A High-Order Particle-in-Cell Method for Low Density Plasma Flow and the Simulation of Gyrotron Resonator Devices

A thesis accepted by the
Faculty of Aerospace Engineering and Geodesy
of the Universität Stuttgart
in partial fulfillment of the requirements for the degree of
Doctor of Engineering Sciences (Dr.-Ing.)

by

Andreas Stock

born in Leipzig

Main-referee	:	Prof. Dr. rer. nat. habil. Claus-Dieter Munz
Co-referee	:	Prof. Dr.-Ing. John Jelonnek
Date of defence	:	26th of April 2013

Institute of Aerodynamics und Gas Dynamics
University of Stuttgart

2013

For Kristina, Thomas and Monika.

Preface

First of all, I want to thank my doctoral supervisor Prof. Dr. Claus-Dieter Munz for providing excellent working conditions, for the scientific freedom and for the chance of working on an exciting interdisciplinary field of research. Many thanks to Prof. Dr. John Jelonnek for being the co-referee for this thesis. Special thanks to Dr. Rudolf Schneider for the great support and the many important inputs.

A lot of thanks to all colleagues from the working group of Prof. Munz for many inspiring scientific discussions and the cooperative and excellent working atmosphere. Special thanks to Dr. Jonathan Neudorfer for having a great time in the Particle-in-Cell code-development resulting in significant scientific output.

Many thanks to Dr. Stefan Kern, Dr. Stefan Illy and Andreas Schlaich from the Institute for Pulsed Power and Microwave Technology for the fruitful scientific discussions on the field of gyrotrons.

Also many thanks to the colleagues of the Institute of Space Systems, Torsten Stindl and Marcel Pfeiffer, for their great collaboration in several projects.

Finally, many thanks to my student assistants, Georg Pirrung, Marc Riedlinger, Dan Fries, and Philip Ortwein, who contributed directly or indirectly to my work at the IAG.

The work was partially financed by the Deutsche Forschungsgemeinschaft (DFG). The computation time was funded by the High-Performance Computing Center Stuttgart (HLRS) and the European Fusion Development Agreement (EFDA) in association with the Jülich Supercomputing Center (JSC).

Stuttgart, 26nd of May 2013

Andreas Stock

Contents

Symbols

Nomenclature conventions:

bold symbols (e.g. $\boldsymbol{j}$)	vector
calligraphic symbols (e.g. $\mathcal{V}$)	matrix
normal symbols (e.g. ρ)	scalar
superscript ˜	nodal ansatz function
superscript ˆ	modal ansatz function

Symbol	Unit	Name
A		amplitude
a_i		Adams-Bashforth coefficients
α		ratio between transversal and parallel velocity components
α_s		exponent of polynomial shape function
B	[T]	magnetic flux density (scalar)
$\boldsymbol{B}$	[T]	magnetic flux density (vector)
$\boldsymbol{B}_0$	[T]	magnetic flux density at $t = 0$
B_x	[T]	x-component of magnetic flux density
B_y	[T]	y-component of magnetic flux density
B_z	[T]	z-component of magnetic flux density
B_s		Beta function (index s refers to the usage within the shape function)
b_i		special MRAB coefficients
c_0	[m/s]	speed of light in vacuum ($c_0 = 299792458$)
$\boldsymbol{D}$	[As/m^2]	electric displacement field
d_i		interpolation coefficients
Δt	[s]	time step
Δt_{serial}	[s]	serial time step
$\vert\Delta \boldsymbol{x}\vert$		spatial measure for timestep calculation
$\boldsymbol{E}$	[V/m]	electric field strength (vector)
$\boldsymbol{E}_0$	[V/m]	electric field strength at $t = 0$
E_x	[V/m]	x-component or electric field strength
E_y	[V/m]	y-component or electric field strength
E_z	[V/m]	z-component or electric field strength

E_r	[V/m]	radial component or electric field strength
E_{El}	[J]	electric field energy
E_{kin}	[J]	kinetic energy
E_{Mag}	[J]	magnetic field energy
E_{pot}	[J]	potential energy
ε_0	[As/Vm]	vacuum permittivity ($\varepsilon_0 = 8.8541878176 \times 10^{-12}$)
ε_r		electric permittivity factor
ε_d		depression factor for hyperbolic divergence cleaning technique
$\boldsymbol{F}$	[N]	force
$\boldsymbol{F}_h$		numerical intercell flux
$\boldsymbol{F}_L$	[N]	Lorentz force
f		probability density function
$f(t, u)$		right-hand-side function
$\boldsymbol{f}_{NME}, \boldsymbol{f}_{ACW}, \boldsymbol{f}_{LM}$		righ-hand-side function in MRAB-PHM scheme
$\boldsymbol{g}$		source vector in the PHM system
γ		relativistic factor
Γ		boundary of domain Ω
Γ_A		absorbing boundary of domain Ω
Γ_C		perfectly conducting boundary of domain Ω
$\boldsymbol{H}$	[A/m]	magnetic field strength
$h(NME), h(ACW), h(LM)$		histories in MRAB-PHM scheme
$\boldsymbol{j}$	[A/m^2]	current density
j_x	[A/m^2]	x-component of current density
j_y	[A/m^2]	y-component of current density
j_z	[A/m^2]	z-component of current density
$\mathbb{K}_\beta$		physical flux matrix for dimension $\beta = 1, 3$
k		wave number
$L_i, L_i(\boldsymbol{x})$		Lagrange basis function
m	[kg]	mass
μ_0	[Vs/Am]	vacuum permeability ($\mu_0 = 4\pi \times 10^{-7}$)
μ_r		magnetic permeability factor
N		spatial convergence order of the DG scheme
N_m		number of modal DOF

N_n		number of nodal DOF
N_p		number of particles
n		degree of interpolating AB polynomial
n_{procs}		number of processes (in context of parallelization)
n_s		number density (of species s)
o_i, g_i, s_i		LSERK4 coefficients (see tab. F.1)
Ω		computational domain
Ω_c	[rad/s]	cyclotron frequency
ω	[rad/s]	angular frequency
$\omega_{p,i}$	[rad/s]	plasma frequency of i-th species
ω_{RF}	[rad/s]	radio frequency (of an EM field)
ω_s	[rad/s]	plasma frequency (of species s)
$\boldsymbol{p}$	[Ns]	relativistic momentum
p		polynomial degree
p_n		interpolation polynomial of degree n
$\phi_i, \phi_i(x)$		modal basis/test function
φ_L	[°]	phase angle of a gyrating electron
ψ_i		test function
Φ, Ψ		scalar potential for divergence cleaning
q	[As] = [C]	electric charge
$\boldsymbol{q}(x)$		test function
$\boldsymbol{R}_h$		residual vector of approximative solution
R_e	[mm]	electron hollow beam radius (guiding-centre-radius)
R_s	[m]	cutoff radius of the shape function
r_L	[mm]	Larmor radius
ρ	[As/m^3]	charge density
T	[s]	period
t	[s]	time
Θ_1, Θ_2	[°]	angle for electron hollow beam initiation
U_{acc}	[V]	accelerating voltage
$\boldsymbol{u}$		state vector in a PDE
$\tilde{\boldsymbol{u}}$		vector of nodal degrees of freedom
$\hat{\boldsymbol{u}}$		vector of modal degrees of freedom
$\boldsymbol{u}_h$		approximation of a state vector in a PDE

$\mathcal{V}$		Vandermonde matrix
$\boldsymbol{v}$	[m/s]	velocity vector: $\boldsymbol{v} \in \mathbb{R}^3$
$\boldsymbol{v}_p$	[m/s]	particle velocity magnitude
v_d	[m/s]	drift velocity
v_{th}	[m/s]	thermal velocity
$v_\perp$	[m/s]	transversal velocity magnitude
$v_{\vert\vert}$	[m/s]	parallel velocity magnitude
$\boldsymbol{x}$	[m]	space vector: $\boldsymbol{x} \in \mathbb{R}^3$
χ, κ		divergence correction speed factors of the PHM system (3.5)-(3.8)
∇		nabla operator
∇_x		nabla operator w.r.t. space $\left[\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right]^T$
∇_v		nabla operator w.r.t. velocity$\left[\frac{\partial}{\partial v_x}, \frac{\partial}{\partial v_y}, \frac{\partial}{\partial v_z}\right]^T$
$\nabla\times$		curl operator
$\nabla\cdot$		divergence operator
$\frac{\partial}{\partial t}$		time derivative operator
$\int_k$		integral over element k
$\int_{\partial k}$		integral over the surface of element k

Abbreviations

AB	**A**dams-**B**ashforth (method)
ACI	**A**fter **C**avity **I**nteraction
ADER	**A**rbitrary High-Order **Der**ivatives Approach
CBM	**C**artesian **B**ackground **M**esh
CFD	**C**omputational **F**luid **D**ynamics
CFL	**C**ourant-**F**riedrichs-**L**ewy
CK	**C**auchy-**K**ovalesvskaya
CPU	**C**entral **P**rocessing **U**nit
DG	**D**iscontinuous **G**alerkin (Method)
DGSEM	**D**iscontinuous **G**alerkin **S**pectral **E**lement **M**ethod
DG-PIC	**D**iscontinuous **G**alerkin **P**article-**i**n-**C**ell (Method)
DOF	**D**egree(s) **o**f **F**reedom
EM	**e**lectro**m**agnetic (field)
EOC	**E**xperimental **O**rder (of) **C**onvergence
FD	**F**inite **D**ifference (Method)
FDTD	**F**inite **D**ifference **T**ime **D**omain (Method)
FE	**F**inite **E**lement (Method)
FV	**F**inite **V**olume (Method)
GLM	**G**eneralized **L**agrange **M**ultiplier
HALO	**H**ighly **A**daptive **L**ocal **O**perator
HPC	**H**igh **P**erformace **C**omputing
IP	**I**nterpolation **P**oint
LM	**L**agrange **M**ultiplier
LSERK	**L**ow **S**torage **E**xlicit **R**unge-**K**utta Method
LTS	**L**ocal **T**ime **S**tepping
MPI	**M**essage **P**assing **I**nterface
MRAB	**M**ulti-**R**ate **A**dams-**B**ashforth
MRK	**M**ulti-Rate **R**unge-**K**utta
ODE	**O**rdinary **D**ifferential **E**quation
PDE	**P**artial **D**ifferential **E**quation
PDF	**P**article **D**ensity **F**unction
PE	**P**arallel **E**fficiency
PIC	**P**article-**in-C**ell (Method)

RF	**R**adio **F**requency
RHS	**R**ight-**H**and-**S**ide
RK	**R**unge **K**utta
RKDG	**R**unge **K**utta **D**iscontinuous **G**alerkin
RP	**R**iemann **P**roblem
RP1/2/3	**R**ecord **P**oint 1/2/3
SELFT	**SELF** consistent **T**imedependent (code for gyrotron design by [Kern, 1996])
SOTA	**S**tate **o**f **t**he **A**rt
SU	**S**peed**u**p
TE	**T**ransversal **E**lectric
TM	**T**ransversal **M**agnetic
TSE	**T**aylor-**S**eries-**E**xpansion
TSI	**T**wo-**S**tream **I**nstability
w.r.t.	**w**ith **r**espect **t**o

Kurzfassung

In dieser Arbeit wird ein parallelisiertes, transientes und dreidimensionales Discontinuous Galerkin Particle-in-Cell Verfahren hoher Ordnung entwickelt und für die Simulation von Gyrotron-Resonatoren auf Hochleistungsrechnern verwendet.

Das Discontinuous Galerkin Verfahren - eine Finite-Elemente Methode - ist eine schnelle und effiziente Methode zur numerischen Lösung der in dieser Arbeit verwendeten Maxwell-Gleichungen. Der Discontinuous Galerkin Ansatz ermöglicht eine einfache Benutzung von unstrukturierten Gittern, welche zur Simulation komplex-geformter technischer Geräte benötigt werden. Zudem ermöglicht das Discontinuous Galerkin Verfahren die Approximation einer Wellenlänge mit deutlich weniger Freiheitsgraden als andere Methoden (z. B. die Finite-Differenzen Methode). Aufgrund der sehr lokalen Abhängigkeiten zwischen den Elementen, bietet das Discontinuous Galerkin Verfahren exzellente Parallelisierungseigenschaften. Diese Eigenschaften sind essentiell für eine effiziente numerische Behandlung des Vlasov-Maxwell-Systems mit der Particle-in-Cell Methode. Dieses System beschreibt die selbstkonsistente Wechselwirkung zwischen geladenen Teilchen und dem elektromagnetischen Feld.

Als zentrale Anwendung dieser Arbeit werden mit dem Discontinuous Galerkin Particle-in-Cell Verfahren verschiedene Gyrotron-Resonatoren auf Hochstleistungsrechnern simuliert. Gyrotrons sind Hochleistungs-Millimeterwellen-Quellen zur Elektron-Zyklotronresonanz-Heizung von magnetisch eingeschlossenen Fusionsplasmen, wie zum Beispiel im Wendelstein 7-X Experimental-Fusionsreaktor.

Gegenüber herkömmlichen Simulationstools, welche in der Auslegung von Gyrotron-Resonatoren verwendet werden, bietet die Particle-in-Cell Methode den Vorteil, dass sie bezogen auf die Modellierung der Wechselwirkung, der Geometrie und des Eigenwellen-Spektrums keine signifikanten physikalischen Vereinfachungen beinhaltet. Daher ist sie die beste Methode zur Validierung herkömmlicher Simulationstools, welche die genannten Vereinfachungen haben. Bisher war die Particle-in-Cell Methode durch den hohen Rechenaufwand und die dadurch resultierenden hohen Rechenzeiten auf Demonstrationsrechnungen beschränkt. Unter Verwendung des parallelisierten Discontinuous Galerkin Particle-in-Cell Verfahren hoher Ordnung wird in dieser Arbeit erstmalig gezeigt, dass dreidimensionale und transiente Simulationen von Gyrotron-

Resonatoren mit hohen Moden-Indizes für Forschungs- und Entwicklungs-Zwecke effizient auf Hochleistungsrechnern durchgeführt werden können.

Anhand eines 30 GHz Gyrotron-Resonators wird das in dieser Arbeit entwickelte Discontinuous Galerkin Particle-in-Cell Verfahren verifiziert und mit dem SELFT Code - einem herkömmlichen Resonator Simulations-Code - verglichen. Darüber hinaus wird das Discontinuous Galerkin Particle-in-Cell Verfahren zur Simulation des Resonators des 1 MW, 140 GHz, $TE_{28,8}$-Mode Gyrotrons verwendet, welches zur Plasma-Heizung im Wendelstein 7-X Experimental-Fusionsreaktor verwendet werden soll.

Aufgrund der hohen Anzahl von Freiheitsgraden zur Diskretisierung des Rechengebietes sowie der hohen Anzahl von Teilchen können diese Simulationen nur auf Hochleistungsrechnern mit ausreichend Speicher und Rechenleistung durchgeführt werden. Deshalb wurde in dieser Arbeit erstmalig eine Parallelisierung der Shape-Function Depositions-Methode auf unstrukturierten Gittern entwickelt. Diese erlaubt die Kopplung hoher Ordnung von geladenen Teilchen und dem elektromagnetischen Feld.

Zur weiteren Verbesserung des Discontinuous Galerkin Particle-in-Cell Code, wurde eine Mehr-Raten Adams-Bashforth Zeitintegrations-Methode beliebig hoher Ordnung für die hyperbolische Divergenz-Korrektur entwickelt und verifiziert. Diese neue Methode berücksichtigt erstmalig die unterschiedlichen Zeitskalen in der hyperbolischen Divergenz-Korrektur und ermöglicht einen effizienteren Zeit-Integrations-Algorithmus als bei herkömmlichen Zeit-Integrations-Verfahren.

Die in dieser Arbeit durchgeführten Simulationen ermöglichen neue physikalische Erkenntnisse zur komplexen Teilchen-Feld-Wechselwirkung in Gyrotrons. Der Discontinuous Galerkin Particle-in-Cell Code kann zur Analyse von transienten Phänomenen, wie der fehlerhaften Strahlausrichtung, Modenkonkurrenz oder Wechselwirkungen am Ausgang des Resonators, der sogenannte After-Cavity-Interaction, verwendet werden. Diese Phänomene können mit herkömmlichen Codes nicht, oder nur unzureichend simuliert werden. Darüber hinaus wird in dieser Arbeit demonstriert, dass der Discontinuous Galerkin Particle-in-Cell Code hoher Ordnung erstmalig eine gekoppelte Simulation der Wellenleiterantenne (Launcher) und des Resonators eines Gyrotrons ermöglicht. Mit herkömmlichen Codes konnten diese Gyrotron-Komponenten nur getrennt voneinander simuliert werden, wobei die komplexe Wechselwirkung am Resonatorausgang und der Einfluss des Elektronenhohlstrahls im Launcher nicht erfasst werden konnten. Discontinuous Galerkin Particle-in-Cell Simulationen hoher Ordnung ermöglichen damit die vollständige Analyse der Wechselwirkung im Gyrotron. Darüber hinaus weisen sie den Weg hin zur vollständigen Simulation des Gyrotrons mit all seinen Komponenten, welche unter Berücksichtigung zukünftiger Hochleistungsrechner-Kapazitäten mit einem Discontinuous Galerkin Particle-in-Cell Verfahren hoher Ordnung, wie es in dieser Arbeit vorgestellt wurde, möglich sein wird.

Abstract

Within this thesis a parallelized, transient, three-dimensional, high-order discontinuous Galerkin Particle-in-Cell solver is developed and used to simulate the resonant cavity of a gyrotron.

