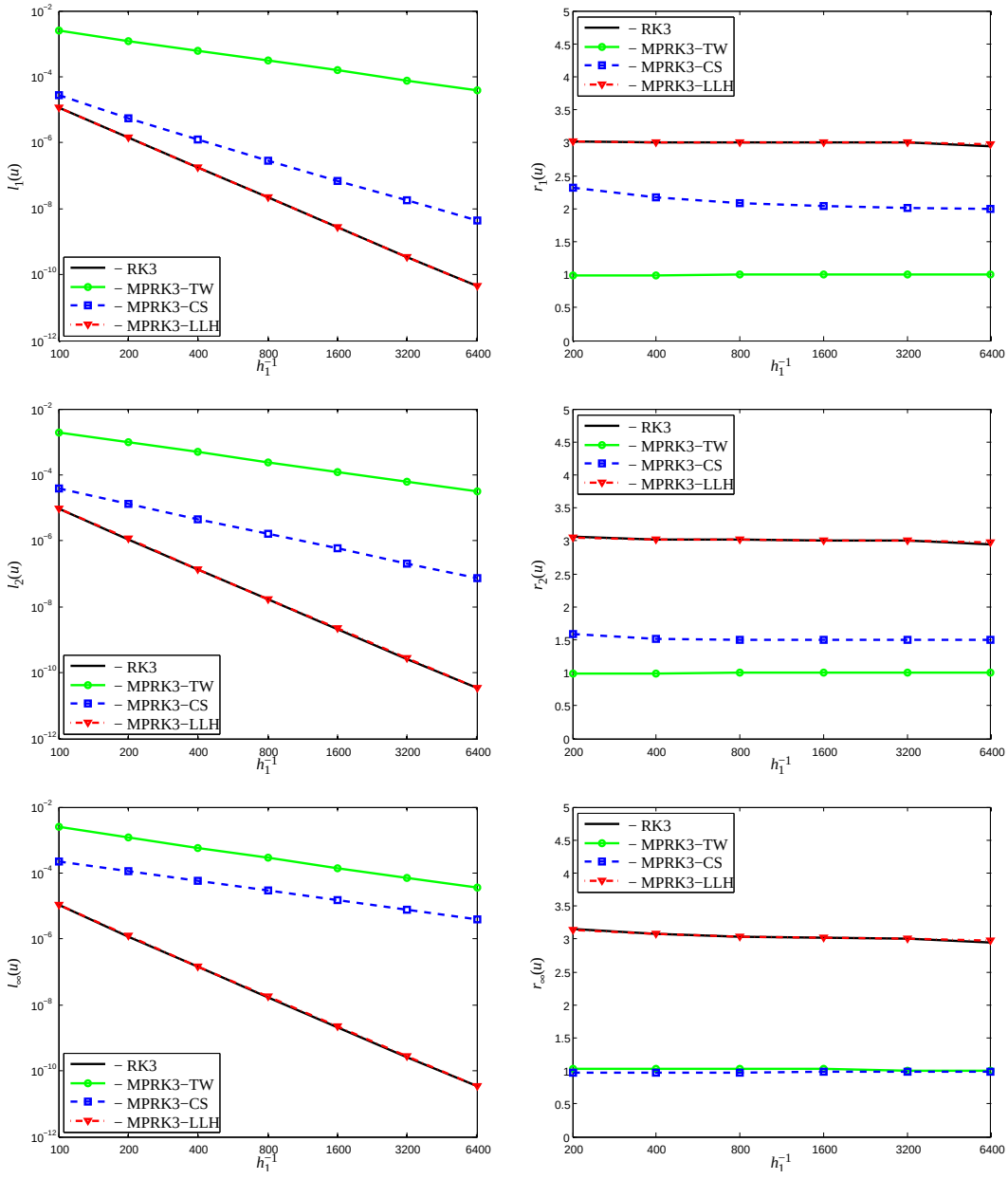


Figure 5.3: Comparison of third order MPRK schemes for the linear advection equation with initial data $u(x,0) = \sin(\pi x)$ at $T = 1$, CFL=1.

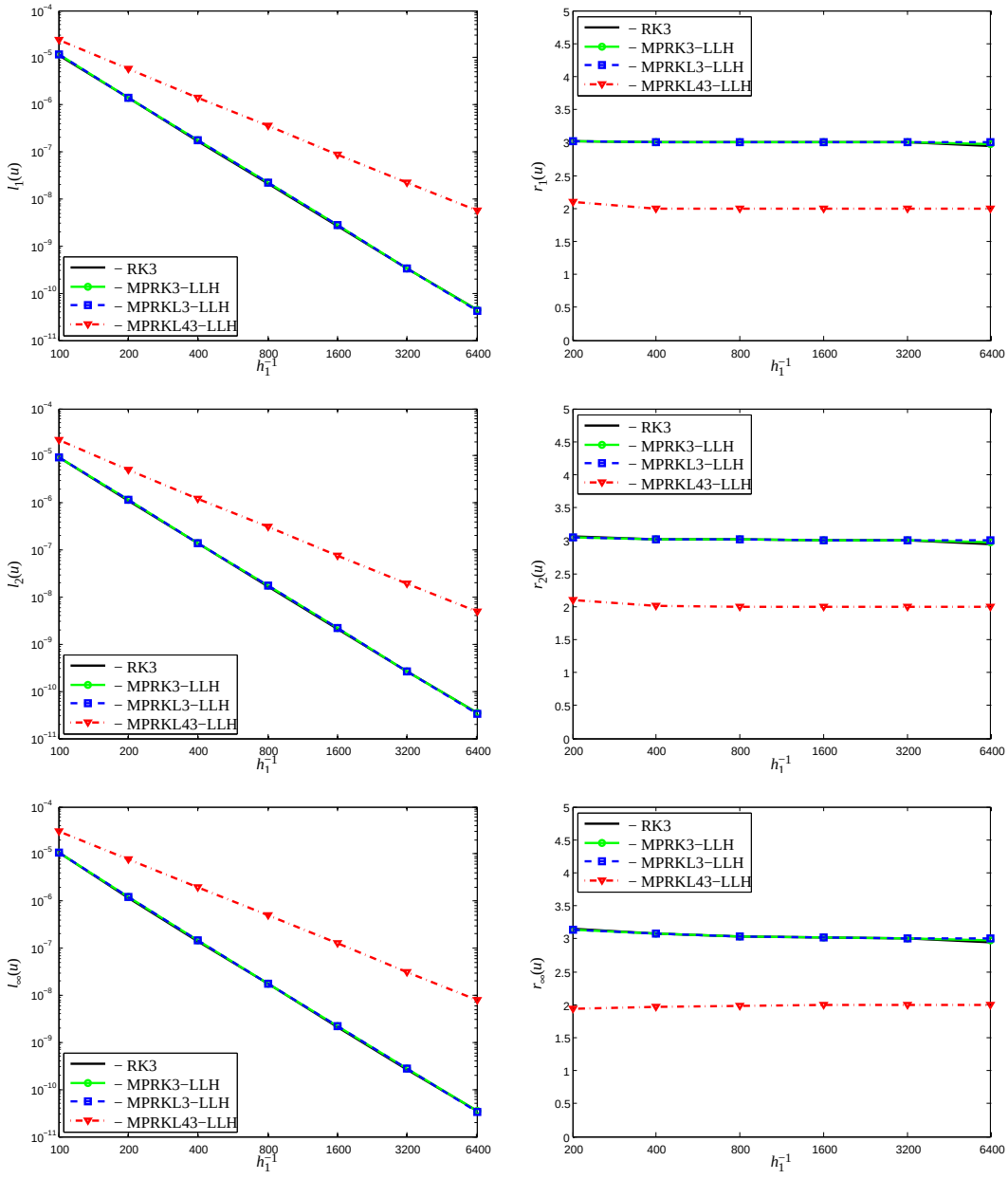


Figure 5.4: Comparison of third order MPRK-LLH schemes with SSP RK3, SSP LRK3 and SSP LRK43 as base methods for the linear advection equation with initial data $u(x,0) = \sin(\pi x)$ at $T = 1$, CFL=1.

While the order conditions can be derived for MPRK schemes of any order, even the third order coupling conditions are hard to satisfy. The number of these conditions increases quickly for higher orders, and it seems impractical to consider conditions beyond the third order.

All multirate schemes based on a single base method are either locally inconsistent (MPRK-CS), or non-conservative (MPRK-TW and MPRK-LLH). As it was shown in this chapter MPRK-TW scheme is consistent with SSP RK2 as a base method only, while MPRK-LLH scheme remains consistent regardless of the base method.

The multirate partitioning considered in this chapter was based on grid points, that is the projections $P_1 L$ and $P_2 L$ are used to split the right hand side of into $L_1 + L_2$. The splitting could also be based on the fluxes ([80, 112]). Splitting by flux guarantees conservation of the scheme, but can lead to inconsistencies.

Analysis of this chapter is crucial for the implementation of multirate Runge-Kutta (MRK) schemes to Maxwell's equations on three-dimensional meshes. Since we apply a third order approximation scheme in space, the same accuracy is needed in time. Originally, MRK-TW scheme was chosen to be implemented for its simplicity and flexibility in time-step ratios. But it was then found that any third order extension of the scheme fails to satisfy even second order accuracy conditions. Therefore other multirate strategies based on RK time-stepping and arbitrary time-steps ratios were compared. From the analysis of this chapter it follows that MRK-LLH is the only scheme that satisfies our criteria for an efficient multirate scheme. It combines the flexibility in time-step ratios and third order accuracy. For three-dimensional Maxwell's equations two methods, namely MRK-TW and MRK-LLH with SSP RK3 base scheme, were compared. Accuracy of both approaches is studied numerically on three-dimensional test problems in Chapter 7, and results agree with the theory presented in this chapter.

Chapter 6

Multirate schemes for Maxwell's equations

In this chapter, the implementation of multirate schemes based on Runge-Kutta time integrations to solve 3D Maxwell's equations is discussed. Details of implementation of two multirate approaches from Chapter 5 are presented. The key parts of the implementation include partition of the computational domain into multirate groups based on a distribution of local time-steps, construction of a coupling buffer and the algorithm for each method.

6.1 Local time-stepping in CEM overview

Since the second order leap-frog time scheme is classically used in computational electromagnetics, several local-time stepping (LTS) versions of the scheme can be found in the literature [30, 95, 59, 10]. Other approaches include LTS versions of multi-stage (RK, predictor-corrector) [10, 44] and multi-step (Adams-Bashforth) explicit methods [53, 60], Cauchy-Kovalevskaja procedures [126], and locally implicit time integration [35]. As an alternative, domain decomposition methods [47] can be used to evolve the solution with different time-steps or schemes on different subdomains.

In [28, 29] Collino, Fouquet and Joly developed and analyzed so-called conservative space-time mesh refinement method for the one-dimensional wave equation. It is based on the finite difference Yee's scheme, and uses special discrete transmission conditions to

obtain unknown values at the interface between two subdomains with different time-steps. These conditions are constructed using the conservation of discrete energy and solved at each global time step. The time step ratio between two LTS subdomains sharing the same interface is equal to 2 in this approach. Extension of the method to 3D Maxwell's equation with reinterpretation of the Yee's scheme as a mixed finite element scheme was done in [30] by the same authors. The advantage of the space-time mesh refinement method is that it guarantees the stability of the scheme by enforcing the conservation of discrete energy. But it requires solution of a linear system at the interface between two grids at each time step. This becomes more and more computationally expensive as we increase the number of multirate domains in 3D space.

Fumeaux et al. in [44] proposed local-time stepping based on predictor-corrector scheme for Maxwell's equations. In their procedure the ratio between time-steps of adjacent subdomains has to be equal to 2 in order to perform the coupling procedure. The half step value for the coarser grid required for coupling with the finer grid is obtained as a mean between the last two available values from the predictor and corrector steps.

By combining the symplectic Störmer-Verlet scheme, which is the leap-frog scheme reorganized into 3 steps, with DG method Piperno in [101] proposed another local time-stepping method. He proved conservation of discrete energy of the proposed approach for two levels of refinement. In [95] Montseny et al. followed the same idea to develop a leap-frog based LTS scheme. In both cases time increments proportional to 2 are used and the latest available solution is used for coupling at the interface between domains with different time-steps.

In [126] LTS technique based on arbitrary-high order derivatives (ADER) DG method was developed. In ADER approach the Cauchy-Kovalevskaya procedure is used to replace the time derivatives by space derivatives in Taylor series in time. In this case the solution is updated in time by one step. Therefore, unlike RK, there is no additional consistency challenge due to the presence of stage values in the local time-stepping procedure.

In [36] Diaz and Grote derived an arbitrary (even) high order LTS method for the second order wave equation. Their method is based on an extension of the second-order leap-frog scheme by a modified equation approach [121] and is implemented with continuous and discontinuous Galerkin finite element. The time-step ratio in the proposed scheme is equal

to some $p \in \mathbb{N}$. Their method was proven to conserve the discrete energy under some CFL condition. The same LTS approach was implemented for 2D Maxwell's equations by Grote and Mitkova in [59] with second and fourth order time integration for a non-conducting medium ($\sigma = 0$) and second order scheme for conducting medium. The same authors also developed an explicit LTS method based on Adams-Bashforth multi-step schemes in [60], which allows an order higher than 2 for conducting medium. Another implementation of LTS method based on Adams-Bashforth multi-step scheme can be found in [51].

Recently another LTS technique (Causal-Path LTS) utilizing multi-stage time schemes has been proposed by Angulo et al. in [10], which was applied to Maxwell's equations using the fourth order RK and second order leap-frog as base time integration schemes. Their LTS approach requires a computation of the stage value of neighbors in order to advance the solution on a given subdomain. Therefore, the idea is similar to the one proposed by Tang and Warnecke in [125] and allows arbitrary time-step ratios.

In this thesis two LTS approaches based on RK time integrations are implemented. One is based on projection of the solution to provide coupling at LTS interfaces [125], and another one uses interpolation of stage values for the same purpose [86]. This chapter presents the details of implementation of these techniques to three-dimensional Maxwell's equations.

6.2 Multirate groups

Consider a computational domain with mesh $\overline{\Omega}_T = \cup_{i=1}^{N} \overline{T}_i$. In this section partition of Ω_T into multirate groups is discussed. Let $\{\Delta\tau_i\}_{i=1}^{N}$ be a set of characteristic stable time steps obtained by [18]

$$\Delta\tau_i \leq \frac{|T_i|}{c_i \sum_{j \in \mathscr{I}_i} |S_{ij}|} \tag{6.1}$$

for each cell T_i of a given mesh Ω, where $\mathscr{I}_i$ is the set of indexes of the neighbors of the element T_i, S_{ij} and $|S_{ij}|$ are the faces shared by elements T_i and T_j and its area. Let

$$\Delta t_{\min} = \min_i \{\Delta\tau_i\}, \text{ and } \Delta t_{\max} = \max_i \{\Delta\tau_i\}.$$

Then a set of K local time steps $\{\Delta t_k\}_{k=1}^{K} \in [\Delta t_{\min}\,,\,\Delta t_{\max})$ can be defined to form K multirate groups as follows

$$D^{(k)} = \begin{cases} \{T_i \in \Omega, \Delta\tau_i \in [\Delta t_k\,,\,\Delta t_{k+1})\}\,, & k = 1,...,K-1, \\ \{T_i \in \Omega, \Delta\tau_i \in [\Delta t_k\,,\,\Delta t_{\max})\}\,, & k = K. \end{cases} \tag{6.2}$$

Each multirate group consists of elements of bulk group $D^{(k)}_{bulk}$ and inner buffer group $D^{(k)}(0)$. The bulk group $D^{(k)}_{bulk}$ includes all elements of $D^{(k)}$ that are sufficiently far from the boundary $\Gamma_k = \partial D^{(k)} \cap \left(\cup_{l=1,l\neq k}^{K} \partial D^{(l)}\right)$. Therefore time integration in the bulk group does not depend on values from the neighboring multirate groups. The size of the inner buffer $D^{(k)}(0)$ depends on the order of finite volume approximation and consist of elements of $D^{(k)}$ whose finite volume stencils contain elements from other multirate groups. We also define the outer buffer groups, which consist of elements from adjacent to $D^{(k)}$ multirate groups required in the coupling procedure. A two-dimensional illustration of different groups is shown in Figure 6.1. It should be noted that the size of outer buffer groups is different for each multirate strategy and, in general, consists of one or more rows of elements adjacent to Γ_k.

Unlike the one-dimensional case, in three space dimensions many choices for multirate partition are available. For example, one could define N multirate groups with time-steps $\Delta\tau_i$ but this approach is far from optimal in the FV framework. Unlike the DG method which is based on a polynomial reconstruction on each cell, the FV method uses an increasing number of cells in the stencil to obtain higher order accuracy. Hence taking individual local time steps will lead to a huge computational overhead due to the size of buffer groups. Therefore one should organize elements into large enough multirate groups (6.2) so that the speedup from multirate time integration surpasses the computational overhead in buffer groups.

One of the important aspects of multirate time integration is synchronization after a number of local time integrations. This ensures that the final solution for all multirate groups can be obtained for a given final time T, and is also important for consistency. Let Δt be the global time step, at which the solution in all multirate groups is synchronized, and the final time is achieved after N^t global time integrations, i.e., $T = N^t \Delta t$. Let m be

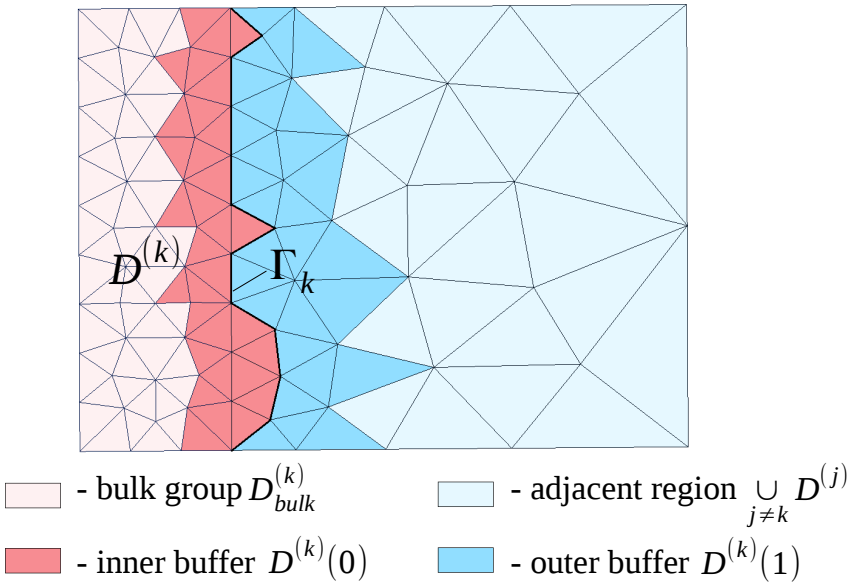


Figure 6.1: The structure of the multirate group $D^{(k)}$.

the number of multirate stages in one global time-step Δt. Then for each multirate group $D^{(k)}$ the local time at l-th multirate stage can be defined by $t_k^{n,l}$, $l \in \{1,...,m\}$. The global time $t^{n,l}$ can then be found from local times $t_k^{n,l}$. Both local and global times depend on the multirate scheme. Details of how local and global times are updated will be given later for each method separately.

The most common choice for the definition of local time steps, which satisfies the synchronization requirement, is to take

$$\Delta t_k = a^{k-1}\Delta t_{\min}, \quad k = 1,...,K, \tag{6.3}$$

where a is some integer, usually 2 or 3 [44, 95, 101, 113]. The number of multirate groups can be determined by

$$K = \left\lfloor \log_a \frac{\Delta t_{\max}}{\Delta t_{\min}} \right\rfloor + 1, \tag{6.4}$$

and all local times are synchronized after each $\Delta t = a^{K-1}\Delta t_{\min}$. We will call this type of partition into multirate groups the power partition.

Another simple way to define local time steps is by

$$\Delta t_k = \frac{K}{K-k}\Delta t_{\min}, \quad k = 0, ..., K-1, \tag{6.5}$$

where

$$K = \left\lfloor \frac{\Delta t_{\max}}{\Delta t_{\min}} \right\rfloor. \tag{6.6}$$

In this case all local times are synchronized after each $\Delta t = K\Delta t_{\min}$. The drawback of this partition as it creates too many multirate groups for large K. Therefore an optimization step is required to remove the unnecessary groups. In numerical experiments the following variation of the fractional partition was found to be efficient

$$\{\Delta t_k\}_{k=1}^{K} \in \left\{2^k \Delta t_{\min}\right\} \cup \left\{3 \cdot 2^{k-1} \Delta t_{\min}\right\}_{k=0}^{K-1},$$

here K is given by (6.4) with $a = 2$. This partition is used to construct an optimal partition by varying the parameters Δt_{min} and K and removing the unnecessary groups. The outline of the optimization procedure is the following

1. Using the values of $\Delta t_{\min}$ and K defined by (6.6) the following two parameters for the new multirate partition are introduced

$$\Delta t^*_{\min} = \alpha \Delta t_{\min}, \quad \alpha \in [0.8, 1], \tag{6.7}$$

$$K^* = \beta K, \quad \beta \in [0.8K, 1.2K]. \tag{6.8}$$

2. For a randomly chosen pair $\left(\Delta t^*_{\min}, K^*\right)$ using a certain search procedure

   (a) construct a multirate partition;

   (b) remove unnecessary multirate groups: if a subdomain with Δt_k consists of a few isolated elements, add it to the subdomain with Δt_{k-1};

   (c) estimate the speedup with resulting multirate partition.

3. Go to step 2 if convergence criteria is not satisfied. After convergence, take the best estimated partition as the final one.

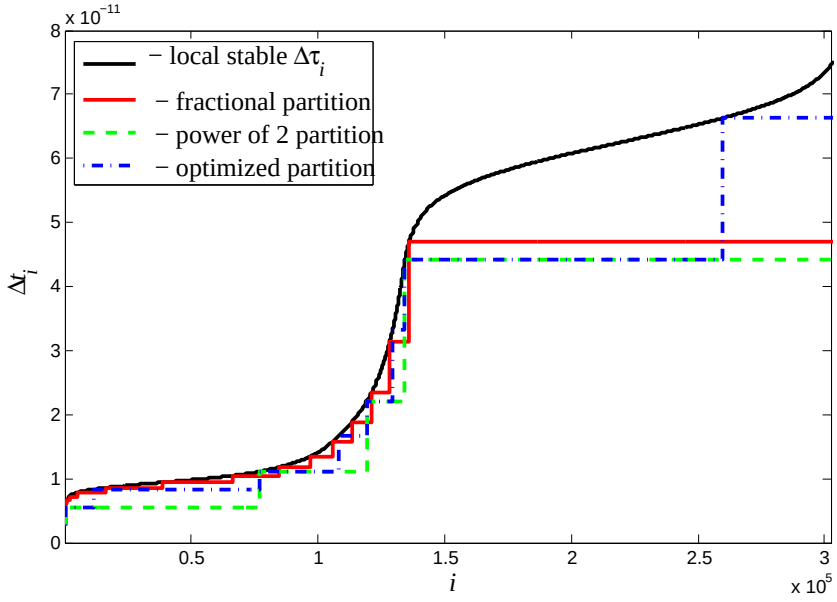


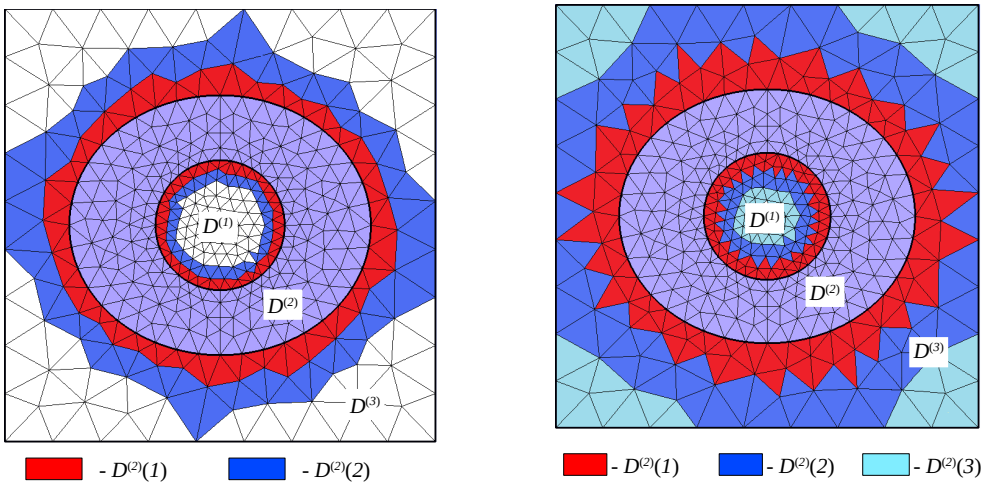
Figure 6.2: An example of distributions of local time-steps based on different partitions for a problem of scattering from a PEC sphere on non-uniform mesh presented in Chapter 7.

In the present work an improved controlled random search algorithm by [92] was used as a searching procedure. The resulting partition will have $\Delta t_1 = \Delta t^*_{\min}$, $\Delta t = K^* \Delta t^*_{\min}$, and the total number of multirate groups $M \leq K^*$. The convergence criteria is based on theoretical speedup formula given by

$$S = \frac{T \Delta t_{\min}^{-1} sN}{T \sum_{k=1}^{K} \Delta t_k^{-1} sN_{D^{(k)}}}, \tag{6.9}$$

where N is the total number of mesh elements, s is the number of Runge-Kutta stages, and $N_{D^{(k)}}$ is the number of elements in $D^{(k)}$ multirate group. During the initialization, the local time-step partition is computed and subdomains are determined.

An example demonstrating different partitions is shown on Figure 6.2. It reveals that the fractional partition (6.5) has too many local time-steps that are very close in magnitude, making this partition inefficient without an optimization step. It also demonstrates that the optimized partition gives a more efficient distribution of local-time steps compared to the power of 2 and fractional partitions.



(a) second order scheme (b) third order scheme

Figure 6.3: Outer buffer groups for the multirate group $D^{(2)}$ for the approximation using (a) MRK2-TW and MUSCL schemes, (b) MRK3-TW and WENO3 schemes.

6.3 Tang-Warnecke scheme

6.3.1 Outer buffer groups

Computations in the inner buffer of each multirate group $D^{(k)}$ involve values from adjacent multirate groups. The coupling in Tang-Warnecke multirate scheme is done by projecting the solution in the outer buffer using Runge-Kutta scheme with adapted time-step. The number of elements in the outer buffer group is defined by the stencil of the spatial scheme and the number of Runge-Kutta stages. For s -stage Runge-Kutta method we define s outer buffer subgroups $D^{(k)}(q)$, $q = 1,...,s$. Each subgroup $D^{(k)}(q)$ consists of r layers of elements, where r is the order of spatial reconstruction. Then the outer buffer is nothing but the union $\cup_{q=0}^{s} D^{(k)}(q)$. In this work multirate schemes based on SSP RK2 and SSP RK3 are considered with the same order space schemes. The first case requires two outer buffer groups $D^{(k)}(1) \cup D^{(k)}(2)$ consisting of two layers of elements, and second case requires three outer buffer groups $D^{(k)}(1) \cup D^{(k)}(2) \cup D^{(k)}(3)$ consisting of three layers of elements. A two-dimensional example of outer buffer groups is given in Figure 6.3.

6.3.2 Time integration

To describe the multirate algorithm based on Tang-Warnecke scheme, which will be referred to as MRK-TW, consider the source free case of the semi-discrete system of Maxwell's equations (2.82) which can be written as

$$\boldsymbol{U}_t = L\boldsymbol{U}. \tag{6.10}$$

Consider the partition into K multirate groups with time-steps Δt_k defined by any partition method. Let m be the number of local time updates (multirate stages) from $t^n = t^{n,0}$ to $t^{n+1} = t^n + \Delta t = t^{n,m}$ between synchronizations. For example, in Figure 6.5 we have 6 updates of local times, therefore $m = 6$. Local times $t_k^{n,l}$, $1 \le l \le m$, associated with each multirate group $D^{(k)}$ are updated at the beginning of the time cycle by

$$t_k^{n,l} = \begin{cases} t_k^{n,l-1} + \Delta t_k, & \text{if } t_k^{n,l-1} = t^{n,l-1}, \\ t_k^{n,l-1}, & \text{if } t_k^{n,l-1} > t^{n,l-1}, \end{cases} \tag{6.11}$$

then the global time corresponding to the l-th multirate stage is obtained by

$$t^{n,l} = \min_k t_k^{n,l}. \tag{6.12}$$

An example of local and global times for MRK-TW scheme is given in Figure 6.4.

At the beginning of each multirate stage l for multirate groups $D^{(k)}$ with $t_k^{n,l} = t^{n,l}$ and their outer buffers $\cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap \left(\cup_{r=1}^{s+1-q} D^{(k)}(r)\right)\right)$ we define the initial stage values by

$$\boldsymbol{W}^{(1)} = \begin{cases} \boldsymbol{U}_k^{n,l-1}, & \text{on } D^{(k)}, \\ \boldsymbol{U}_j^{n,l^*}, & \text{on } \cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap \left(\cup_{r=1}^{s} D^{(k)}(r)\right)\right), \end{cases} \tag{6.13}$$

here $l^* \le l-1$ is the last multirate stage with $t_k^{n,l^*} = t_j^{n,l^*}$. The q-th stage value of Runge-Kutta scheme on multirate groups $D^{(k)}$ is then computed by

$$\boldsymbol{U}_k^{(q)} = \boldsymbol{U}_k^{n,l-1} + \Delta t_k \sum_{r=1}^{q-1} a_{qr} L_k \boldsymbol{W}^{(r)},\ q = 2, ..., s, \tag{6.14}$$

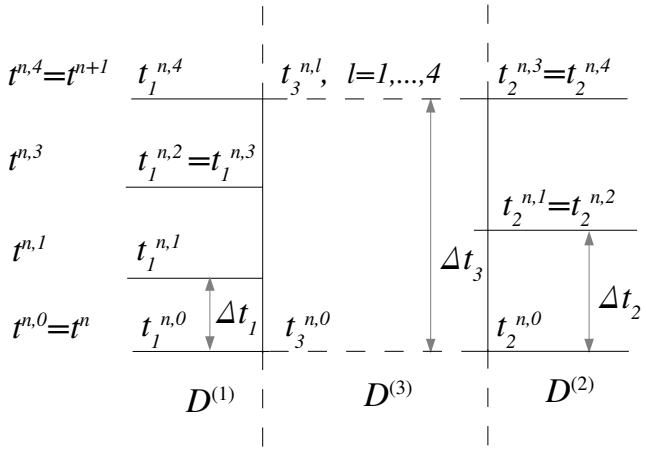


Figure 6.4: Example of local and global times for MRK-TW scheme.

and coupling values denoted by $\boldsymbol{V}_j^{(q)}$ are computed in the outer buffer of $D^{(k)}$ by

$$\boldsymbol{V}_j^{(q)} = \boldsymbol{U}_j^{n,l^*} + \Delta t_{k,j} \sum_{r=1}^{q-1} a_{qr} L_j \boldsymbol{W}^{(r)}, \tag{6.15}$$

where $\Delta t_{k,j} = t_k^{n,l} - t_j^{n,l^*}$ and

$$\boldsymbol{W}^{(q)} = \begin{cases} \boldsymbol{U}_k^{(q)}, & \text{on } D^{(k)}, \\ \boldsymbol{V}_j^{(q)}, & \text{on } \cup_{j=1,j\neq k}^{K} \left(D^{(j)} \cap \left(\cup_{r=1}^{s+1-q} D^{(k)}(r) \right) \right), \end{cases} \quad q = 2,...,s.$$

Below the steps of the MRK-TW algorithm for a given multirate stage l, $l \in \{1,...,m\}$, of the n-th global time-step are presented.