The high-order discontinuous Galerkin approach - a Finite-Element type method - provides a fast and efficient algorithm to numerically solve Maxwell's equations used within this thesis. Besides its outstanding dissipation and dispersion properties, the discontinuous Galerkin approach easily allows for using unstructured grids, as required to simulate complex-shaped engineering devices. The discontinuous Galerkin approach approximates a wavelength with significantly less degrees of freedom compared to other methods, e.g. Finite Difference methods. Furthermore, the parallelization capabilities of the discontinuous Galerkin framework are excellent due to the very local dependencies between the elements. These properties are essential for the efficient numerical treatment of the Vlasov-Maxwell system with the Particle-in-Cell method. This system describes the self-consistent interaction of charged particles and the electromagnetic field.

As central application within this thesis gyrotron resonators are simulated with the discontinuous Galerkin Particle-in-Cell method on high-performance-computers. The gyrotron is a high-power millimeter wave source, used for the electron cyclotron resonance heating of magnetically confined fusion plasma, e.g. in the Wendelstein 7-X experimental fusion-reactor.

Compared to state-of-the-art simulation tools used for the design of gyrotron resonators the Particle-in-Cell method does not use any significant physically simplifications w.r.t. the modelling of the particle-field-interaction, the geometry and the wave-spectrum. Hence, it is the method of choice for validation of current simulation tools being restricted by these simplifications. So far, the Particle-in-Cell method was restricted to be used for demonstration calculations only, because of its huge computational demand and the thereby resulting long calculation time. Using the presented high-order discontinuous Galerkin Particle-in-Cell scheme on high-performance-computers, this thesis demonstrates for the first time that full-wave and transient research- and design-simulations of gyrotron resonators with high mode-indices can be efficiently performed.

For benchmark issues the developed discontinuous Galerkin Particle-in-Cell scheme is verified with a 30 GHz resonant cavity and the results are compared to the SELFT code, which is a state-of-the-art design code for resonators. The discontinuous Galerkin Particle-in-Cell scheme is used to simulate the resonator of the 1 MW, 140 GHz, $\mathrm{TE}_{28,8}$-mode gyrotron, used for plasma heating of the Wendelstein 7-X fusion-reactor.

Due to the huge number of degrees of freedom and particles, this type of simulation can only be performed on high-performance-computers with enough memory and computational power. Hence, the discontinuous Galerkin Particle-in-Cell code is improved by a new parallelization approach for the high-order shape-function deposition method on unstructured grids, allowing for a high-order coupling between the particles and the electromagnetic field.

To further improve the discontinuous Galerkin Particle-in-Cell code, a multi-rate time-stepping method, based on an Adams-Bashforth approach, for the hyperbolic divergence cleaning is developed and verified. This new approach considers the different time scales occurring in the hyperbolic divergence cleaning allowing for a more efficient time-stepping-algorithm then standard time-stepping-schemes. A rule for the construction of arbitrary-order multi-rate time-stepping methods has been derived.

The presented simulations provide new physical insights to the complex particle-field-interaction appearing in gyrotrons. The discontinuous Galerkin Particle-in-Cell scheme can be used to analyse transient phenomena, such as beam-miss-alignment, mode competition or interaction appearing at the end of the resonator, so called after-cavity-interactions. Those phenomena cannot or can only hardly be simulated with current design codes. Furthermore, this work demonstrates the ability of the discontinuous Galerkin Particle-in-Cell code to simulate the coupled resonator and quasi-optical antenna (launcher) of a gyrotron. With current codes these components could only be simulated separately, neglecting the complex interactions at the resonator output port and the influence of the electron hollow beam in the launcher. The high-order discontinuous Galerkin Particle-in-Cell simulations enable a holistic analysis of the interaction in a gyrotron. Furthermore, they pave the road to a complete simulation of a gyrotron, considering all its components, with a high-order discontinuous Galerkin Particle-in-Cell scheme - as presented here - on future high-performance-computers.

1. Introduction

The introduction is organized as follows:

- First, the state-of-the-art in the central topics of this thesis is presented.
- Second, a brief overview to the objectives is presented.
- Finally, the thesis structure is given.

1.1. State of the Research in Relevant Topics

1.1.1. Numerical Plasma-Physics via the Particle-in-Cell Method

Numerical plasma simulation has a history reaching back to the 1950s. A central role in the computational plasma physics plays the Particle-in-Cell (PIC) method, which has been first mentioned by [Harlow, 1955]. The PIC method is a coupled Lagrangian and Eulerian framework to solve the Maxwell-Vlasov system (a detailed description will be given in section 3.1). Since then, plasma codes have been developed to simulate phenomena of laboratory plasma [Dawson, 1962, Armstrong, 1967, Morse and Nielson, 1969, Langdon and Birdsall, 1970].

A very popular PIC scheme was proposed by [Boris, 1970]. Based on the Leapfrog particle pusher, it is still building the foundation of state-of-the-art PIC codes [Goplen et al., 1995, Blahovec et al., 2000, VORPAL, 2012].

The two textbooks by [Hockney and Eastwood, 1981] and [Birdsall and Langdon, 1991] are standard works in the field of numerical plasma physics using the PIC method. All four authors are pioneers from the 1960s to 1970s era of the PIC method development.

Restricted to Cartesian grids due to the Finite-Difference-method-based space discretization [Yee, 1966], the PIC method did not establish as a design tool for complex plasma devices in the beginning. Instead, it was used for demonstration calculations for complex plasma phenomena and laboratory plasma, such as the Weibel instability [Morse and Nielson, 1971]. PIC codes for unstructured grids, enabling to simulate complex geometries, have been developed later in the 1990s based on new discretization methods, such as the Finite Volume (FV) method [Munz et al., 1999].

Still, the simulation of complex plasma devices is a challenging task for state-of-the-art PIC codes. Recent studies with the commercial codes VORPAL [Lin and Smithe, 2012], MAGIC [Kumar et al., 2010] or CST [Ashutosh and Jain, 2012] prove this rating. These examples are still demonstration calculations. The runtime for the computations is far out of the amount of time that can be spend for design and research calculations.

Non-commercial PIC codes - mainly being developed in US military research laboratories - which are specialized on certain topics have been successfully used for design related tasks. After 20 years of code development, verification and validation, the ICEPIC code [Blahovec et al., 2000] is meanwhile used as design and optimization tool for magnetron devices [Peterkin Jr. et al., 2012]. The Aleph PIC-DSMC code at the Sandia National Labs, with a similar development history as ICEPIC, performs 3D arc breakdown simulations of a Child-Langmuir diode with billions of particles and elements [Hopkins et al., 2012]. Another notable area of application, where PIC codes are used in a more design related manner, is the prediction of solar events and their effects on satellites, also known as space weather [Lapenta, 2012].

For the sake of completeness it has to be noted that the PIC method is only applied when the continuity assumption for a plasma is not satisfied and the behavior of a plasma can only be described by the most general kinetic model of a plasma, i.e. the Valsov equation [Vlasov, 1945]. For a plasma that is dense enough to be treated as a conducting fluid (e.g. solar plasma, fusion plasma, high-pressure plasma) the magnetohydrodynamics (MHD) model is used to describe the physical behavior [Gurnett and Bhattacharjee, 2005]. The model combines the equations of fluid dynamics, i.e. Euler or Navier-Stokes equations, with Maxwell's equations of electrodynamics. Computational methods for the MHD equations are based on the same ideas as for the computational fluid dynamics. In theory, for a denser plasma also the PIC method can be applied, but practically it is not used due to the growing computational demand for a growing density.

1.1.2. High Performance Computing in the Context of the Particle-in-Cell Method

High Performance Computing (HPC) is the utilization of the growing computational capacities of supercomputers, such as the Cray XE6 "Hermit" of the

High-Performance-Computing-Centre Stuttgart (HLRS), Germany. While in the 20^{th} century the CPU clock rate was continuously growing and thereby enhanced the execution speed of the codes, in the last decade the clock rate did not grow significantly. Instead the number of CPUs increased, since it became cheaper to produce them. Following the prediction of Moore's Law [Moore, 1965] current supercomputers double their computational power each 18-20 month by doubling the number of CPUs. Therefore, SOTA PIC codes have to utilize this power. In the past decade, mainly the distributed memory parallelization by MPI [MPI-Forum, 2009] was used. Notable MPI parallelizations of PIC codes are ICEPIC [Bettencourt and Greenwood, 2008], VORPAL [VORPAL, 2012] or our DG-PIC code [Neudorfer et al., 2011].

Nowadays shared memory systems become more and more relevant. This is due to the architecture of the supercomputer clusters. At a certain amount of CPUs, it is reasonable to share the memory between a small number of CPUs, e.g. 4,8,16 or 32 CPU cores. This enhances the communication between the cores. A consequence of this evolution is the decreasing memory per CPU core. An established shared memory parallelization standard is OpenMP [Chapman et al., 2007]. The most radical shared memory parallelization can be achieved by graphic processing units (GPUs). There, several hundred cores share the same memory. Also for the PIC scheme GPU implementations exist [Burau et al., 2010].

All mentioned parallelization approaches point out the necessity of parallelization for utilizing todays super-computing resources. A good overview of the topic of parallelization of PIC codes is given in [Neudorfer, 2012].

1.1.3. High-Order Methods in Numerical Plasma Simulation

The development of an efficient PIC code is not only a question of utilizing the growing computational capabilities provided by supercomputer clusters, but also a question of using new and more efficient numerical methods. Within this thesis, we focus on the high-order discontinuous Galerkin (DG) method, motivated by its merits in the field of computational fluid dynamics [Cockburn et al., 2000, Hesthaven and Warburton, 2008]. The DG method is a Finite Element type (FE), high-order method combining the advantages of the FV approach (i.e. its locality) and the FE approach (i.e. the high-order polynomial representation of the state within an element). The DG approach was introduced for the PIC framework by [Jacobs and Hesthaven, 2006]. Since then only few groups followed the idea of using high-order methods within the PIC framework, e.g. [Campos Pinto et al., 2009] with a curl-conform FE-based

high-order PIC method, or [Candel et al., 2009] with a FE-based high-order PIC code.
The main advantages of these high-order methods are summarized in the following:

- Easy utilization of unstructured grids enabling to simulate geometrically complex devices (except for Spectral Element (SE) methods).
- Due to high-order polynomials describing the state within an element, less elements are required than with low-order approaches.
- For the DG approach only neighboring element data are required, which results in low dependencies and in a very efficient parallelization of the scheme.
- The high-order DG method possesses very good dissipation and dispersion properties [Gassner and Kopriva, 2011]. This is especially important for Maxwell's equations, since electromagnetic (EM) waves can be propagated nearly without any numerical damping.
- A high-order DG-PIC method can efficiently make use of high-order polynomial shape functions to represent particles on the computational grid [Jacobs and Hesthaven, 2006]. In [Stindl et al., 2011] it was shown that this technique can reduce the number of computational macro particles drastically, making the scheme more efficient.

1.1.4. Multi-Rate Time-Stepping Methods

To model physical phenomena in numerics, systems of partial differential equations (PDEs) or ordinary differential equations (ODEs) are used. In these systems often different physical components act on different timescales. They can differ by many orders of magnitude. Such systems are called stiff. In a standard explicit time-stepping method the timestep for all components is restricted by the maximum propagation speed due to the Courant-Friedrichs-Lewy (CFL) condition [Courant et al., 1928]. This results in a lack of efficiency, since the slow components could be stepped with a larger timestep than the fast components. Instead, it would be much more efficient to consider the specific timestep of each component. Solving such systems by stepping each component on its specific timescale is called multi-rate time integration and has been described first by [Gear, 1974]. Multi-rate linear multistep methods go back to the work of [Wells, 1982, Gear and Wells, 1984]. Pioneering work on

this subject has also been contributed by [Andrus, 1979] about ODE systems separated into subsystems.
Multi-rate methods are used in applications where stiff differential equation systems appear, e.g. electric circuit simulation [Kværnø and Rentrop, 1999], power system dynamics [Crow and Chen, 1994] and control with real-time simulation [Andrus, 1979,Gomm, 1981]. Multi-rate methods are also used to solve PDE systems. In [Engstler and Lubich, 1997] a Richardson extrapolation multi-rate approach for the smoothed particle hydrodynamics (SPH) method was introduced. An implicit-explicit (IMEX) multi-rate integration approach was used for the PIC method by [Jacobs and Hesthaven, 2009]. In [Schlegel et al., 2009] a multi-rate Runge-Kutta (MRK) method to solve an advection equation was proposed. Other MRK approaches are presented in [Andrus, 1993,Günther et al., 2001,Savcenco et al., 2007,Grote et al., 2012]. In [Bouzarth and Minion, 2010] a multi-rate spectral deferred correction method using Gauss integration points in time was introduced, to solve a system of ODEs modeling regularized stokeslets. Especially, in the context of Maxwell's equations the work of [Grote and Mitkova, 2010,Grote and Mitkova, 2011] has to be mentioned. They investigated the possibilities of local time-stepping (LTS) and multi-rate methods for DG based discretization methods w.r.t. locally refined grids.
In contrary to the classic LTS approach, which mainly focuses on the domain decomposition w.r.t. the timestep size, multi-rate schemes use a time domain decomposition. A time domain decomposition is a decomposition of the ODE/PDE system into different components, e.g. for the PIC method we have an EM field evolving with the speed of light, c_0, and particles moving with a speed much less than c_0. Here, it does not make sense to decompose the computational domain into fast and slow areas, but we have to split the equations into a fast and a slow component. In [Stock, 2009,Klöckner, 2010] a multi-rate Adams-Bashforth (MRAB) time-stepping approach for the PIC method is derived. There, a speed-up by a factor of 10 could be achieved at the cost of moderate dissipation.

1.1.5. The Gyrotron: A High-Power Millimeter Wave Source

In the field of high-power millimeter wave sources, gyrotrons play a crucial role, since other millimeter wave sources like klystrons, magnetrons, extended interaction tubes and cross-field amplifiers are restricted to low output power due to their decreasing size for millimeter- and smaller wave-lengths [Thumm, 2010]. The gyrotron utilizes the electron cyclotron maser (ECM) instability to generate high power millimeter waves. For a weakly relativistic helical electron beam in a longitudinal magnetic field, the kinetic energy of the electrons can be transferred to the electromagnetic (EM) field due to azimuthal phase bunching when the radio-frequency (RF) ω_{RF} of the EM field is slightly larger than the electron cyclotron frequency Ω_c. Todays gyrotron research focuses on frequencies between 35 and 240 GHz, with output powers from a tenth of a MW up to several MW for fusion related applications.

Gyrotrons are used as electromagnetic radiation sources with a wide range of applications. For instance, high-power micro- and millimeter wave sources are applied for electron cyclotron resonance heating, electron cyclotron current drive, stability control and diagnostic measurements of magnetically confined plasmas for energy generation by controlled thermonuclear fusion [Thumm, 2011]. Other possible fields of applications range from electron cyclotron resonance ion sources over sub-millimeter wave and THz spectroscopy to materials processing and plasma chemistry [Thumm, 2010].

1.1.6. Numerical Gyrotron Simulation

The theory for the computation of gyrotron interaction was first developed by [Vlasov et al., 1969] in Russia, later in the US by [Fliflet et al., 1982] and [Cai et al., 1992] and their coworkers in Germany [Borie and Jödicke, 1992].

These fundamentals form the basis of the SELFT [1] code by [Kern, 1996] at the Karlsruhe Institute of Technology (KIT). SELFT is a simulation code for the gyrotron interaction, based on a slow-variables approach. Currently, it is serving as a rapid design tool for gyrotron resonators at KIT and plays a central role in the design of state-of-the-art gyrotrons. SELFT gains its computational speed due to several analytical simplifications, such as:

- neglecting field components, which are assumed to have minor effects on the particle-field interaction,
- using reduced frequency spectra, which disable the code to cover unexpected interactions between modes,
- modeling of the voltage depression by a simplified model,
- artificial excitation of modes in the initial phase, and tuning the response of the system towards an expected mode.

[1] SELFT refers to “SELF consistent Time dependent”

Even though being fast, the underlying method suffers from the problem of being limited to simplifications and assumptions, which are not known to be valid for some aspects of the resonator physics, such as high taper-angles, different wall materials, an inhomogeneous magnetic field or realistic particle distributions.

In this context, also the EURIDICE code by [Avramides et al., 2012] and the MAGY code by [Botton et al., 1998] have to be mentioned, following a similar approach like the SELFT code. All mentioned codes can be classified as computational fast design codes for gyrotron cavity simulations.

The PIC method for the Vlasov-Maxwell system is a valid approach describing the entire self-consistent particle-field interaction properly enough to simulate the relevant physics of the resonator without any restrictions compared to the computationally fast codes. A drawback of the PIC approach is, that it requires huge computational demands and thus, cannot serve as a fast design tool. However, PIC simulations can give a significantly better understanding of the physics, and furthermore, can validate the analytic simplifications of codes like SELFT.

Earlier verification studies in the 1990s for SELFT were conducted with a two-and-a-half-dimensional, rotationally symmetric finite difference time domain (FDTD) based PIC code, T2hPIC, by [Illy, 1997], being restricted to rotationally symmetric modes. Another work by [Barroso et al., 1999] numerically investigated a 32 GHz $\mathrm{TE}_{0,2,1}$ mode resonator with a two-and-a-half-dimensional PIC code.

Growing computational capabilities in the past decade also allow for using three-dimensional full-wave PIC codes. Recent work on the three-dimensional simulation of a 127.5 GHz $\mathrm{TE}_{24,8}$ mode gyrotron resonator with MAGIC - a commercial FDTD-PIC code [Goplen et al., 1995] - has been performed by [Kumar et al., 2010]. Very recently a 42 GHz resonator has been simulated with the CST code [CST, 2012] by [Ashutosh and Jain, 2012], showing promising results. The Tech-X company presented at 2012 ICOPS a simulation of a 110 GHz $\mathrm{TE}_{22,8}$ gyrotron resonator [Lin and Smithe, 2012], performed with their VORPAL code [VORPAL, 2012]. All of these presented calculations can be classified as demonstration simulations, having more impact on research than on design and optimization of gyrotrons.

1.2. Objectives

The aim of this thesis is to further improve a parallel high-order discontinuous-Galerkin-method-based Particle-in-Cell (DG-PIC) code, in order to efficiently simulate large scale plasma devices, such as a resonant cavity of high-power gyrotrons for fusionplasma heating.
The key objectives within this scope can be summarized as follows:

1. development and verification of a new multi-rate time integration technique, based upon the multi-step Adams-Bashforth method, for the numerical simulation of the purely hyperbolic Maxwell equations

2. development of an efficient parallelization of the high-order shape function deposition method for the PIC approach on unstructured grids

3. verification of the high-order DG-PIC code

4. large-scale PIC computations of a gyrotron resonator on HPC clusters

5. demonstration that the parallel DG-PIC code can be used for design considerations

1.3. Outline

This thesis is organized as follows: In chapter 2 the Vlasov-Maxwell system and the gyrotron physics are presented. Chapter 3 briefly introduces the Particle-in-Cell method, the numerical treatment of Maxwell's equations and the discontinuous Galerkin approach. The second part of this chapter is dedicated to the theory and development of a new multi-rate time integration method for the purely hyperbolic Maxwell equations, and the parallelization of the shape function deposition for the PIC scheme on unstructured grids. In section 3.7 the HALO code framework, used within this thesis, and its basic structure is introduced. The verification test cases for the DG-PIC code are presented in chapter 4. The benchmark simulation and analysis of a 30 GHz resonant cavity and the simulation of the 140 GHz resonator of the gyrotron designed for the electron resonance heating of the Wendelstein 7-X stellarator, in chapter 4, are the central building blocks of the demonstration that the presented DG-PIC code can be applied to complex large-scale plasma devices. The conclusions chapter 5 finalizes this thesis and gives an outlook to future research activities emerging from this work.

2. Governing Equations

2.1. The Vlasov-Maxwell System

The most general expression of the physical behavior of a collisionless, collective interacting plasma is given by the collisionless Boltzmann equation, also known as the Vlasov equation [Vlasov, 1945],

$$\left(\frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} + \frac{\boldsymbol{F}}{m} \cdot \nabla_{\mathbf{v}}\right) f = 0, \tag{2.1}$$

where $f = f(\mathbf{x}, \boldsymbol{v}, t)$ is a probability density function, describing in time t the density of a particle species in six-dimensional phase space $(\boldsymbol{v}, \boldsymbol{x})$, and $\boldsymbol{F}$ is a force acting on the particles having the mass m and the charge q. In the scope of this thesis, we assume that no collisions between particles occur. If collisions shall be considered, additional terms have to be added on the right-hand side of equation (2.1). In a self-consistent electromagnetic system, $\boldsymbol{F}$ is defined by the Lorentz force,

$$\boldsymbol{F}_L = q\left[\boldsymbol{E} + \mathbf{v} \times \boldsymbol{B}\right], \tag{2.2}$$

where $\boldsymbol{E}(\mathbf{x}, t), \boldsymbol{B}(\mathbf{x}, t)$ are the space and time dependent electric field and magnetic induction, respectively. The EM field evolution in space and time is described by Maxwell's equations [Jackson, 1999, p.2],

$$\frac{\partial \boldsymbol{E}}{\partial t} = c_0^2 \nabla \times \boldsymbol{B} - \frac{\boldsymbol{j}}{\varepsilon_0}, \tag{2.3}$$

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla \times \boldsymbol{E}, \tag{2.4}$$

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

$$\nabla \cdot \boldsymbol{E} = \frac{\rho}{\varepsilon_0}, \tag{2.6}$$

where the charge and current densities are given by

$$\rho(\mathbf{x}, t) = q \int_{\mathbb{R}^3} f(\mathbf{x}, \mathbf{v}, t)\ d^3 v, \tag{2.7}$$

and

$$\boldsymbol{j}(\mathbf{x},t) = q \int_{\mathbb{R}^3} \mathbf{v} f(\mathbf{x},\mathbf{v},t)\ d^3v, \tag{2.8}$$

respectively, incorporating the probability density function f of Vlasov's equation (2.1). The vacuum permittivity ε_0 is related to the vacuum permeability μ_0 and the speed of light in vacuum c_0 according to $\varepsilon_0\mu_0 c_0^2 = 1$. Sometimes Maxwell's equations are defined in terms of the electric displacement field $\mathbf{D}$ and the magnetic field $\mathbf{H}$ which are connected to $\boldsymbol{E}$ and $\boldsymbol{B}$ via the electric permittivity ε_r and magnetic permeability μ_r, by

$$\mathbf{B} = \mu_r\mu_0\mathbf{H}, \tag{2.9}$$

$$\mathbf{D} = \varepsilon_r\varepsilon_0\mathbf{E}. \tag{2.10}$$

For vacuum $\varepsilon_r = \mu_r = 1$. In case of fluids, gases or solids ε_r and μ_r have the function of material constants. Maxwell's equations are a set of two different types of partial differential equations (PDEs). The first type is given by the two linear hyperbolic evolution equations (2.3) and (2.4), known as Ampère's and Faraday's law, respectively. They describe the temporal evolution of the EM field and the dynamic interplay between the electric and magnetic field. The linearity is due to the constant propagation speed of the EM waves, i.e. the speed of light c_0. The second type is given by the two divergence equations (2.5) and (2.6), known as Gauss's law for magnetism and Gauss's law of electrostatics, respectively. While (2.5) describes the absence of magnetic monopoles, (2.6) describes the static relation between the charge density and the electric field, i.e. that a charge (e.g. an electron or ion) is the source of an electric field.

Maxwell's equations are constructed in such a way that every electromagnetic field fulfills the charge conservation equation,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \boldsymbol{j} = 0, \tag{2.11}$$

which can be derived from (2.6) and (2.3). This derivation of (2.11) is explicitly given in the appendix section A.

Together equations (2.1)-(2.8) form the Vlasov-Maxwell system, yielding a self-consistent description of a collisionless electromagnetic plasma.

2.2. Gyrotron Physics

A gyrotron is a high-power millimeter wave source. A detailed description of the purpose of a gyrotron is given in section 1.1.5. Here, we deal with the physics of the gyrotron, i.e. the excitation of high-power millimeter waves.

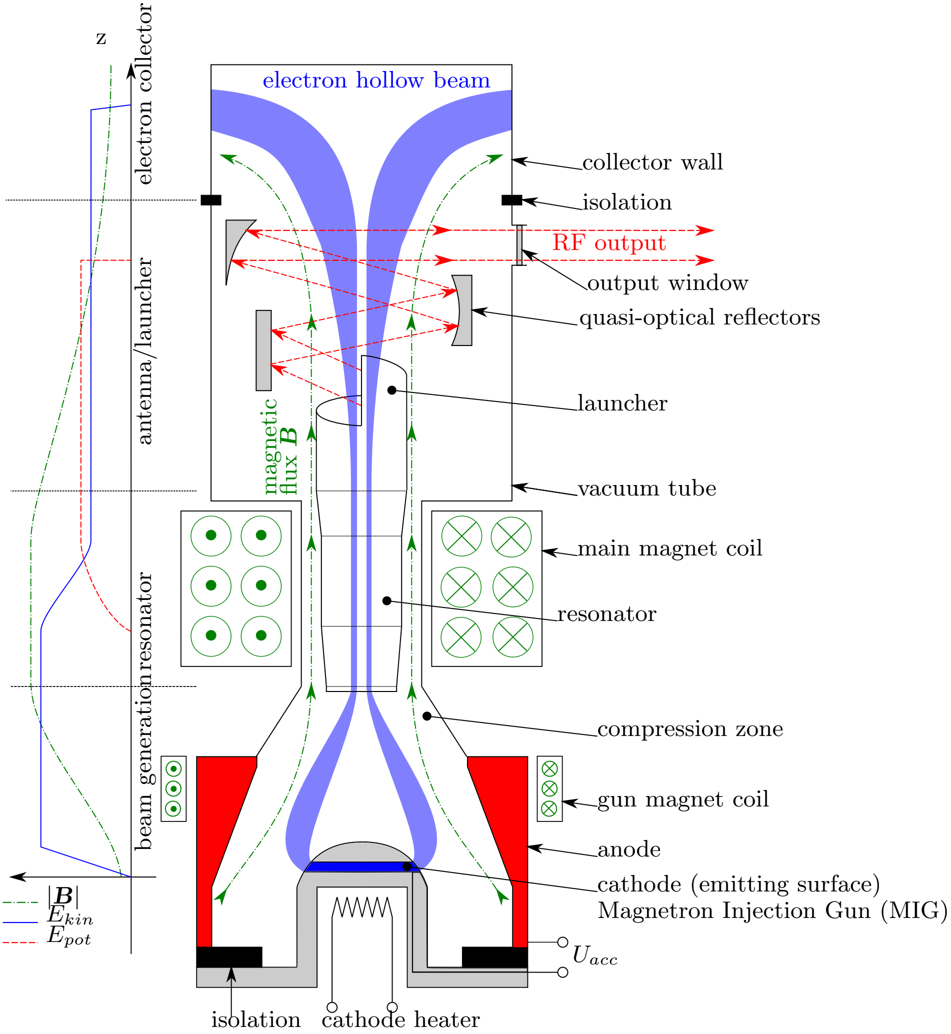


Fig. 2.1.: Schematic configuration of a gyrotron with radial RF outcoupling according to [Illy, 1997]. The diagram on the left shows simplified energy profiles of the electron beam and the RF field to illustrate the energy transfer in the resonator. The figure is from [Riedlinger, 2011].

Figure 2.1 shows the schematic drawing of a gyrotron which can be divided into four sections:

1. magnetron injection gun (MIG)
2. resonator
3. launcher antenna and output window
4. collector

From the MIG an electron hollow beam is emitted. Due to the compression of an axial magnetic field by the gun magnet coils, the electrons follows cycloidal trajectories towards the resonator. In the resonant cavity, the beam interacts with the transversal part of the TE mode in the waveguide. This interaction is based on the resonance effect, appearing if the electron cyclotron frequency Ω_c is nearly as large as the frequency ω_{RF}[1] of the EM wave. The radio-frequency ω_{RF} is depending on the geometry of the waveguide, i.e. the radius of the resonator. Hence, a specific radius favors the excitation of a certain ω_{RF} [Jackson, 1999]. The cyclotron (Lamor) frequency is defined as

$$\Omega_c = \frac{|q|B}{\gamma m}, \tag{2.12}$$

where q is the charge and m the mass of the particle, gyrating transversal to the magnetic field lines with the magnetic flux density of the magnitude B. For particles at relativistic speed v_p, the relativistic factor

$$\gamma = \frac{1}{\sqrt{1 - \frac{\mathrm{v}_p^2}{c_0^2}}}, \tag{2.13}$$

has to be recognized.

The resonance effect is called electron cyclotron maser (ECM) instability. It is a relativistic effect, requiring electrons gyrating at relativistic energy. If the particle gyrates nearly at the same frequency as ω_{RF}, they are azimuthally bunched into a certain phase. Figure 2.2 shows the situation for a gyrating electron in a transversal electric field. Since Ω_c is slightly smaller than ω_{RF} the electrons change their phase angle φ_g towards the electric field vector $\boldsymbol{E}$.

[1] here, referred as radio-frequency (RF)

The electric field decelerates and accelerates the electron at different φ_g. At $\varphi_g = 90°$ the particle is decelerated and at $\varphi_g = 180°$ it is accelerated. When it is decelerated, its relativistic factor decreases and Ω_c grows. Thus, it stays longer aligned in phase with the electric field vector. When accelerated, Ω_c gets smaller and the electron moves faster out of the phase. Considering many electrons gyrating on the circle, they will obviously accumulate in the decelerating phase.

In the decelerating phase, the electron transfers kinetic energy to the electric field and in the accelerating phase, it takes energy out of the field. Due to the bunching in the decelerating phase more energy is transfered from the electrons to the electric field, than vice versa. This results in a net energy transfer from the electron beam (kinetic energy) to the electric field (potential energy).

Figure 2.2 also reveals why mostly TE modes are excited. The electric field vector $\boldsymbol{E}$ is aligned in the rotational plane of the electrons. Hence, in the resonator only TE modes are excited.

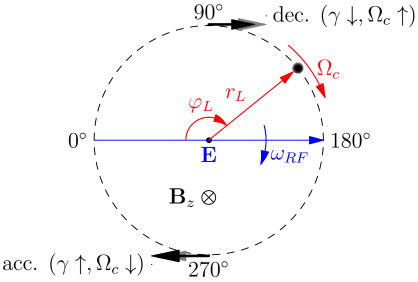

90°
dec. $(\gamma\downarrow, \Omega_c\uparrow)$
Ω_c
r_L
φ_L
0°
180°
E
ω_{RF}
$\mathbf{B}_z \otimes$
acc. $(\gamma\uparrow, \Omega_c\downarrow)$
270°

Fig. 2.2.: Relativistic gyration of an electron in a transversal electric field.

For a successful resonance and energy transfer, the resonator dimensions have to be chosen such that the required TE mode with a certain ω_{RF} can propagate. Furthermore, the magnetic field strength B_z is used to tune Ω_c of the beam. The resonance condition is only optimal in the resonator. The longer the resonator is, the more energy can be transfered. But, if the length is too large, an overbunching occurs, i.e. more electrons are accelerated and take energy out of the electric field. Thus, the length also has to be tuned, in order to achieve the maximum energy transfer.

An important detail is that only kinetic energy of the transversal electron motion ($v_\perp$) can be converted into potential field energy. Thus, it is desired to make the velocity ratio

$$\alpha = \frac{v_\perp}{v_{||}}, \tag{2.14}$$

as large as possible. This ratio is limited by the magnetic mirror effect that would reflect electrons before reaching the resonator at the end of the beam tunnel.

The entry and exit of the resonator are delimited by two conic sections, i.e. the downtaper and the uptaper, respectively. They ensure that the excited TE mode can only propagate towards the launcher. After the resonator, it is not desired that an interaction between the beam and the EM field occurs. However, it can happen that an after-cavity-interaction (ACI) appears [Zapevalov and Moiseev, 2004]. As this phenomenon is lowering the output power and efficiency of a gyrotron, it is undesired. Design optimization is trying to eliminate this effect.

After the uptaper the quasi optical launcher antenna reshapes the high-order TE mode to a low-order mode with a high Gaussian content [Flamm et al., 2011]. This lowers the losses due to diffraction when the mode is propagated out of the gyrotron through a system of mirrors. Finally the mode leaves the gyrotron through the output window, which can be a made of diamond [Thumm, 2011].

Due to the decreasing of the magnetic field strength after the resonator, the resonance condition is not matched any more in the launcher and the beam diverts. After the launcher a collector absorbs the electrons.

According to [Illy, 1997], in gyrotron design the following parameters are used to achieve an efficient setup:

- the geometry of the resonator (length and radius),
- the desired TE mode,
- the magnitude of the applied magnetic field B_z
- the beam parameter (voltage, current, velocity ratio α, beam radius).

A detailed analysis of the effects of the different parameter is given in [Kern, 1996, Illy, 1997].

3. Numerical Methods

3.1. Numerical Plasma Simulation: The Particle-in-Cell Method

A powerful method to numerically treat the non-linear Vlasov-Maxwell system in six-dimensional phase space is the Particle-in-Cell (PIC) method [Birdsall and Langdon, 1991].

The particle density function f in Vlasov's equation (2.1) is discretized in a Lagrangian manner by using particles in phase space, $\boldsymbol{x}$ and $\boldsymbol{v}$. From the Lorentz force (2.2) the equations of motion for the particles can be derived, i.e.

$$\boldsymbol{F}_L = \frac{d\boldsymbol{p}}{dt} = \frac{d\boldsymbol{m_0}\gamma\mathbf{v}}{dt} = q(\boldsymbol{E} + \mathbf{v} \times \boldsymbol{B}), \tag{3.1}$$

$$\frac{d\boldsymbol{x}}{dt} = \mathbf{v}, \tag{3.2}$$

where $\boldsymbol{p}$ is the relativistic momentum consisting of $\mathbf{v}$, m_0 and γ representing the particle speed, mass at rest and the relativistic factor (2.13), respectively.

For the sake of completeness it has to be mentioned that practically the relativistic momentum $\boldsymbol{p}$ is not used as a conservative variable. Rather the primitive variables $\mathbf{v}$ and $m_0\gamma$ are used, since not a force but an acceleration is used for describing the particle motion. Hence, the acceleration given by

$$\mathbf{a} = \frac{d\mathbf{v}}{dt} = \frac{q}{m_0\gamma}(\boldsymbol{E} + \mathbf{v} \times \boldsymbol{B}) \tag{3.3}$$

is used in the time-stepping method.

The EM field is described in an Eulerian framework, being discretized with Finite Element, Finite Difference, Finite Volume or mixed Finite Element and Finite Volume time domain methods. Here, we use the latter one, i.e. a discontinuous Galerkin method which is described in detail in section 3.3. Both frameworks are combined by particle-mesh coupling techniques. A single PIC cycle is schematically depicted in figure 3.1.