Algorithm 1 (MRK-TW):

1. Compute local times $t_k^{n,l}$, $k = 1,...,K$ by (6.11);
2. compute global time $t^{n,l}$ by 6.12;
3. for all $D^{(k)}$, $k = 1,...,K$, with $t_k^{n,l} = t^{n,l}$
   (a) assign initial stage values $\boldsymbol{W}^{(1)}$ using (6.13);

(b) for $q = 2, ..., s$

i. compute the stage values $\boldsymbol{U}_k^{(q)}$ on $D^{(k)}$ using (6.14);

ii. compute the coupling values $\boldsymbol{V}_j^{(q)}$ on $\cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap \left(\cup_{r=1}^{s+1-q} D^{(k)}(r)\right)\right)$ using (6.15) and form

$$\boldsymbol{W}^{(q)} = \begin{cases} \boldsymbol{U}_k^{(q)}, & \text{on } D^{(k)}, \\ \boldsymbol{V}_j^{(q)}, & \text{on } \cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap \left(\cup_{r=1}^{s+1-q} D^{(k)}(r)\right)\right); \end{cases}$$

(c) compute the final Runge-Kutta step by

$$\boldsymbol{U}_k^{n,l} = \boldsymbol{U}_k^{n,l-1} + \Delta t_k \sum_{q=1}^{s} b_q L_k \boldsymbol{W}^{(q)} \qquad \text{on } D^{(k)}.$$

Here we note that if $t_k^{n,l} = t_j^{n,l}$ additional computations in the outer buffer are not required, since the stage values are obtained in two subdomains simultaneously. The main contribution to the computational overhead in MRK-TW scheme is due to additional flux evaluations ($L\boldsymbol{U}$) in outer buffers which grows with order of a scheme. The next scheme doesn't have this drawback, since it is based on interpolation of RK stage values in outer buffers.

An illustration of the above algorithm for the second order case is given in Figure 6.5, where we have 4 multirate groups and 6 multirate stages.

6.4 Liu-Li-Hu linear scheme

6.4.1 Coupling buffer

The coupling in the Liu-Li-Hu linear multirate scheme multirate scheme, which is referred to as MRK-LLH here, is done by a modification of the latest RK stage values in the outer buffer. No additional time integrations, and therefore FV reconstructions, are required by this scheme. Hence the size of the outer buffer depends only on the stencil size for spatial approximation. Using the notations from the previous section the outer buffer for MRK-LLH scheme consists of $D^{(k)}(1)$. An example is given in Figure 6.6.

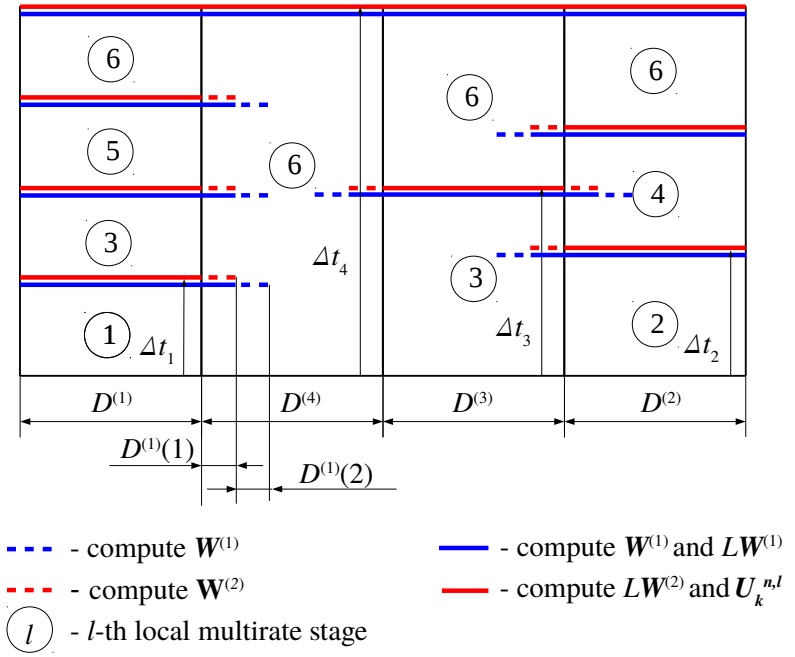


Figure 6.5: Example of MRK2-TW algorithm for a time slab $\left(t^n, t^{n+1}\right]$.

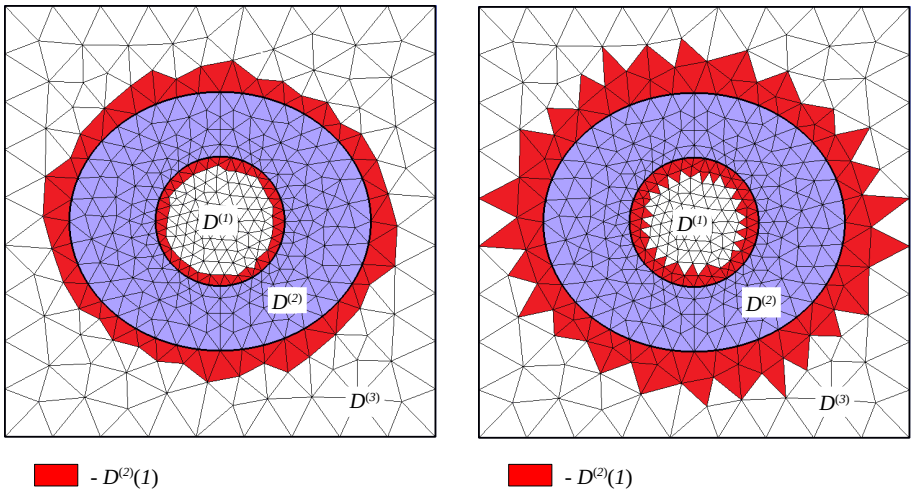


Figure 6.6: Outer buffer groups for the multirate group $D^{(2)}$ for the approximation using (a) MRK2-LLH and MUSCL schemes, (b) MRK3-LLH and WENO3 schemes.

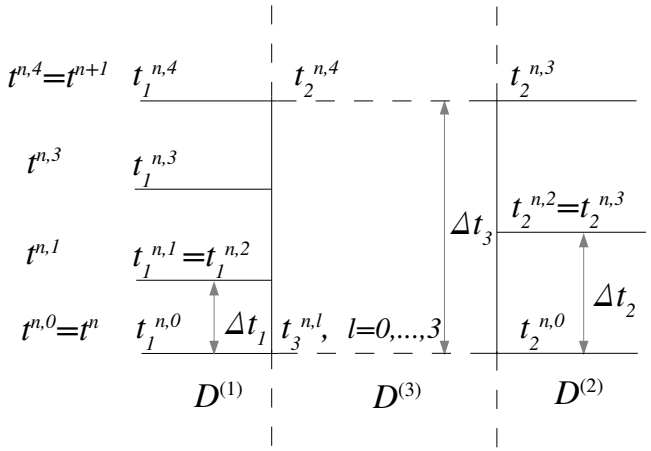


Figure 6.7: Example of local and global times for MRK2-LLH scheme.

6.4.2 Time integration

Consider again the partition into K multirate groups with time-steps Δt_k, which define the multirate time-stepping with global time-step Δt consisting of m multirate stages. Unlike in MRK-TW scheme, here local times $t_k^{n,l}$, $l \in \{0,...,m-1\}$, associated with each multirate group $D^{(k)}$ are updated at the end of the l-th stage by

$$t_k^{n,l+1} = \begin{cases} t_k^{n,l} + \Delta t_k, & \text{if } t_k^{n,l} + \Delta t_k = t^{n,l+1}, \\ t_k^{n,l}, & \text{if } t_k^{n,l} + \Delta t_k > t^{n,l+1}, \end{cases} \tag{6.16}$$

where

$$t^{n,l+1} = \min_k \left(t_k^{n,l} + \Delta t_k \right). \tag{6.17}$$

An example of local and global times for MRK-LLH scheme is given in Figure 6.7.

At each multirate stage l for every $D^{(k)}$ with $t_k^{n,l} = t^{n,l}$ the coupling RK stage values $\boldsymbol{V}_j$ are computed in the outer buffer $\cup_{j=1,j\neq k}^{K} \left(D^{(j)} \cap D^{(k)}(1) \right)$ by

$$\boldsymbol{V}_j^{(q)} = \begin{cases} \sum_{r=1}^{q} \left[\boldsymbol{C}\boldsymbol{T}_{\Delta t_j} \boldsymbol{T}_{\Delta t_k}^{-1} \boldsymbol{C}^{-1} \right]_{qr} \boldsymbol{U}_j^{(r)}, & \text{if } t_j^{n,l} = t_k^{n,l}, \\ \sum_{r=1}^{s} \left[\boldsymbol{C}\boldsymbol{T}_{\Delta t_j} \boldsymbol{H}_{\Delta t_{k,j}^{n,l}} \boldsymbol{T}_{\Delta t_k}^{-1} \boldsymbol{C}^{-1} \right]_{qr} \boldsymbol{U}_j^{(r)}, & \text{if } t_j^{n,l} < t_k^{n,l}, \end{cases} \tag{6.18}$$

where $\Delta t_{k,j}^{n,l} = t_k^{n,l} - t_j^{n,l}$, $\boldsymbol{U}_j^{(r)}$ are the RK stage values on $D^{(j)}$ at $t_j^{n,l}$, and matrices $\boldsymbol{C}$, $\boldsymbol{T}_{\Delta t}$, and $\boldsymbol{H}_{\Delta t}$ are defined by (5.44), (5.45) and (5.56) respectively. Then the time integration is performed on $D^{(k)}$. Below the steps of the MRK-LLH algorithm for a given multirate stage l, $l \in \{0,...,m-1\}$, of the n-th global time-step are presented.

Algorithm 2 (MRK-LLH):

1. for all $D^{(k)}$, $k = 1,...,K$, with $t_k^{n,l} = t^{n,l}$

   (a) compute $\boldsymbol{V}_j^{(1)}$ on $\cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap D^{(k)}(1)\right)$ by (6.18) and form the initial stage value vector by

$$\boldsymbol{W}^{(1)} = \begin{cases} \boldsymbol{U}_k^{(1)} & \text{on } D^{(k)}, \\ \boldsymbol{V}_j^{(1)} & \text{on } \cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap D^{(k)}(1)\right); \end{cases}$$

   (b) for $q = 2,...,s$

      i. compute the stage values $\boldsymbol{U}_k^{(q)}$ on $D^{(k)}$

$$\boldsymbol{U}_k^{(q)} = \boldsymbol{U}_k^{n,l} + \Delta t_k \sum_{r=1}^{q-1} a_{qr} L_k \boldsymbol{W}^{(r)};$$

      ii. compute the coupling stage values $\boldsymbol{V}_j^{(q)}$ on $\cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap D^{(k)}(1)\right)$ by (6.18) and form

$$\boldsymbol{W}^{(q)} = \begin{cases} \boldsymbol{U}_k^{(q)} & \text{on } D^{(k)}, \\ \boldsymbol{V}_j^{(q)} & \text{on } \cup_{j=1,j\neq k}^{K}\left(D^{(j)} \cap D^{(k)}(1)\right); \end{cases}$$

   (c) compute the final Runge-Kutta step by

$$\boldsymbol{U}_k^{n,l+1} = \boldsymbol{U}_k^{n,l} + \Delta t_k \sum_{q=1}^{s} b_q L_k \boldsymbol{W}^{(q)} \qquad \text{on } D^{(k)};$$

2. compute $t^{n,l+1} = \min_k \left(t_k^{n,l} + \Delta t_k\right)$;

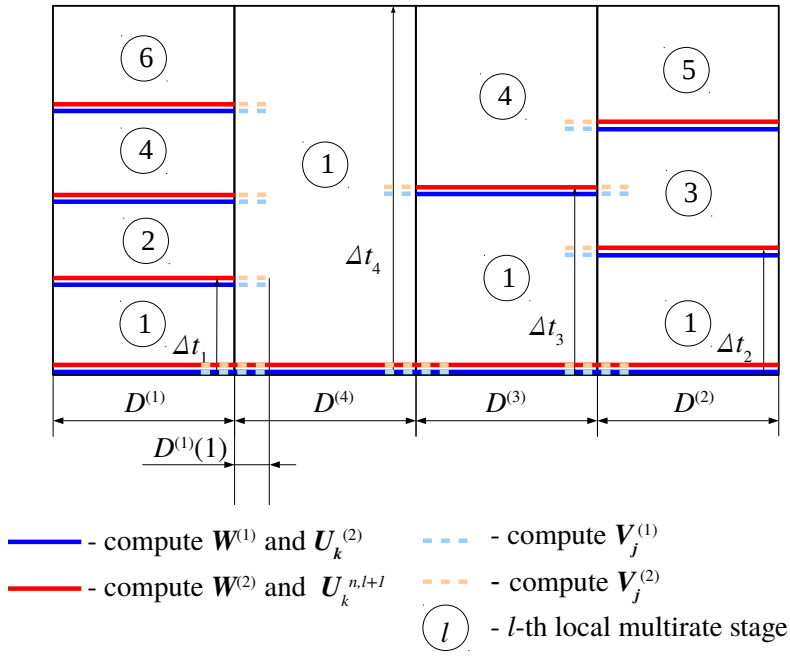


Figure 6.8: Example of MRK2-LLH algorithm for a time slab $\left(t^n, t^{n+1}\right]$.

3. compute local times for $k = 1, \ldots, K$ by

$$t_k^{n,l+1} = \begin{cases} t_k^{n,l} + \Delta t_k, & \text{if } t_k^{n,l} + \Delta t_k = t^{n,l+1}, \\ t_k^{n,l}, & \text{if } t_k^{n,l} + \Delta t_k > t^{n,l+1}. \end{cases}$$

In the above algorithm we store all RK stage values $\boldsymbol{U}_j^{(q)}$ in outer buffers $\cup_{j=1, j\neq k}^{K} \left(D^{(j)} \cap D^{(k)}(1) \right)$ of each $D^{(k)}$ in order to perform coupling at every multirate stage l. No additional evaluations of fluxes by FV scheme are needed to provide coupling in this case, since no RK steps are performed in the outer buffer. The computational overhead for MRK-LLH depends only on the number of interpolations of RK stage values in outer buffers. To conclude the section we present an illustration of the above algorithm for the same example as in previous section in Figure 6.8.

6.5 Chapter summary

The implementation of two different multirate techniques based on explicit SSP Runge-Kutta method for multidimensional problems on unstructured meshes is presented in this chapter. General guidelines for generating multirate groups based on different definitions of local time-steps are discussed.

MRK-TW method is simpler to implement since it is based on a projection of the solution to provide a coupling between two multirate groups with different time-steps. No extra storage and adaptation of stage values at the interface are needed. But the projection of the solution in elements of the outer buffer requires the most expensive operation ($\boldsymbol{LU}$ evaluation) on these extra elements, which decreases the actual speedup of the multirate approach. Another problem is that the stage values in outer buffers are not always internally consistent with inner buffer, which causes the loss of accuracy. As it was shown in the previous chapter the MRK-TW scheme is second order accurate with SSP RK2, but only first order accurate with SSP RK3. These theoretical results are tested further on three-dimensional problems in the next chapter.

MRK-LLH scheme is more accurate but the coupling is slightly more difficult to implement. Since the coupling is based on adjustments of RK stage values in the outer buffer, the scheme is consistent at interfaces between two multirate groups. Also no additional RK steps are required in outer buffers, therefore their size is much smaller for MRK-LLH than for MRK-TW. But since in MRK-LLH method more adjustments of stage values are needed for coupling, the speedup from the scheme is not greater than for MRK-TW method.

Chapter 7

Numerical experiments

In this chapter the performance of both the type II WENO scheme presented in Chapter 4 and multirate Runge-Kutta schemes from Chapter 6 are evaluated numerically. The first example is the problem of scattering from a perfectly conducting (PEC) sphere which uses the scattered FVTD formulation. Then a series of examples are performed for a problem of plane-wave propagation in a parallel-plate waveguide utilizing the total field FVTD formulation. And, finally, series of simulations are carried out for a plane-wave reflection/transmission at a dielectric interface.

The three-dimensional schemes described in this thesis were implemented in C++. Tetrahedral meshes for all problems considered in this chapter are generated using the open source software Gmsh version 2.7.1 [1, 50]. This work also utilizes some of the modules of the FVTD engine developed earlier in [41], namely the mesh, physical parameters, and interface that opens the Gmsh generated tetrahedral mesh and stores a project file with physical parameters for the problem. The mesh module is refactored with some new functions required by new schemes are added to mesh classes. The new flux splitting formulation in normalized form and sources were implemented by Dr. Dmitry Firsov. The author's contribution to the code is the implementation of new algorithms described in Chapters 4 and 6, namely single and multirate Runge-Kutta schemes of second and third order and third order polynomial and WENO schemes. To find pseudo-inverse matrices with SVD decomposition for WENO reconstructions the open source Armadillo C++ linear algebra library [107] was used. Matlab [2] was utilized for two-dimensional plots of time-domain

solutions. For three-dimensional visualizations the open source software ParaView [6, 3] was used. Numerical experiments were carried out on Intel i7-4790k with 4.4 GHz quad core CPU with 16 GB of RAM.

7.1 Scattering from a PEC sphere

The problem of plane wave scattered by a conducting sphere is one of the classic problems in electromagnetics for which an analytic series solution is known. This problem is used as a reference to measure scattering properties of other objects. In CEM it is used to validate the solution obtained by a given numerical method. An analytical solution for scattering from PEC is available in frequency-domain as a series expressed by spherical Hankel and Bessel functions and associated Legendre polynomials. The time-domain solution can be obtained by applying inverse Fourier transforms to the solution.

7.1.1 Analytic solution in frequency domain

The analytic series solution for the electromagnetic field scattered from a perfectly conducting sphere can be found in many standard textbooks on electromagnetics. Here the solution given by Harrington in [64] and Balanis in [13] is used. It is found by solving the scalar Helmholtz wave equation in spherical coordinates, which is obtained from the time-harmonic Maxwell's equations.

Consider a conducting sphere of radius a illuminated by the plane wave that is x-polarized and is traveling in z direction. The geometry of the problem is shown in Figure 7.1. The electrical field of an incident wave of radial frequency $\omega = 2\pi f$, where f is a time-harmonic frequency [64], is given by

$$\mathscr{E}_x^I = E_0 e^{-i\beta z} = E_0 e^{-i\beta r\cos\theta}, \tag{7.1}$$

$$\mathscr{E}_y^I = 0, \tag{7.2}$$

$$\mathscr{E}_z^I = 0, \tag{7.3}$$

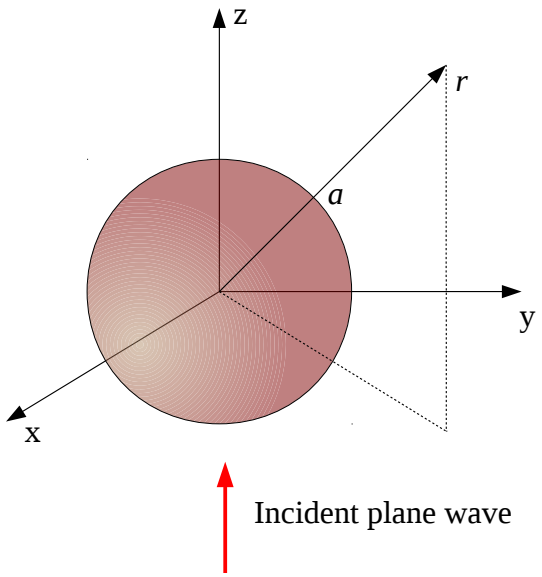


Figure 7.1: Scattering from a PEC sphere: incident plane wave on a PEC sphere.

where $\beta = \omega\sqrt{\mu_0\varepsilon_0}$ is the wave number. The spherical components of the scattered electric field are then found to be [13]

$$\mathscr{E}_r^S = -iE_0\cos\phi\sum_{n=1}^{\infty} b_n\left[\hat{H}_n^{(2)''}(\beta r)+\hat{H}_n^{(2)}(\beta r)\right]P_n^1(\cos)\,\theta, \tag{7.4}$$

$$\mathscr{E}_\theta^S = \frac{E_0}{\beta r}\cos\phi\sum_{n=1}^{\infty}\left[ib_n\hat{H}_n^{(2)'}(\beta r)\sin\theta P_n^{1'}(\cos\theta)-c_n\hat{H}_n^{(2)}(\beta r)\frac{P_n^1(\cos\theta)}{\sin\theta}\right], \tag{7.5}$$

$$\mathscr{E}_\phi^S = \frac{E_0}{\beta r}\sin\phi\sum_{n=1}^{\infty}\left[ib_n\hat{H}_n^{(2)'}(\beta r)\frac{P_n^1(\cos\theta)}{\sin\theta}-c_n\hat{H}_n^{(2)'}(\beta r)\sin\theta P_n^{1'}(\cos\theta)\right], \tag{7.6}$$

where $\hat{H}_n^{(2)}$ are n-th order spherical Hankel functions of the second kind, P_n^1 are n-th order associated Legendre polynomials of the first kind, and differentiation is performed with respect to the argument. The coefficients a_n, b_n and c_n in the series are obtained for a given sphere of radius $r = a$ by

$$a_n = \frac{j^{-n}(2n+1)}{n(n+1)},\quad b_n = -a_n\frac{\hat{J}_n'(\beta a)}{\hat{H}_n^{(2)'}(\beta a)},\quad c_n = -a_n\frac{\hat{J}_n(\beta a)}{\hat{H}_n^{(2)}(\beta a)}, \tag{7.7}$$

where $\hat{J}_n$ are spherical Bessel functions. It should be noted that the solution (7.4-7.6) has a singularity at $\theta = \pi$. In this case the following relationships are used

$$\frac{P_n^1(\cos\theta)}{\sin\theta} \to \frac{(-1)^n n(n+1)}{2}, \quad \theta \to \pi, \tag{7.8}$$

$$P_n^{1'}(\cos\theta) \to \frac{(-1)^n n(n+1)}{2}, \quad \theta \to \pi. \tag{7.9}$$

Cartesian components of the electric field are recovered from (7.4-7.6) by the transformation

$$\left[\mathscr{E}_x^S, \mathscr{E}_y^S, \mathscr{E}_z^S\right]^T = \begin{bmatrix} \sin\theta\cos\phi & \cos\theta\cos\phi & -\sin\phi \\ \sin\theta\sin\phi & \cos\theta\sin\phi & \cos\phi \\ \cos\theta & \sin\theta & 0 \end{bmatrix} \begin{bmatrix} \mathscr{E}_r^S \\ \mathscr{E}_\theta^S \\ \mathscr{E}_\phi^S \end{bmatrix}.$$

7.1.2 Time-domain solution

Consider a PEC sphere with radius $a = 0.5$ [m]. The x component of the electric field of the incident plane wave E_x^I is given by the derivative of the Gaussian pulse

$$E_x^I = -2\frac{t-t_0}{b^2} A e^{-\frac{(t-t_0)^2}{b^2}}, \tag{7.10}$$

where

$$A = 1.7489 \times 10^{-9}\,[V/m]\,,\ b = 1.5 \times 10^{-9}\,[s]\,,\ t_0 = 6 \times 10^{-9}\,[s]\,. \tag{7.11}$$

The pulse given by (7.10-7.11) and its spectrum are shown in Figure 7.2.

Then E_0 in (7.1) is obtained from E_x^I for a finite number of frequencies ω by taking the Fourier transform. To obtain the analytic time-domain solution first the solution in frequency-domain (7.4-7.6) is computed using a finite number of elements in the summation, then the real part of its inverse Fourier transform is taken. This is done for certain observation points.

7.1.3 Numerical validation of WENO scheme

As it was shown in Chapter 3, the accuracy of WENO scheme depends on the value of the small parameter ε in the definition of non-linear weights. For smooth solutions the best

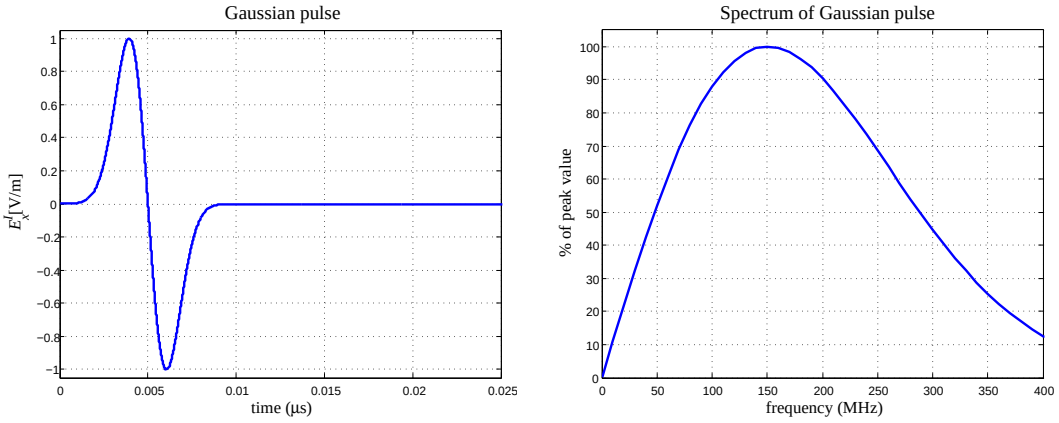


Figure 7.2: Scattering from a PEC sphere: Incident field E_x^I.

accuracy in the one-dimensional case was achieved for $\varepsilon = ah^k$, where $k \leq 2$. In the following series of experiments we would like to check how the value of ε affects the solution by WENO scheme on three-dimensional unstructured meshes. Initial numerical experiments with fixed choice of $\varepsilon = 10^6$ in the formula for non-linear weights (4.48) demonstrated very inaccurate solutions. Therefore, following the theory presented in Chapter 3 a small parameter ε is defined as a function of linear cell size h defined by (4.13) on each element T_i. For simplicity the index i is omitted, but it is assumed that ε and is different on each T_i. It is important that either the scaling parameter a is used for non-normalized FVTD formulation, or $a = 1$ otherwise. If ε is not scaled properly, then the solution accuracy can not be controlled using the theory of Chapter 3.

For the numerical simulation the free space domain is truncated at the distance of 2[m] from PEC sphere surface, this distance is enough for a short time computation. The geometry of the numerical problem is given in Figure 7.3. Smaller elements with the linear size 0.05 are taken in the region near PEC surface enclosed between the spheres of radii R_1 and R_2. In the region between the spheres of radii R_2 and R_3 the linear cell size gradually increases from 0.05 to 0.15. The mesh elements have relatively uniform size 0.15 in the region enclosed between the spheres of radii R_3 and R_4. The generated mesh consists of 196370 elements with 3010 of them containing a PEC face.

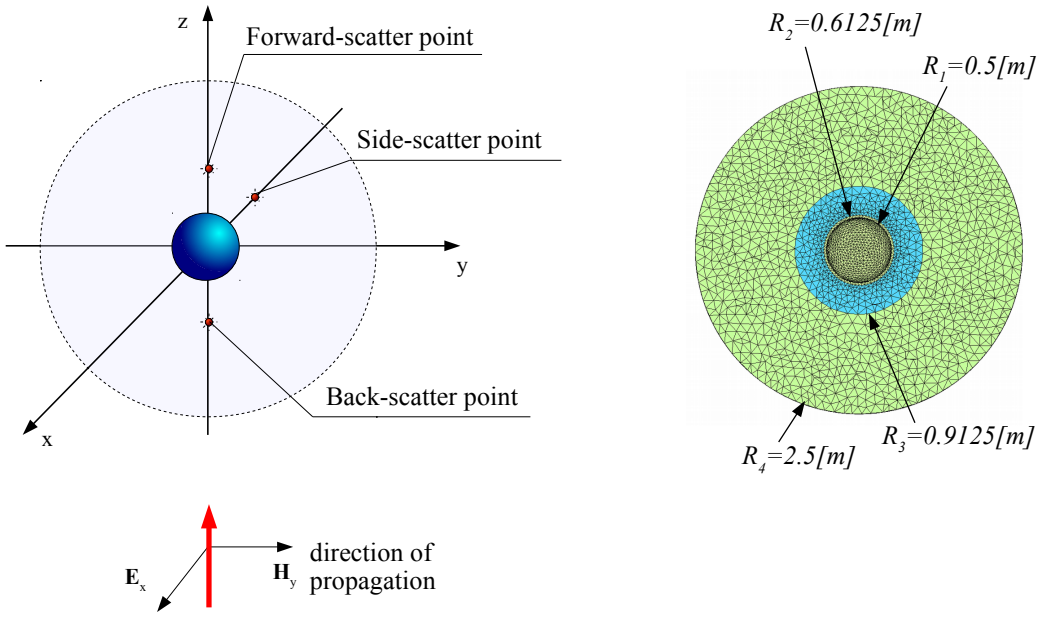


Figure 7.3: Scattering from PEC sphere: problem geometry and mesh.