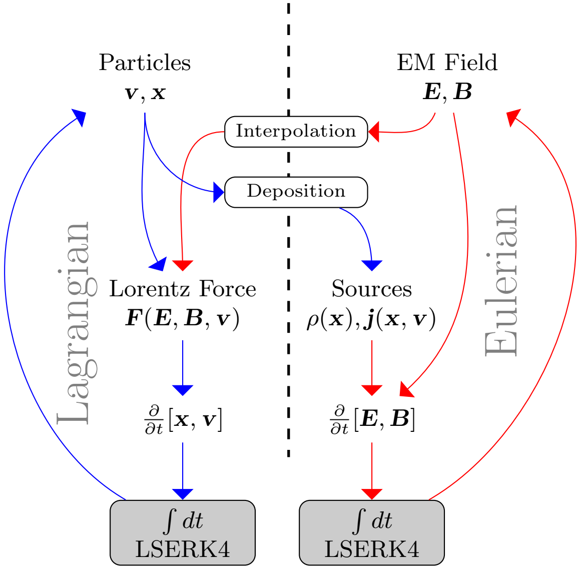

Particles
$\boldsymbol{v}, \boldsymbol{x}$
EM Field
$\boldsymbol{E}, \boldsymbol{B}$
Interpolation
Deposition
Lagrangian
Eulerian
Lorentz Force
$\boldsymbol{F}(\boldsymbol{E}, \boldsymbol{B}, \boldsymbol{v})$
Sources
$\rho(\mathbf{x}), \boldsymbol{j}(\mathbf{x}, \boldsymbol{v})$
$\frac{\partial}{\partial t}[\mathbf{x}, \boldsymbol{v}]$
$\frac{\partial}{\partial t}[\boldsymbol{E}, \boldsymbol{B}]$
$\int dt$
LSERK4
$\int dt$
LSERK4

Fig. 3.1.: Particle-in-Cell cycle.

The sources of Maxwell's equations - ρ and $\boldsymbol{j}$ - are directly derived from the phase space distribution of the particles, which in general, can consist of multiple species (e.g. electrons, ions, etc.). The mapping of the particles on the computational grid, to compute ρ and $\boldsymbol{j}$ is called deposition. For computational efficiency the particles are represented by macro-particles, i.e. one macro-particle has the mass and charge of a certain number of real particles. The factor describing this number is called macro-particle-factor (MPF). These macro-particles often are treated as clouds rather than points. The clouds are described by shape functions that have the size of the element the particle is located in. Standard approaches for the deposition are the nearest barycenter (NBC) and the volume-weighting (VW) technique [Birdsall and Langdon, 1991,Hockney and Eastwood, 1981]. Both approaches map the macro-particle to an element. In case of the NBC technique the charge is deposited on the entire grid cell without a gradient. For high-order methods this means that all interpolation points within an element have the same values, yielding a zero-order weighting of the particle. In case of the VW technique a geometrical first-order weighting is done. With the VW approach, charge conserving PIC schemes can be built [Eastwood, 1991, Villasenor and Buneman, 1992]. For unstructured high-order methods, these approaches are not suitable since they introduce noise in the polynomial description of ρ and $\boldsymbol{j}$ due to their lack of smoothness. Furthermore, the non-smooth shape of a macro-particle generated by the NBC or VW approaches enhances finite grid heating and instabilities [Birdsall and Langdon, 1991].

A high-order shape function utilizes the high-order features of the spatial discretization, i.e. a smooth polynomial description of ρ and $\boldsymbol{j}$. Within this thesis, we use a polynomial shape function suggested in [Jacobs and Hesthaven, 2009], yielding

$$S_{pol,3D}(r_p) = \frac{1}{2B_s(1.5, \alpha_s + 1)\pi R_s^3} \left[1 - \left(\frac{r_p}{R_s}\right)^2\right]^{\alpha_s}, \tag{3.4}$$

with $r_p = [0, R_s]$ for the radius of a macro-particle, and $B_s(x, y)$ denotes the beta function. $S_{pol,3D}$ has a unit integral and is cheap to evaluate. Figure 3.2 shows the idea of shape functions on unstructured grids. The cutoff radius R_s of the shape function defines which interpolation points on the grid are affected by the particle. Depending on the distance between the particle and the grid point, the shape function (3.4) gives a normalized density, which has to be multiplied with the charge of the particle. Other than the NBC and VW method, the shape function makes the particle independent of the grid. While for both other methods the particle only affects the grid cell where it is located in, the shape function projects the particle also over the element boundaries into neighboring elements.

Figure 3.3 shows the 2D projected plots of the polynomial shape function (3.4) for different α_s. The higher α_s gets, the sharper the particle is weighted around its centre. In general, it is expected that the physical behavior of a particle is represented by a delta function, but in numerical context a smoother behavior is desired to reduce the noise.

To complete the self-consistent interplay between the particles and the EM field, the Lorentz force (2.2) is computed by interpolation of the EM field at actual the particle positions. Classic interpolation approaches use the inverse of the NBC and VW method [Birdsall and Langdon, 1991]. For the NBC method all particles within an element are affected by the same EM field, e.g. the mean value or the value at the barycentre. for the high-order DG method, used in this

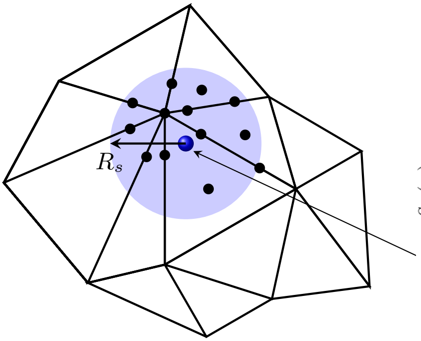


Fig. 3.2.: Particle (blue dot) with shape function influence radius R_s is deposited on the interpolation point on the computational grid (black dots).

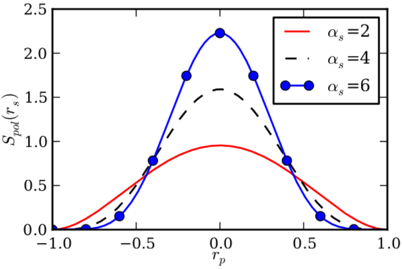


Fig. 3.3.: 2D plot of polynomial shape function (3.4) for different α_s.

thesis, we evaluate the high-order polynomial at the particle position. Together with the shape function, this yields a consistent high-order coupling between the particles and the EM field.

Finally, the particles and the EM field are advanced in time, taking into account the Lorentz force (2.2) by a time integration method, commonly the classical Leapfrog method [Boris, 1970] or higher-order Runge-Kutta methods [Jacobs and Hesthaven, 2006]. The time integration method is described in detail in section 3.4.

3.2. Numerical Treatment of Maxwell's Equations

The numerical treatment of Maxwell's equations can be divided into two categories:

- Electrostatics: Only Gauss's law of electrostatics (2.6) is solved and only the electrostatic field is considered. A numerical solver for a Poisson type equation is required. Electrostatics can be performed in one, two and three space dimensions.

- Electromagnetics: Both Ampère's and Faraday's law (2.3)-(2.4) are solved to compute the transient temporal evolution of the electric and magnetic field. Due to the curl operator, one-dimensional electromagnetic simulations are not possible. At least a two-dimensional setup is required, which again can be subdivided in two categories:
  - TE: Transverse electric setup, considering only $[E_x, E_y, B_z]$.
  - TM: Transverse magnetic setup, considering only $[B_x, B_y, E_z]$.

  For a three-dimensional setup all field components are considered, i.e. stated as full-wave property.

In this thesis, we deal with the second category in three space dimensions. If an initial EM field $[\boldsymbol{E}_0, \boldsymbol{B}_0]$ satisfies (2.5) and (2.6), and if the charge conservation (2.11) is fulfilled for $t \geq t_0$, then the temporal evolution is described by the two hyperbolic equations (2.3) and (2.4) only. Unfortunately, in the discrete case three types of numerical errors can occur making the numerical solution erroneous:

I $\nabla \cdot (\nabla \times \boldsymbol{u}) = 0$ is only satisfied up to a certain discretization error.

II $\boldsymbol{E}_0$ and $\boldsymbol{B}_0$ might not match (2.6) and (2.5), or at least up to a certain discretization error.

III The charge conservation (2.11) is violated on the discrete level. This can easily happen with time-varying sources (e.g. moving charges, as appearing in the PIC method).

To solve this problem, two classes of approaches have been proposed:

I The *projection method* (also known as Poisson correction) relies on a Helmholtz decomposition of the electric field [Birdsall and Langdon, 1991, p. 359-361]. In the frame of the standard projection approach, a Poisson equation has to be solved, which requires large computational effort and is cumbersome for the implementation on parallel platforms.

II The *generalized Lagrange multiplier* (GLM) approach uses Lagrange multipliers to incorporate the divergence constraints (2.5) and (2.6) into the

> hyperbolic evolution equations (2.3) and (2.4). There exist three different formulations, the hyperbolic-elliptic constrained formulation, proposed by [Assous et al., 1993], the hyperbolic-parabolic form, introduced by [Marder, 1987], and the purely hyperbolic Maxwell (PHM) system, introduced by [Munz et al., 2000b, Munz et al., 2000a]. The latter one is known as hyperbolic divergence cleaning.

The hyperbolic divergence cleaning (HDC) technique is a good choice w.r.t. numerical complexity, locality and parallelization, since it can be solved explicitly, and beyond that, yields qualitatively similar results to the projection method [Jacobs and Hesthaven, 2006]. It is easy to implement in an explicit discontinuous Galerkin scheme, which will be used within this thesis. The HDC is based on a purely hyperbolic extension of Maxwell's equations, called the purely hyperbolic Maxwell (PHM) equations,

$$\frac{\partial \boldsymbol{E}}{\partial t} = c_0^2 \nabla \times \boldsymbol{B} - \chi c_0^2 \nabla \Psi - \frac{\boldsymbol{j}}{\varepsilon_0}, \tag{3.5}$$

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla \times \boldsymbol{E} - \kappa \nabla \Phi, \tag{3.6}$$

$$\frac{\partial \Phi}{\partial t} = -\kappa c_0^2 \nabla \cdot \boldsymbol{B}, \tag{3.7}$$

$$\frac{\partial \Psi}{\partial t} = \chi \left(-\nabla \cdot \boldsymbol{E} + \frac{\rho}{\varepsilon_0} \right), \tag{3.8}$$

where the Lagrange multipliers Φ and Ψ are introduced as correctional scalar potential fields, having the magnitude of the arising error in the divergence constraints (2.5)-(2.6). These scalar potentials possess the eigenvalues $[\chi, \kappa] \times c_0$ with $[\chi, \kappa] \geq 1$ [Munz et al., 2000a], indicating that the introduced artificial correcting waves are faster than c_0. Such a proceeding is a numerically motivated artificial construction [Lee et al., 2004]. Note, that in the following $\kappa = \chi$, and we only will refer to χ. Theoretically, it can happen that $\chi \neq \kappa$, but in practical applications this is not done.

Figure 3.4 shows the principle of the hyperbolic divergence cleaning for Ampère's law. Upon the divergence error afflicted electric field wave I, the correcting wave II is superimposed. The resulting wave III is divergence free.

The big advantage of the HDC technique is that besides the automatic enforcement of the divergence constraints, the charge conservation is satisfied. This free feature often has to be enforced separately at high computational costs [Campos Pinto et al., 2009]. Furthermore, we consider the electrostatic

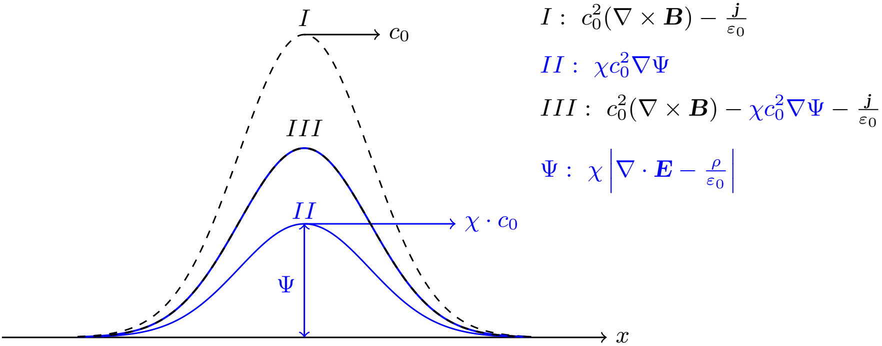

I
c_0
III
II
$\chi \cdot c_0$
Ψ
x
$I: \; c_0^2(\nabla \times \boldsymbol{B}) - \frac{\boldsymbol{j}}{\varepsilon_0}$
$II: \; \chi c_0^2 \nabla \Psi$
$III: \; c_0^2(\nabla \times \boldsymbol{B}) - \chi c_0^2 \nabla \Psi - \frac{\boldsymbol{j}}{\varepsilon_0}$
$\Psi: \; \chi \left| \nabla \cdot \boldsymbol{E} - \frac{\rho}{\varepsilon_0} \right|$

Fig. 3.4.: Hyperbolic divergence cleaning for the electric field.

field component, giving the solution of the electrostatic part of Maxwell's equations as well.

In discrete case the HDC technique suffers the problem that errors computed in step n will be subtracted from the EM field in step $n+1$. This is due to the fact that the evolution of the scalar potentials Φ and Ψ first computes the error and then grows to the required value in the proceeding timestep. This can cause instabilities in the numerical scheme, since too high or too low damping is applied. In [Jacobs and Hesthaven, 2006] a damping approach is suggested, yielding

$$\frac{\partial \Psi}{\partial t} = \chi \left(-\nabla \cdot \boldsymbol{E} + \frac{\rho}{\varepsilon_0} \right) - \varepsilon_d \Psi, \tag{3.9}$$

where $\varepsilon_d \ll 1$. We use another approach, i.e. a damping of the scalar potentials before their gradients are computed, yielding

$$\frac{\partial \boldsymbol{E}}{\partial t} = c_0^2 \nabla \times \boldsymbol{B} - \chi c_0^2 \nabla \left(\varepsilon_d \Psi \right) - \frac{\boldsymbol{j}}{\varepsilon_0}, \tag{3.10}$$

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla \times \boldsymbol{E} - \kappa \nabla \left(\varepsilon_d \Phi \right), \tag{3.11}$$

where $\varepsilon_d = [0.98, 0.99] \in \mathbb{R}$ has shown sufficient damping to stabilize the scheme.
Without considering the divergence constraints in Maxwell's equations and only using Ampère's and Faraday's law, the results become erroneous and unstable. In section 4.1.3 the effect of the hyperbolic divergence cleaning is investigated in detail.

3.3. Space Discretization: The Discontinuous Galerkin Method

In this section the high-order discontinuous Galerkin (DG) method used for the spatial discretization of Maxwell's equations will be explained. First, a brief overview on space discretization in the PIC context explains the peculiarity of our high-order DG-PIC approach. Then, a brief introduction to interpolation basics is given, in order to pave the road for a derivation of the variational form of the used DG method.

3.3.1. Space Discretization in Context of the Particle-in-Cell Method

The most commonly used space discretization method for PIC schemes is the FD-based method introduced by [Yee, 1966]. Since in the early years of the computational plasma physics only basic plasma phenomena, such as instabilities [Melrose, 1989, Morse and Nielson, 1971] were analysed by PIC codes, there was no relevance for unstructured grids. Mainly, electrostatics solvers where used. Later, when computational capabilities grew, the applications became more complex and also transient electromagnetic codes became relevant. Even today commercial codes like MAGIC [Goplen et al., 1995] or VORPAL [VORPAL, 2012] use the FD approach. The very powerful research code ICEPIC [Blahovec et al., 2000] is a classical FDTD code, based on structured Cartesian grids allowing to simulate complex geometries by approximating boundaries by step functions. Implicit PIC codes [Chen et al., 2011, Lapenta, 2012] also make use of FD discretization.
With the growing computational capabilities and the prevailing of FE methods in other computational physics domains (e.g. fluid dynamics, structural mechanics), also PIC codes made use of FE methods to solve Maxwell's equations [Bui et al., 2005, Campos Pinto et al., 2009, Candel et al., 2009]. When using FE methods for the space discretization of Maxwell's equations, one has

to choose the function spaces properly - known as the *exact sequence* [Sonnendrücker, 2011] - in order to preserve the specific features of Maxwell's equations at the discrete level. This problem can be avoided by using discontinuous Galerkin (DG) methods. Allowing for discontinuities between elements, the function space is already chosen properly to preserve the specific features of Maxwell's equations [Hesthaven and Warburton, 2002]. The DG method has earned its merits in the computational fluid dynamics (CFD) community at the beginning of the 21^{st} century [Cockburn et al., 2000, Hesthaven and Warburton, 2008], where it meanwhile is a state-of-the-art method for high-order CFD applications. It is strongly related to Spectral and Spectral Element (SE) methods [Hesthaven et al., 2007, Kopriva, 2009], which do not yet play an important role in the PIC context. In [Jacobs and Hesthaven, 2006] a PIC scheme based on a nodal DG approach was introduced first. In this context, also Finite Volume (FV) methods for the PIC scheme has to be mentioned [Munz et al., 1999]. Since the DG approach combines the advantages of the FE and the FV approaches in a perfect manner, also developments on the FV discretization for Maxwell's equations have influenced the work of DG-PIC methods (e.g. appropriate Riemann solvers). While the FV method requires non-local stencils over multiple neighboring elements to achieve high-order information, the DG method uses a high-order polynomial within an element and only requires the information of the neighboring elements. Due to this locality the DG method possesses outstanding parallelization capabilities with nearly perfect scaling on SOTA supercopmuters [Neudorfer, 2012]. Besides the high-order accuracy discretization, the DG approach possesses very good dispersion and dissipation properties, especially, for linear PDEs [Gassner and Kopriva, 2011], such as the PHM system.

3.3.2. Interpolation Issues

In the DG method the computational domain is divided into k non-overlapping elements. These elements generally are triangles or quadrangles, in two space dimensions, and tetrahedrons, hexahedrons, pyramids or prisms in three space dimensions. Formulations for other polyhedrons exist, but play no crucial role in the following context. In each element k the solution $\boldsymbol{u}(\boldsymbol{x},t), \boldsymbol{x} \in \mathbb{R}^3$, is approximated by an interpolation function,

$$\boldsymbol{u}(\boldsymbol{x},t) \approx \boldsymbol{u}_h(\boldsymbol{x},t). \tag{3.12}$$

Two common interpolation ansatz functions are used:

- For the *modal* ansatz the approximate solution is represented via a modal basis function $\{\phi_i(\mathbf{x})\}|_{i=1,N_m}$, yielding

$$\boldsymbol{u}_h(\mathbf{x},t) = \sum_{i=1}^{N_m} \hat{u}_i(t)\phi_i(\mathbf{x}), \tag{3.13}$$

  where in three space dimensions $N_m = (p+1)(p+2)(p+3)/6$ degrees of freedom (DOF) are required to approximate a polynomial of degree p [Gassner et al., 2009]. This ansatz is called *modal* since the physical interpretation of each $\{\phi_i\}$ term is the eigenmodes of the basis function, e.g. $\phi_i = \sin(\frac{ik}{\pi})$. Each DOF tells us how much a certain eigenmode contributes to the approximating function.

- For the *nodal* approach, the approximate solution is represented via a Lagrange basis function $\{L_i(\mathbf{x})\}|_{i=1,N_n}$, yielding

$$\boldsymbol{u}_h(\mathbf{x},t) = \sum_{i=1}^{N_n} \tilde{u}(\mathbf{x}_i,t)L_i(\mathbf{x}), \tag{3.14}$$

  where N_n is the number of DOF. The Lagrange basis has the property $L_i(x_j) = \delta_{i,j}$, i.e. the DOF $\tilde{\mathbf{u}}(\mathbf{x}_i,t)$ is the solution at the nodal interpolation points (IP) $\mathbf{x}_i$. That means that the polynomial has not to be evaluated at the interpolation points, since the DOF are already the physical state. While for a modal basis N_m does not differ for different type of elements (hexahedron, tetrahedron, etc.), for a nodal basis N_n depends on the type of the element. Furthermore, N_n depends on the quality of the interpolation, which can be expressed by the Lebesgue constant, and the choice of the nodal interpolation points $\mathbf{x}_i$. In [Gassner et al., 2009] a detailed analysis on the choice of $\mathbf{x}_i$ and N_n w.r.t. the Lebesgue constant is given.

The connection between the modal and the nodal DOF is given by

$$\mathcal{V}\hat{\mathbf{u}} = \tilde{\mathbf{u}} \quad , \text{ where } \ \mathcal{V}_{i,j} = \phi_j(\mathbf{x}_i) \tag{3.15}$$

is the Vandermonde matrix.
Note, that in general $N_n \neq N_m$. In the special case of the tetrahedron $N_n = N_m$, for our choice of $\mathbf{x}_i$. Table 3.1 lists up N_m and N_n for different element types in our DG scheme. In the following often the spatial (convergence) order $N = p + 1$ will be mentioned. It will be stated that a N^{th}-order scheme has

			N_n			
p	N	N_m	hexahedron	tetrahedron	pyramid	prism
1	2	4	8	4	5	6
2	3	10	26	10	14	18
3	4	20	56	20	30	39
4	5	35	99	35	55	72
5	6	56	160	56	91	120
6	7	84	244	84	140	185
7	8	120	352	120	204	270
8	9	165	485	156	285	378
9	10	220	648	220	385	511

Tab. 3.1.: Number of modal and nodal DOF for different types of elements using a polynomial of the degree p.

been used, i.e. that a polynomial with the degree $p = N - 1$ has been used. In the appendix section C the interpolation points are shown for different element types at different polynomial degrees.

Even though, commonly a nodal DG by [Hesthaven and Warburton, 2008] is used, also modal DG methods are known, such as the ADER DG formulation by Dumbser [Taube et al., 2009]. Within this thesis, a mixed nodal and modal approach, as presented in [Gassner et al., 2009], is used. Hence, we introduced the above basis functions. In the following section we will derive the variational form for the PHM system. There, also the details of the mixed nodal and modal scheme are presented and a discussion on the advantages is given.

3.3.3. The Discontinuous Galerkin Approach for the Purely Hyperbolic Maxwell Equations

The PHM equations (3.5)-(3.8) can be expressed in compact form as,

$$\frac{\partial \boldsymbol{u}}{\partial t} + \sum_{\beta=1}^{3} \mathbb{K}_\beta \frac{\partial \boldsymbol{u}}{\partial x_\beta} = \boldsymbol{g}, \tag{3.16}$$

where $\boldsymbol{u}$ is the solution, $\mathbb{K}_\beta$ is the advection flux matrix (sometimes referred to as physical flux) and $\boldsymbol{g}$ is the source term. Details on the rewriting into the compact form (3.16) is given in the appendix section B.1. We approximate the solution $\boldsymbol{u} \approx \boldsymbol{u}_h$ and introduce the residual

$$\boldsymbol{R}_h = \frac{\partial \boldsymbol{u}_h}{\partial t} + \sum_{\beta=1}^{3} \mathbb{K}_\beta \frac{\partial \boldsymbol{u}_h}{\partial x_\beta} - \boldsymbol{g}_h. \tag{3.17}$$

The Galerkin approach is one way in FE methods how to vanish $\boldsymbol{R}_h$. For this purpose we introduce the test function $\psi_i(\mathbf{x})|_{i=1,N_q}$ which is a member of the same function space as the space of the basis function of $\boldsymbol{u}_h$. N_q is equal to the number of basis functions used for the ansatz function. The Galerkin approach requires that the residual is orthogonal to all test functions, yielding that the projection of the residual on the test function is zero, i.e.

$$\int \boldsymbol{R}_h \ \psi_i \ d\mathbf{x} = 0. \tag{3.18}$$

This leads to N_q equations for N_q unknowns.
Performing the projection of the residual on the test function in (3.17), we retrieve $i = 1, N_q$ equations of the form

$$\int_k \frac{\partial \boldsymbol{u}_h}{\partial t} \psi_i d\mathbf{x} + \int_k \sum_{\beta=1}^{3} \mathbb{K}_\beta \frac{\partial \boldsymbol{u}_h}{\partial x_\beta} \psi_i d\mathbf{x} = \int_k \boldsymbol{g}_h \psi_i \ d\mathbf{x}. \tag{3.19}$$

In order to shift the spatial derivative from the interpolating function onto the test function, we perform an integration by parts, yielding the weak form

$$\int_k \frac{\partial \boldsymbol{u}_h}{\partial t} \psi_i \ d\mathbf{x} + \int_{\partial k} \boldsymbol{F}_h(\boldsymbol{u}_i, \boldsymbol{u}_e, \mathbf{n}) \ \psi_i \ ds - \int_k \sum_{\beta=1}^{3} \mathbb{K}_\beta \boldsymbol{u}_h \frac{\partial \psi_i}{\partial x_\beta} \ d\mathbf{x} = \int_k \boldsymbol{g}_h \psi_i \ d\mathbf{x}, \tag{3.20}$$

where $\boldsymbol{F}_h$ denotes the (numerical) intercell flux, depending on the interior state $\boldsymbol{u}_i$ and the exterior state $\boldsymbol{u}_e$ and the outward pointing normal vector $\boldsymbol{n}$ on the surface of the element.
We do not require any continuity for the interfaces. This results in a Riemann problem (RP) for the intercell flux calculation [Toro, 1997]. Due to the linearity of Maxwell's equations, the Riemann problem can be solved at low costs by a flux-vector splitting approach. A detailed derivation of this Riemann solver can be found in the appendix section B.2. For non-linear Riemann problems, e.g. for the Euler or the Navier-Stokes equations, iterative or approximative Riemann solvers are used. These approximative Riemann solvers can be used

for Maxwell's equations as well, e.g. the Lax-Friedrichs Riemann solver [Jacobs and Hesthaven, 2006].

Very closely related to the solution of the RP for the intercell flux, is the treatment of the boundary conditions (BCs). A detailed description of the BC treatment can be found in the appendix section B.3.

In this thesis a modal DG scheme with nodal integration is used, which is described in detail in [Gassner et al., 2009]. The mixed nodal and modal scheme has the advantage that for the evaluation of the integrals, i.e. physical and numerical fluxes, the values of the nodal solution at the IPs $\boldsymbol{x}_i$ directly can be used. For a pure modal variant a conversion between nodal and modal DOF via the Vandermonde matrix (3.15) is required. Furthermore, the modal approach reduces the number of DOF per grid cell to N_m, compared to N_n for the nodal variant. Another advantage of this mixed approach is that we can compute the Gauss integration, which is required to evaluate the integral expressions appearing in the DG form, exactly and store it in an initial phase of the simulation. In short, the mixed nodal and modal approach combines the advantages of both basis functions, which are:

- advantage of a modal basis: less DOF than for a nodal approach,
- advantage of a nodal basis: quadrature-free approach, since the Gauss integration can be initially pre-computed.

The derivation and implementation of the DG scheme is not part of this thesis. In this sense, we use the DG code framework described in section 3.7 as an "off-the-shelf" module where only the PHM equations have to be implemented into, i.e. the Riemann solver, the physical flux and the boundary conditions. However, in the appendix section D an overview to the discrete DG scheme for the PHM equations is given.

3.4. Time Discretization

In the previous section, the space discretization was briefly presented. In this section, we discuss the time discretization methods used for advancing the EM field and particles in time. First, we review the EM field evolution in time by a high-order Runge-Kutta method, followed by a discussion on the timestep choice. Then, we discuss the consistent time integration of the particles and the EM field in the PIC scheme. This section is closed with a brief discussion on an alternative single-step, Taylor-series-based, high-order time-integration approach for the PIC method.

3.4.1. Temporal Evolution of the EM Field

Since the PHM system is a system of hyperbolic evolution equations, the temporal discretization has to be considered. From equation (D.8) we obtain a set of ordinary differential equations (ODE), which can be solved with any ODE solver. The DG method stability properties restrict us to high-order time discretization methods. Usually a third- or fourth-order Runge-Kutta method is used for the DG scheme, e.g. the RKDG schemes by [Cockburn et al., 2000, Cockburn and Shu, 2001, Hesthaven and Warburton, 2008].
Within this thesis, a fourth-order low-storage explicit Runge-Kutta (LSERK4) scheme from [Carpenter and Kennedy, 1994] is used for the temporal evolution of the EM fields. It is a five-stage, two-register scheme starting with $\boldsymbol{u}_0$ at t_n and ending with $\boldsymbol{u}_5$ at t_{n+1}. The algorithm is

$$\begin{aligned} \mathbf{w}_i &= o_i \mathbf{w}_{i-1} + \Delta t f(t_{i-1}, \boldsymbol{u}_{i-1}), \\ \boldsymbol{u}_i &= \boldsymbol{u}_{i-1} + g_i \mathbf{w}_i \end{aligned} \tag{3.21}$$

for the stage $i = 1, \cdots, 5$, where $\boldsymbol{u}_0 = \boldsymbol{u}_n$. The scheme is self-starting since $o_1 = 0$. Thus w_0 can also be set 0. The substage time is given by $t_i = t_n + s_i \Delta t$, which is artificial and has no physical meaning. This is also the reason why temporal exact boundary conditions can only be prescribed first-order accurate [Carpenter et al., 1993]. The coefficients o_i, g_i, s_i are given in table appendix F.1. The important feature of this scheme is the fact that only two registers are used, i.e. that only one additional state-vector $\mathbf{w}_i$ has to be stored besides $\boldsymbol{u}_i$. This makes the scheme very efficient w.r.t. the memory demand, and besides, it has a larger stability region than a classic 4^{th}-order Runge-Kutta scheme (see [Hesthaven and Warburton, 2008, p. 94]).

3.4.2. Timestep Calculation

In the PIC scheme the timestep calculation is influenced by two factors, i.e. the propagation velocity of the EM field, c_0, and the speed of the particles. No particle will exceed the speed of light. Therefore, the dominating velocity is c_0. In the PHM equations additionally the divergence correcting component with the propagation speed χc_0 (with $\chi \geq 1$) has to be considered. For an explicit time integration method the stable timestep for the PHM system is given by

$$\Delta t = \mathrm{CFL} \frac{|\Delta \boldsymbol{x}|}{|\chi c_0|}, \tag{3.22}$$

where $|\Delta \boldsymbol{x}|$ is a measure for the spatial dimension of the smallest grid cell and the CFL factor is the Courant-Friedrichs-Lewy number [Courant et al., 1928], which is a margin factor chosen < 1 for an explicit scheme. In three dimensions and on unstructured grids $|\Delta \boldsymbol{x}|$ has no standard formula to be calculated with. It is open to still ongoing research what measure is best for calculating the most reasonable value [Hesthaven and Warburton, 2008, Hindenlang, 2008].
For DG methods also a $\frac{1}{p^2}$ dependency for Δt exists [Hesthaven and Warburton, 2008, p. 97], where p is the degree of the interpolating polynomial. Additionally, the timestep is reduced by a geometrical factor, depending on how non-uniform the grid is, which can be considered as a part of the measure $|\Delta \boldsymbol{x}|$. Together, these factors decrease the timestep drastically with increasing spatial order. In our scheme, we use pre-computed factors for different spatial orders. These coefficients have been pre-computed by the analysis of the eigenvalues of the DG operator of a linear scalar advection equation.

3.4.3. Temporal Coupling of the Particles and the Electromagnetic Field

Figure 3.5 shows a schematic workflow diagram of the multistage LSERK4 scheme. On each substep the DG scheme is used to evaluate $f(t_i, \boldsymbol{u}_i)$, in order to achieve the new stage $\boldsymbol{u_i}$. Furthermore, in the PIC context the particles need to be localized and the charge and current densities have to be calculated as source terms for the PHM equations. The particles also have to be advanced in time to get their new position and velocity on the substages. The only consistent way to do this, is to advance the particles with the same LSERK4 scheme. Thereby, we guarantee a consistent coupling of the particles and the EM field in time. The advantage is that we use a high-order scheme for the particles as well. The disadvantage is that we need to calculate ρ and $\boldsymbol{j}$ on every substage, which can be very costly depending on the number of particles and the complexity of the computational domain.

3.4.4. Particle Treatment in Time

Besides the EM field, also the particles have to be discretized in time. Equations (3.1) and (3.2) describe their behavior in time. A typical method used within the PIC community is the time-centred explicit Leapfrog scheme of 2^{nd}-order [Birdsall and Langdon, 1991]. It is energy-conserving, which is very important for the particle treatment. Mainly it is applied together with a finite difference Yee scheme [Yee, 1966], known as Boris scheme [Boris, 1970]. The peculiarity of

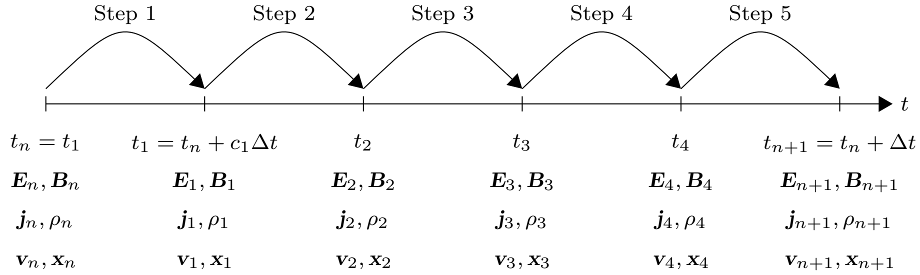

Step 1
Step 2
Step 3
Step 4
Step 5
t
$t_n = t_1$
$t_1 = t_n + c_1\Delta t$
t_2
t_3
t_4
$t_{n+1} = t_n + \Delta t$
$\boldsymbol{E}_n, \boldsymbol{B}_n$
$\boldsymbol{E}_1, \boldsymbol{B}_1$
$\boldsymbol{E}_2, \boldsymbol{B}_2$
$\boldsymbol{E}_3, \boldsymbol{B}_3$
$\boldsymbol{E}_4, \boldsymbol{B}_4$
$\boldsymbol{E}_{n+1}, \boldsymbol{B}_{n+1}$
$\boldsymbol{j}_n, \rho_n$
$\boldsymbol{j}_1, \rho_1$
$\boldsymbol{j}_2, \rho_2$
$\boldsymbol{j}_3, \rho_3$
$\boldsymbol{j}_4, \rho_4$
$\boldsymbol{j}_{n+1}, \rho_{n+1}$
$\mathbf{v}_n, \mathbf{x}_n$
$\mathbf{v}_1, \mathbf{x}_1$
$\mathbf{v}_2, \mathbf{x}_2$
$\mathbf{v}_3, \mathbf{x}_3$
$\mathbf{v}_4, \mathbf{x}_4$
$\mathbf{v}_{n+1}, \mathbf{x}_{n+1}$

Fig. 3.5.: Schematic workflow of the five stage and 4^{th}-order Runge-Kutta scheme.

the Leapfrog scheme is that the $\mathbf{x}$ and $\boldsymbol{v}$ coordinate of the particles are shifted half a timestep to each other. Figure 3.6 shows a schematic drawing of the Leapfrog scheme workflow and the substages involved when using the LSERK4 scheme for the EM field. Since $\boldsymbol{x}$ and $\boldsymbol{v}$ are only updated once per timestep, consequently ρ and $\boldsymbol{j}$ can only be updated once. This means that ρ and $\boldsymbol{j}$ are constant during the LSERK4 substages, which leads to inconsistencies with the high-order field evolution. Furthermore, we do not have $\boldsymbol{E}$ and $\boldsymbol{B}$ at $t_{n+1/2}$, making it hard to calculate the Lorentz force. These reasons make it impossible to use the LSERK4 scheme and the Leapfrog scheme consistently together. In our case the only alternative for a consistent particle and field discretization in time is to use the LSERK4 scheme for both the EM field and the particles, as described in section 3.4.3.

3.4.5. An Alternative High-Order Time Discretization Approach for the Particle-in-Cell Method

In this section we review an alternative single-step, high-order time discretization method based on a Taylor-series-expansion (TSE) and its applicability to the PIC method.