As it was stated in Chapter 4 the quality of mesh cells affects the values of linear weights and, in most cases, negative linear weights are unavoidable. For mild negative weight a splitting technique is applied, but for a small number of Gauss points with very negative weights an alternative reconstruction has to be used to avoid instabilities. In the experiments presented in this subsection the following criteria is applied to linear weights

$$\max_{l}(|\gamma_l|) < \zeta, \tag{7.12}$$

where ζ is chosen to be equal to 2. This threshold on linear weights gives 2.078% of elements where the third order polynomial reconstruction is applied. More details on how the value of ζ affects the results will be discussed in the next section. The solution using the scattered FVTD formulation using WENO scheme with $\varepsilon = h,h^2,h^4$ as well as the linear scheme are compared to the analytic solution at the observation points (see Figure 7.3) located at a distance of 0.65[m] from PEC surface. The results for the E_x field component are presented in Figures 7.4, 7.5 and 7.6. The pointwise comparison of error in time reveals

	Side-scatter $(1.15,0,0)$	Side-scatter $(-1.15,0,0)$	Forward-scatter $(0,0,-1.15)$	Back-scatter $(0,0,-1.15)$
Linear	3.4646e-3	2.6937e-3	3.0290e-2	4.2612e-3
WENO3, $\varepsilon = h$	3.5060e-3	2.6895e-3	2.9756e-2	4.2369e-3
WENO3, $\varepsilon = h^2$	3.7354e-3	2.7809e-3	3.1922e-2	5.4391e-3
WENO3, $\varepsilon = h^4$	6.2229e-3	5.7343e-3	4.8555e-2	1.2554e-2

Table 7.1: Scattering from PEC sphere: $\max_{t^n} \left| E_x(t^n) - E_x^{Analytic}(t^n) \right|$ at observation points for the solution by third order linear and WENO schemes.

that WENO scheme with $\varepsilon = h^4$ gives the worst results, while the errors for $\varepsilon = h$,h^2 are close to the ones from the linear scheme. The maximum errors in the solution for E_x at observation points are presented in Table 7.1 indicating that the values of ε equal to h and h^2 are the most suitable choices for the problem.

7.1.4 Numerical validation of multirate schemes

The purpose of the experiments in this subsection is to compare the results for a PEC sphere obtained by multirate schemes with the analytic solution. First a mesh with higher cell size ratio than in the previous example is generated. This is done with help of four spherical surfaces shown on Figure 7.7. The average linear cell size between the spheres of radii R_1 and R_2 is 0.0225[m], and between the spheres of radii R_3 and R_4 it is 0.15[m]. The size of elements gradually increases from 0.0225[m] to 0.15[m] in the area between the spheres of radii R_2 and R_3. This partition generates a mesh with linear cell size ratio $1:6.667$ with the smallest elements located on the PEC boundary. The generated mesh consists of 307544 elements with 14374 of them containing a PEC face.

The experiments with both power of 2 (P2) and optimized partition (OP) are carried using the second and third order Tang-Warnecke and Liu-Li-Hu schemes. The partition into P2 multirate groups is shown on Figure 7.8. It contains only 199 elements with $\Delta t = \Delta t_{\min}$, which are scattered around PEC surface and are not visible on the figure. The ratio between the smallest and largest time-steps is $1:16$. The solution on the majority of elements evolves with time-steps $2\Delta t_{\min}$ and $16\Delta t_{\min}$, this gives a theoretical speedup equal to 4.21. The optimized partition is shown in Figure 7.9. It contains 126 elements with a minimum

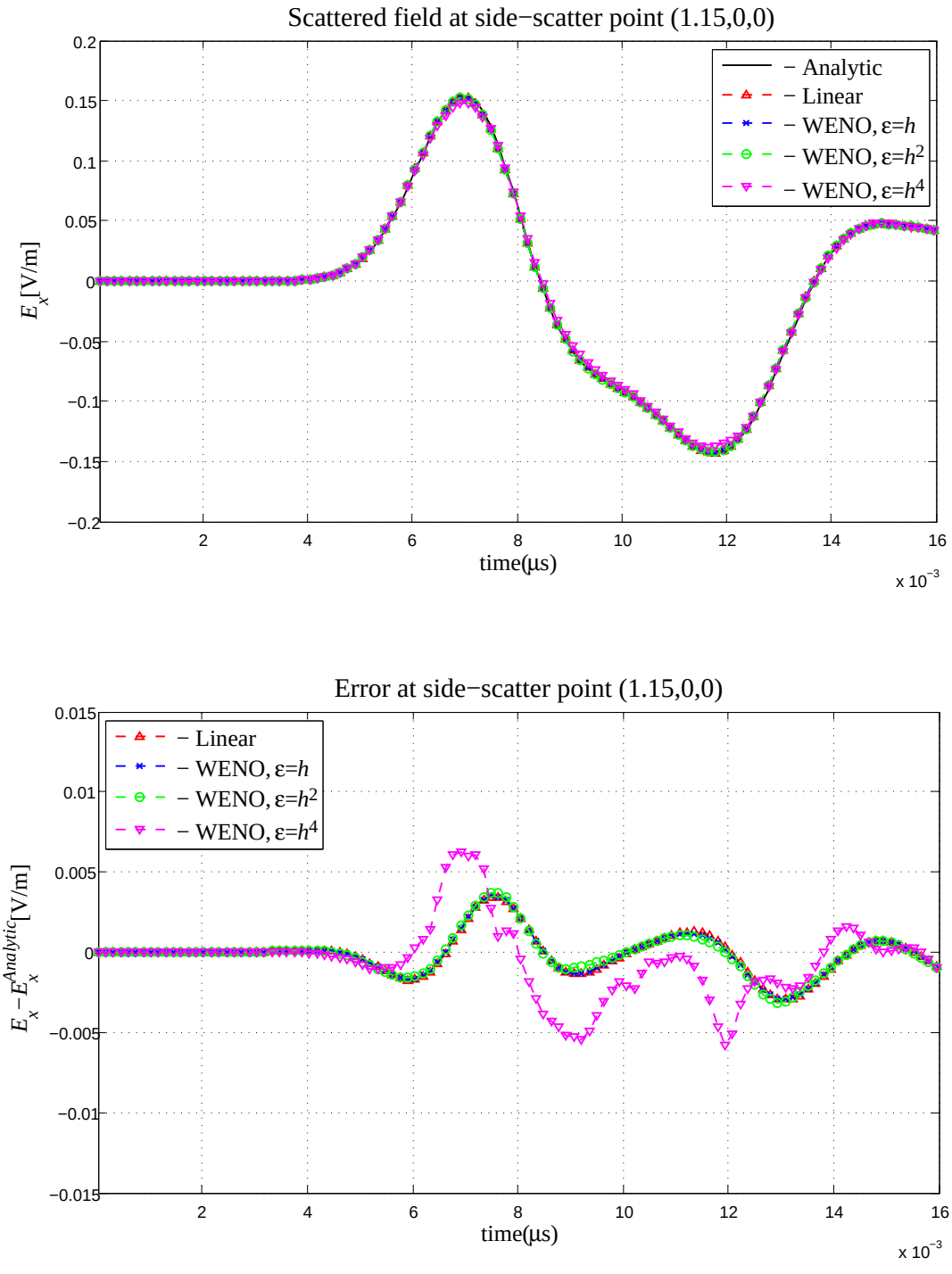


Figure 7.4: Scattering from PEC sphere: time-domain solution at side-scatter point (1.15,0,0) using third order linear and WENO schemes.

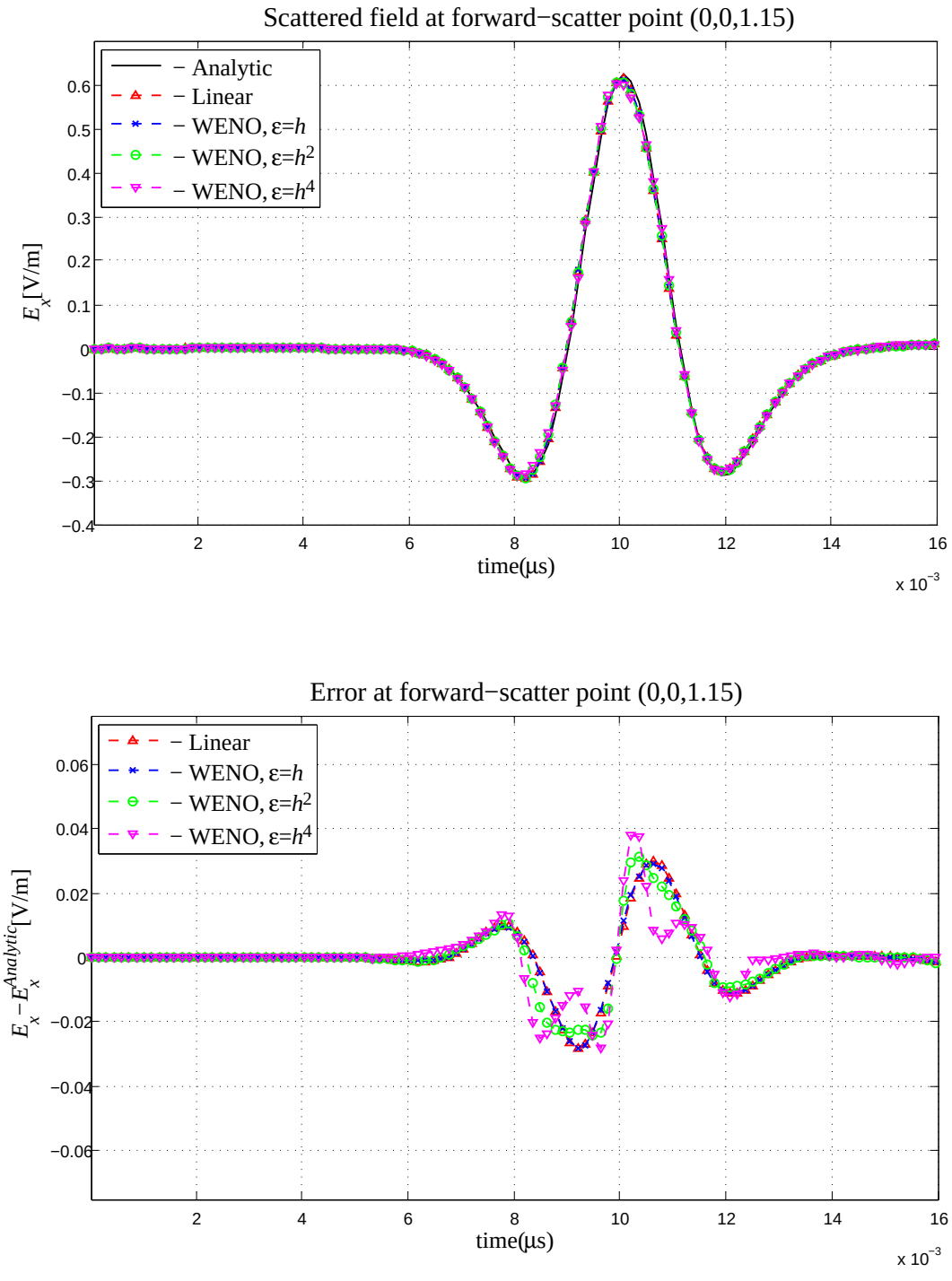


Figure 7.5: Scattering from PEC sphere: time-domain solution at forward-scatter point (0,0,1.15) using third order linear and WENO schemes.

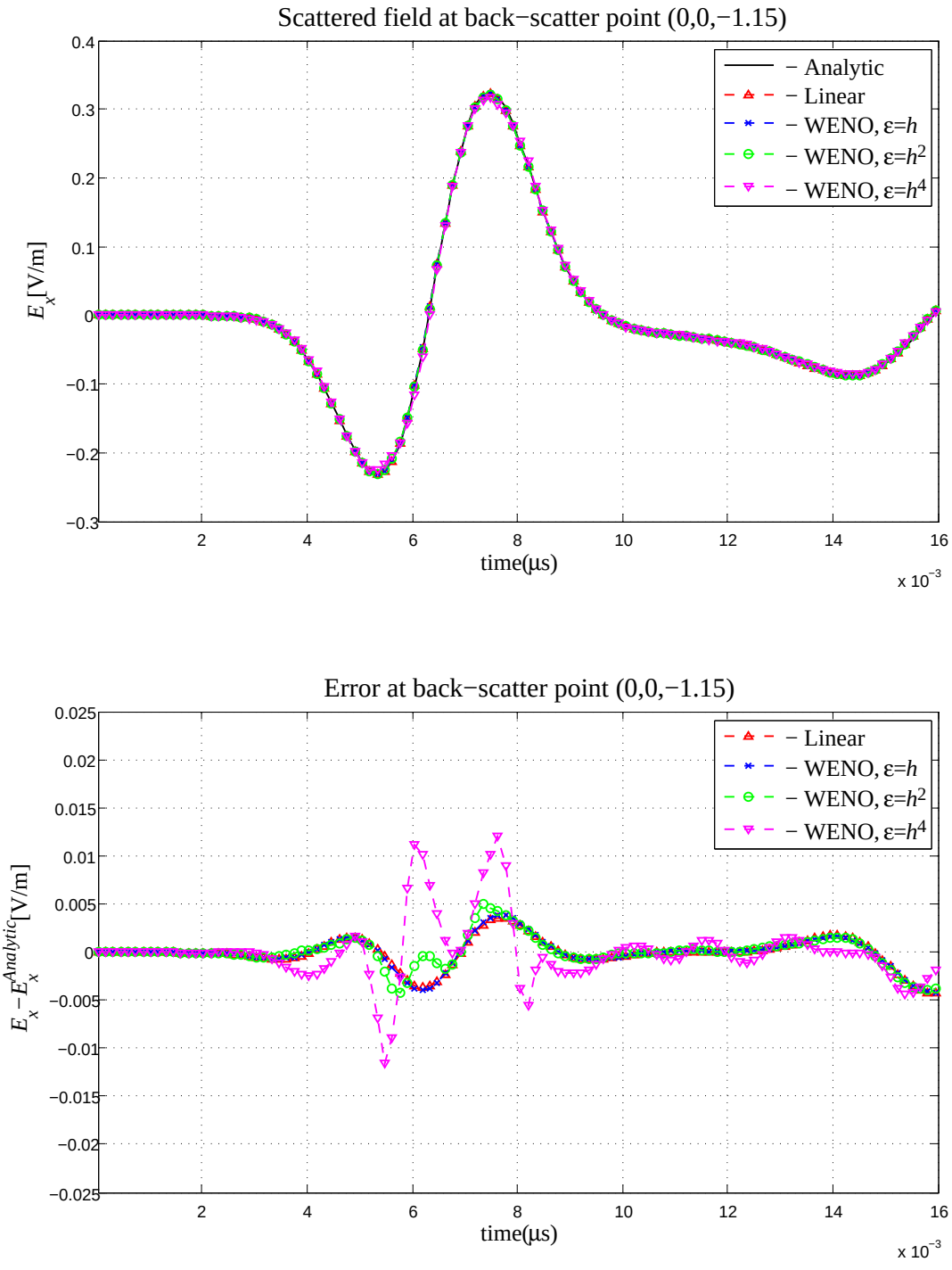


Figure 7.6: Scattering from PEC sphere: time-domain solution at back-scatter point (0,0,-1.15) using third order linear and WENO schemes.

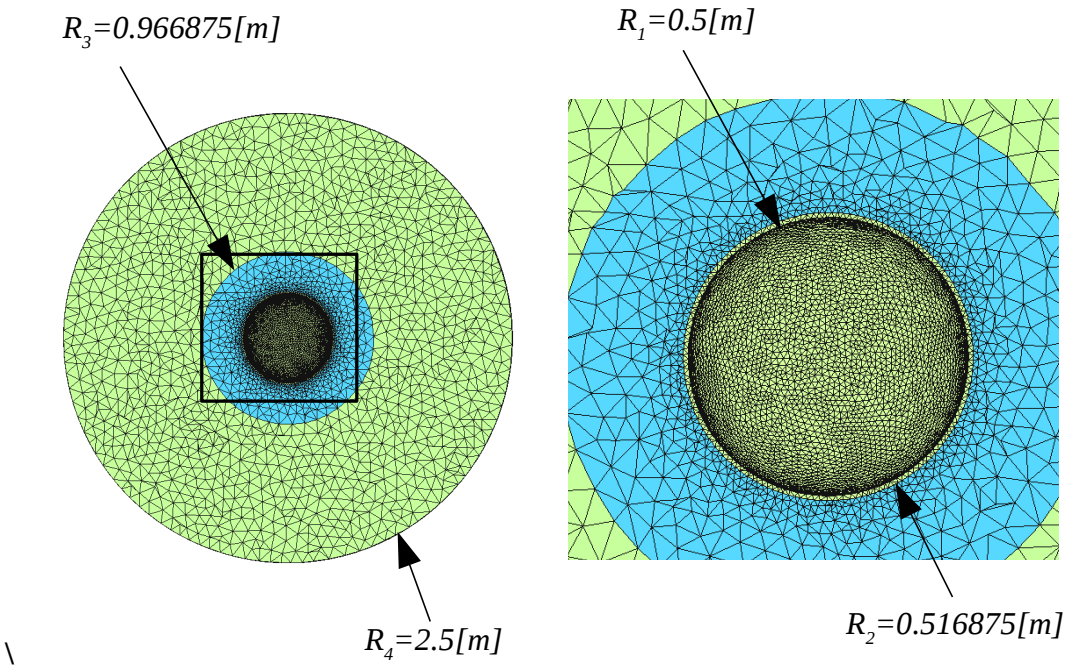
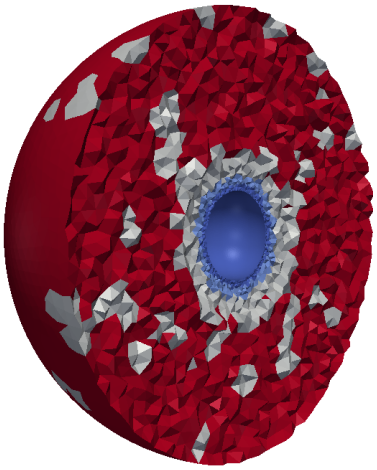


Figure 7.7: Scattering from PEC sphere: mesh for multirate partition.

time-step $\Delta t_{\min}$, and the time-step ratio is 1 : 12. The solution on the majority of cells evolves with time-steps $3\Delta t_{\min}$ and $12\Delta t_{\min}$, and the theoretical speedup is estimated to be 5.29. Since the theoretical speedup does not take into account the time for coupling between the MRK groups, the actual numerical speedup will be smaller and dependent on the scheme.

First the performance of second order multirate schemes is compared. The second order MUSCL scheme [17, 77] is used for space discretization with the SSP RK2 method in time. In Figures 7.10, 7.11 and 7.12 the solution at the observation points (see Figure 7.3) located at a distance of 0.65[m] from PEC surface is shown together with the pointwise errors in time. The results demonstrate that both multirate methods maintain the same accuracy when compared with the non-multirate version of SSP RK2 method regardless of the partition. These results agree with theoretical results for one-dimensional case presented in Chapter 5. In Table 7.2 the maximum errors at observation points are compared.

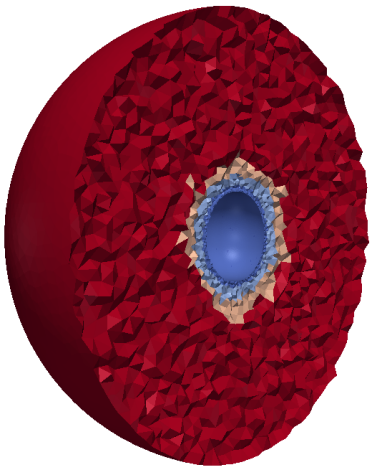
The same experiments are then carried with SSP RK3 base method in time and third order polynomial scheme in space (polynomial reconstruction on big stencils). The results



Multirate domains:

Δt_{min}: 199
$2\Delta t_{min}$: 116763
$4\Delta t_{min}$: 35325
$8\Delta t_{min}$: 38406
$16\Delta t_{min}$: 153456

Figure 7.8: Scattering from PEC sphere: power of 2 domain multirate partition for the mesh shown on Figure 7.7 with linear cells size ratio 1 : 6.667.



Multirate domains with $\Delta t^*_{min} = 0.88\Delta t_{min}$:

Δt^*_{min}: 126
$2\Delta t^*_{min}$: 1313
$3\Delta t^*_{min}$: 114144
$6\Delta t^*_{min}$: 38106
$9\Delta t^*_{min}$: 12952
$12\Delta t^*_{min}$: 177508

Figure 7.9: Scattering from PEC sphere: example of optimized multirate partition for the mesh shown on Figure 7.7 with linear cells size ratio 1 : 6.667.

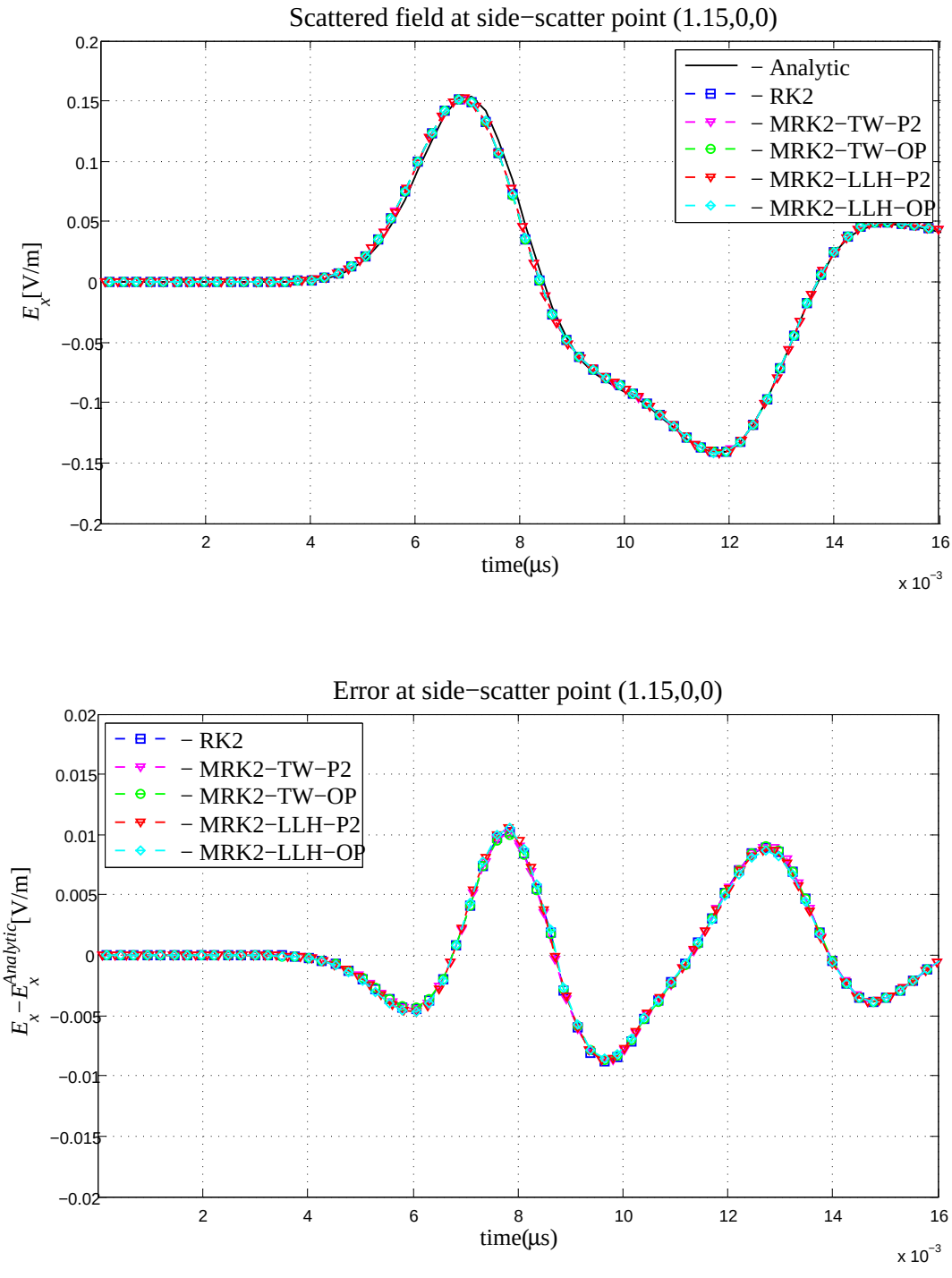
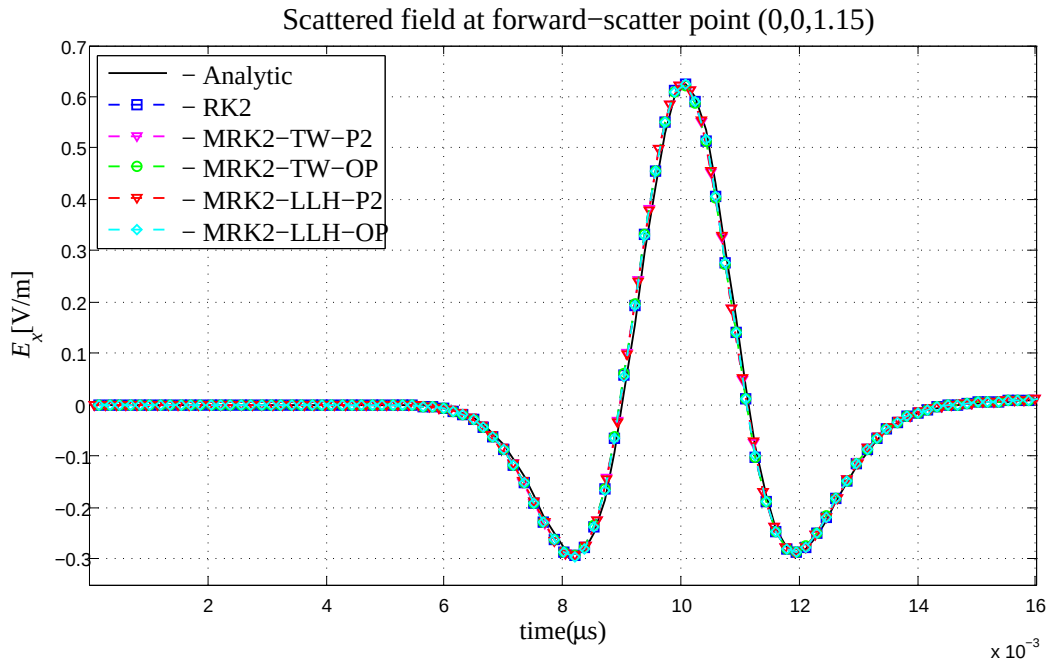


Figure 7.10: Scattering from PEC sphere: time-domain solution at side-scatter point (1.15,0,0) using RK2 and MRK2 schemes.



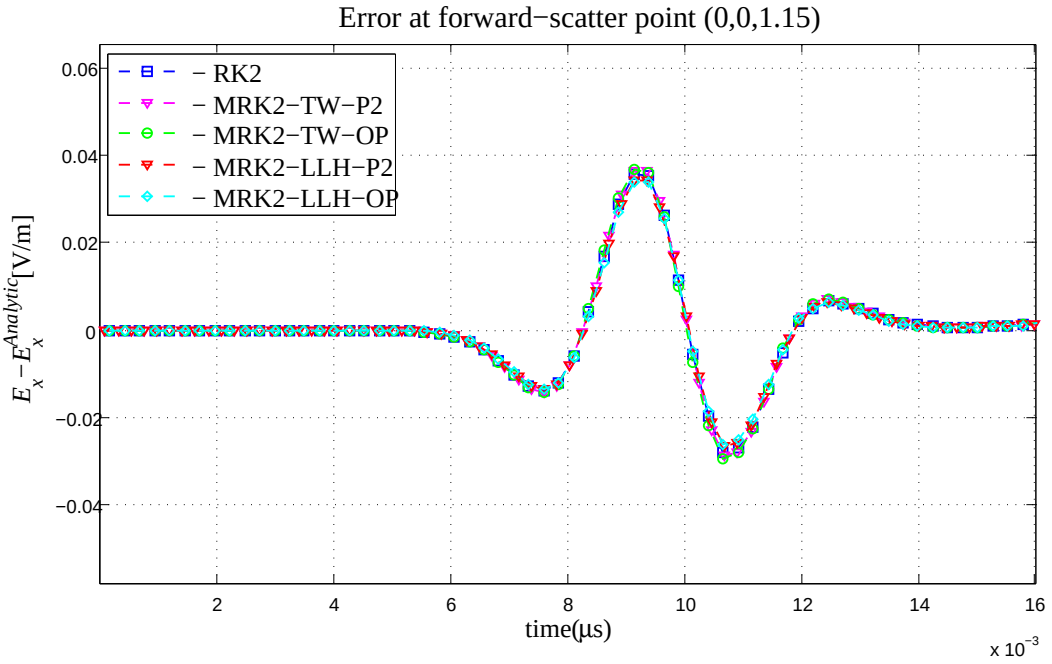


Figure 7.11: Scattering from PEC sphere: time-domain solution at forward-scatter point (0,0,1.15) using RK2 and MRK2 schemes.