The arbitrary high-order derivatives (ADER) approach has been successfully applied to Maxwell's equations [Taube et al., 2009]. It is a TSE-based coupled space-time approach which cannot be expressed in the sense of the method of lines. Another Taylor-series-based approach, which potentially can be used for the PHM system, has been proposed by [Lörcher et al., 2007], i.e. the space-

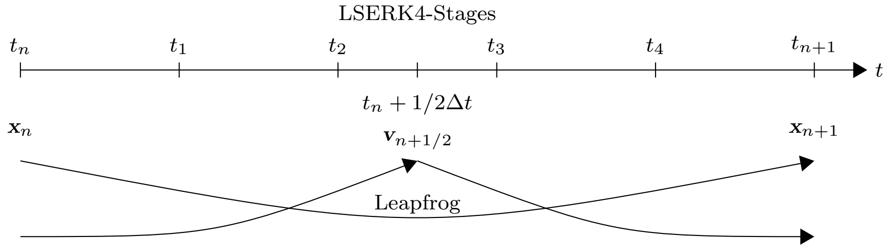


Fig. 3.6.: Schematic workflow of the Leapfrog scheme for the particles in addition with the LSERK4 scheme for the EM field.

time-extension (STE) DG method. Both approaches are true one-step methods, which gain their high-order ability by high-order spatial derivatives, being used as time derivatives in the sense of the Cauchy-Kovalesvskaya (CK) procedure. They only differ in the inter-cell-flux-coupling. While the ADER approach uses a generalized Riemann problem to derive higher-order derivatives of the inter-cell-fluxes, the STE approach uses a Gauss quadrature in time, where at each Gauss point a RP is solved. The state at the Gauss points is derived from a Taylor-series, only using cell-interior information.

The main idea of these schemes is a space-time-Taylor-series-extension of the EM field. A consistent coupled time integration approach for the particles should therefore also base on such a TSE approach. In [Quandt, 2010] a high-order TSE-based particle time-integration method has been developed. Unfortunately no coupling between the particles and the EM field in a self-consistent way has been derived. Instead, only examples with a prescribed EM field have been studied, where no self-consistent interplay between the particles and the EM field took place.

Therefore, in the following we shall describe a self-consistent PIC cycle based on a TSE. We start with the TSE of the EM field. The RHS of the EM field depends on the sources ρ and $\boldsymbol{j}$. Hence, the particle position and velocity have an influence on the EM field. On the other hand the particle motion directly depends on the EM field due to the Lorentz force (2.2). Explicitly said: To gain a time derivative of the particle state, we require time derivatives of the EM field (see [Quandt, 2010, p. 35-43]). For time derivatives of the EM field we need time derivatives of the particle state, in order to compute the influence of the source terms. Obviously, the TSE of the EM field and the particle state depend in such a way on each other that we have an implicit dependency loop. It is a puzzling task to formulate the entire self consistent loop for an arbitrary-order TSE approach. For incorporating the source terms we refer to [Campos Pinto et al., 2009] where a TSE-based time integration of the source terms is described. Together with [Taube et al., 2009] and [Quandt, 2010] it seems to be possible to gain a self-consistent, but implicit, TSE-based time stepping scheme. In the scope of this thesis, this topic will not be investigate deeper and the realization will be postponed to future work.

3.5. A Multi-Rate Adams-Bashforth Approach for the Hyperbolic Divergence Cleaning Technique

We aim to increase the efficiency of the hyperbolic divergence cleaning method by applying a multi-rate time-stepping method proposed by [Gear and Wells, 1984] and further developed in [Stock, 2009, Klöckner, 2010]. An overview to the SOTA in multi-rate time-stepping methods is given in section 1.1.4.
In the following sections a multi-rate Adams-Bashforth (MRAB) approach for the PHM equations is derived. First a brief review of the basic principles of the multistep Adams-Bashforth time integration method is given. Then, we present the operator splitting considered for the MRAB approach of the PHM equations, an algorithm for the implementation will be discussed, and the derived scheme will be verified by an example, having an analytic solution for Maxwell's equations. This is followed by an assessment of the results and the efficiency of the method, compared to the LSERK4 method, standard used within our DG framework.

3.5.1. Review of the Multistep Adams-Bashforth Method

The linear first-order ODE,

$$y'(t) = f(y(t), t), \tag{3.23}$$

can be solved by the separation of variables, yielding

$$y(t_i + \Delta t) = y(t_i) + \int_{t_i}^{t_i+\Delta t} f(y(t), t)dt, \tag{3.24}$$

where Δt denotes a certain timestep.

The temporal integral on the right side can numerically be solved by various integration methods, e.g. using the left Riemann sum, results in Euler's method,

$$y(t_i + \Delta t) = y(t_i) + \Delta t \; f(y(t_i), t_i). \tag{3.25}$$

A more accurate method can be formulated by approximating f with an interpolation polynomial $p_n(t)$ of n^{th}-degree, resulting in

$$y(t_i + \Delta t) = y(t_i) + \int_{t_i}^{t_i+\Delta t} p_n(t) dt. \tag{3.26}$$

To build p_n we can only use values of f which are already known, i.e. values of previous timesteps. Hence we require,

$$p_n(t_{i-j}) = f(t_{i-j}), \quad \text{for} \;\; j = 0, 1, \cdots, n. \tag{3.27}$$

Using the Lagrange basis polynomials

$$L(t_j) = \prod_{\substack{0 \leq m \leq k \\ m \neq j}} \frac{t - t_m}{t_j - t_m}, \tag{3.28}$$

we can write p_n as

$$p_n = \sum_{j=0}^{n} f(t_{i-j}) L(t_j). \tag{3.29}$$

The resulting polynomial can be integrated over the interval $[t_i, t_i + \Delta t]$. For arbitrary degree n we can pre-compute coefficients $a_j|_{j=0,n}$ requiring

$$\int_{t_i}^{t_i+\Delta t} p_n(t) dt = \Delta t \left[a_{i-n} f(t_{i-n}) + a_{i-n+1} f(t_{i-n+1}) + \cdots + a_{i-1} f(t_{i-1}) + a_i f(t_i) \right]. \tag{3.30}$$

The coefficients a_j for different degrees n are listed in table F.2. With the coefficients a_j we can write a n^{th}-order integration scheme, yielding

$$y(t_i + \Delta t) = y(t_i) + \Delta t \sum_{j=0}^{n} a_j f(t_{i-j}). \tag{3.31}$$

This type of multistep scheme is called Adams-Bashforth (AB) method, which was first mentioned in [Bashforth, 1883]. Figure 3.7 shows the basic principle of the AB method for a 2^{nd}-order interpolation polynomial, which is the extrapolation of p_2 towards $t_i + \Delta t$ and the integration over the timestep Δt.

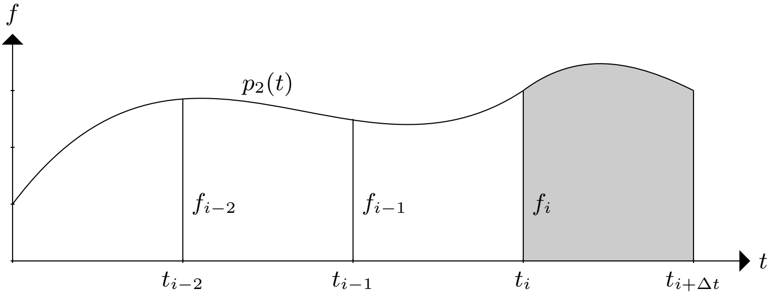


Fig. 3.7.: 2^{nd}-order AB method with three sampling points at t_i, t_{i-1}, t_{i-2} and the integration of the interpolating polynomial $p_2(t)$ over the interval $[t_i, t_{i+\Delta t}]$.

3.5.2. An Operator Splitting Approach for the PHM Equations

The following idea is based on the operator splitting approach proposed in [Munz et al., 2000b] and extended by the temporal integration part based upon the MRAB approach.

The PHM equations (3.5)-(3.8) form a hyperbolic system, where the EM wave is propagating with the speed of light, c_0, and the divergence cleaning component is propagating with χc_0. The timestep size is restricted by the largest eigenvalue of the system, i.e. $|\chi c_0|$, with $\chi > 1$. Due to the CFL condition, the entire equation may only be advanced stable in time when the timestep is restricted by this eigenvalue.

For our multi-rate scheme we consider an operator (field) splitting approach based on a right-hand-side (RHS) decomposition. The RHS of (3.5) and (3.6) can be split into:

1. a divergence error afflicted component, given by the native Maxwell equations (NME), yielding
$$\frac{\partial \boldsymbol{E}_{NME}}{\partial t} = c_0^2 \nabla \times \boldsymbol{B} - \frac{\boldsymbol{j}}{\varepsilon_0}, \tag{3.32}$$
$$\frac{\partial \boldsymbol{B}_{NME}}{\partial t} = -\nabla \times \boldsymbol{E}, \tag{3.33}$$

2. and a divergence correcting component, i.e. the artificial correcting waves (ACW),
$$\frac{\partial \boldsymbol{E}_{ACW}}{\partial t} = -\chi c_0^2 \nabla \phi, \tag{3.34}$$
$$\frac{\partial \boldsymbol{B}_{ACW}}{\partial t} = -\kappa \nabla \psi. \tag{3.35}$$
$$\tag{3.36}$$

The superposition of these two components results in the corrected Maxwell equations (CME) ,

$$\frac{\partial \boldsymbol{E}_{CME}}{\partial t} = \frac{\partial \boldsymbol{E}_{NME}}{\partial t} + \frac{\partial \boldsymbol{E}_{ACW}}{\partial t}, \tag{3.37}$$
$$\frac{\partial \boldsymbol{B}_{CME}}{\partial t} = \frac{\partial \boldsymbol{B}_{NME}}{\partial t} + \frac{\partial \boldsymbol{B}_{ACW}}{\partial t}, \tag{3.38}$$

with the state vector

$$\boldsymbol{u}_{CME} = [E_{x,CME}, E_{y,CME}, E_{z,CME}, B_{x,CME}, B_{y,CME}, B_{z,CME}]^T. \tag{3.39}$$

Furthermore, we can express $\boldsymbol{u}_{CME}$ as a composition of these two components, i.e.

$$\boldsymbol{u}_{CME} = \boldsymbol{u}_{NME} + \boldsymbol{u}_{ACW}. \tag{3.40}$$

To complete the set of the PHM equations we label equations (3.7) and (3.8) as the Lagrange multiplier (LM) components with the state

$$\boldsymbol{u}_{LM} = [\Phi, \Psi]^T. \tag{3.41}$$

Expressing the RHS in terms of functions, we can show the dependencies on the variables (3.39) and (3.41), as following:

$$\frac{\partial \boldsymbol{u}_{CME}}{\partial t} = \boldsymbol{f}_{NME}(\boldsymbol{u}_{CME}) + \boldsymbol{f}_{ACW}(\boldsymbol{u}_{CME}, \boldsymbol{u}_{LM}), \tag{3.42}$$

$$\frac{\partial \boldsymbol{u}_{LM}}{\partial t} = \boldsymbol{f}_{LM}(\boldsymbol{u}_{CME}, \boldsymbol{u}_{LM}), \tag{3.43}$$

where

$$\boldsymbol{f}_{NME} = \begin{pmatrix} c_0^2 \nabla \times \boldsymbol{B}_{CME} \\ -\nabla \times \boldsymbol{E}_{CME} \end{pmatrix}, \boldsymbol{f}_{ACW} = \begin{pmatrix} -\chi c_0^2 \nabla \Phi \\ -\kappa \nabla \Psi \end{pmatrix}, \tag{3.44}$$

$$\text{and} \quad \boldsymbol{f}_{LM} = \begin{pmatrix} -\kappa c_0^2 \nabla \cdot \boldsymbol{B}_{CME}, \\ \chi \left(-\nabla \cdot \boldsymbol{E}_{CME} + \frac{\rho}{\varepsilon_0} \right) \end{pmatrix}. \tag{3.45}$$

Note, that we always use the corrected state $\boldsymbol{u}_{CME}$ as input for the RHS when the EM field is required. We do not use the states $\boldsymbol{u}_{ACW}$ or $\boldsymbol{u}_{NME}$, but only the RHS expressions. This is not very intuitive, but in the following we shall see that this issue will become much more reasonable.

Equations (3.42)-(3.43) are still the PHM equations (3.5)-(3.8), but decomposed in a form that is suitable for the following MRAB time-stepping approach.

3.5.3. Time Discretization with a Multi-Rate Adams-Bashforth Method

In this section, the fundamental building blocks, required to build a MRAB scheme for the PHM equations, are presented.

We split the RHS as described in the previous section and integrate each part at its specific timescale. The step-ratio between the fast and slow components is given by the integer multiple of the correction speed, i.e.

$$h \cdot \chi = H, \tag{3.46}$$

where h and H is the small and the large timestep, respectively. During a large timestep H, the correcting wave (ACW) has to be stepped χ times, while the NME part only needs to be stepped once. The following timesteps are used for the different RHSs:

- ACW: small timestep h

- NME: large timestep H
- LM: small timestep h
- CME: composition of fast and slow RHS; small timestep h

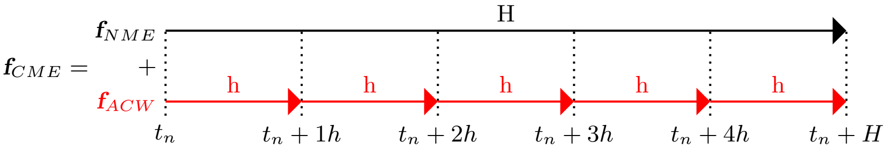


Fig. 3.8.: Two-rate PHM system with fast component (ACW) and slow component (NME) for $\chi = 5$.

Figure 3.8 shows the different timescales for the two RHSs for $\chi = 5$. In (3.42) we defined that always $\mathbf{u}_{CME}$ is the input of the RHS functions and that $\mathbf{f}_{CME}$ is a composition of $\mathbf{f}_{ACW}$ and $\mathbf{f}_{NME}$. In this sense the time update of $\mathbf{u}_{CME}$ can be written as a composition of the integrals over $\mathbf{f}_{ACW}$ and $\mathbf{f}_{NME}$ at its specific timescale, yielding

$$\mathbf{u}_{CME}|_{t_{n+k\cdot h}} = \mathbf{u}_{CME}|_{t_n+(k-1)\cdot h} + \int_{t_n}^{t_{n+H}} \mathbf{f}_{CME}\ dt \tag{3.47}$$

$$= \mathbf{u}_{CME}|_{t_n+(k-1)\cdot h} + \int_{t_n}^{t_{n+\chi h}} \mathbf{f}_{ACW}\ dt + \int_{t_n}^{t_{n+H}} \mathbf{f}_{NME}\ dt. \tag{3.48}$$

Using the MRAB approach, we can write the discrete form of the integral, yielding

$$\mathbf{u}_{CME}|_{t_{n+k\cdot h}} = \mathbf{u}_{CME}|_{t_n+(k-1)\cdot h} + h\sum_{i=0}^{N_{AB}} a_{i,k} \cdot h(ACW)_i + H\sum_{i=0}^{N_{AB}} b_{i,k} \cdot h(NME)_i, \tag{3.49}$$

for $k = 1, \chi$,

where $a_{i,k}$ are the classical AB coefficients and $b_{i,k}$ are special interpolation coefficients for a dedicated set of RHS history points in time; $h(ACW)_i$ and $h(NME)_i$ denote the $\boldsymbol{f}_{ACW}$ and $\boldsymbol{f}_{NME}$ history, respectively. The coefficients $b_{i,k}$ are special coefficients, which can be calculated - in a quite general manner - as described in the appendix section E. Since they have to be calculated for every set of χ and N_{AB} (and even every substage) we skip to explicitly specify them here. Instead, we refer to a flexible code environment, which allows to pre-compute these coefficients for every special set of χ and N_{AB}.
For the update of $\boldsymbol{u}_{LM}$ we can use a standard AB approach, advancing $\boldsymbol{u}_{LM}$ on the small timescale,

$$\boldsymbol{u}_{LM}|_{t_{n+k\cdot h}} = \boldsymbol{u}_{LM}|_{t_n+(k-1)\cdot h} + h\sum_{i=0}^{N_{AB}} a_{i,k}\cdot h(LM)_i, \tag{3.50}$$

where $a_{i,k}$, again, are the classical AB coefficients for the order N_{AB} and $h(LM)_i$ is the history of $\boldsymbol{f}_{LM}$. For the evaluation of $\boldsymbol{f}_{LM}$ additionally ρ has to be provided on substep level. For the PIC scheme an evaluation of ρ on substep level would be very expensive. As a consequence, the particles would have to be stepped with the timestep h, which makes the procedure even more costly. Alternatively, when ρ is evaluated only on the large timescale, it can be extrapolated onto substep level in the same manner as $\boldsymbol{u}_{NME}$, yielding

$$\rho_{t_n+k\cdot h} = \sum_{i=1}^{N_{AB}} d_{i,k}h(\rho)_i \quad \text{for} \quad k=1,\ldots,\chi, \tag{3.51}$$

where $d_{i,k}$ are extrapolation coefficients and $h(\rho)_i$ is the history of ρ. The corrected state $\boldsymbol{u}_{CME}$ will be used in $\boldsymbol{f}_{LM}$, as well as in $\boldsymbol{f}_{ACW}$ on substep level. The following histories are required:

- $h(NME), h(\rho)$: updated on large timescale H,
- $h(ACW), h(LM)$: updated on small timescale h.

With these definitions, we are able to build a MRAB scheme for the RHS decomposed PHM system.

3.5.4. Algorithm and Implementation

In this section the algorithm for the multi-rate scheme will be presented. We use a 2^{nd}-order approximation in time and a correction speed of $2c_0$, i.e. $\chi = 2$.