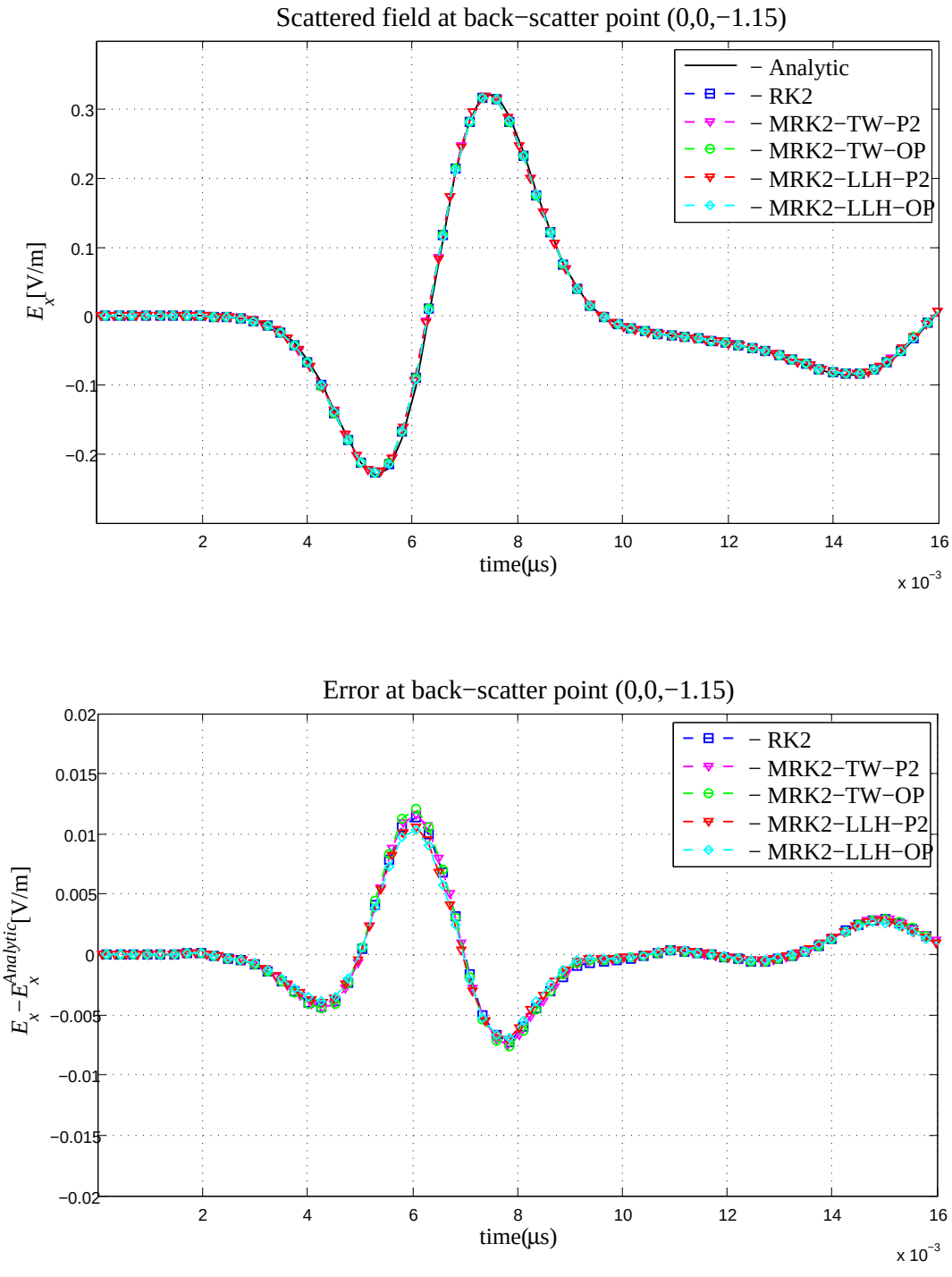


Figure 7.12: Scattering from PEC sphere: time-domain solution at back-scatter point (0,0,-1.15) using RK2 and MRK2 schemes.

	Side-scatter (1.15,0,0)	Side-scatter (-1.15,0,0)	Forward-scatter (0,0,1.15)	Back-scatter (0,0,-1.15)
RK2	1.0148e-2	1.2075e-2	3.6344e-2	1.1324e-2
MRK2-TW-P2	1.0137e-2	1.2087e-2	3.7161e-2	1.1598e-2
MRK2-TW-OP	9.9520e-3	1.2208e-2	3.7547e-2	1.2027e-2
MRK2-LLH-P2	1.0512e-2	1.1696e-2	3.4975e-2	1.0576e-2
MRK2-LLH-OP	1.0509e-2	1.1709e-2	3.4739e-2	1.0361e-2

Table 7.2: Scattering from PEC sphere: $\max_{t^n} \left| E_x(t^n) - E_x^{Analytic}(t^n) \right|$ at observation points for RK2 and MRK2.

	Side-scatter (1.15,0,0)	Side-scatter (-1.15,0,0)	Forward-scatter (0,0,1.15)	Back-scatter (0,0,-1.15)
RK3	2.7371e-3	2.3150e-3	2.7221e-2	8.9647e-3
MRK3-TW-P2	4.8248e-3	4.7309e-3	3.8499e-2	1.5816e-2
MRK3-TW-OP	4.8441e-3	4.6981e-3	3.8677e-2	1.5721e-2
MRK3-LLH-P2	2.7356e-3	2.3305e-3	2.7559e-2	8.9820e-3
MRK3-LLH-OP	2.7370e-3	2.3172e-3	2.7246e-2	8.9735e-3

Table 7.3: Scattering from PEC sphere: $\max_{n} \left| E_x(t^n) - E_x^{Analytic}(t^n) \right|$ at observation points for RK3 and MRK3.

for observation points are shown in Figures 7.13, 7.14 and 7.15. As it can be seen the pointwise error for MRK3-TW method is noticeably larger near peak values of the solution than for other schemes. This shows that the solution is more polluted by coupling between the multirate groups in the case of MRK3-TW scheme. At the same time, MRK3-LLH scheme gives results with the same errors as a singlerate RK3 scheme. These results agree with the analysis in Chapter 5. The maximum errors at observation points for the third order MRK schemes are shown in Table 7.3. In Figures 7.16, 7.17 and 7.18 we present the errors from both second and third order multirate schemes with optimized partition at the observation points located at a distance of 0.5[m] from PEC surface.

The speedup achieved by both schemes for different order and partition are shown in Table 7.4. One can see that different partition strategies lead to different speedup values. In all cases optimized partition gives better speedup than power of 2 partition. Also in most cases the speedup achieved by MRK-TW scheme is a little better compared to MRK-LLH,

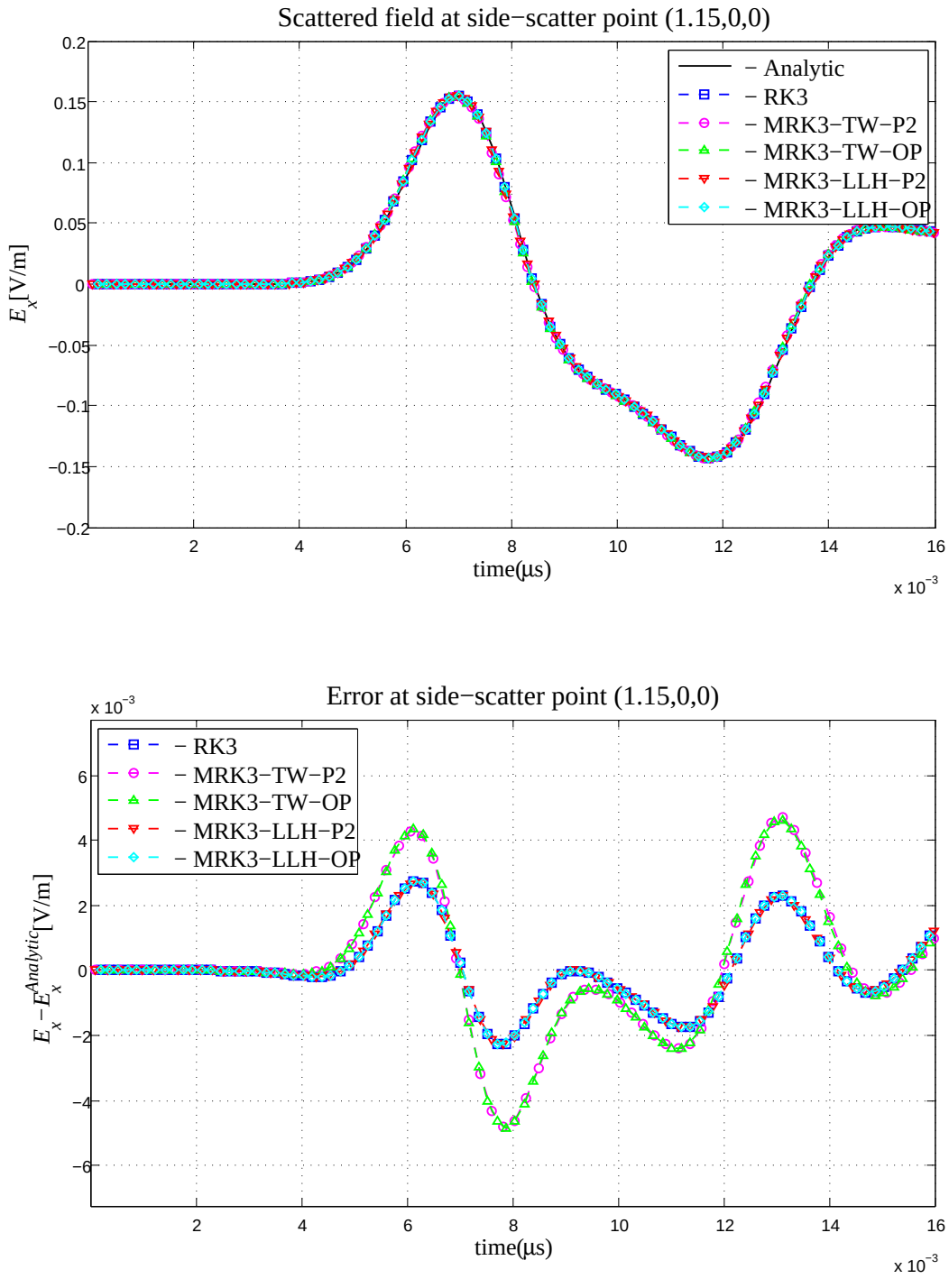
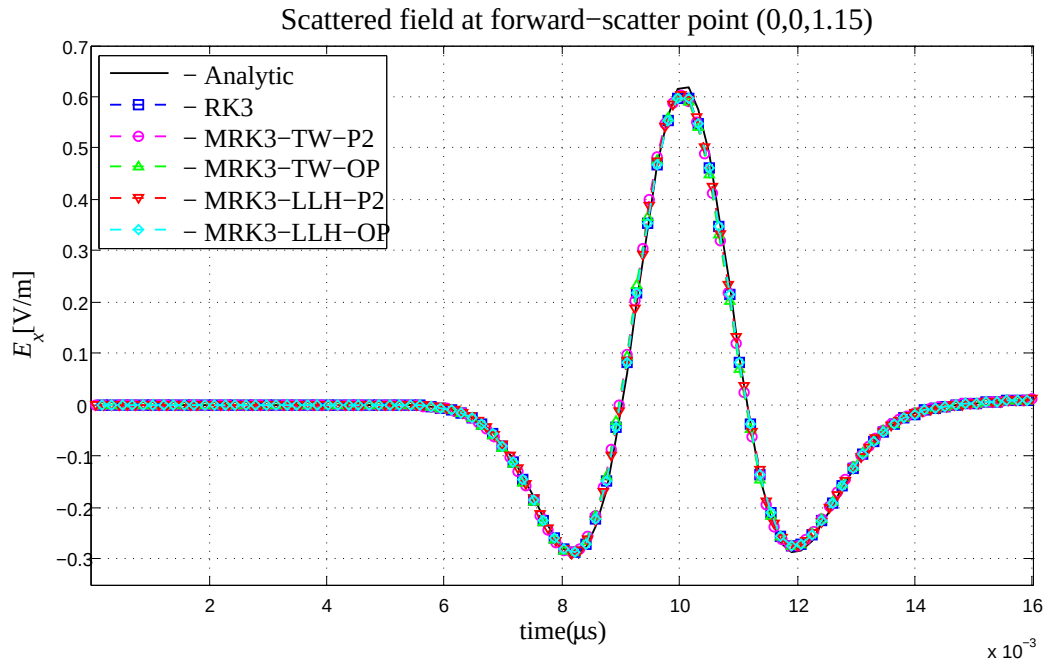


Figure 7.13: Scattering from PEC sphere: time-domain solution at side-scatter point (1.15,0,0) using RK3 and MRK3 schemes.



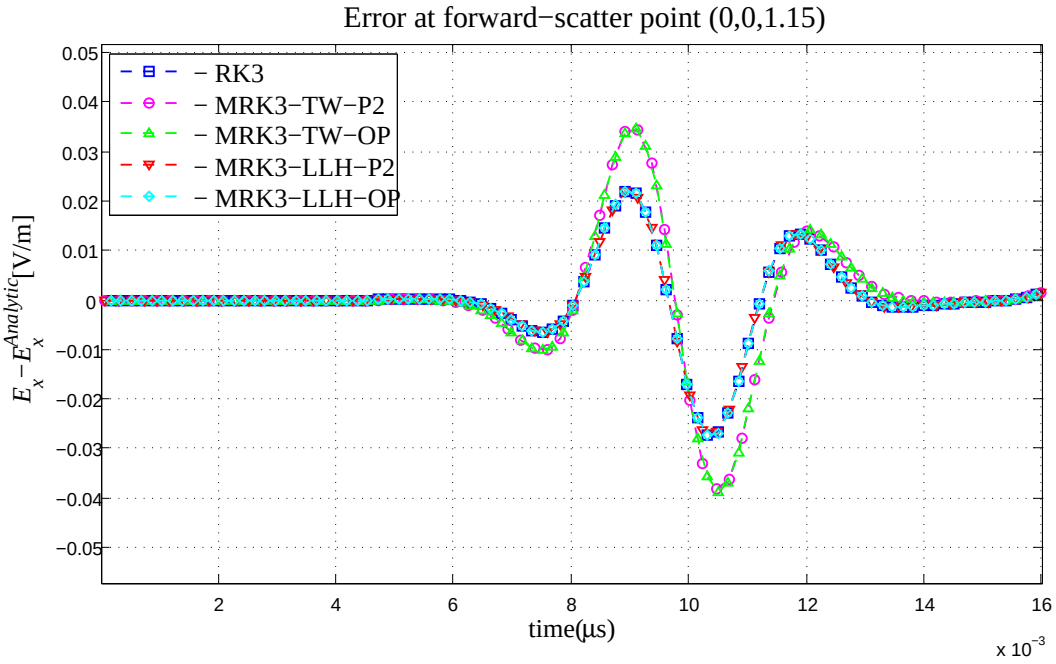


Figure 7.14: Scattering from PEC sphere: time-domain solution at forward-scatter point (0,0,1.15) using RK3 and MRK3 schemes.

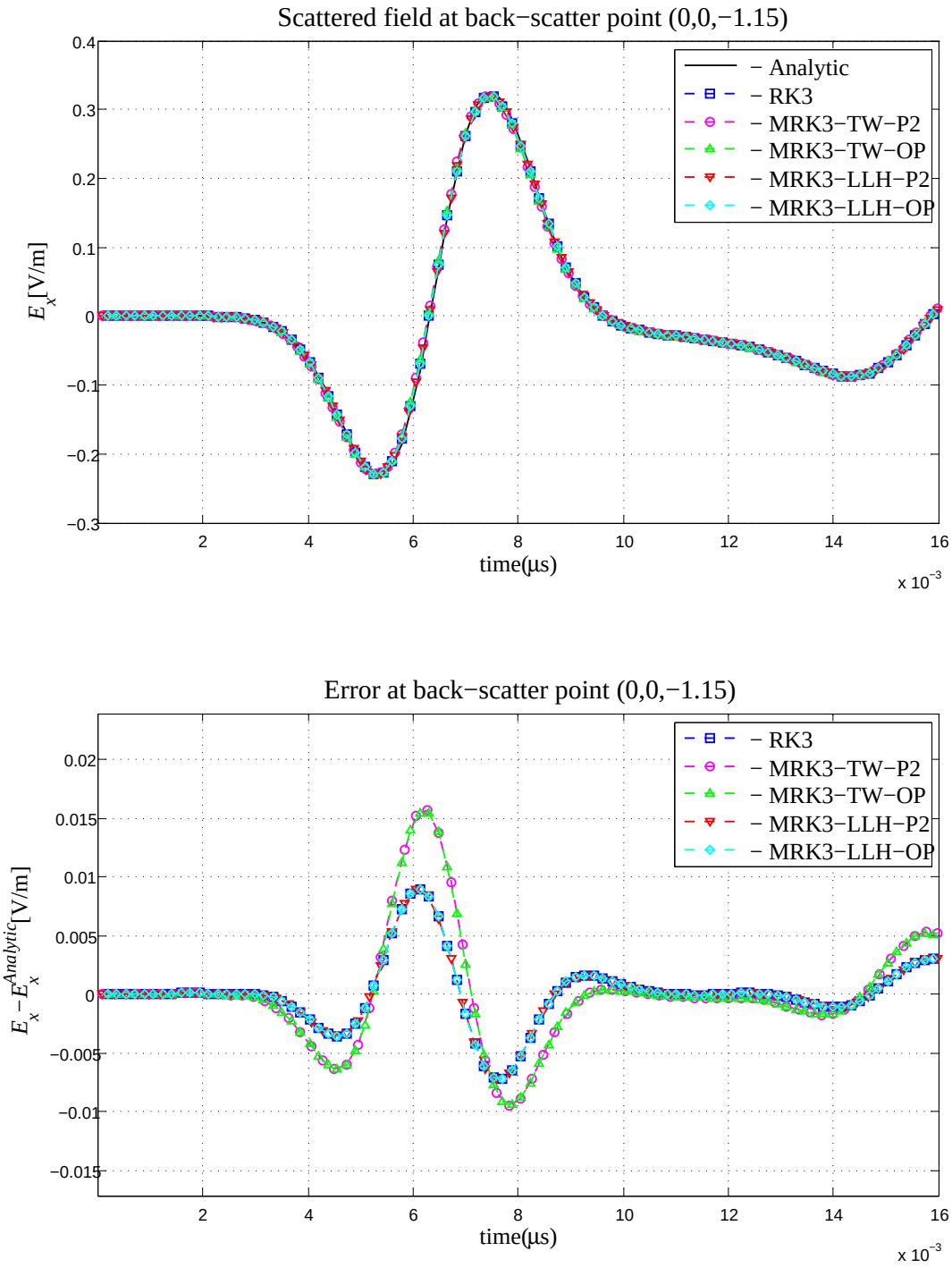


Figure 7.15: Scattering from PEC sphere: time-domain solution at back-scatter point (0,0,-1.15) using RK3 and MRK3 schemes.

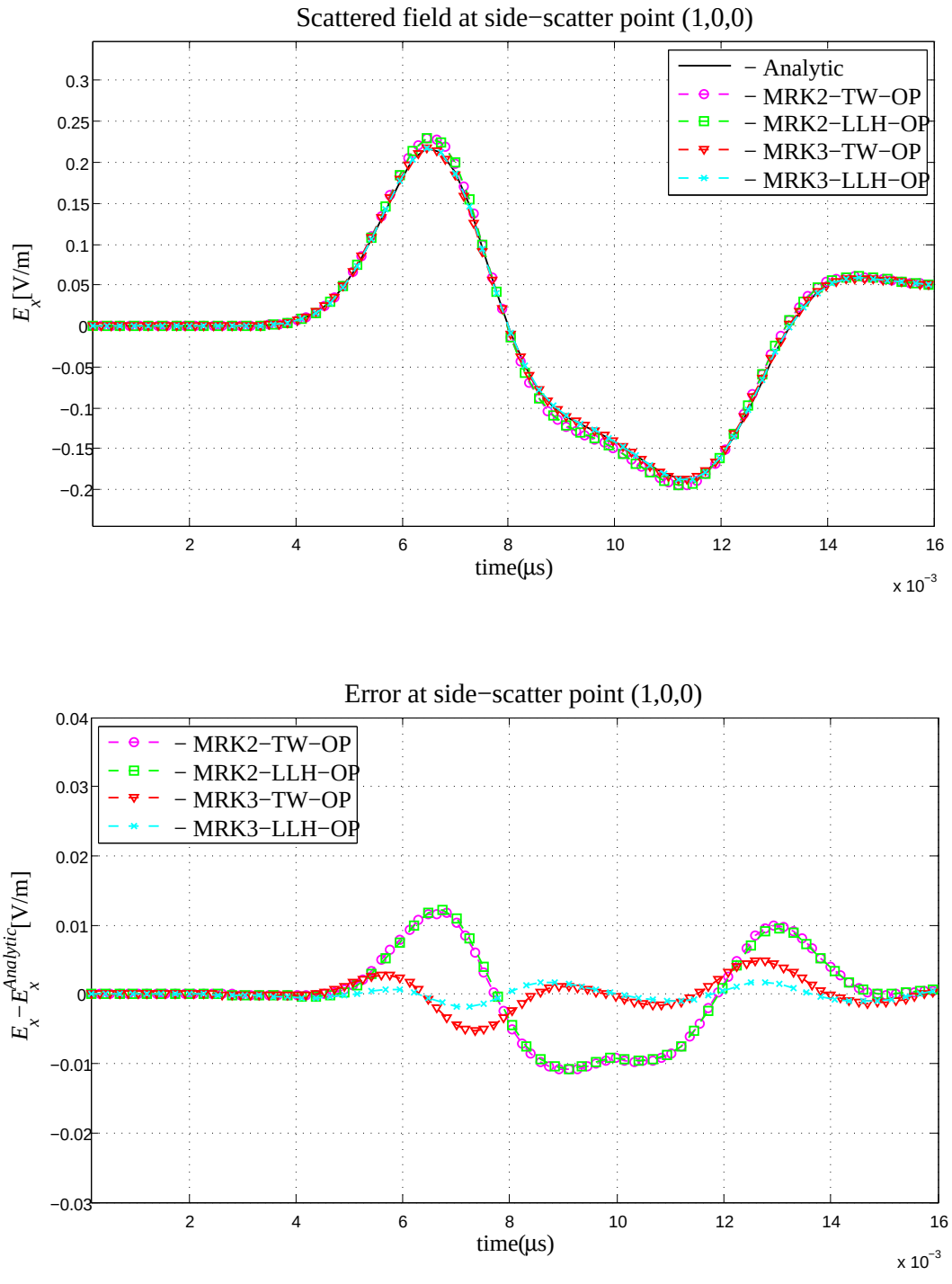


Figure 7.16: Scattering from PEC sphere: time-domain solution at side-scatter point (1,0,0) using MRK2 and MRK3 schemes.

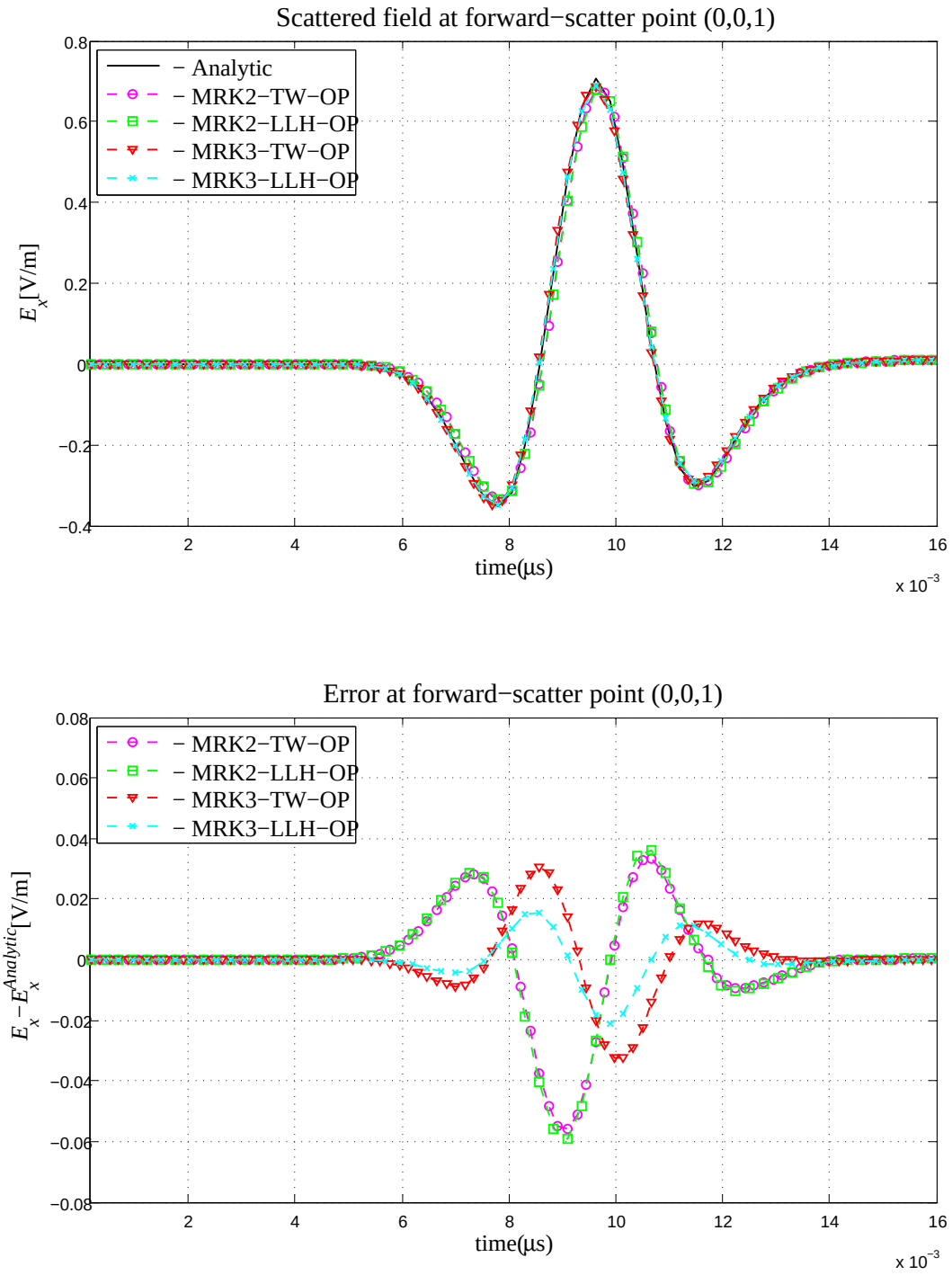


Figure 7.17: Scattering from PEC sphere: time-domain solution at forward-scatter point (0,0,1) using MRK2 and MRK3 schemes.

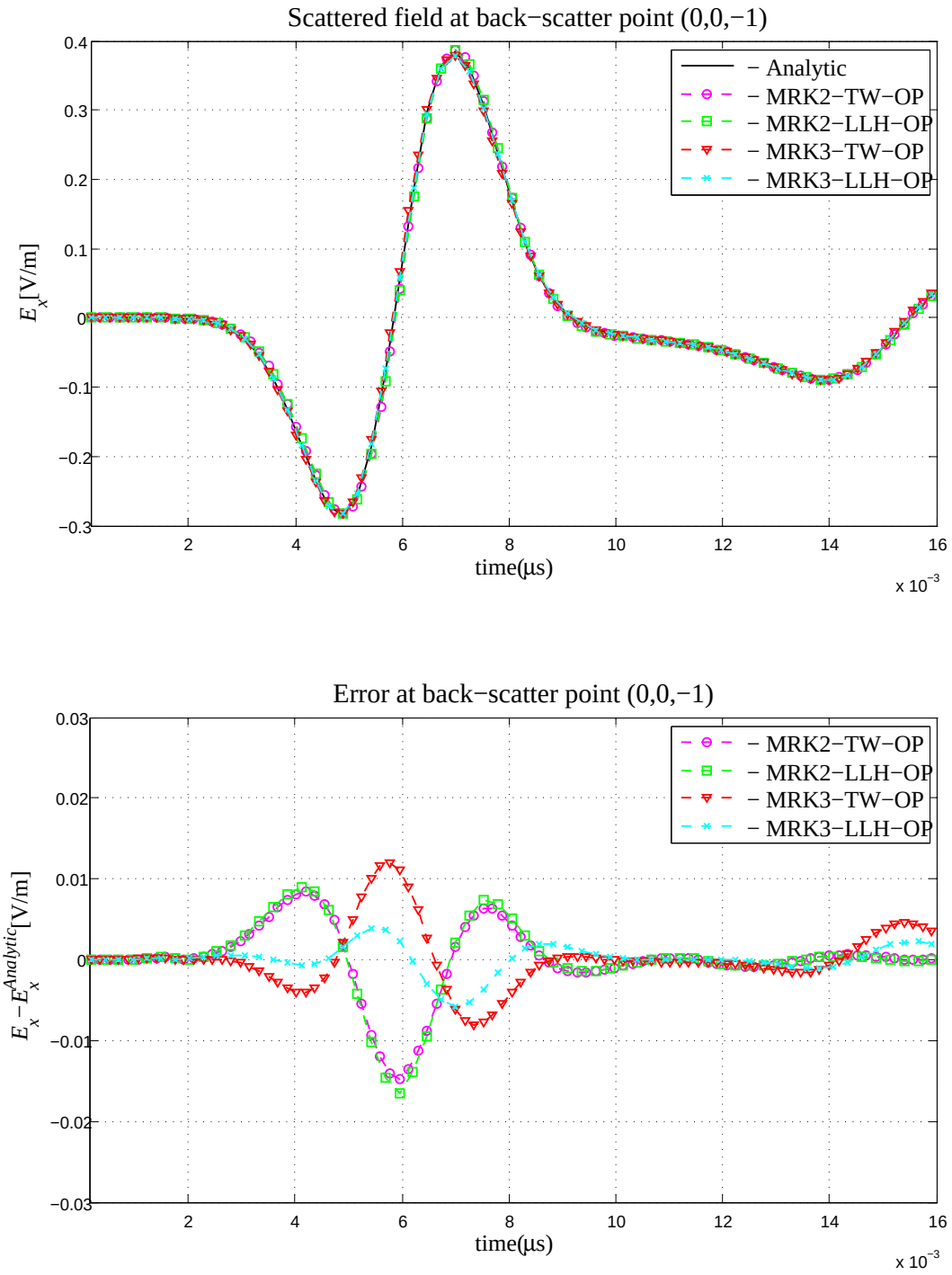


Figure 7.18: Scattering from PEC sphere: time-domain solution at back-scatter point (0,0,-1) using MRK2 and MRK3 schemes.

	theoretical	MRK2-TW	MRK2-LLH	MRK3-TW	MRK3-LLH
P2	4.21	3.34	2.867	3.18	3.11
OP	5.29	3.566	3.144	3.53	3.735

Table 7.4: Scattering from PEC sphere: speedup gained by MRK schemes for the mesh shown on Figure 7.7 with linear cells size ratio 1:6.667.

except for MRK3 with an optimized partition. This can be explained by the extra coupling required for consistency between multirate groups with the same local time. The scheme by Tang-Warnecke does not require coupling in this case. The potential speedup achieved by multirate schemes is further investigated in the next section with a mesh consisting of elements with large linear cell size ratio.

7.2 Parallel-plate waveguide

In this section the problem of a plane wave propagation in a parallel-plate waveguide similar to the one in [44] is considered. Computational domain is represented by a cube with two faces parallel to xy-plane being PEC plates, two faces parallel to zx-plane being PMC plates, and planes $x = -1$ and $x = 1$ representing the waveguide ports. A plane-wave excited on the port with $x = -1$ and propagating in x direction is given by the boundary conditions

$$E_z^{in} = f(t), \tag{7.13}$$

$$H_y^{in} = -f(t)\sqrt{\frac{\varepsilon}{\mu}}, \tag{7.14}$$

$$E_x^{in} = E_y^{in} = H_x^{in} = H_z^{in} = 0, \tag{7.15}$$

where $\varepsilon = \varepsilon_0 \approx 8.854 \cdot 10^{-12} \frac{As}{Vm}$, $\mu = \mu_0 = 4\pi \cdot 10^{-7} \frac{Vs}{Am}$. The geometry of the problem is shown in Figure 7.19.

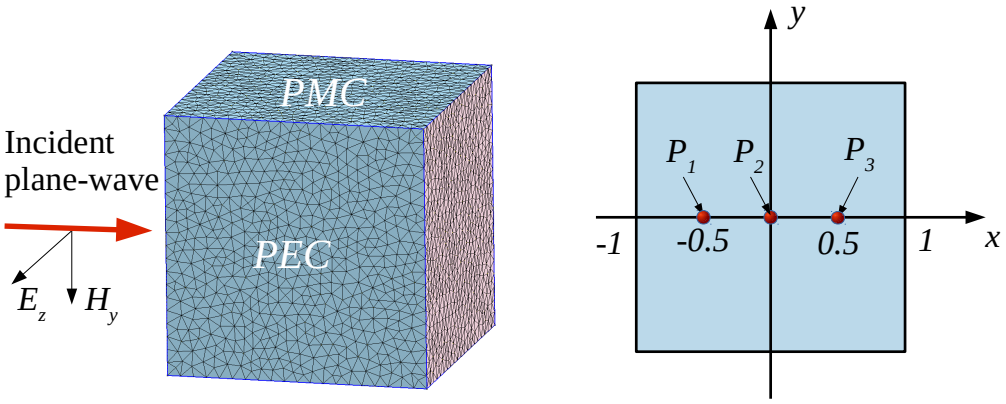


Figure 7.19: Propagation in a parallel-plate waveguide: geometry of the problem.

7.2.1 Propagation in a cube with uniform mesh

The purpose of the first set of experiments is to study how the value of the threshold ζ defined in (7.12) affects the number of non-WENO reconstructions and errors of the solution. For that consider an incoming plane-wave (7.13-7.15) with $f(t)$ given by the Gaussian pulse

$$f(t) = e^{-\frac{(t-t_0)^2}{b^2}}, \tag{7.16}$$

with

$$b = 1.2 \times 10^{-9}\,[s]\,,\; t_0 = c_0^{-1}\,[s]\,, \tag{7.17}$$

where $c_0 = (\mu_0 \varepsilon_0)^{-\frac{1}{2}}$ is the speed of light in vacuum. Experiments are performed on three meshes with relatively uniform linear cell sizes equal to 0.05, 0.1 and 0.2[m] using WENO scheme with $\varepsilon = h^2$. On each mesh the percentage of non-WENO reconstructions for $\zeta = 1,2,3,4$ is determined and the L^1 error given by

$$l_1(\boldsymbol{U}) = \frac{\sum_{i=1}^{N} |T_i| \sum_{j=1}^{3} \left(\sqrt{\varepsilon} \left| \overline{E}_i^j \right| + \sqrt{\mu} \left| \overline{H}_i^j \right| \right)}{\sqrt{\varepsilon_0} \sum_{i=1}^{N} |T_i|} \tag{7.18}$$

is computed at time $T = 2c_0^{-1}$. The results of computations are presented in Tables 7.5 and 7.6. The percentage of non-WENO reconstructions for $4 \leq \zeta < 10$ is less than 1 and varies

# of cells	% of non-WENO reconstructions			
	$\zeta = 1$	$\zeta = 2$	$\zeta = 3$	$\zeta = 4$
8138	6.84	2.25	1.28	0.89
64530	6.71	2.08	1.17	0.80
550576	6.62	2.02	1.14	0.78

Table 7.5: Propagation in a parallel-plate waveguide: the number of cells with non-WENO reconstruction for various values of the threshold ζ and mesh sizes.

# of cells	L^1 error at $T = 2c_0^{-1}$			
	$\zeta = 1$	$\zeta = 2$	$\zeta = 3$	$\zeta = 4$
8138	2.183629e-2	2.191221e-2	2.202186e-2	2.245129e-2
64530	6.648114e-3	6.713686e-3	6.858187e-3	7.035359e-3
550576	1.671580e-3	1.682986e-3	1.705448e-3	1.758959e-3

Table 7.6: Propagation in a parallel-plate waveguide: L^1 error for various values of the threshold ζ and mesh sizes, solution obtained with $\varepsilon = h^2$.

insignificantly, since the are only a few linear weights in that range. But the presence of this small number of bad linear weights creates instability for the same time-step criteria, for which the scheme is stable with $\zeta < 4$. The L^1 error also grows slightly with ζ. It can be noticed that the percentage of non-WENO reconstructions is somewhat smaller for finer meshes. From the experiments it can be concluded that for optimal performance the value of ζ should be in the range between 2 and 3.

In Table 7.7 the time and storage of the third order polynomial and WENO schemes are compared. The results show that WENO schemes requires roughly six times more memory space compared to the polynomial scheme, since it stores the information on small stencils in addition to the big stencil. The CPU time per time-step for WENO scheme is approximately 20-30 times larger than for the polynomial scheme. Therefore from a practical point of view WENO scheme should only be used when its non-oscillatory properties benefit the solution. One way to reduce computational cost is to switch between polynomial and WENO reconstruction depending on the values of smoothness indicators. This has not been studied rigorously, but we found that a naive criteria, such as $\max_l SI_l > \frac{\varepsilon}{2}$ for WENO reconstruction reduces the time by a factor between 2 and 3 without compromising either smooth or discontinuous solution.

# of cells	Storage [GB]		$CPU_{\Delta t}$[s]		
	Polyn.	WENO	Polyn.	WENO	WENO, if $\max_l SI_l > \frac{\varepsilon}{2}$
8138	0.048	0.157	0.0243	0.4478	0.1858
64530	0.245	1.2	0.1765	4.2390	1.5288
550576	1.8	11.0	1.4706	40.411	14.032

Table 7.7: Propagation in a parallel-plate waveguide: comparison of storage and CPU time per time-step requirements for the 3rd order polynomial and WENO schemes.

# of cells	WENO			Linear	MUSCL
	$\varepsilon = h$	$\varepsilon = h^2$	$\varepsilon = h^4$		
8138	2.237577e-2	2.191221e-2	2.373869e-2	2.235277e-2	3.032525e-2
64530	6.336795e-3	6.713686e-3	8.238667e-3	6.072867e-3	1.578373e-2
550576	1.570713e-3	1.682986e-3	2.353663e-3	1.450966e-3	7.906960e-3

Table 7.8: Propagation in a parallel-plate waveguide: L^1 errors at $T = 2c_0^{-1}$ for WENO with $\varepsilon = h$,h^2,h^4, linear and MUSCL schemes.

In Table 7.8 the L^1 errors are shown for WENO scheme with various ε together with linear and MUSCL schemes. While errors decrease slower for smaller ε, they are still comparable with the ones for linear scheme and are smaller than for MUSCL scheme. Time-domain solutions by WENO with various choices of ε at three observation points are shown in Figures 7.20, 7.21 and 7.22, and enlarged views of peak values are presented in Figure 7.23. As in one-dimensional case the best resolution of peaks is achieved with $\varepsilon = h$ for which the non-linear weights are the closest to linear, while $\varepsilon = h^4$ significantly distorts the solution near critical points.

Next several experiments with a discontinuous signal traveling in the x direction are presented. The purpose of this example is to demonstrate the advantage of WENO scheme over linear scheme for discontinuous solutions and to validate the theoretical results for ε on three-dimensional simulations. Consider the incoming field on the waveguide port (7.13-7.15) given by

$$f(t) = H(t - t_s) H(t_e - t),$$

where $H(t)$ is the Heaviside step function and

$$t_s = \frac{1}{4} c_0^{-1} [s], \quad t_e = \frac{7}{4} c_0^{-1} [s].$$

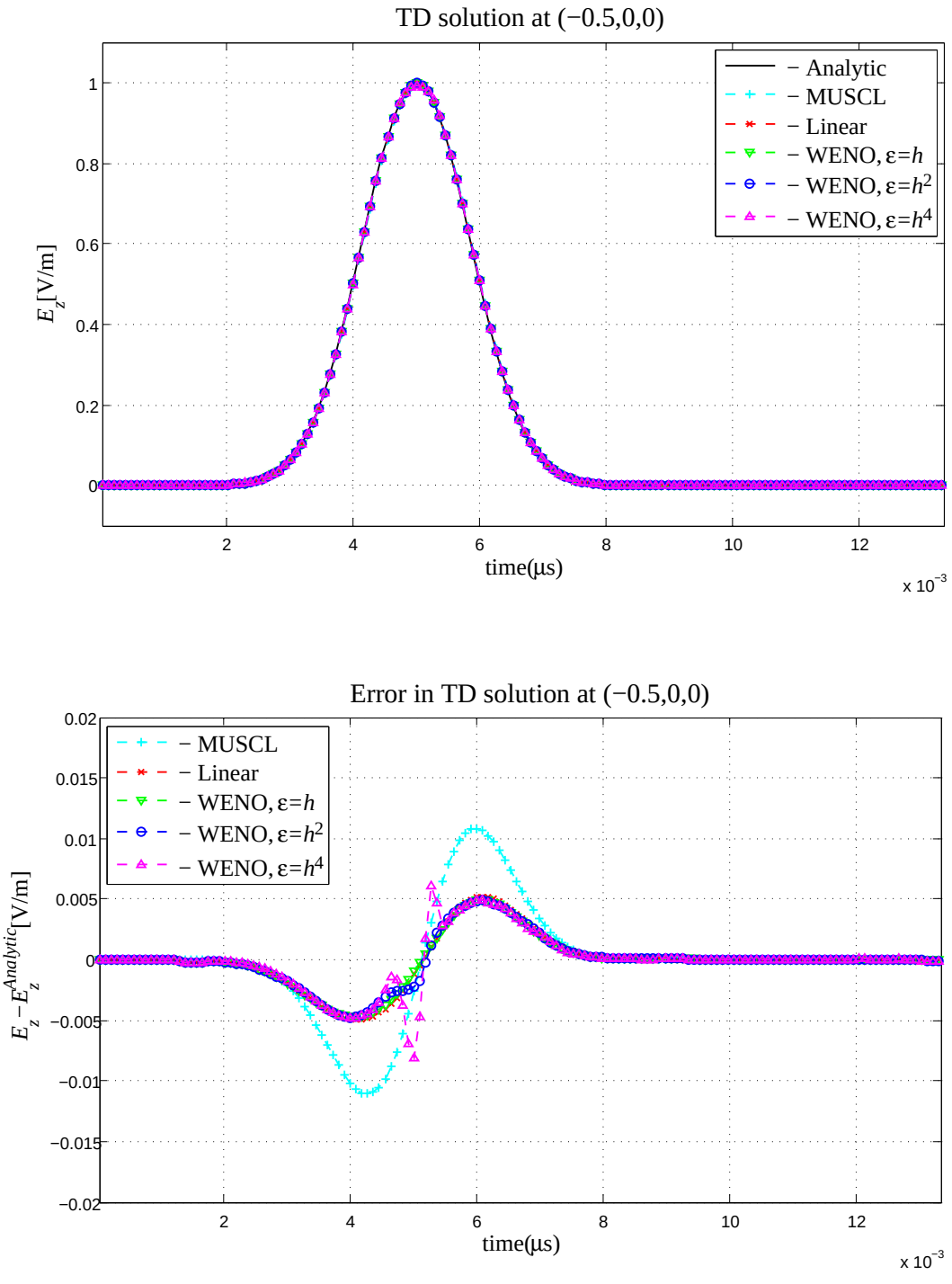


Figure 7.20: Propagation in a parallel-plate waveguide: time-domain solution and errors in time for the propagation of Gaussian pulse at the observation point $P_1 = (-0.5, 0, 0)$.

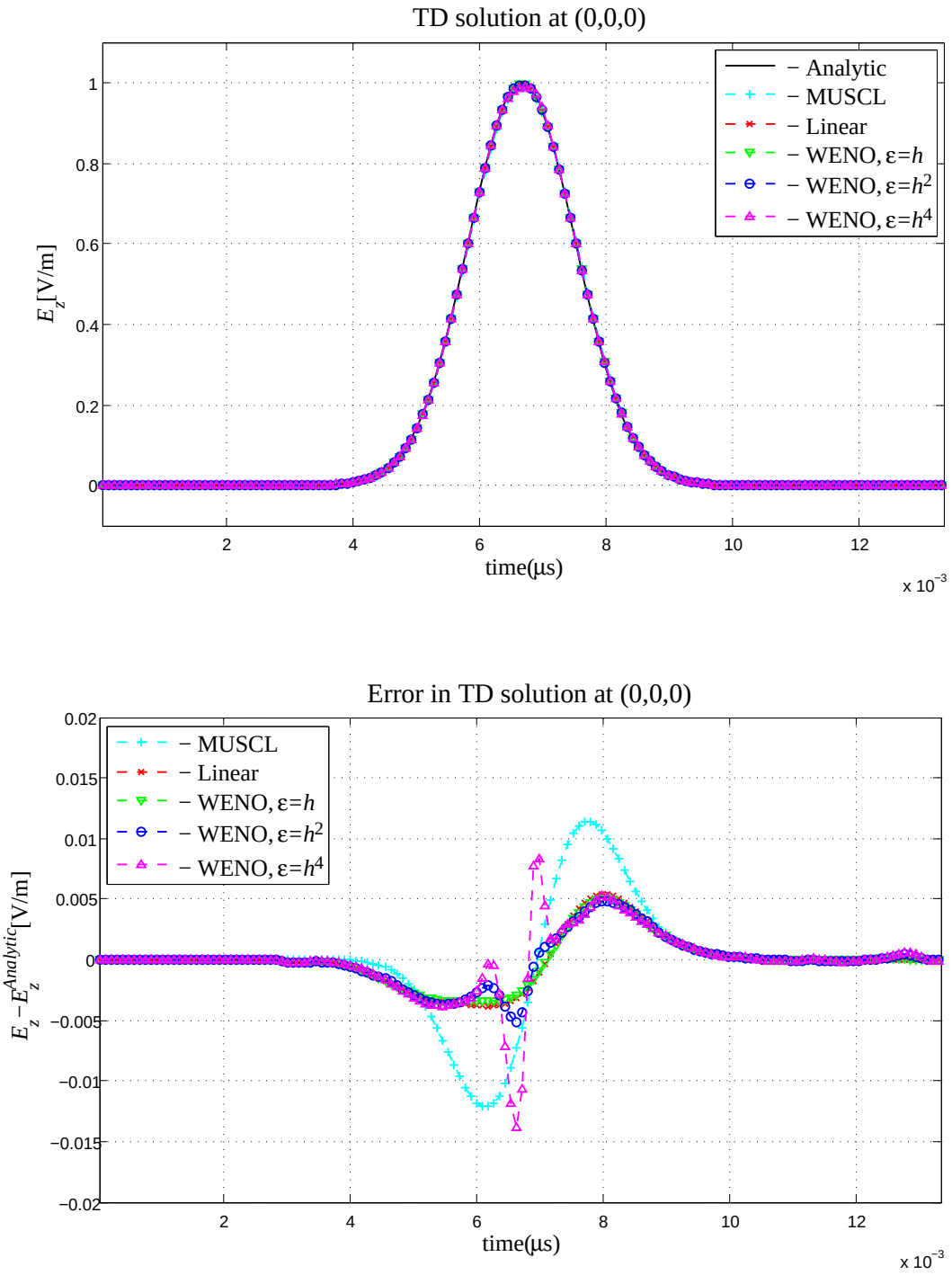


Figure 7.21: Propagation in a parallel-plate waveguide: time-domain solution and errors in time for the propagation of Gaussian pulse at the observation point $P_2 = (0,0,0)$.

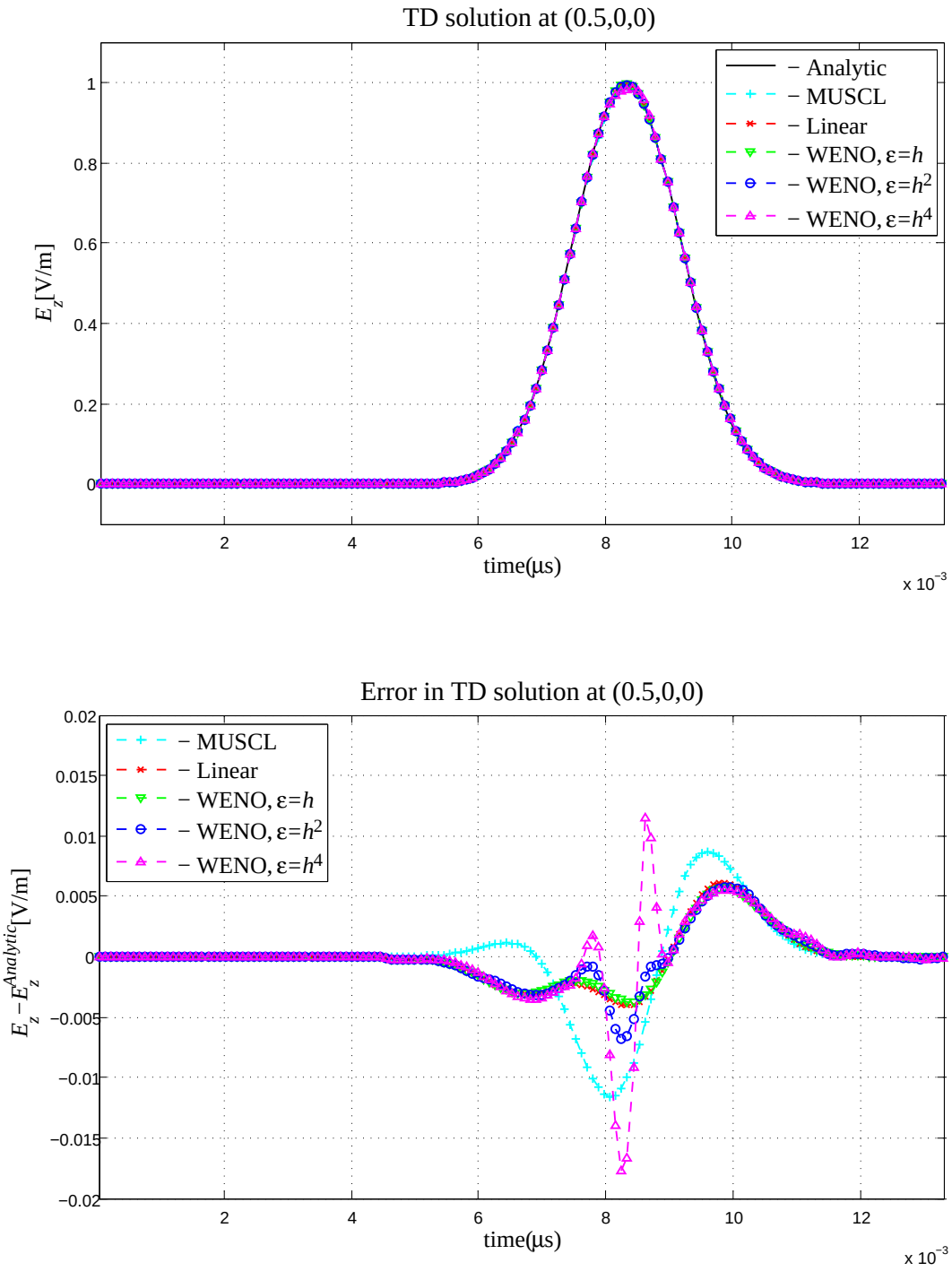


Figure 7.22: Propagation in a parallel-plate waveguide: time-domain solution and errors in time for the propagation of Gaussian pulse at the observation point $P_3 = (0.5, 0, 0)$.

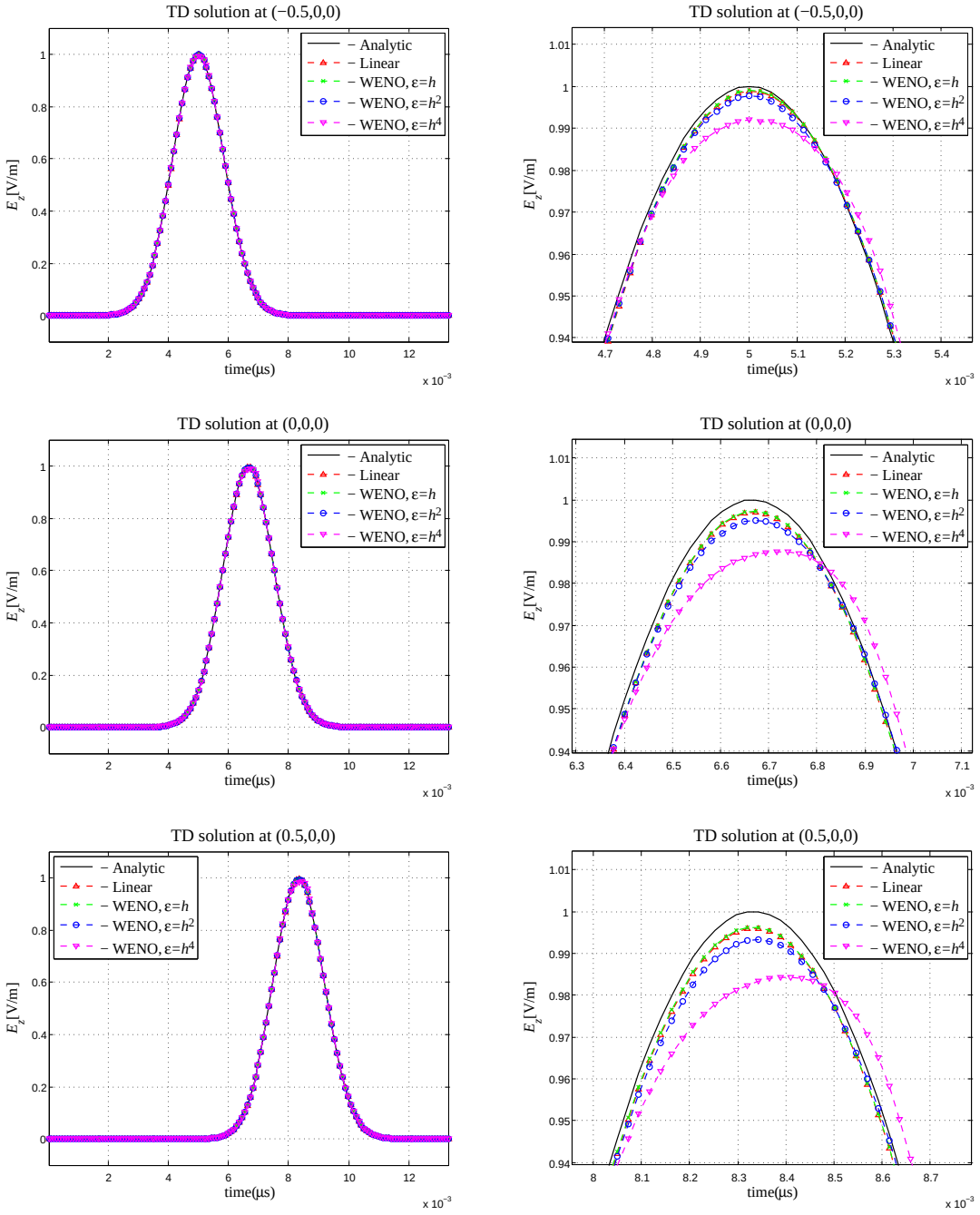


Figure 7.23: Propagation in a parallel-plate waveguide: enlarged view of the time-domain solution near critical point for the propagation of Gaussian pulse at observation points.

For the experiments we use the finest mesh from the previous example. The results of simulations using third order linear and WENO schemes at the observation points are presented in Figures 7.24, 7.25 and 7.26. As in the one-dimensional case essentially non-oscillatory results are achieved with $\varepsilon \leq h^2$. There is a significant pollution of the solution in the smooth region obtained by linear scheme. At the same time, only slight oscillations are present when $\varepsilon = h^2, h^4$ and they do not propagate into the smooth region as much as for linear scheme or WENO with $\varepsilon = h$.

7.2.2 Plane-wave propagation in an extremely inhomogeneous mesh

This experiment is similar to the example presented in [44]. A plane-wave given by the Gaussian pulse (7.17-7.16) propagates in a cubic computational domain with large inhomogeneity in cell sizes. The purpose of this experiment is to examine potential speedup achieved by multirate schemes presented in this work. As in [44] an extremely large inhomogeneity in cell sizes is artificially produced with help of two spherical surfaces of radii R_1 and R_2 that define three domains in a cube (see Figure 7.27). The first domain is defined by a sphere of radius $R_1 = 1/500$ with the center at the origin and partitioned into elements of linear size $h_1 = R_1/1.6$. The second domain is defined by the region enclosed between the spheres of radii R_1 and $R_2 = 1/5$ with the mesh size varying from h_1 on the small sphere to $h_2 = 1/10$ on the large sphere. The third domain is defined by the region between the large sphere and the cube boundary and partitioned into cells of relatively uniform size $1/10$. As a result the linear cells size ratio in the mesh is 1:80 with the total number of cells equal to 80183 and less than 200 of elements with minimal size. The schematic representation of the geometry and the resulting mesh are shown in Figure 7.27.

The computational domain is partitioned into 8 multirate groups with the maximum time-step ratio $1 : 87$ and 0.316 percent of elements in the multirate group with the smallest time-step. Theoretical speedup for this partition is estimated to be 60.34. Numerical speedup achieved by MRK3-TW and MRK3-LLH schemes for optimal multirate is presented in Table 7.9. The results demonstrate the ability to achieve much greater speedup than in [44] due to flexibility of MRK-TW and MRK-LLH schemes in defining the multirate groups. In this particular example we find that the achieved numerical speedup is

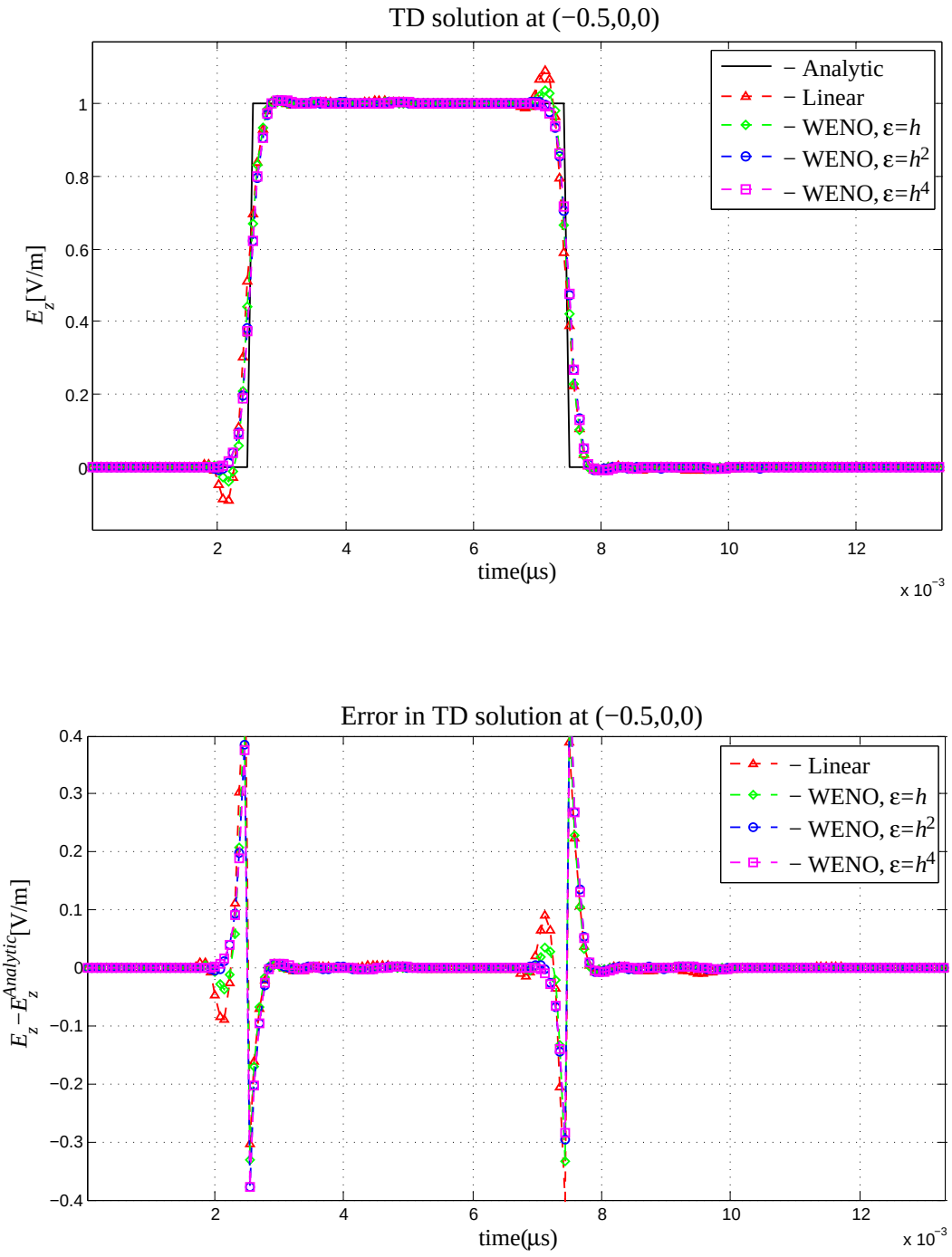


Figure 7.24: Propagation in a parallel-plate waveguide: time-domain solution and errors in time for the propagation of discontinuous pulse at the observation point $P_1 = (-0.5, 0, 0)$.