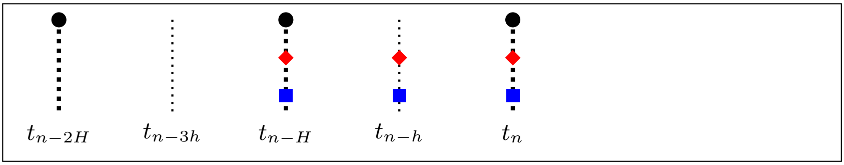
To make it easier to understand the different steps of the algorithm we use a diagram type that has been developed in [Stock, 2009, Klöckner, 2010].
A key feature of these diagrams is a color and symbol key used for the histories shown below.

● $h(NME)$: Native Maxwell Equations

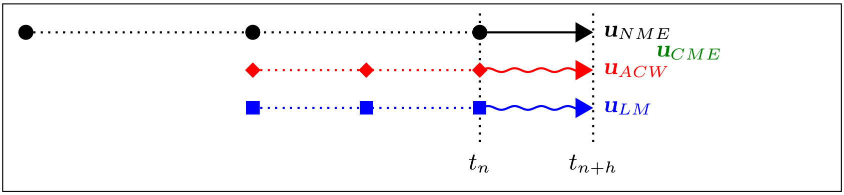
◆ $h(ACW)$: Artificial Correction Waves

■ $h(LM)$: Lagrange Multipliers

The initial setup of the histories is as following,



The histories indicate also the points in time where the RHS is evaluated. In a first step all states are extrapolated to t_{n+h},



yielding

$$\begin{aligned}
\mathbf{u}_{CME}|_{t_{n+h}} &= \mathbf{u}_{CME}|_{t_n} + \int_{t_n}^{t_{n+h}} f_{ACW}\ dt + \int_{t_n}^{t_{n+h}} f_{NME}\ dt, && (3.52)\\
&= \mathbf{u}_{CME}|_{t_n} + h\sum_{i=0}^{2} a_i\ h(ACW)_i + H\sum_{i=0}^{2} b_i\ h(NME)_i, && (3.53)
\end{aligned}$$

$$\text{and } \mathbf{u}_{LM}|_{t_{n+h}} = \mathbf{u}_{LM}|_{t_n} + \int_{t_n}^{t_{n+h}} f_{LM}\ dt = \mathbf{u}_{LM}|_{t_n} + h\sum_{i=0}^{2} a_i\ h(LM)_i, \tag{3.54}$$

where a_i are classic AB coefficients and b_i are special coefficients, which are pre-computed for this special setting in order to save time during execution (for b_i see table F.3, substep 1). Figure 3.9 shows the extrapolation and integration for $\mathbf{f}_{NME}$ in this step. The "natural" timestep of this component is H but it is only integrated over a small step h. Therefore, special coefficients b_i have to be computed.

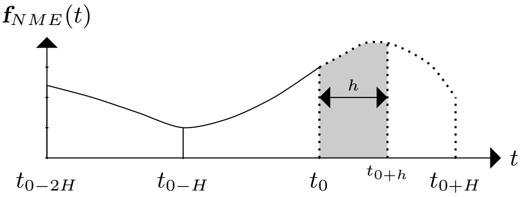


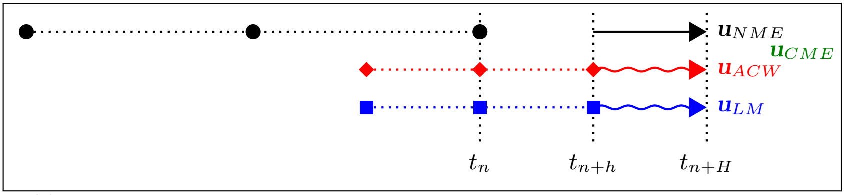
Fig. 3.9.: Extrapolation and integration of $\mathbf{f}_{NME}$ in the first substep.

With the new state at t_{n+h} we can compute the RHS functions,

$$\mathbf{f}_{ACW}(\mathbf{u}_{CME}, \mathbf{u}_{LM})|_{t_{n+h}}, \tag{3.55}$$

$$\mathbf{f}_{LM}(\mathbf{u}_{CME}, \mathbf{u}_{LM})|_{t_{n+h}}, \tag{3.56}$$

and update the histories $h(ACW)$ and $h(LM)$. In the next step, we integrate the state to the final stage t_{n+H},



yielding

$$\mathbf{u}_{CME}|_{t_{n+H}} = \mathbf{u}_{CME}|_{t_{n+h}} + \int_{t_{n+h}}^{t_{n+H}} \mathbf{f}_{ACW}\ dt + \int_{t_{n+h}}^{t_{n+H}} \mathbf{f}_{NME}\ dt, \tag{3.57}$$

$$= \mathbf{u}_{CME}|_{t_{n+h}} + h\sum_{i=0}^{2} a_i\ h(ACW)_i + H\sum_{i=0}^{2} b_i\ h(NME)_i, \tag{3.58}$$

$$\mathbf{u}_{LM}|_{t_{n+H}} = \mathbf{u}_{LM}|_{t_{n+h}} + \int_{t_{n+h}}^{t_{n+H}} \mathbf{f}_{LM}\ dt = \mathbf{u}_{LM}|_{t_{n+h}} + h\sum_{i=0}^{2} a_i\ h(LM)_i, \tag{3.59}$$

where again b_i are special coefficients suited for the situation shown in figure 3.10 (for b_i see table F.3, substep 2) and a_i are classical AB coefficients.

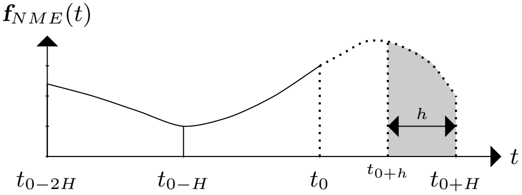


Fig. 3.10.: Extrapolation and integration of $\boldsymbol{f}_{NME}$ in the final substep.

Finally, the RHSs are computed at t_{n+H},

$$\boldsymbol{f}_{ACW}(\boldsymbol{u}_{CME}, \boldsymbol{u}_{LM})|_{t_{n+H}}, \tag{3.60}$$

$$\boldsymbol{f}_{LM}(\boldsymbol{u}_{CME}, \boldsymbol{u}_{LM})|_{t_{n+H}}, \tag{3.61}$$

$$\boldsymbol{f}_{NME}(\boldsymbol{u}_{CME})|_{t_{n+H}}, \tag{3.62}$$

and the histories $h(ACW)$, $h(LM)$ and $h(NME)$ are updated. Algorithm 1 summarizes the entire MRAB-PHM scheme for arbitrary N_{AB} and χ in a pseudo code environment.

Algorithm 1 Time domain decomposed MRAB-PHM time-stepping algorithm for arbitrary N_{AB} and χ.

```
By some start-up method, obtain:
- $\boldsymbol{u}_{CME}|_{t_n}$ and $\boldsymbol{u}_{LM}|_{t_n}$
- start-up history: $h_{NME}$, $h_{ACW}$, $h_{LM}$
———— Start Multi-Rate Method ————
for step $i = 1$ to $\chi$ do
    $\boldsymbol{u}_{CME}|_{t_{n+i\cdot h}} = \boldsymbol{u}_{CME}|_{t_{n+(i-1)\cdot h}} + \int_{t_{n+(i-1)\cdot h}}^{t_{n+i\cdot h}} \boldsymbol{f}_{NME} dt + \int_{t_{n+(i-1)\cdot h}}^{t_{n+i\cdot h}} \boldsymbol{f}_{ACW} dt$
    $\boldsymbol{u}_{LM}|_{t_{n+i\cdot h}} = \boldsymbol{u}_{LM}|_{t_{n+(i-1)\cdot h}} + \int_{t_{n+(i-1)\cdot h}}^{t_{n+i\cdot h}} \boldsymbol{f}_{LM} dt$
    Evaluate $\boldsymbol{f}_{ACW}(\boldsymbol{u}_{CME}, \boldsymbol{u}_{LM})|_{t_{n+i\cdot h}}$ and $\boldsymbol{f}_{LM}(\boldsymbol{u}_{CME}, \boldsymbol{u}_{LM})|_{t_{n+i\cdot h}}$
end for
Evaluate $\boldsymbol{f}_{NME}(\boldsymbol{u}_{CME}, \boldsymbol{u}_{LM})|_{t_{n+H}}$
```

Note, that even if the histories $h(ACW)$ and $h(NME)$ are used, the states $\boldsymbol{u}_{ACW}$ and $\boldsymbol{u}_{NME}$ are not stored explicitly. Instead, they are directly added up to $\boldsymbol{u}_{CMS}$. This is important w.r.t. a low memory consumption.

The presented algorithm can be recast to arbitrary N_{AB} and χ. It is fully consistent w.r.t. the approximation order in time, which has been tested with an equivalent 2×2 ODE system. The experimental order of convergence (EOC) could not be shown for the PHM system. For an ODE system using the same MRAB scheme the EOC could be verified. Why the EOC could not be shown for the PDE system can have various reasons. Also for the classical AB and LSERK4 method it was not able to gain the EOC. To find the temporal EOC in a hyperbolic equation solver is a difficult task. It has to be ensured that the error introduced by the spatial discretization is negligible. On the other hand, decreasing timesteps introduce numerical damping, which degrades the quality of the solution. However, for all time-integration schemes the EOC could be calculated with an ODE system. Hence, the implementation of the time integration scheme could be verified indirectly.

In [Klöckner, 2010], a variety of at least 20 MRAB schemes for the PIC method was found by changing the order of integration of the different components. This variety seems not to be applicable for the PHM system, since a decomposition in fast and slow variables does not work in the same way. A deeper investigation of the PHM system RHS decomposition might reveal other schemes as well but will be deferred to future work.

3.5.5. Implementation in the Discontinuous Galerkin Framework

In the last two sections the RHS decomposition and the algorithm were presented. In this section, we focus on the implementation into the DG framework. This is important since different aspects of the DG framework are affected by the RHS decomposition, i.e.

- the volume integral,
- the surface integral,
- the Riemann solver,
- and the boundary conditions.

Volume Integral: The physical flux matrices $\mathbb{K}_\beta$ (B.1)-(B.3) can be decomposed in the following way.

- *Native Maxwell Equations (NME):*

$$\mathbb{K}_1 \frac{\partial \mathbf{u}_{CME}}{\partial x} = \begin{pmatrix} 0 \\ c_0^2 \partial_x B_z \\ -c_0^2 \partial_x B_y \\ 0 \\ -\partial_x E_z \\ \partial_x E_y \end{pmatrix}, \tag{3.63}$$

$$\mathbb{K}_2 \frac{\partial \mathbf{u}_{CME}}{\partial y} = \begin{pmatrix} -c_0^2 \partial_y B_z \\ 0 \\ c_0^2 \partial_y B_x \\ \partial_y E_z \\ 0 \\ -\partial_y E_x \end{pmatrix}, \tag{3.64}$$

$$\mathbb{K}_3 \frac{\partial \mathbf{u}_{CME}}{\partial z} = \begin{pmatrix} c_0^2 \partial_z B_y \\ -c_0^2 \partial_z B_x \\ 0 \\ -\partial_z E_y \\ \partial_z E_x \\ 0 \end{pmatrix}. \tag{3.65}$$

Additionally, the current densities have to be added to the volume integral,

$$\frac{1}{\varepsilon_0} \begin{pmatrix} j_x \\ j_y \\ j_z \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{3.66}$$

This part of the volume integral is evaluated in a separate subroutine and stored into the history $h(NME)$.

- *Divergence Correcting System (DCS):* The divergence correcting system is the composition of the ACW and LM RHS. Since both parts run on the small timescale, they can be processed together. The physical flux reads,

$$\mathbb{K}_1 \frac{\partial \begin{pmatrix} \mathbf{u}_{CME} \\ \mathbf{u}_{LM} \end{pmatrix}}{\partial x} = \begin{pmatrix} c_0^2 \chi \partial_x \Phi \\ 0 \\ 0 \\ \kappa \partial_x \Psi \\ 0 \\ 0 \\ c_0^2 \kappa \partial_x B_x \\ \chi \partial_x E_x \end{pmatrix}, \tag{3.67}$$

$$\mathbb{K}_2 \frac{\partial \begin{pmatrix} \mathbf{u}_{CME} \\ \mathbf{u}_{LM} \end{pmatrix}}{\partial y} = \begin{pmatrix} 0 \\ c_0^2 \chi \partial_y \Phi \\ 0 \\ 0 \\ \kappa \partial_y \Psi \\ 0 \\ c_0^2 \kappa \partial_y B_y \\ \chi \partial_y E_y \end{pmatrix}, \tag{3.68}$$

$$\mathbb{K}_3 \frac{\partial \begin{pmatrix} \mathbf{u}_{CME} \\ \mathbf{u}_{LM} \end{pmatrix}}{\partial z} = \begin{pmatrix} 0 \\ 0 \\ c_0^2 \chi \partial_z \Phi \\ 0 \\ 0 \\ \kappa \partial_x \Psi \\ c_0^2 \kappa \partial_z B_z \\ \chi \partial_z E_z \end{pmatrix}. \tag{3.69}$$

Additionally the charge density has to be added,

$$\frac{1}{\varepsilon_0} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \tilde{\rho}\chi \end{pmatrix}, \tag{3.70}$$

where the tilde indicates that ρ has to be extrapolated. This part of the volume integral is evaluated in a separate subroutine and stored into the histories $h(ACW)$ and $h(LM)$. Note, that the upper six entries are stored in $h(ACW)$ and the last two in $h(LM)$.

Surface Integral: For the surface integral the Riemann solver has to be decomposed. We use the same flux vector splitting approach as presented in section B.2.

- *Native Maxwell Equations (NME):*
  We set $\chi = 0$ in the Riemann solver for the PHM system and take only the upper 6×6 sub-matrix of $\mathbb{A}^{\pm}$ (B.11). The solution of the RP at the surface interpolation point $\mathbf{x}_i$ is

$$\boldsymbol{u}_{NME}(\mathbf{x}_i) = (\mathbb{A}^{+}_{NME}\boldsymbol{u}_{CME,L}(\mathbf{x}_i) + \mathbb{A}^{-}_{NME}\boldsymbol{u}_{CME,R}(\mathbf{x}_i)), \tag{3.71}$$

where

$$\mathbb{A}^{+} = \begin{pmatrix} c_0(1-n_1^2) & -c_0n_1n_2 & -c_0n_1n_3 & 0 & c_0^2n_3 & -c_0^2n_2 \\ -c_0n_1n_2 & c_0(1-n_2^2) & -c_0n_2n_3 & -c_0^2n_3 & 0 & c_0^2n_1 \\ -c_0n_1n_3 & -c_0n_2n_3 & c_0(1-n_3^2) & c_0^2n_2 & -c_0^2n_1 & 0 \\ 0 & -n_3 & n_2 & c_0(1-n_1^2) & -c_0n_1n_2 & -c_0n_1n_3 \\ n_3 & 0 & -n_1 & -c_0n_1n_2 & c_0(1-n_2^2) & -c_0n_2n_3 \\ -n_2 & n_1 & 0 & -c_0n_1n_3 & -c_0n_2n_3 & c_0(1-n_3^2) \end{pmatrix}, \tag{3.72}$$

and

$$\mathbb{A}^{-} = \begin{pmatrix} -c_0(1-n_1^2) & c_0n_1n_2 & c_0n_1n_3 & 0 & c_0^2n_3 & -c_0^2n_2 \\ c_0n_1n_2 & -c_0(1-n_2^2) & c_0n_2n_3 & -c_0^2n_3 & 0 & c_0^2n_1 \\ c_0n_1n_3 & c_0n_2n_3 & -c_0(1-n_3^2) & c_0^2n_2 & -c_0^2n_1 & 0 \\ 0 & -n_3 & n_2 & -c_0(1-n_1^2) & c_0n_1n_2 & c_0n_1n_3 \\ n_3 & 0 & -n_1 & c_0n_1n_2 & -c_0(1-n_2^2) & c_0n_2n_3 \\ -n_2 & n_1 & 0 & c_0n_1n_3 & c_0n_2n_3 & -c_0(1-n_3^2) \end{pmatrix}. \tag{3.73}$$

- *Divergence Correcting Systems (DCS):*
  The RP of the divergence correcting system has the solution

$$\begin{pmatrix} \boldsymbol{u}_{CME} \\ \boldsymbol{u}_{LM} \end{pmatrix}(\mathbf{x}_i) = \left(\mathbb{A}^{+}_{DCS}\begin{pmatrix} \boldsymbol{u}_{L,CME} \\ \boldsymbol{u}_{L,LM} \end{pmatrix}(\mathbf{x}_i) + \mathbb{A}^{-}_{DCS}\begin{pmatrix} \boldsymbol{u}_{R,CME} \\ \boldsymbol{u}_{R,LM} \end{pmatrix}(\mathbf{x}_i)\right) \tag{3.74}$$

where

$$\mathbb{A}^{\pm}_{DCS} = \begin{pmatrix} \pm\mathcal{D} & \boldsymbol{0} & \begin{matrix} 0 & c_0^2n_1\chi \\ 0 & c_0^2n_2\chi \\ 0 & c_0^2n_3\chi \end{matrix} \\ \boldsymbol{0} & \pm\mathcal{D} & \begin{matrix} n_1\kappa & 0 \\ n_2\kappa & 0 \\ n_3\kappa & 0 \end{matrix} \\ \begin{matrix} 0 & 0 & 0 \\ n_1\chi & n_2\chi & n_3\chi \end{matrix} & \begin{matrix} c_0^2n_1\kappa & c_0^2n_2\kappa & c_0^2n_3\kappa \\ 0 & 0 & 0 \end{matrix} & \begin{matrix} \pm\kappa c_0 & 0 \\ 0 & \pm\chi c_0 \end{matrix} \end{pmatrix}, \tag{3.75}$$

with

$$\mathcal{D} = \begin{pmatrix} c_0\chi n_1^2 & c_0\chi n_1n_2 & c_0\chi n_1n_3 \\ c_0\chi n_1n_2 & c_0\chi n_2^2 & c_0\chi n_2n_3 \\ c_0\chi n_1n_3 & c_0\chi n_2n_3 & c_0\chi n_3^2 \end{pmatrix}. \tag{3.76}$$

Boundary Conditions: The periodic boundary condition (BC) is straightforward as the one used in a single-rate method. For an open BC, we have to split the RP and consider the different states. As absorbing BC, we use the Silver-Müller BC (see appendix section B.3.2). For the two new systems, we have to compute the solution of the RP. The procedure to compute the solution of the RP at the boundary is described in detail in section B.3. Therefore, we will only show the results of the inverse RP and derive therefrom the BC.