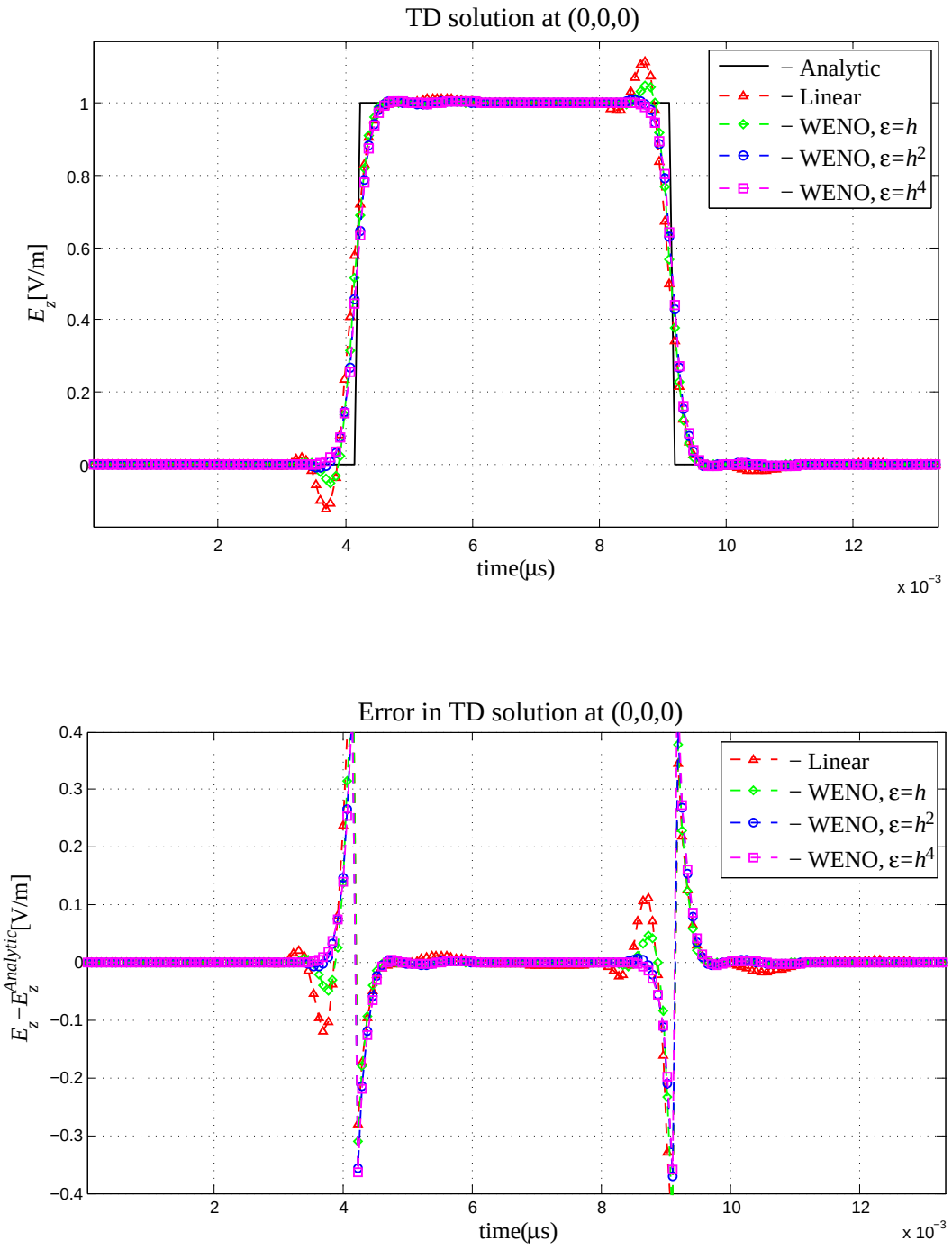


Figure 7.25: Propagation in a parallel-plate waveguide: time-domain solution and errors in time for the propagation of discontinuous pulse at the observation point $P_2 = (0, 0, 0)$.

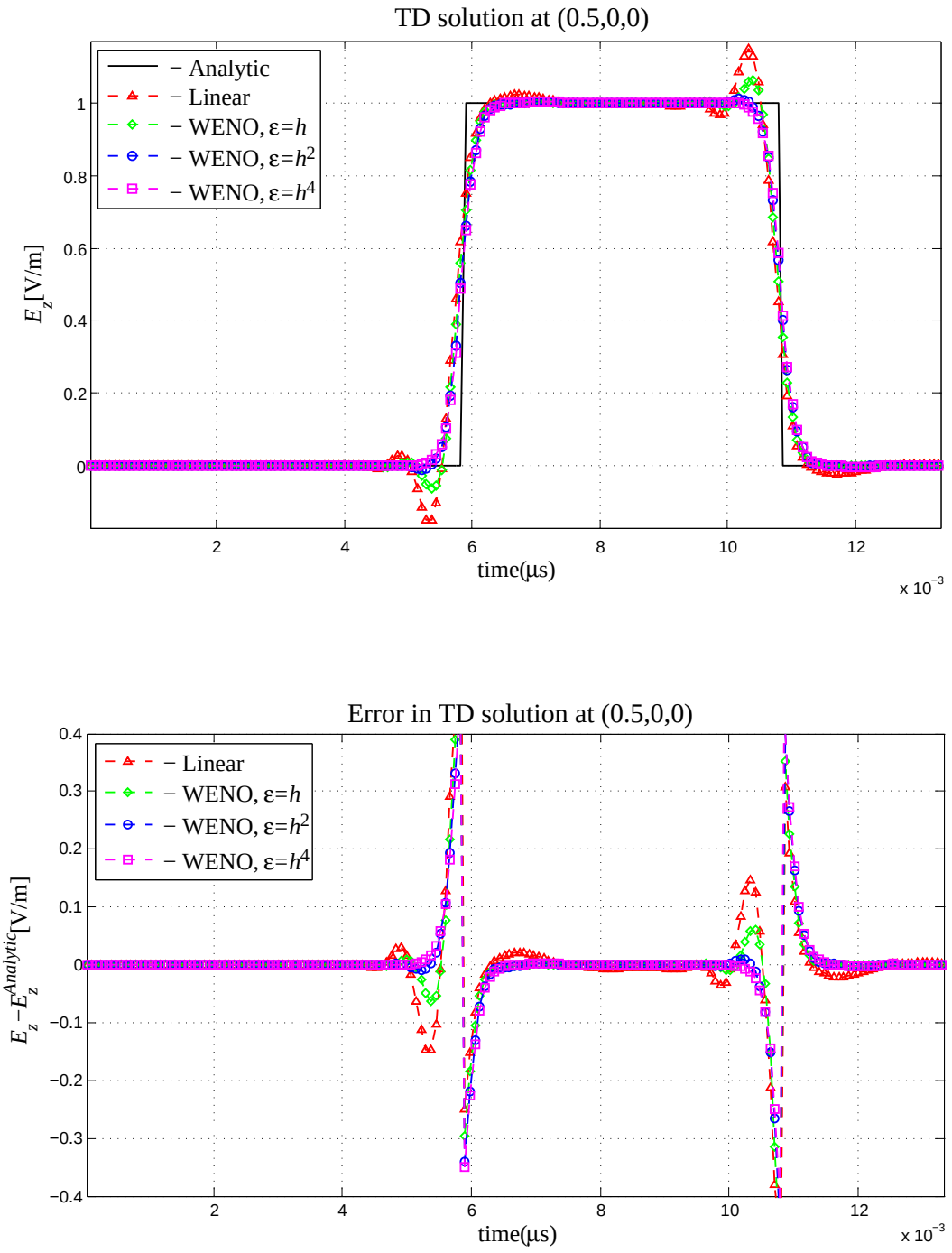


Figure 7.26: Propagation in a parallel-plate waveguide: time-domain solution and errors in time for the propagation of discontinuous pulse at the observation point $P_3 = (0.5, 0, 0)$.

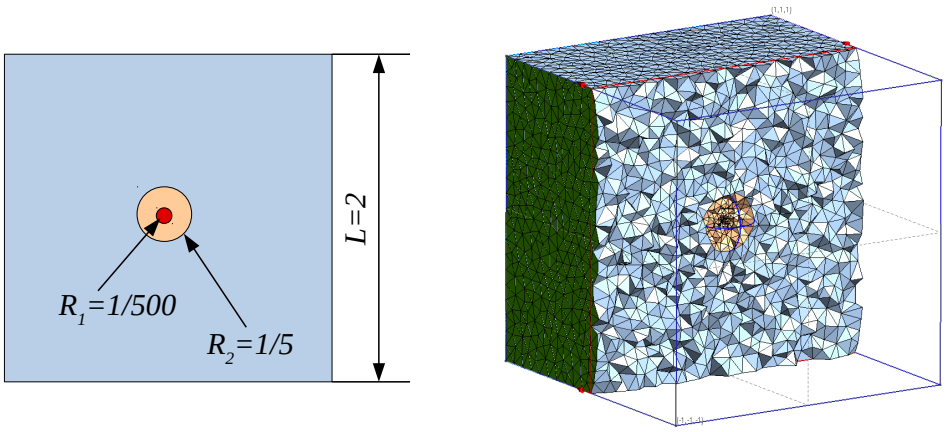


Figure 7.27: Plane-wave propagation in an extremely inhomogeneous mesh: mesh with cell size ratio 1:80.

	speedup	N_{Lu}
MRK3-TW	37.22	1796646
MRK3-LLH	35.38	1629000

Table 7.9: Plane-wave propagation in an extremely inhomogeneous mesh: speedup factors and number of flux evaluations per global time-step N_{Lu} by the third order MRK time-stepping for a plane-wave propagation.

slightly better for MRK-TW scheme, but this depends on a particular mesh and partition. The number of flux evaluations per one global time-step is also shown in Table 7.9. As expected, this number is smaller for MRK3-LLH, but since the speedup is also affected by the time required for the adjustment of stage values in LLH scheme, the scheme is sometimes slightly slower than MRK3-TW.

To see how the error of the solution is affected by extreme mesh we compare numerical solutions obtained by RK3, MRK3-TW and MRK3-LLH schemes. The results of computations at three observation points located on the x axis at $x = -0.3$ (before the inhomogeneous region), $x = 0$ (inside the finest mesh), and $x = 0.3$ (after the inhomogeneous region) are shown in Figures 7.28, 7.29 and 7.30 and maximum errors at these points are presented in Table 7.10. One can see that MRK3-LLH scheme achieves the smallest maximum errors at all observation points even compared to the singlerate scheme. The maximum error grows as the signal travels from the coarsest region to the finest and back to the coarsest

	$\max_{t^n} \lvert E_z(t^n) - E_z^{Analytic}(t^n) \rvert$ at			L^1 error	L^∞ error
	$(-0.3,0,0)$	$(0,0,0)$	$(0.3,0,0)$		
RK3	9.4945e-3	1.1960e-2	1.4687e-2	9.0780e-3	1.8921e-2
MRK3-TW	2.2795e-2	2.3302e-2	3.4321e-2	1.2764e-2	2.9345e-2
MRK3-LLH	6.9558e-3	5.9577e-3	6.8932e-3	7.4495e-3	1.2258e-2

Table 7.10: Plane-wave propagation in an extremely inhomogeneous mesh: pointwise, L^1 and L^∞ errors at $T = 2c_0^{-1}$ of the solution by RK3 and MRK3 schemes.

for RK3 and MRK3-TW schemes, while for MRK3-LLH scheme the error stays within the same limits. The error for RK3 and MRK3-TW increases by roughly a factor of 1.5. L^1 and L^∞ errors at the time when the peak is in the finest region is also the smallest for MRK3-LLH scheme.

7.3 Plane-wave reflection/transmission at a dielectric interface

Since electromagnetic problems often consider waves propagating through inhomogeneous media, the example in this section involves a domain consisting of free space and a dielectric with isotropic, linear non-dispersive properties. The part of the domain where $x < 0$ represents a free space ($\varepsilon_1 = \varepsilon_0$, $\mu_1 = \mu_0$) and the region where $x > 0$ represents a dielectric medium with $\varepsilon_2 = \varepsilon_r \varepsilon_0$, $\mu_2 = \mu_0$. The incident, reflected and transmitted waves travel in the direction normal to the dielectric boundary. The problem geometry is shown in Figure 7.31.

The incoming plane-wave (7.13-7.15) excited on the free space side at $x = -1$ is given by the Gaussian pulse (7.16-7.17).

7.3.1 Analytic solution in frequency domain

Solutions for transmitted and reflected waves at a dielectric interface can be found in many books on electromagnetics, see for example Balanis [13]. Given the incident plane-wave

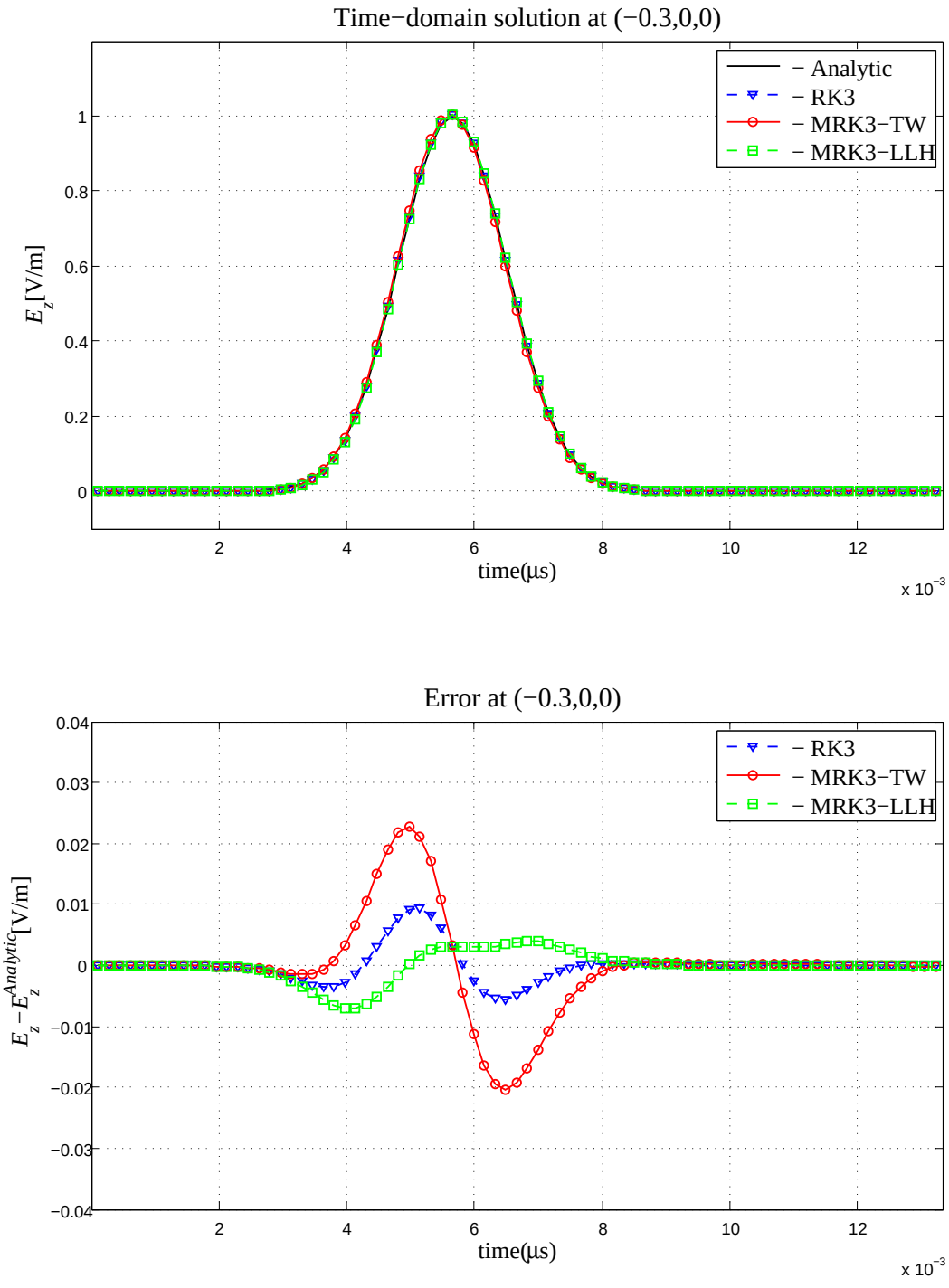


Figure 7.28: Plane-wave propagation in an extremely inhomogeneous mesh: time-domain solution and error at the observation point $P_1 = (-0.3, 0, 0)$.

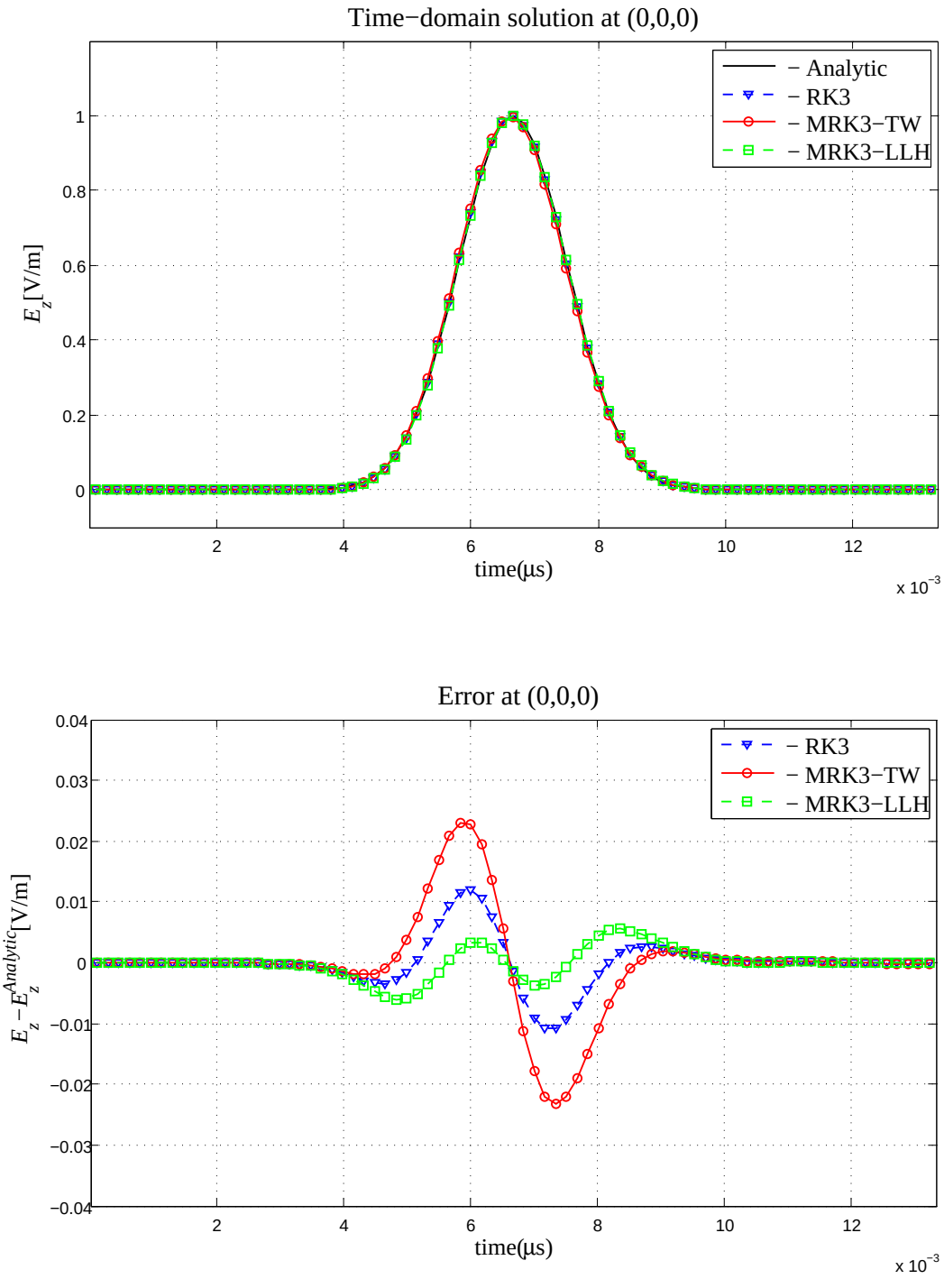


Figure 7.29: Plane-wave propagation in an extremely inhomogeneous mesh: time-domain solution and error at the observation point $P_2 = (0,0,0)$.

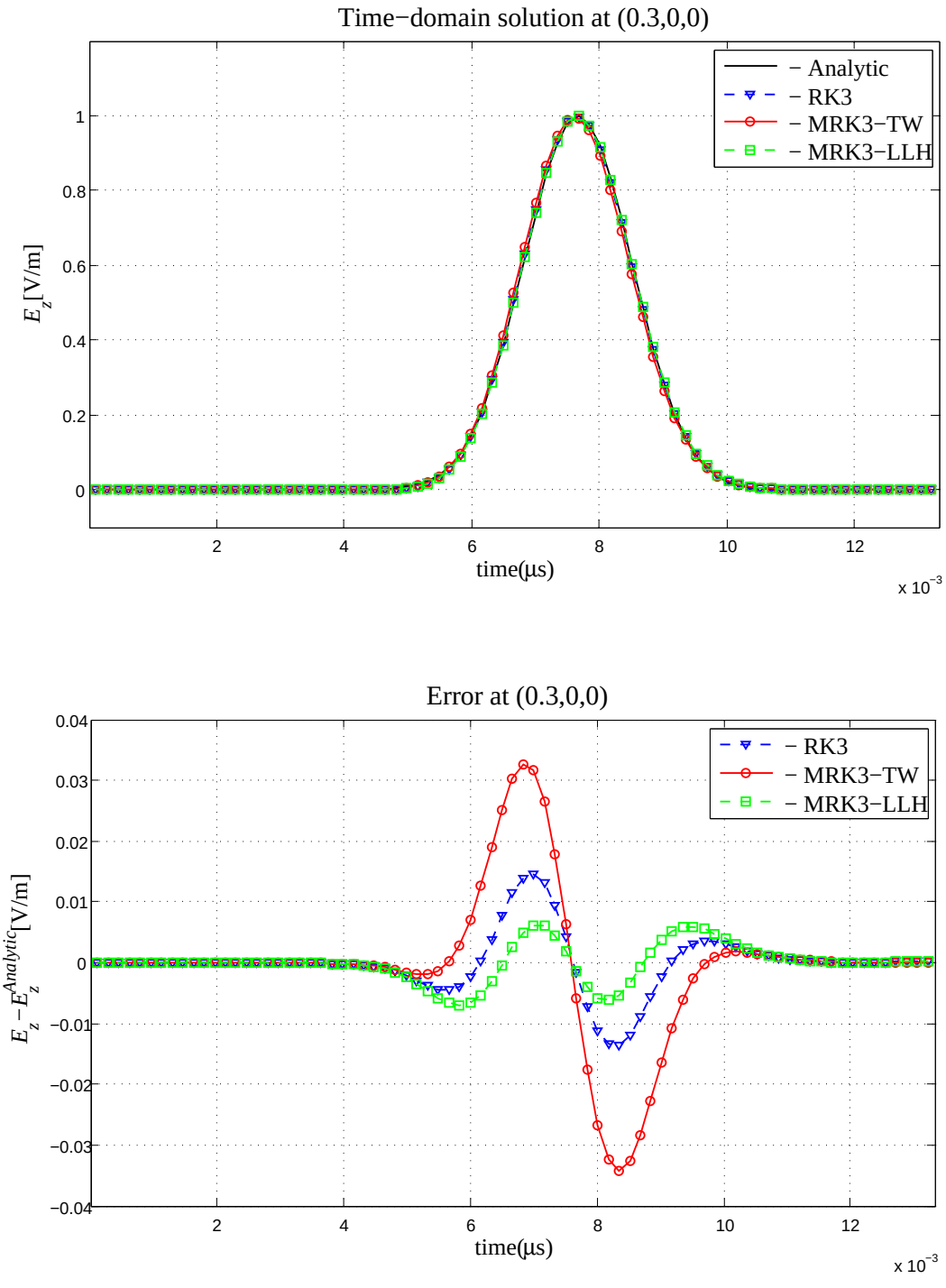


Figure 7.30: Plane-wave propagation in an extremely inhomogeneous mesh: time-domain solution and error at the observation point $P_3 = (0.3, 0, 0)$.

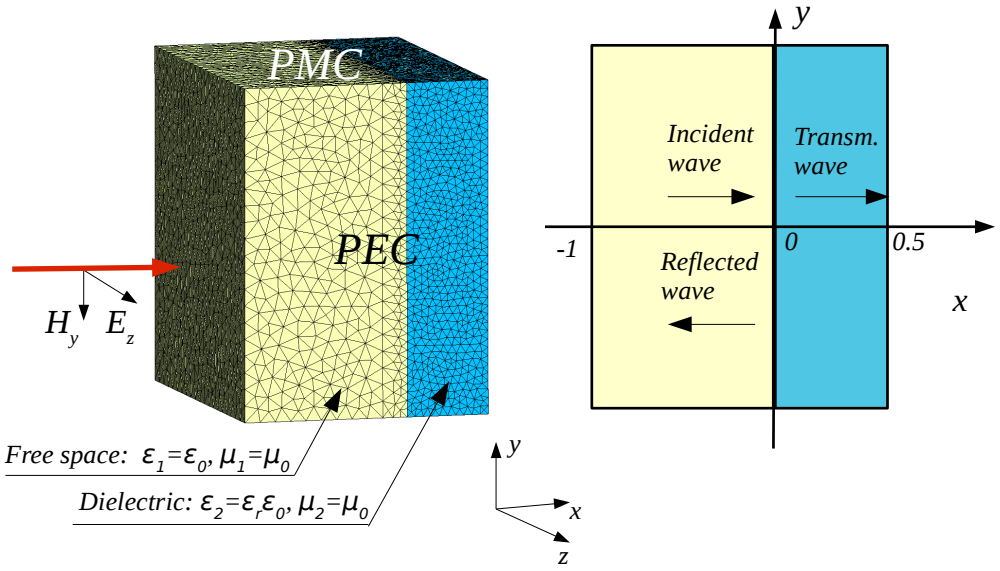


Figure 7.31: Plane-wave reflection/transmission at a dielectric interface: problem geometry and mesh.

propagating in x direction

$$\mathscr{E}_z^{in} = E_0 e^{-i\beta_1 x} = E_0 e^{-i\beta_0 x}, \tag{7.19}$$

$$\mathscr{H}_y^{in} = -\frac{E_0}{Z_1} e^{-i\beta_0 x} = -\frac{E_0}{Z_0} e^{-i\beta_0 x}, \tag{7.20}$$

$$\mathscr{E}_x^{in} = \mathscr{E}_y^{in} = \mathscr{H}_x^{in} = \mathscr{H}_z^{in} = 0, \tag{7.21}$$

where $\beta_0 = \omega\sqrt{\varepsilon_0\mu_0} = \frac{\omega}{c}$ is the wave number, and $Z_1 = Z_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}}$ is the intrinsic impedance of free space. The solution for the fields transmitted through the dielectric interface is given by

$$\mathscr{E}_z^{tr} = \tau E_0 e^{-i\beta_2 x} = \tau E_0 e^{-i\sqrt{\varepsilon_r}\beta_0 x}, \tag{7.22}$$

$$\mathscr{H}_y^{tr} = -\tau\frac{E_0}{Z_2} e^{-i\beta_2 x} = -\sqrt{\varepsilon_r}\tau\frac{E_0}{Z_0} e^{-i\sqrt{\varepsilon_r}\beta_0 x}, \tag{7.23}$$

$$\mathscr{E}_x^{tr} = \mathscr{E}_y^{tr} = \mathscr{H}_x^{tr} = \mathscr{H}_z^{tr} = 0, \tag{7.24}$$

where $Z_2 = \sqrt{\frac{\mu_2}{\varepsilon_2}} = \sqrt{\frac{\mu_0}{\varepsilon_r \varepsilon_0}}$, and τ is the transmission coefficient $\tau = \frac{2Z_2}{Z_2+Z_1} = \frac{2/\sqrt{\varepsilon_r}}{1/\sqrt{\varepsilon_r}+1}$, and for the reflected wave fields it is

$$\mathscr{E}_z^{refl} = \Gamma E_0 e^{\beta_1 x} = \Gamma E_0 e^{\beta_0 x}, \tag{7.25}$$

$$\mathscr{H}_y^{refl} = \Gamma \frac{E_0}{Z_0} e^{\beta_1 x} = \Gamma \frac{E_0}{Z_0} e^{\beta_0 x}, \tag{7.26}$$

$$\mathscr{E}_x^{refl} = \mathscr{E}_y^{refl} = \mathscr{H}_x^{refl} = \mathscr{H}_z^{refl} = 0, \tag{7.27}$$

where $\Gamma = \frac{Z_2-Z_1}{Z_2+Z_1} = \frac{1/\sqrt{\varepsilon_r}-1}{1/\sqrt{\varepsilon_r}+1}$ is the reflection coefficient.

7.3.2 Numerical solution

Consider dielectric material with $\varepsilon_r = 4$ and two meshes with the linear cell size ratio in free space and dielectric equal to 2 (Figure 7.31). The results for the time-domain solutions at three observation points located in free space, at a dielectric boundary and inside dielectric material are shown in Figures 7.32, 7.33 and 7.34. In Table 7.11 L^1 errors of the solution at two different times corresponding to the peak value at a dielectric interface and inside the dielectric are presented. In the presence of dielectric boundary there is a discontinuity in the parameters of Maxwell's equations, therefore the solution has a discontinuity at the interface between the two media in the first derivative with respect to the space variables in the direction of propagation x. In the one-dimensional case the polynomial scheme produces size $O(h)$ oscillations in the solution for this type of singularity on uniform meshes, and these oscillations are more pronounced for large contrasts in the coefficients. But if the cell size ratio in regions with different coefficients is equal to the ratio between the speeds of propagation in them then the corner is resolved by polynomial scheme without noticeable oscillations. On the other hand WENO scheme gives non-oscillatory solution in both cases for $\varepsilon \leq h^2$. In this example the dielectric contrast is mild and linear cell sizes are proportional to the factor between the speeds of propagation in two media. Therefore both polynomial and WENO schemes have similar performance.

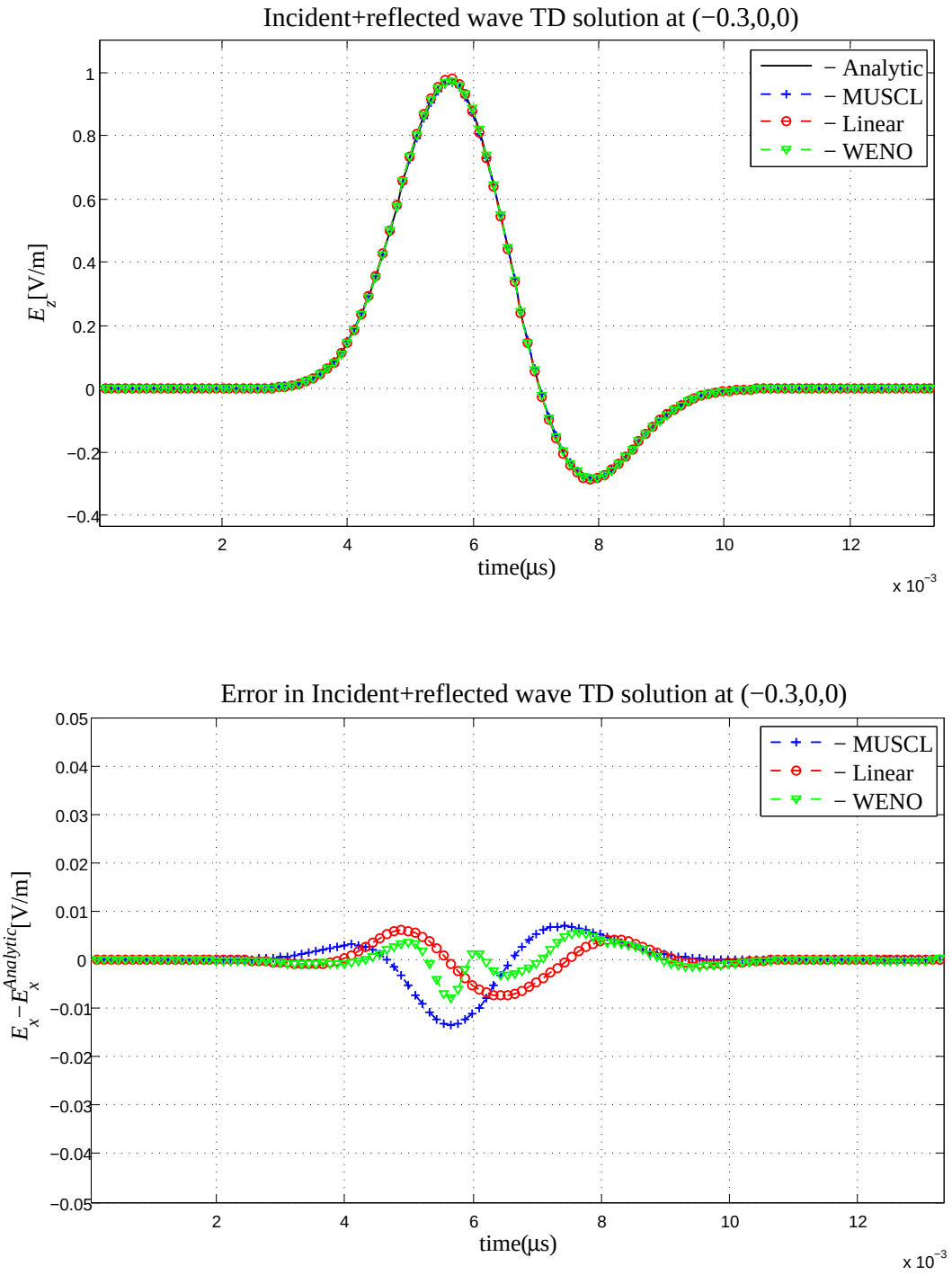


Figure 7.32: Plane-wave reflection/transmission at a dielectric interface: time-domain solution by second and third schemes at the observation point inside free space $P_1 = (-0.3, 0, 0)$.

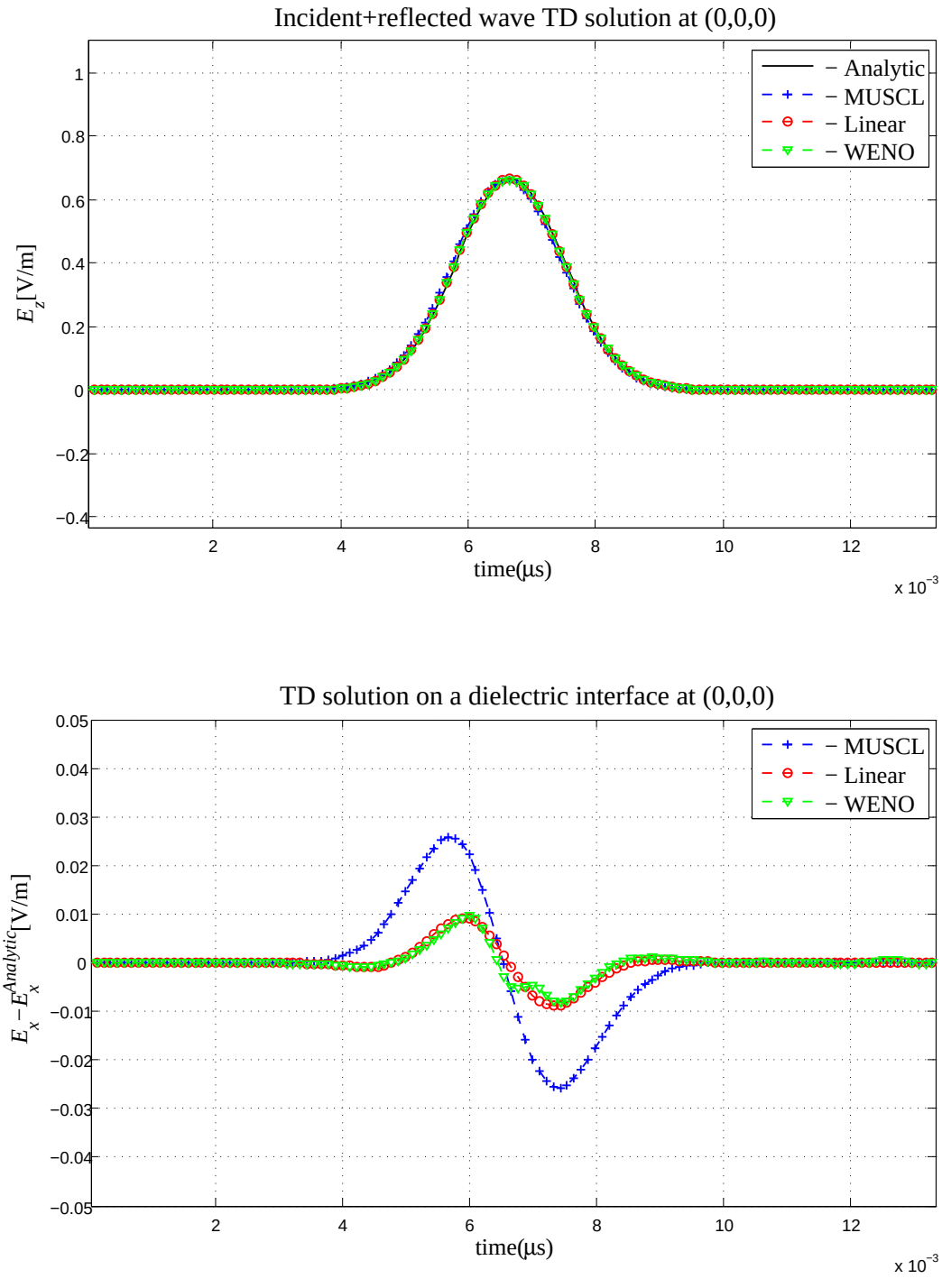
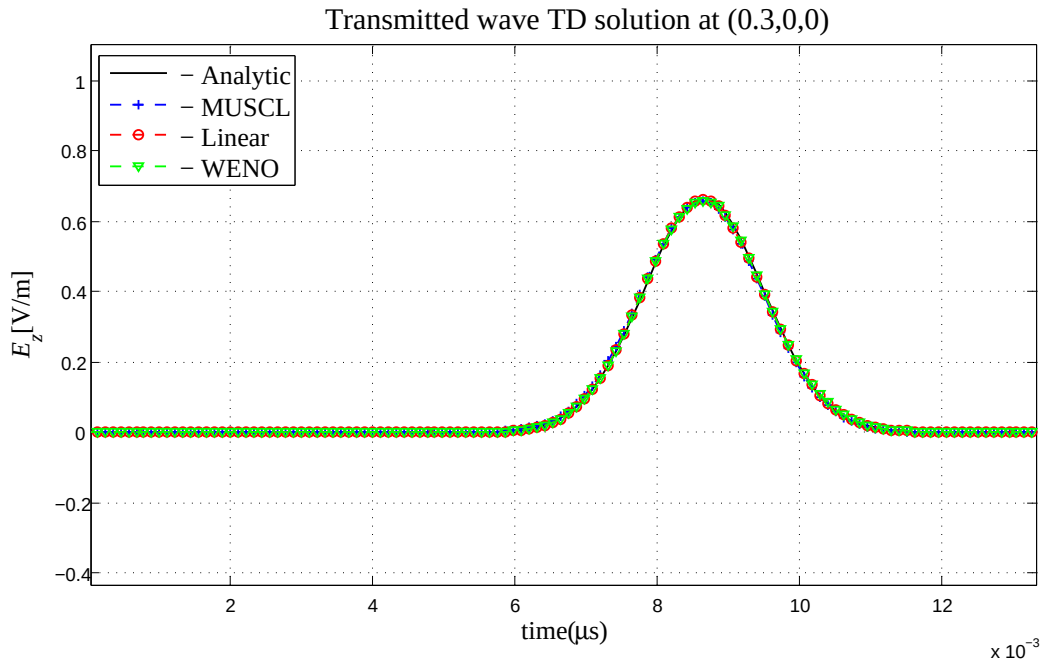


Figure 7.33: Plane-wave reflection/transmission at a dielectric interface: time-domain solution by second and third order schemes at the observation point at a dielectric interface $P_2 = (0,0,0)$.



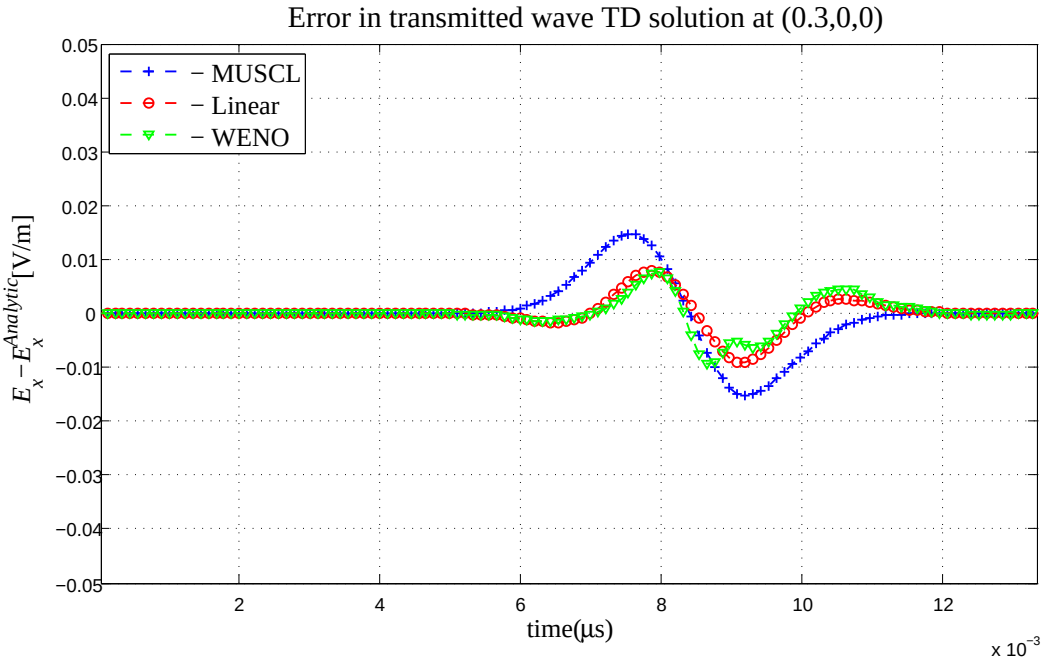


Figure 7.34: Plane-wave reflection/transmission at a dielectric interface: time-domain solution by second and third order schemes at the observation point inside dielectric $P_3 = (0.3, 0, 0)$.

# of cells	L^1 error at $T = 2c_0^{-1}$			L^1 error at $T = 2.5c_0^{-1}$		
	MUSCL	Linear	WENO	MUSCL	Linear	WENO
65900	2.0705e-2	8.0107e-3	8.9319e-3	2.0851e-2	1.3988e-2	1.5548e-2
497785	1.0220e-2	2.2526e-3	2.6461e-3	1.0961e-2	4.3745e-3	5.0034e-3

Table 7.11: Plane-wave reflection/transmission at a dielectric interface: L^1 and L^∞ errors.

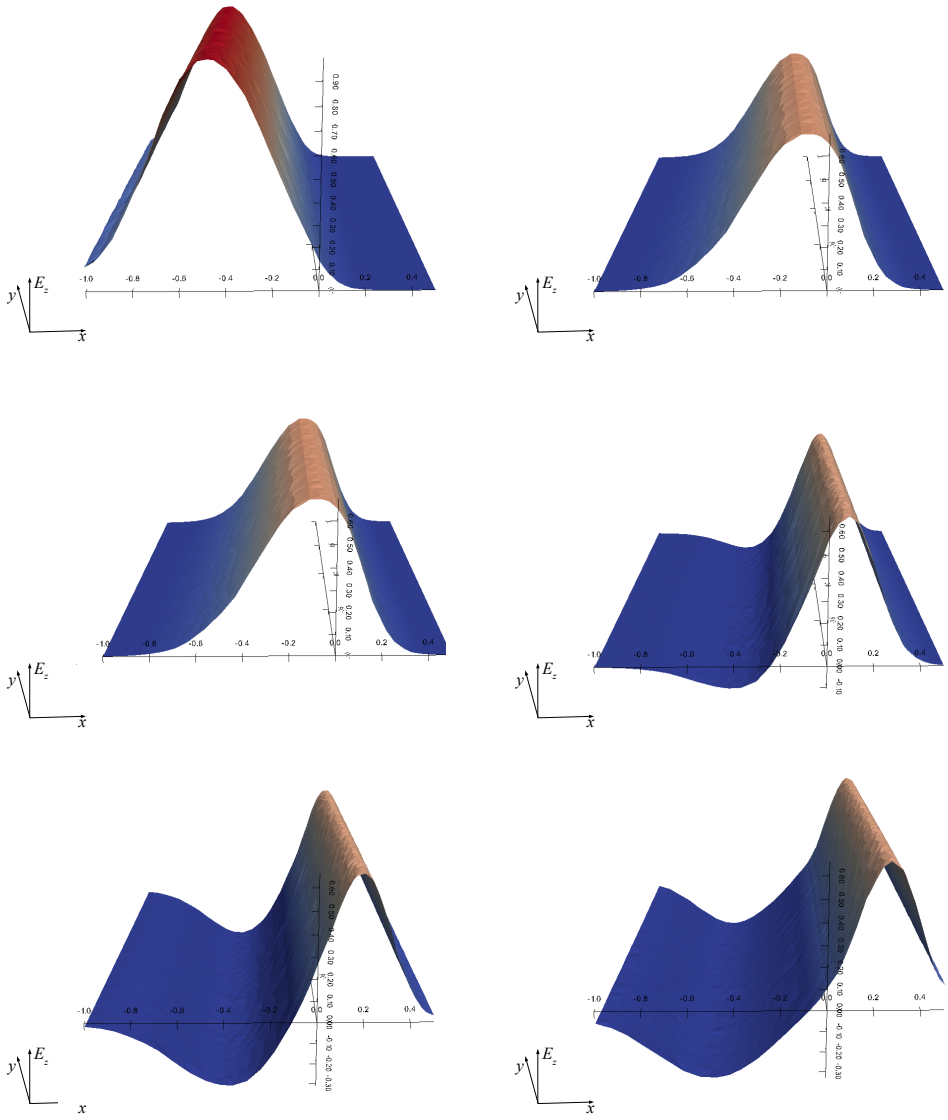


Figure 7.35: Plane-wave reflection/transmission at a dielectric interface: E_z in the xy-plane at various times between $1.5c_0^{-1}$ and $2.5c_0^{-1}$ by WENO scheme.

7.4 Chapter summary

Numerical results for both total-field and scattered-field formulations of the FVTD scheme with the type II WENO and MRK schemes are presented in this chapter. For the type II WENO scheme it was confirmed that the value of ε in the definition of non-linear weights has the same effect on the solution as in one-dimensional case. Numerical experiments confirm the advantages of type II WENO scheme over polynomial scheme by better stability and reduced oscillations for solutions with discontinuities. The main drawback of type II WENO scheme is its inability to find good linear weights for all quadrature points for arbitrary unstructured meshes. Some criteria are necessary to rule out reconstructions utilizing very negative linear weights. The percentage of very negative linear coefficients was found to be higher than in two-dimensional case reported in [89]. As a result a full type II WENO approximation that does not compromise the stability of the scheme on unstructured meshes is not possible. A small percentage of quadrature points required an alternative reconstruction, which was implemented by third order polynomial reconstruction. Numerical experiments with discontinuous pulse show the clear advantage of non-oscillatory properties of WENO over the polynomial scheme. Some works also indicate that WENO schemes are useful for problems of propagation of electromagnetic waves in multilayered linear dispersive media [110], which are not considered in this work.

Both multirate Runge-Kutta schemes demonstrate similar performance in terms of speedup. Flexibility in multirate partition was shown to be useful in maximizing the numerical speedup of MRK schemes. Both schemes allow an arbitrary partition in their application, as a result a higher than previously reported speedup was achieved in our experiments by optimizing partitions. In terms of accuracy both multirate schemes perform similarly with SSP RK2 base method but not with SSP RK3. Better accuracy is observed with MRK3-LLH scheme compared to MRK3-TW which agrees with theoretical results for one-dimensional problems. It was also observed that smaller errors may be achieved by the multirate scheme compared to singlerate scheme on mesh with high contrast in element sizes. One explanation for this is that the number of time integrations with a multirate scheme is significantly smaller in the coarsest region compared to the number of time integrations by a singlerate scheme. As a result the error caused by a large cell size does not

accumulate as much, as in the case of a very large number of uniform time-steps. On the other hand numerical errors are also affected by larger time-steps in coarse regions. Therefore, better accuracy of a multirate scheme compared to a singlerate scheme is not always observed. In any case, based on the numerical results in this work we can conclude that solutions obtained by MRK3-LLH are at least as accurate as the ones obtained by RK3.

Theoretical results on the stability of multirate schemes were not presented in this study. It should be noted that coupling strategies at multirate interfaces as well as a distribution of local time-steps may have different effect on the stability. Nevertheless, in numerical experiments both MRK-TW and MRK-LLH achieved stable results with the local time-step criteria (6.1).

Chapter 8

Summary and outlook

8.1 Summary

In this work we present an implementation of the third order type II WENO scheme for tetrahedral meshes and multirate time-stepping based on Runge-Kutta scheme for the solution of time-domain formulation of Maxwell's equations.

Accuracy analysis of one-dimensional WENO3 scheme. In order to implement the type II WENO scheme efficiently for three-dimensional problems we first perform a one-dimensional analysis of the dependence of accuracy on the value of the small parameter ε in the definition of WENO weights which was inspired by a similar study in [11]. It was found that the value ε proportional to the square of the cell size (h^2), proposed in [11], gives the best compromise to ensure a quality solution for both smooth and discontinuous parts, where smooth solutions benefit from larger values of ε ($\varepsilon \sim h$), while oscillations near discontinuities are better treated with $\varepsilon \sim h^4$. It is also important to scale the parameter ε according to characteristic values of the solution, otherwise the accuracy is also compromised. The numerical experiments presented in Chapter 7 confirm the validity of one-dimensional analysis for the three-dimensional case.

Implementation of third order finite volume WENO schemes to the solution of Maxwell's equations on tetrahedral meshes: In this thesis we study the application

of the type II WENO scheme on tetrahedral meshes to the solution of Maxwell's equations. The main difference from the one-dimensional case is that the linear weights are no longer all positive due to mesh irregularities. Hence the linear reconstruction does not always agree with the polynomial reconstruction on the same stencil. This affects the accuracy of the resulting WENO reconstruction and at the same time very negative linear weights produce instabilities. For stability purposes reconstructions with very negative linear weights were replaced by a third order polynomial reconstruction. Based on the accuracy analysis presented for the one-dimensional case in Chapter 3, we chose the small parameter ε in the formula for the non-linear weights (4.48) as a function of h_i defined by (4.13). The value of ε has to be scaled properly, if the problem is solved in non-normalized form. Similar to the results in one-dimensional case we observed that smooth solutions benefit from $\varepsilon = h_i$, while oscillations due to discontinuities are better treated with $\varepsilon = h_i^4$.

Accuracy analysis of MPRK schemes for linear problems. In order to find a third order accurate local time-stepping strategy for the solution of three-dimensional Maxwell's equations we review different multirate schemes based on Runge-Kutta time integration. Simple extensions of available multirate strategies by using higher order base method do not lead to a higher order multirate schemes. As it was shown in Chapter 5 it can even drop the accuracy, as in case of Tang-Warnecke scheme. To analyze the accuracy of different multirate approaches the so-called order conditions were used. Since the number of these conditions grows rapidly with order, it is very hard to satisfy all of them for the case of accuracy higher than second order. The focus of this work is on problems described by linear equations. It was shown in this thesis that fewer order conditions are necessary in this case. Using the linear order conditions we found that only one of the considered multirate schemes satisfies third order coupling between multirate groups when 3-stage third order RK schemes are used as a base method. These findings were confirmed by numerical experiments for one-dimensional linear advection equation. In three-dimensional experiments the third order multirate scheme (MRK-LLH) produces smaller pointwise errors than the single-rate Runge-Kutta scheme on meshes with large contrast in cell sizes.

Implementation of multirate Runge-Kutta strategies to the solution of Maxwell's equations. Two different multirate strategies based on Runge-Kutta schemes were implemented for the solution of Maxwell's equations, namely, the approach by Tang and Warnecke [125] (MRK-TW), and by Liu, Li and Hu [86] (MRK-LLH). The implementation includes the partition of the computational domain into multirate groups, the definition of the inner and outer buffers for each multirate group depending on the coupling strategy, and the algorithm for local time-stepping based on the multirate approach. Both multirate schemes (MRK-TW and MRK-LLH) allow arbitrary ratio between the time-steps of neighboring multirate groups. We use this advantage to define the optimal partition into multirate groups to achieve the best speedup possible.

8.2 Contributions

To the best of our knowledge this is the first implementation of the type II third order WENO scheme on tetrahedral meshes to the solution of Maxwell's equations [82]. Since only linear, isotropic and non-dispersive media is considered here, the main advantage of WENO was shown for a discontinuous pulse signal. But we believe that more complex models of media can benefit from WENO as well.

The effect of the choice of small parameter ε in the definition of WENO weights was addressed in literature previously [11, 21], and the choice $\varepsilon = h^2$ was studied in [11]. The novelty of this work is a theoretical and numerical comparison of the choices $\varepsilon = h^k$, $k = 1, ..., 4$, in classical third order WENO scheme on smooth and discontinuous solutions. These results were crucial for the implementation of WENO schemes on three-dimensional unstructured meshes.

Multirate formulations of Runge-Kutta schemes are implemented for three-dimensional Maxwell's equations [83]. The novelty of this work is the three-dimensional implementation and one-dimensional analysis of MRK schemes for linear problems. It is shown that the number of order conditions required for general non-linear problems is reduced in the linear case. Using the third order conditions for linear problems it is shown that the accuracy of MRK scheme depends on both the coupling procedure and the choice of the base method. In [86] the authors stated that their multirate procedure has the order of the base

scheme for linear problems The analysis in this work confirms this for RK3 schemes with three stages. The four stage SSP RK43 scheme satisfies second order conditions only. The Butcher form of Liu-Li-Hu [86] multirate scheme was derived in this work. It allowed to apply order conditions to identify the base RK schemes that produce third order multirate coupling for linear problems. Numerical results confirming the theoretical results are also presented. In three-dimensional experiments MRK3-LLH also showed the most accurate results. Optimal domain partition into multirate groups with arbitrary time-step ratios is proposed to increase the speedup achievable by local time-stepping. Flexibility in multirate coupling of schemes implemented in this work allowed much greater speedup (*more than a factor of 30*) on highly inhomogeneous meshes than previously reported in the literature.

8.3 Future improvements

8.3.1 Boundary conditions for FVTD

Higher order absorbing boundary conditions. In this work only the first order absorbing boundary condition was used. The attractive features of the Silver-Müller boundary condition is its simplicity and very small computational cost. But it is accurate only for normal incidence. Alternative treatments of open boundaries include the perfectly matched layers (PML) [109, 46, 79] and boundary conditions based on integral equations (IE) [42].

Curved surface boundary. In order to obtain the solution with higher order accuracy the boundary of the computational domain needs to be accurately resolved. In this study only tetrahedral elements with flat faces are considered. In this case the number of elements near small geometrical features with curved boundaries has to be sufficient to get the benefits of higher order numerical scheme. Namely, the first order error generated in the boundary region has to be proportional to the higher order error inside the computational domain. Alternatively, one can consider elements containing faces with non-zero curvature. In this case the boundary Gauss quadrature points and weights must correspond exactly to the

curved boundary, and integration along the curved surface has to consider the variation in the outward normal vector [93].

8.3.2 WENO schemes.

Type I and II WENO hybrid. As it was discovered numerically in this work the main drawback of the type II WENO scheme is its inability to find good linear weights for all quadrature points on arbitrary unstructured meshes. Even for relatively uniform meshes the occurrence of points with very negative linear weights is around 2 percent. Also it was found that even a relatively mild negative weights ($-5 < \gamma_l < -1$) affect the accuracy of the linear reconstruction and thus the WENO reconstruction as well. As it was suggested in [89] reconstructions at quadrature points with negative linear weights can be replaced by a more expensive but also more flexible type I WENO scheme. This will help to maintain the overall accuracy of the scheme without compromising the non-oscillatory properties (as in case of polynomial substitution).

WENO and polynomial hybrid schemes. Since the WENO3 scheme is significantly more expensive compared to the same order polynomial scheme, it is impractical for realistic applications. To get the advantages of the non-linear scheme without compromising the efficiency, it should be used only in regions near singularities. Elsewhere, it is more practically efficient to use a much cheaper polynomial reconstruction. It can be implemented by assigning different schemes to different parts of the computational domain or using a criteria to switch between the schemes. In order to implement the criteria a more sensitive than a classical smoothness indicator used in this work is needed, that is able to distinguish between flat and rough parts of the solution. Works on improvement of classic WENO scheme by modifying the smoothness indicator as well as non-linear weights can be found for one-dimensional and structured multidimensional case (see for example [21]).

Post processing for high order recovery of discontinuous solutions. When solving differential equations numerically with high order schemes the numerical error at the discontinuity is carried into the smooth regions degenerating the overall accuracy of the solution. The WENO schemes solve this problem by assigning bigger weights to stencils which do

not contain discontinuities and very small weights to stencils with discontinuities. As a result the full order of accuracy is achieved in smooth regions of discontinuous solution for scalar problems. For hyperbolic systems of equations with discontinuous solutions solved by WENO schemes, the errors from discontinuities lower the accuracy of pointwise errors in smooth regions [119]. But the designed order of accuracy is still preserved in integrated quantities of the numerical solution when the system and the scheme are linear [94, 91]. Therefore a post processing of the numerical solution can help achieve the designed order of accuracy of pointwise solution in smooth regions [55].

Different basis functions for WENO. Another modification of the type II WENO scheme can be accomplished with the help of other basis functions such as polyharmonic splines [5] and radial basis functions (RBF). Since Gaussian pulses are often used in CEM models, Gaussian RBFs may be a potential choice for Maxwell's equations. Implementation of different basis functions may produce WENO scheme with a lower number of small stencils reducing the computational cost of the approximation.

Comparative study of WENO3 and DG. Since the order of accuracy of FV scheme is increased by adding layers of elements into a stencil, it becomes cumbersome to implement boundary conditions as well as to extend the type II WENO scheme to higher orders. A study comparing the third order WENO scheme on unstructured meshes with the popular DG method would be of big interest in the future.

Maximum-principle satisfying WENO on unstructured meshes. Another extension of work with WENO method is the development of maximum-principle satisfying scheme by means of limiters that do not destroy the high order accuracy on smooth solutions. Such limiting strategy for WENO FV schemes on rectangular meshes was proposed by Zhang and Shu [134].

8.3.3 Multirate time-integration

Stability analysis of MPRK schemes. Stability analysis of multirate schemes is not presented in this work. Some stability related results for multirate schemes considered here

can be found in literature. Local entropy condition and maximum principle for MRK-TW scheme with the forward Euler base scheme are presented in [125]. Properties such as positivity, maximum principle preserving and total variation boundedness (TVB) under local CFL number were shown for MRK2-CS in [31]. Monotonicity conditions for MRK2-TW and MRK2-CS were studied in [75]. Stability analysis for MRK-LLH scheme was investigated on the one-dimensional wave equation with DG space discretization by studying the distribution of eigenvalues of the amplification matrix.

As far as we know there is no general criterion found for the stability of the multirate Runge-Kutta schemes. A strategy for doing linear stability analysis of one-step multirate schemes was proposed by Kværnø in [84]. This approach was demonstrated for a simple Euler scheme, but can be adapted to MRK schemes. The idea of the approach is to consider the following test problem

$$u_t = \Lambda u = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \ \Lambda \in \mathbb{R}^{2\times 2},$$

with

$$\lambda_{11}, \lambda_{22} < 0 \quad \text{and} \quad \gamma = \frac{\lambda_{12}\lambda_{21}}{\lambda_{11}\lambda_{22}} < 1,$$

where λ_{12} and λ_{21} are needed to estimate the influence of coupling on stability, and the parameter γ is used to measure the coupling between the equations. A compound step of an MRK scheme can then be expressed by

$$\begin{bmatrix} u_1^{n+1} \\ u_2^{n+1} \end{bmatrix} = K \begin{bmatrix} u_1^{n} \\ u_2^{n} \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} u_1^{n} \\ u_2^{n} \end{bmatrix},$$

Then the stability region is studied by determining the conditions for which the spectral radius of K is less than one.

IMEX schemes. Another way to limit the influence of very small elements on CPU time is to use co-called implicit-explicit (IMEX) formulations. In this case the region with the finest resolution can be treated implicitly, while the solution on the rest of the domain can be advanced in time using an explicit multirate scheme [61, 35].

Appendix A

Accuracy of WENO3 on non-uniform grid

Consider the general one-dimensional mesh $a = x_{\frac{1}{2}} < ... < x_{i-\frac{1}{2}} < x_{i+\frac{1}{2}} < x_{i+\frac{3}{2}} < ... < x_{N+\frac{1}{2}} = b$ with non-uniform cell size $h_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ on I_i. The linear polynomials $p_{1,i}^{(1)}(x)$ and $p_{1,i}^{(2)}(x)$ defined on small stencils $S_i^{(1)} = \{I_{i-1}, I_i\}$ and $S_i^{(2)} = \{I_i, I_{i+1}\}$ can be obtained using (3.19) as follows

$$p_{1,i}^{(1)}(x) = \overline{u}_i + \frac{2h_i}{h_{i-1}+h_i}\left[\overline{u}_i - \overline{u}_{i-1}\right]\xi, \tag{A.1}$$

$$p_{1,i}^{(2)}(x) = \overline{u}_i + \frac{2h_i}{h_i+h_{i+1}}\left[-\overline{u}_i + \overline{u}_{i+1}\right]\xi. \tag{A.2}$$

From the above and the smoothness indicators for both small stencils can obtained

$$SI_{1,i} = h_i \int_{I_i} \left(p_{1,i}^{(1)}(x)\right)_x^2 dx = \frac{4h_i^2}{\left(h_{i-1}+h_i\right)^2}\left[-\overline{u}_{i-1} + \overline{u}_i\right]^2, \tag{A.3}$$

$$SI_{2,i} = h_i \int_{I_i} \left(p_{1,i}^{(2)}(x)\right)_x^2 dx = \frac{4h_i^2}{\left(h_i+h_{i+1}\right)^2}\left[-\overline{u}_i + \overline{u}_{i+1}\right]^2. \tag{A.4}$$

Then the estimates similar to the uniform case can be proved in the following proposition

Theorem 13. *Let $u(x) \in C^3$ on the big stencil S_i. Then the smoothness indicators (A.3-A.4) have the following properties*

If $u'(x) \neq 0$ for all $x \in S_i$, then

$$SI_{l,i} = \alpha(x_i) h_i^2 + O\left(h_i^3\right), \qquad l \in \{1,2\} \tag{A.5}$$

for some locally Lipschitz continuous $\alpha_l(x)$, and

$$SI_{l_1,i} - SI_{l_2,i} = \beta_{l_1,l_2}(x_i) h_i^3 + O\left(h_i^4\right), \qquad l_1 \neq l_2,\ l_1,l_2 \in \{1,2\} \tag{A.6}$$

for some locally Lipschitz continuous $\beta_{l_1,l_2}(x)$.

If $u(x)$ has a point $x^* \in S_i \setminus \{x_i\}$ such that $u'(x^*) = 0$, then

$$SI_{l,i} = \alpha_l(x_i) h_i^4 + O\left(h_i^5\right), \tag{A.7}$$

and

$$SI_{l_1,i} - SI_{l_2,i} = \beta_{l_1,l_2}(x_i) h_i^4 + O\left(h_i^5\right), \tag{A.8}$$

for some locally Lipschitz continuous $\alpha_l(x)$ and $\beta_{l_1,l_2}(x)$ with $l_1 \neq l_2$, $l_1,l_2 \in \{1,2\}$.

Proof. Using the Taylor series of $U(x)$ about x_i we get

$$\begin{aligned}
SI_{1,i} =& \frac{4h_i^2}{(h_{i-1}+h_i)^2}\left[\overline{u}_i - \overline{u}_{i-1}\right]^2 \\
=& \frac{4h_i^2}{(h_{i-1}+h_i)^2}\left(\frac{U\left(x_i+\frac{h_i}{2}\right) - U\left(x_i-\frac{h_i}{2}\right)}{h_i} - \frac{U\left(x_i-\frac{h_i}{2}\right) - U\left(x_i-\frac{h_i}{2}-h_{i-1}\right)}{h_{i-1}}\right)^2 \\
=& \frac{4h_i^2}{(h_{i-1}+h_i)^2}\left(\frac{1}{2}u'(x_i)(h_i+h_{i-1}) - \frac{1}{6}u''(x_i)\left(\frac{1}{2}h_i^2 + \frac{3}{2}h_i h_{i-1} + h_{i-1}^2\right) + O\left(h_i^3\right)\right)^2 \\
=& \left(u'(x_i)h_i - \frac{1}{3}u''(x_i)\left(\frac{1}{2}h_i + h_{i-1}\right)h_i + O\left(h_i^2\right)\right)^2
\end{aligned}$$

and

$$
\begin{aligned}
SI_{2,i} =& \frac{4h_i^2}{(h_i+h_{i+1})^2}\left[-\overline{u}_i+\overline{u}_{i+1}\right]^2 \\
=& \frac{4h_i^2}{(h_i+h_{i+1})^2}\left(-\frac{U\left(x_i+\frac{h_i}{2}\right)-U\left(x_i-\frac{h_i}{2}\right)}{h_i}+\frac{U\left(x_i+\frac{h_i}{2}+h_{i+1}\right)-U\left(x_i+\frac{h_i}{2}\right)}{h_{i+1}}\right)^2 \\
=& \frac{4h_i^2}{(h_i+h_{i+1})^2}\left(\frac{1}{2}u'(x_i)(h_i+h_{i+1})+\frac{1}{6}u''(x_i)\left(2\left(\frac{h_i}{2}\right)^2+3\frac{h_i}{2}h_{i+1}+h_{i+1}^2\right)+O\left(h_i^3\right)\right)^2 \\
=& \left(u'(x_i)h_i+\frac{1}{3}u''(x_i)\left(\frac{1}{2}h_i+h_{i+1}\right)h_i+O\left(h_i^2\right)\right)^2
\end{aligned}
$$

Let $h_{i-1}=\kappa_1 h_i$ and $h_{i+1}=\kappa_2 h_i$, then

$$
SI_{l,i}=\left(u'(x_i)h_i+\delta_l u''(x_i)h_i^2+O\left(h_i^3\right)\right)^2, \tag{A.9}
$$

where $\delta_l=\left(\frac{2l}{3}-1\right)\left(\frac{1}{2}+\kappa_l\right)$. Therefore we deduce (A.5) and (A.6) with $\alpha(x_i)=\left[u'(x_i)\right]^2$ and $\beta_{l_1,l_2}(x_i)=2\left(\delta_{l_1}-\delta_{l_2}\right)u'(x_i)u''(x_i)$.

Now consider the case when $u'(x^*)=0$, for some $x^*\in S_i\setminus\{x_i\}$. Let $x_i-x^*=\kappa h_i$ with $0<|\kappa|<\frac{3}{2}$. Using

$$
u'(x_i)=u''(x_i)\kappa h_i+O\left(h_i^2\right).
$$

In (A.9) we get

$$
SI_{l,i}=\left(\eta_l u''(x_i)h_i^2+O\left(h_i^3\right)\right)^2,
$$

where $\eta_l=\left(\kappa+\left(\frac{2l}{3}-1\right)\left(\frac{1}{2}+\kappa_l\right)\right)$. Therefore we obtain the estimates (A.7) and (A.8) with $\alpha_l(x_i)=\left[\eta_l u''(x_i)\right]^2$ and $\beta_{l_1,l_2}(x_i)=\left(\eta_{l_1}^2-\eta_{l_2}^2\right)\left[u''(x_i)\right]^2$. □

Bibliography

[1] Gmsh website. [Online]. Available: `http://geuz.org/gmsh/`.

[2] MathWorks website. [Online]. Available: `http://www.mathworks.com/`.

[3] ParaView website. [Online]. Available: `http://www.paraview.org/`.

[4] R. Abgrall. On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation. *Journal of Computational Physics*, 114(1):45–58, 1994.

[5] T. Aboiyar, E. H. Georgoulis, and A. Iske. Adaptive ADER methods using kernel–based polyharmonic spline WENO reconstruction. *SIAM Journal on Scientific Computing*, 32(6):3251–3277, 2010.

[6] J. Ahrens, B. Geveci, and C. Law. ParaView: an end-user tool for large data visualization. *The Visualization Handbook*, pages 717–732, 2005.

[7] P. Albrecht. The Runge-Kutta theory in a nutshell. *SIAM Journal on Numerical Analysis*, 33(5):1712–1735, 1996.

[8] J. F. Andrus. Numerical solution of systems of ordinary differential equations separated into subsystems. *SIAM Journal on Numerical Analysis*, 16(4):605–611, 1979.

[9] J. F. Andrus. Stability of a multi–rate method for numerical integration of ODE's. *Computers & Mathematics with Applications*, 25(2):3–14, 1993.

[10] L. D. Angulo, J. Alvarez, F. L. Teixeira, M. F. Pantoja, and S. G. Garcia. Causal–path local time–stepping in the discontinuous Galerkin method for Maxwell's equations. *Journal of Computational Physics*, 256:678–695, 2014.

[11] F. Aràndiga, A. Baeza, A. M. Belda, and P. Mulet. Analysis of WENO schemes for full and global accuracy. *SIAM Journal on Numerical Analysis*, 49(2):893–915, 2011.

[12] F. Aràndiga, M. C. Martí, and P. Mulet. Weights design for maximal order WENO schemes. *Journal of Scientific Computing*, 60(3):641–659, 2014.

[13] C. A. Balanis. *Advanced Engineering Electromagnetics*. Wiley, 1989.

[14] C. A. Balanis. *Antenna Theory: Analysis and Design*. Wiley, 1997.

[15] P. Batten, C. Lambert, and D. M. Causon. Positively conservative high–resolution convection schemes for unstructured elements. *International Journal for Numerical Methods in Engineering*, 39(11):1821–1838, 1996.

[16] D. Baumann. *A 3–D numerical field solver based on the finite–volume time–domain method*. PhD thesis, Swiss Federal Institute of Technology, Zurich, 2006.

[17] D. Baumann, C. Fumeaux, and R. Vahldieck. Field–based scattering–matrix extraction scheme for the FVTD method exploiting a flux–splitting algorithm. *IEEE Transactions on Microwave Theory and Techniques*, 53(11):3595–3605, 2005.

[18] P. Bonnet, X. Ferrieres, F. Issac, F. Paladian, J. Grando, J. C. Alliot, and J. Fontaine. Numerical modeling of scattering problems using a time domain finite volume method. *Journal of Electromagnetic Waves and Applications*, 11(8):1165–1189, 1997.

[19] R. Borges, M. Carmona, B. Costa, and W. S. Don. An improved weighted essentially non–oscillatory scheme for hyperbolic conservation laws. *Journal of Computational Physics*, 227(6):3191–3211, 2008.

[20] J. C. Butcher. *Numerical Methods for Ordinary Differential Equations*. Wiley, 2003.

[21] M. Castro, B. Costa, and W. S. Don. High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 230(5):1766–1792, 2011.

[22] A. Chatterjee and R.-S. Myong. Efficient implementation of higher-order finite volume time-domain method for electrically large scatterers. *Progress In Electromagnetics Research B*, 17:233–254, 2009.

[23] M.-H. Chen, B. Cockburn, and F. Reitich. High-order RKDG methods for computational electromagnetics. *Journal of Scientific Computing*, 22(1-3):205–226, 2005.

[24] Q. Chen and P. Monk. Introduction to applications of numerical analysis in time domain computational electromagnetism. In *Frontiers in Numerical Analysis-Durham 2010*, pages 149–225. Springer, 2012.

[25] A. Christlieb and B. Ong. Implicit parallel time integrators. *Journal of Scientific Computing*, 49(2):167–179, 2011.

[26] A. J. Christlieb, R. D. Haynes, and B. W. Ong. A parallel space-time algorithm. *SIAM Journal on Scientific Computing*, 34(5):233–248, 2012.

[27] B. Cockburn, F. Li, and C.-W. Shu. Locally divergence-free discontinuous Galerkin methods for the Maxwell equations. *Journal of Computational Physics*, 194(2):588–610, 2004.

[28] F. Collino, T. Fouquet, and P. Joly. A conservative space–time mesh refinement method for the 1–d wave equation. Part I: Construction. *Numerische mathematik*, 95(2):197–221, 2003.

[29] F. Collino, T. Fouquet, and P. Joly. A conservative space–time mesh refinement method for the 1–D wave equation. Part II: Analysis. *Numerische mathematik*, 95(2):223–251, 2003.

[30] F. Collino, T. Fouquet, and P. Joly. Conservative space–time mesh refinement methods for the FDTD solution of Maxwell's equations. *Journal of Computational Physics*, 211(1):9–35, 2006.

[31] E. M. Constantinescu and A. Sandu. Multirate timestepping methods for hyperbolic conservation laws. *Journal of Scientific Computing*, 33(3):239–278, 2007.

[32] C. Dawson and R. Kirby. High resolution schemes for conservation laws with locally varying time steps. *SIAM Journal on Scientific Computing*, 22(6):2256–2284, 2001.

[33] C. Deng, W. Yin, S. Chai, and J. Mao. A Comparative study of the DGTD algorithm and the FVTD algorithm in Computational Electromagnetics. In *2010 Third International Joint Conference on Computational Science and Optimization (CSO)*, volume 1, pages 56–59. IEEE, 2010.

[34] N. Deore and A. Chatterjee. A cell–vertex finite volume time domain method for electromagnetic scattering. *Progress In Electromagnetics Research M*, 12:1–15, 2010.

[35] S. Descombes, S. Lanteri, and L. Moya. Locally implicit time integration strategies in a discontinuous Galerkin method for Maxwell's equations. *Journal of Scientific Computing*, 56(1):190–218, 2013.

[36] J. Diaz and M. Grote. Energy conserving explicit local time–stepping for second–order wave equations. *SIAM Journal on Scientific Computing*, 31(3):1985–2014, 2009.

[37] M. Dumbser and M. Käser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221(2):693–723, 2007.

[38] M. Dumbser, M. Käser, V. A. Titarev, and E. F. Toro. Quadrature–free non–oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *Journal of Computational Physics*, 226(1):204–243, 2007.

[39] L. J. Durlofsky, B. Engquist, and S. Osher. Triangle based adaptive stencils for the solution of hyperbolic conservation laws. *Journal of Computational Physics*, 98(1):64–73, 1992.

[40] F. Edelvik and G. Ledfelt. Explicit hybrid time domain solver for the Maxwell equations in 3D. *Journal of Scientific Computing*, 15(1):61–78, 2000.

[41] D. Firsov, J. LoVetri, I. Jeffrey, V. Okhmatovski, C. Gilmore, and W. Chamma. High-order FVTD on unstructured grids using an object–oriented computational engine. *ACES Journal*, 22(1):71–82, 2007.

[42] D. K. Firsov and J. LoVetri. FVTD–integral equation hybrid for Maxwell's equations. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 21(1–2):29–42, 2008.

[43] C. Fumeaux, D. Baumann, G. Almpanis, E. P. Li, and R. Vahldieck. Finite–Volume Time-Domain method for electromagnetic modelling: Strengths, limitations and challenges. *International Journal of Microwave and Optical Technology*, 3(3):318–328, 2008.

[44] C. Fumeaux, D. Baumann, P. Leuchtmann, and R. Vahldieck. A generalized local time–step scheme for efficient FVTD simulations in strongly inhomogeneous meshes. *IEEE Transactions on Microwave Theory and Techniques*, 52(3):1067–1076, 2004.

[45] C. Fumeaux, D. Baumann, and R. Vahldieck. Advanced FVTD simulation of dielectric resonator antennas and feed structures. *ACES Journal*, 19(3):155–164, 2004.

[46] C. Fumeaux, K. Sankaran, and R. Vahldieck. Spherical perfectly matched absorber for finite–volume 3–D domain truncation. *IEEE Transactions on Microwave Theory and Techniques*, 55(12):2773–2781, 2007.

[47] M. J. Gander, L. Halpern, and F. Nataf. Optimal Schwarz waveform relaxation for the one dimensional wave equation. *SIAM Journal on Numerical Analysis*, 41(5):1643–1681, 2004.

[48] R. Garg. *Analytical and Computational Methods in Electromagnetics*. Artech House, 2008.

[49] C. W. Gear and D. R. Wells. Multirate linear multistep methods. *BIT Numerical Mathematics*, 24(4):484–502, 1984.

[50] C. Geuzaine and J.-F. Remacle. Gmsh: A 3–D finite element mesh generator with built–in pre-and post–processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[51] N. Godel, S. Schomann, T. Warburton, and M. Clemens. GPU accelerated Adams–Bashforth multirate discontinuous Galerkin FEM simulation of high–frequency electromagnetic fields. *IEEE Transactions on Magnetics*, 46(8):2735–2738, 2010.

[52] S. K. Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.

[53] N. Goedel, S. Schomann, T. Warburton, and M. Clemens. Local timestepping discontinuous Galerkin methods for electromagnetic RF field problems. In *3rd European Conference on Antennas and Propagation, 2009*, pages 2149–2153. IEEE, 2009.

[54] J. B. Goodman and R. J. LeVeque. On the accuracy of stable schemes for 2d scalar conservation laws. *Mathematics of Computation*, 45(171):15–21, 1985.

[55] D. Gottlieb and C.-W. Shu. On the Gibbs phenomenon and its resolution. *SIAM Review*, 39(4):644–668, 1997.

[56] S. Gottlieb, D. I. Ketcheson, and C.-W. Shu. High order strong stability preserving time discretizations. *Journal of Scientific Computing*, 38(3):251–289, 2009.

[57] S. Gottlieb and C.-W. Shu. Total variation diminishing Runge–Kutta schemes. *Mathematics of Computation*, 67(221):73–85, 1998.

[58] S. Gottlieb, C. W. Shu, and E. Tadmor. High order strong stability preserving time discretizations. *SIAM Review*, 43(1):89–112, 2001.

[59] M. J. Grote and T. Mitkova. Explicit local time–stepping methods for Maxwell's equations. *Journal of Computational and Applied Mathematics*, 234(12):3283–3302, 2010.

[60] M. J. Grote and T. Mitkova. High–order explicit local time–stepping methods for damped wave equations. *Journal of Computational and Applied Mathematics*, 239:270–289, 2013.

[61] M. Günther, A. Kværnø, and P. Rentrop. Multirate partitioned Runge–Kutta methods. *BIT Numerical Mathematics*, 41(3):504–514, 2001.

[62] E. Hairer. Order conditions for numerical methods for partitioned ordinary differential equations. *Numerische Mathematik*, 36(4):431–445, 1981.

[63] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Solving Ordinary Differential Equations. Springer, 1993.

[64] R. F. Harrington. *Time–Harmonic Electromagnetic Fields*. McGraw-Hill, 1961.

[65] A. Harten. On a class of high resolution total-variation-stable finite-difference schemes. *SIAM Journal on Numerical Analysis*, 21(1):1–23, 1984.

[66] A. Harten and S. R. Chakravarthy. Multi-dimensional ENO schemes for general geometries. Technical report, DTIC Document, 1991.

[67] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high order accurate essentially non–oscillatory schemes, III. *Journal of Computational Physics*, 71(2):231–303, 1987.

[68] A. Harten and S. Osher. Uniformly high order accurate essentially non–oscillatory schemes, I. *SIAM Journal on Numerical Analysis*, 24(2):279–309, 1987.

[69] A. K. Henrick, T. D. Aslam, and J. M. Powers. Mapped weighted essentially non-oscillatory schemes: achieving optimal order near critical points. *Journal of Computational Physics*, 207(2):542–567, 2005.

[70] V. Hermes, I. Klioutchnikov, and H. Olivier. Linear stability of WENO schemes coupled with explicit Runge–Kutta schemes. *International Journal for Numerical Methods in Fluids*, 69(6):1065–1095, 2012.

[71] J. S. Hesthaven and T. Warburton. Nodal high-order methods on unstructured grids: I. Time-domain solution of Maxwell's equations. *Journal of Computational Physics*, 181(1):186–221, 2002.

[72] P. Hillion. Numerical integration on a triangle. *International Journal for Numerical Methods in Engineering*, 11(5):797–815, 1977.

[73] C. Hu and C.-W. Shu. Weighted essentially non–oscillatory schemes on triangular meshes. *Journal of Computational Physics*, 150(1):97–848, 1999.

[74] M. E. Hubbard. Multidimensional slope limiters for MUSCL-type finite volume schemes on unstructured grids. *Journal of Computational Physics*, 155(1):54–74, 1999.

[75] W. Hundsdorfer, A. Mozartova, and V. Savcenco. Monotonicity conditions for multirate and partitioned explicit Runge–Kutta schemes. In *Recent Developments in the Numerics of Nonlinear Hyperbolic Conservation Laws*, pages 177–195. Springer, 2013.

[76] Z. Jackiewicz and R. Vermiglio. Order conditions for partitioned Runge–Kutta methods. *Applications of Mathematics*, 45(4):301–316, 2000.

[77] I. Jeffrey. *Finite–volume simulations of Maxwell's equations on unstructure grids*. PhD thesis, University of Manitoba, Winnipeg, 2009.

[78] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.

[79] T. Kaufmann, K. Sankaran, C. Fumeaux, and R. Vahldieck. A review of perfectly matched absorbers for the finite-volume time-domain method. *ACES Journal*, 23(3):184–192, 2008.

[80] D. I. Ketcheson, C. B. Macdonald, and S. J. Ruuth. Spatially partitioned embedded Runge–Kutta methods. *SIAM Journal on Numerical Analysis*, 51(5):2887–2910, 2013.

[81] O. Knoth and R. Wolke. Implicit–explicit Runge–Kutta methods for computing atmospheric reactive flows. *Applied numerical mathematics*, 28(2):327–341, 1998.

[82] M. A. Kotovshchikova, D. K. Firsov, and S. H. Lui. Third order finite volume WENO scheme in application to Maxwell's equations on tetrahedral meshes. In preparation.

[83] M. A. Kotovshchikova, D. K. Firsov, and S. H. Lui. Third order multirate Runge-Kutta schemes for accelerating finite volume solution of 3D time-dependent Maxwell's equations. In preparation.

[84] A. Kværnø. Stability of multirate Runge–Kutta schemes. *International Journal of Differential Equations and Applications*, 1:97–105, 2000.

[85] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge, 2002.

[86] L. Liu, X. Li, and F. Q. Hu. Nonuniform time-step Runge–Kutta discontinuous Galerkin method for computational aeroacoustics. *Journal of Computational Physics*, 229(19):6874–6897, 2010.

[87] X.-D. Liu, S. Osher, and T. Chan. Weighted essentially non–oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

[88] Y. Liu and Zhang Y.-T. High order schemes on three–dimensional general polyhedral meshes – application to level set method. *Journal of Scientific Computing*, 5(2–4):836–848, 2009.

[89] Y. Liu and Y.-T. Zhang. A robust reconstruction for unstructured WENO schemes. *Journal of Scientific Computing*, 54(2–3):603–621, 2013.

[90] N. K. Madsen and R. W. Ziolkowski. Numerical solution of Maxwell's equations in the time domain using irregular nonorthogonal grids. *Wave Motion*, 10(6):583–596, 1988.

[91] A. Majda and S. Osher. Propagation of error into regions of smoothness for accurate difference approximations to hyperbolic equations. *Communications on Pure and Applied Mathematics*, 30(6):671–705, 1977.

[92] N. Manzanares-Filho, C. A. A. Moino, and A. B. Jorge. An improved controlled random search algorithm for inverse airfoil cascade design. *Proceedings of the 6th World Congresses of Structural and Multidisciplinary Optimization*, 2005.

[93] C. Michalak and C. Ollivier-Gooch. Accuracy preserving limiter for the high-order accurate solution of the Euler equations. *Journal of Computational Physics*, 228(23):8693–8711, 2009.

[94] M. S. Mock and P. D. Lax. The computation of discontinuous solutions of linear hyperbolic equations. *Communications on Pure and Applied Mathematics*, 31(4):423–430, 1978.

[95] E. Montseny, S. Pernet, X. Ferriéres, and G. Cohen. Dissipative terms and local time–stepping improvements in a spatial high order Discontinuous Galerkin scheme for the time–domain Maxwell's equations. *Journal of Computational Physics*, 227(14):6795–6820, 2008.

[96] M. Motamed, C. B. Macdonald, and S. J. Ruuth. On the linear stability of the fifth–order WENO discretization. *Journal of Scientific Computing*, 47(2):127–149, 2011.

[97] C. D. Munz, R. Schneider, and U. Voss. A finite–volume method for the maxwell equations in the time domain. *SIAM Journal on Scientific Computing*, 22(2):449–475, 2000.

[98] R.-H. Ni. A multiple–grid scheme for solving the Euler equations. *AIAA Journal*, 20:1565–1571, 1982.

[99] C. Ollivier-Gooch and M. Van Altena. A high–order–accurate unstructured mesh finite–volume scheme for the advection–diffusion equation. *Journal of Computational Physics*, 181(2):729–752, 2002.

[100] S. Osher and R. Sanders. Numerical approximations to nonlinear conservation laws with locally varying time and space grids. *Mathematics of Computation*, 41(164):321–336, 1983.

[101] S. Piperno. Symplectic local time–stepping in non–dissipative DGTD methods applied to wave propagation problems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 40(5):815–841, 2006.

[102] S. Piperno, M. Remaki, and L. Fezoui. A nondiffusive finite volume scheme for the three–dimensional Maxwell's equations on unstructured meshes. *SIAM Journal on Numerical Analysis*, 39(6):2089–2108, 2002.

[103] H. T. Rathod and S. V. Hiremath. Boundary integration of polynomials over an arbitrary linear tetrahedron in Euclidean three–dimensional space. *Computer Methods in Applied Mechanics and Engineering*, 153(1):81–106, 1998.

[104] M. Remaki. A new finite volume scheme for solving Maxwell's system. *International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, 19(3):913–931, 2000.

[105] J. R. Rice. Split Runge-Kutta methods for simultaneous equations. *J. Res. Natl. Bur. Standards B. Mathematics and Mathematical Physics*, 64:151–170, 1960.

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

[107] C. Sanderson. Armadillo: An open source C++ linear algebra library for fast prototyping and computationally intensive experiments. 2010.

[108] A. Sandu and E. M. Constantinescu. Multirate explicit Adams methods for time integration of conservation laws. *Journal of Scientific Computing*, 38(2):229–249, 2009.

[109] K. Sankaran, C. Fumeaux, and R. Vahldieck. Cell–centered finite–volume–based perfectly matched layer for time–domain Maxwell system. *IEEE Transactions on Microwave Theory and Techniques*, 54(3):1269–1276, 2006.

[110] S. Sardeshpande and A. Chatterjee. Electromagnetic wave propagation in linearly dispersive media using higher-order WENO scheme. *Journal of Electromagnetic Waves and Applications*, 23(16):2135–2142, 2009.

[111] M. Schlegel, O. Knoth, M. Arnold, and R. Wolke. Multirate Runge–Kutta schemes for advection equations. *Journal of Computational and Applied Mathematics*, 226(2):345–357, 2009.

[112] M. Schlegel, O. Knoth, M. Arnold, and R. Wolke. Numerical solution of multiscale problems in atmospheric modeling. *Applied Numerical Mathematics*, 62(10):1531–1543, 2012.

[113] B. Seny, J. Lambrechts, R. Comblen, V. Legat, and J.-F. Remacle. Multirate time stepping for accelerating explicit discontinuous galerkin computations with application to geophysical flows. *International Journal for Numerical Methods in Fluids*, 71(1):41–64, 2013.

[114] V. Shankar, W. F. Hall, and A. H. Mohammadian. A time–domain differential solver for electromagnetic scattering problems. *Proceedings of IEEE*, 77(5):709–721, 1989.

[115] J. Shi, C. Hu, and C.-W. Shu. A technique of treating negative weights in WENO schemes. *Journal of Computational Physics*, 175(1):108–127, 2002.

[116] C.-W. Shu. TVB uniformly high–order schemes for conservation laws. *Mathematics of Computation*, 49(179):105–121, 1987.

[117] C.-W. Shu. Total–variation–diminishing time discretizations. *SIAM Journal on Scientific and Statistical Computing*, 9:1073, 1988.

[118] C.-W. Shu. *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*. Springer, 1998.

[119] C.-W. Shu. High order weighted essentially nonoscillatory schemes for convection dominated problems. *SIAM Review*, 51(1):82–126, 2009.

[120] C.-W. Shu and S. Osher. Efficient implementation of essentially non–oscillatory shock–capturing schemes. *Journal of Computational Physics*, 77(1):439–471, 1988.

[121] G. R. Shubin and J. B. Bell. A modified equation approach to constructing fourth order methods for acoustic wave propagation. *SIAM Journal on Scientific and Statistical Computing*, 8(2):135–151, 1987.

[122] T. Sonar. On the construction of essentially non–oscillatory finite volume approximations to hyperbolic conservation laws on general triangulations: polynomial recovery, accuracy and stencil selection. *Computer Methods in Applied Mechanics and Engineering*, 140(1):157–181, 1997.

[123] J. L. Steger and R. F. Warming. Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods. *Journal of Computational Physics*, 40(2):263–293, 1981.

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

[125] H.-Z. Tang and G. Warnecke. High resolution schemes for conservation laws and convection–diffusion equations with varying time and space grids. *Journal of Computational Mathematics*, 24(2):121–140, 2006.

[126] A. Taube, M. Dumbser, C. D. Munz, and R. Schneider. A high–order discontinuous Galerkin method with time–accurate local time stepping for the Maxwell equations. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 22(1):77–103, 2008.

[127] P. Tsoutsanis, V. A. Titarev, and D. Drikakis. WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions. *Journal of Computational Physics*, 230(4):1585–1601, 2011.

[128] B. Van Leer. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. *Journal of Computational Physics*, 32(1):101–136, 1979.

[129] R. Wang and R. J. Spiteri. Linear instability of the fifth-order WENO method. *SIAM Journal on Numerical Analysis*, 45(5):1871–1901, 2007.

[130] M. Wirianto, W. A. Mulder, and E. C. Slob. Applying essentially non-oscillatory interpolation to controlled-source electromagnetic modelling*. *Geophysical Prospecting*, 59(1):161–175, 2011.

[131] N. K. Yamaleev and Mark H. Carpenter. A systematic methodology for constructing high-order energy stable WENO schemes. *Journal of Computational Physics*, 228(11):4248–4272, 2009.

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

[133] 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 Transactions on Antennas and Propagation*, 45(3):354–363, 1997.

[134] X. Zhang and C.-W. Shu. Maximum–principle–satisfying and positivity–preserving high-order schemes for conservation laws: survey and new developments. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. The Royal Society, 2011.

[135] Y.-T. Zhang and C.-W. Shu. Third order WENO scheme on three dimensional tetrahedral meshes. *Computer Physics Communications*, 5(2–4):836–848, 2009.