- *Native Maxwell System (NMS):*
  The solution of the RP reads,

$$\boldsymbol{E}_0 = \frac{1}{2}\left[2(\boldsymbol{E}_{zero}\cdot\boldsymbol{n})\boldsymbol{n} + (\boldsymbol{E}_L+\boldsymbol{E}_R) - \left[(\boldsymbol{E}_L+\boldsymbol{E}_R)\cdot\boldsymbol{n}\right]\boldsymbol{n} + c_0(\boldsymbol{B}_L-\boldsymbol{B}_R)\times\boldsymbol{n}\right], \tag{3.77}$$

$$\boldsymbol{B}_0 = \frac{1}{2c}\left[2c(\boldsymbol{B}_{zero}\cdot\boldsymbol{n})\boldsymbol{n} + c_0(\boldsymbol{B}_L+\boldsymbol{B}_R) - c_0\left[(\boldsymbol{B}_L+\boldsymbol{B}_R)\cdot\boldsymbol{n}\right]\boldsymbol{n} - (\boldsymbol{E}_L-\boldsymbol{E}_R)\times\boldsymbol{n}\right], \tag{3.78}$$

  where $\boldsymbol{E}_{zero}$ and $\boldsymbol{B}_{zero}$ on the RHS yields the state for the zero eigenvalues. Inserting (3.77) and (3.78) into the Silver-Müller condition (B.32) yields

$$[\boldsymbol{E}_R - c_0(\boldsymbol{B}_R\times\boldsymbol{n})]\times\boldsymbol{n} = [\epsilon - c_0(\beta\times\boldsymbol{n})]\times\boldsymbol{n}, \tag{3.79}$$

  where we can conclude that $\boldsymbol{E}_R = 0$ and $\boldsymbol{B}_R = 0$.

- *Divergence Correcting Systems (DCS):*
  According to the eigenvalues

$$(\lambda_1,\ldots,\lambda_8) = (-c_0\gamma, c_0\gamma, -c_0\chi, c_0\chi, 0, 0, 0, 0). \tag{3.80}$$

  the solution of the RP on the boundary reads

$$\boldsymbol{E}_0 = \frac{1}{2}\left[2(\boldsymbol{E}_{zero} - (\boldsymbol{E}_{zero}\cdot\boldsymbol{n})\boldsymbol{n}) + c_0(\Phi_L-\Phi_R)\boldsymbol{n} - ((\boldsymbol{E}_L+\boldsymbol{E}_R)\cdot\boldsymbol{n})\,\boldsymbol{n}\right], \tag{3.81}$$

$$\boldsymbol{B}_0 = \frac{1}{2c_0}\left[2c(\boldsymbol{B}_{zero} - (\boldsymbol{B}_{zero}\cdot\boldsymbol{n})\boldsymbol{n}) + (\Psi_L-\Psi_R)\boldsymbol{n} - c_0\left((\boldsymbol{B}_L+\boldsymbol{B}_R)\cdot\boldsymbol{n}\right)\boldsymbol{n}\right], \tag{3.82}$$

$$\Psi_0 = \frac{1}{2}\left[\Psi_R+\Psi_L + c_0(\boldsymbol{B}_L-\boldsymbol{B}_R)\cdot\boldsymbol{n}\right], \tag{3.83}$$

$$\Phi_0 = \frac{1}{2c_0}\left[c_0(\Phi_R+\Phi_L) + (\boldsymbol{E}_L-\boldsymbol{E}_R)\cdot\boldsymbol{n}\right], \tag{3.84}$$

where $\boldsymbol{E}_{zero}$ and $\boldsymbol{B}_{zero}$ on the RHS yields the state for the zero eigenvalues. Inserting (3.81) and (3.82) into (B.32) yields

$$[\boldsymbol{E}_{zero} - c_0(\boldsymbol{B}_{zero} \times \boldsymbol{n})] \times \boldsymbol{n} = [\epsilon - c_0(\beta \times \boldsymbol{n})] \times \boldsymbol{n}, \tag{3.85}$$

where we can conclude that $\boldsymbol{E}_{zero} = 0$ and $\boldsymbol{B}_{zero} = 0$. Due to the equality $\boldsymbol{n} \times \boldsymbol{n} = 0$, the quantities $\boldsymbol{E}_R, \boldsymbol{E}_L, \boldsymbol{B}_R$ and $\boldsymbol{B}_L$ vanish in (B.32) and only $\boldsymbol{E}_{zero} = 0$ and $\boldsymbol{B}_{zero} = 0$ is left over. This is a problem since we don't know the values for $\boldsymbol{E}_R$ and $\boldsymbol{B}_R$. But, these values are required for the solution of Lagrangian multiplier variables boundary condition, yielding

$$\Psi_R = -\Psi_L + c_0(\boldsymbol{B}_R - \boldsymbol{B}_L) \cdot \boldsymbol{n}, \tag{3.86}$$

$$\Phi_R = -\Phi_L + \frac{1}{c_0}(\boldsymbol{E}_R - \boldsymbol{E}_L) \cdot \boldsymbol{n}, \tag{3.87}$$

when setting $\Psi_0 = \Phi_0 = 0$. The main problem is that we cannot set the BCs for $\boldsymbol{E}_R$ and $\boldsymbol{B}_R$.

The analysis of the BCs for both subsystems revealed that for the correcting system no unique condition for $\boldsymbol{E}_R$ and $\boldsymbol{B}_R$ can be found. From [Munz et al., 2000a], we know that $\boldsymbol{E}_R = \boldsymbol{B}_R = 0$ for the PHM system. If the RHS decomposition is consistent, we can expect that the sum of the exterior state of both subsystems is zero. Since we know that $\boldsymbol{E}_R$ and $\boldsymbol{B}_R$ of the native system is zero, we can conclude that $\boldsymbol{E}_R$ and $\boldsymbol{B}_R$ of the divergence correcting system also have to be zero.

3.5.6. Timestep Size Determination

A crucial question for new time-discretization methods is the calculation of the stable timestep for a certain PDE. While for single-rate methods an analytical approach for calculating the stable timestep exists [Hairer et al., 2009], for multi-rate methods no established and proven theory exists. Different approaches have been discussed, e.g. recasting a PDE-problem to an ODE system with the same stiffness-behavior [Stock, 2009], or a parameterization of a 2×2 ODE system in order to analyse the behavior for a multi-rate scheme in the entire parameter domain [Klöckner, 2010]. In principle, both approaches follow the idea that an ODE system can mimic the behavior of a PDE system, such as the PHM equations or the Vlasov-Maxwell system. Nevertheless, analytical attempts to understand the stability properties of these types of multi-rate methods are not known so far.

In this investigation, none of the above mentioned approaches were applied. Instead, we use the stable timestep for the LSERK4 method and iteratively decreased the CFL number until a stable solution for the MRAB scheme has been found. This proceeding has the disadvantage that only for the simulated timespan a stable solution has been found. We cannot tell if for a longer simulation time the solution will remain stable.

3.5.7. Verification Calculations

As stated before, the MRAB scheme has been tested with an 2×2 ODE system, where the EOC could be shown. Now, we want to verify the MRAB scheme for the PHM system and to check the performance.
To verify the implementation, we use an example proposed by [Issautier et al., 1995]. Originally proposed in 2D we extended it to 3D, yielding

$$\rho(\mathbf{x},t) = \sin(\omega t)\left[\sin(\pi y) + \sin(\pi x) + \sin(\pi z)\right],$$

$$j_x(\mathbf{x},t) = (\cos(\omega t) - 1)\left[\pi\cos(\pi x) + \pi^2 x \sin(\pi z)\right] - \cos(\omega t) x \sin(\pi z), \quad (3.88)$$

$$j_y(\mathbf{x},t) = (\cos(\omega t) - 1)\left[\pi\cos(\pi y) + \pi^2 y \sin(\pi x)\right] - \cos(\omega t) y \sin(\pi x), \quad (3.89)$$

$$j_z(\mathbf{x},t) = (\cos(\omega t) - 1)\left[\pi\cos(\pi z) + \pi^2 z \sin(\pi y)\right] - \cos(\omega t) z \sin(\pi y). \quad (3.90)$$

The peculiarity of this example is that it fulfills the charge conservation (2.11), i.e.

$$\nabla \cdot \boldsymbol{j} = -\cos(\omega t)\left[\sin(\pi y) + \sin(\pi x) + \sin(\pi z)\right], \quad (3.91)$$

$$\frac{\partial \rho}{\partial t} = \cos(\omega t)\left[\sin(\pi y) + \sin(\pi x) + \sin(\pi z)\right]. \quad (3.92)$$

Since we prescribe the source terms of Maxwell's equations no particles are involved. Only numerical errors due to $\nabla \cdot (\nabla \times \boldsymbol{u}) \neq 0$ are relevant. As initial conditions we set $\boldsymbol{E}_0 = \boldsymbol{B}_0 = 0$. The domain is $1 \times 1 \times 1[m^3]$ and discretized by $10 \times 10 \times 10$ hexahedral elements. We use a 4^{th}-order space discretization. Additionally, we set $c_0 = \mu_0 = \varepsilon_0 = 1$.
Figure 3.11 shows the potential field energy,

$$E_{pot} = \underbrace{\int_V \frac{1}{2}\varepsilon_0 |\boldsymbol{E}|^2 \, dV}_{\text{Electric Field Energy}} + \underbrace{\int_V \frac{1}{2\mu_0} |\boldsymbol{B}|^2 \, dV}_{\text{Magnetic Field Energy}}, \quad (3.93)$$

over a simulation time of 10 seconds. We use $\chi = 5$ and $N_{AB} = 4$, in the following called MRAB4. The results are compared to a 4^{th}-order single-rate

Adams-Bashforth method (AB4) and a LSERK4 method (see section 3.4.1). All three time-discretization approaches achieve the same results, which verifies the correct implementation.

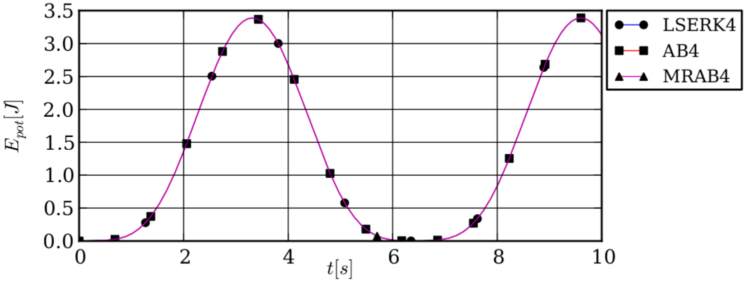


Fig. 3.11.: Potential field energy over time for a 4^{th}-order Runge-Kutta (LSERK4), an Adams-Bashforth (AB4) and a multi-rate Adams-Bashforth (MRAB4) method for $\chi = 5$.

The key question is the size of the stable timestep for the MRAB4 method. The second column of table 3.2 shows the stable timestep for the different methods. The numbers are normalized by the stable timestep of the LSERK4 method. For the MRAB method we show the large timestep H. The computations with the different χ factors were done all with the same timestep H for the MRAB method. For the AB4 and the LSERK4 method the timestep had to be decreased by the factor χ. Note, that also the runtime increases by this factor. The following columns show the runtime normalized by the runtime of the LSERK4 method. This allows directly to compare the efficiency between the MRAB4 and the LSERK4 method.

For $\chi = 1$ a single-rate method is regained and the AB and MRAB method are equal. The time-step for the AB4 and MRAB4 method is about a tenth of the one for the LSERK4 method. This is a typical behavior for AB methods. With increasing order, the stability region is decreasing and the timestep is getting smaller. For RK methods this behavior is inverse. For increasing order, the timestep is increasing as well. For $\chi = 1$ we would expect a factor of two longer runtime for the AB methods compared to the LSERK4 method. This is due to the fact that for one timestep the LSERK4 method requires 5 RHS evaluations, while the AB method would require ten RHS evaluations. But the runtimes

	Δt	runtimes				
		$\chi = 1$	$\chi = 2$	$\chi = 3$	$\chi = 5$	$\chi = 10$
RK4	1.00	1.00	1.00	1.00	1.00	1.00
AB4	0.09	3.95	4.22	4.10	3.90	3.72
MRAB4	0.09	5.04	3.82	3.43	3.00	2.85

Tab. 3.2.: Timestep size and runtimes for the Issautier example with different time-stepping methods.

for the AB4 method are even higher, which could be explained by the overhead (memory requests, subroutine calls, etc.) of the methods. The fact that the MRAB method requires even more time than the AB method supports this explanation. The MRAB method is much more overhead consumptive than a pure AB implementation. For increasing χ the MRAB method wins over the AB method and has an increasing runtime efficiency compared to the LSERK4 method. For $\chi = 10$ the MRAB method still needs about three times longer than the LSERK4 method. These results show that the expected speed-up is not achievable with the new MRAB method.

3.5.8. Conclusions and Outlook

We derived a MRAB scheme for the PHM equations for arbitrary step-ratios χ and time-order. The new MRAB scheme has been verified with an example using a manufactured source terms. Although, we could verify the correct implementation of the new MRAB scheme we could not show that it is more efficient than a RK4 method.

An essential problem when using AB methods is the decreasing timestep for increasing order. The AB schemes have very small stability regions and do therefore not compete very well with RK methods, which have increasing stability regions. To overcome this problem we could consider multi-rate Runge-Kutta methods, which have been investigated in [Schlegel et al., 2009, Grote et al., 2012]. Although, only up to third-order schemes were found, it is an interesting alternative w.r.t. the size of the stability region.

An aspect which might slow down the implementation is the memory demand and the memory access. For the AB method more memory is required, since a RHS history has to be stored. For the MRAB method the memory access pattern is getting even worse since the histories are split.

On the small timescale $\boldsymbol{f}_{LM}$ and $\boldsymbol{f}_{ACW}$ have to be evaluated which takes rather more computational demand than the evaluation of $\boldsymbol{f}_{NME}$. This is a bad ratio between both timescales, since still the most work is done on small timescale. For the MRAB-PIC scheme in [Stock, 2009] the main computational load was in the particle treatment, which was performed only on large timescale. Thus, the load was shifted to the large timescale, which dominated the runtime. For an increasing particle number, the efficiency grew. If we would use the presented MRAB-PHM scheme within a PIC method this effect would be significant.

A combination of the MRAB-PIC scheme from [Stock, 2009] and the here presented MRAB-PHM scheme is worth to be investigated. The resulting three-rate scheme would offer even more potential to save computational time, since the particles would be integrated on an even larger timescale.

It would be interesting to use Gauss points in time instead of equidistant points, as used in the AB approach. In [Bouzarth and Minion, 2010] an explicit multirate time domain decomposition technique was presented, using Gauss points and quadrature rule in time.

Another option would be to treat the correcting system with a lower order in space and time than the native Maxwell equations. Especially for hierarchical basis functions it could be quite straightforward to implement such a scheme.

The major problem of the AB method is the small stability region. It is possible to extend this region by adding further RHS evaluations [Niegemann et al., 2012]. Such an optimization of the stability region would be an option to increase the timestep size and the efficiency of the MRAB scheme.

3.6. Parallelization Approach for the Shape Function Deposition

For the simulation of engineering applications, such as gyrotrons [Thumm, 2011] or fusion reactors [Holtkamp, 2007], a parallelization of the DG-PIC scheme is essential. A full-wave, transient DG-PIC code requires too much computational effort for running these kind of applications in serial. For our DG-PIC code, we use a MPI-based [MPI-Forum, 2009] parallelization with a data and domain decomposition. The computational domain is decomposed and all particles are treated locally on the specific MPI domain. At the MPI domain boundaries the intercell flux and the particles entering and leaving the MPI domain have to be communicated. A detailed review to the particle parallelization concept of the DG-PIC code used within this thesis is given in [Neudorfer, 2012].

Note, that the following figures and explanations are made for a 2D domain due to an easier visualization and comprehension. The extension to 3D in the code is straightforward.

Here, a new parallelization approach for the deposition with the high-order shape function (3.4) has been developed. While classical nearest-barycenter or volume-weighting decomposition techniques only require information of the local MPI domain, the shape function requires communication with neighboring MPI domains. The neighboring domains need particle positions of particles that are not located in their domains in order to perform the deposition.

3.6.1. Preliminaries: Particle Localization Method and Shape Function Deposition

In section 3.1, we introduced the polynomial shape function (3.4) for the deposition of ρ and $\boldsymbol{j}$ on the computational grid. The deposition is linked very strongly with the localization method for the particles. On equidistant Cartesian meshes the localization is straightforward, but on unstructured meshes the localization can be very cumbersome. In this section we present the localization technique for the particles and the link to the deposition with the shape function.

Even if not developed within the scope of this thesis, we explain the deposition technique based on the Cartesian background mesh in detail, to pave the road for the next section. There, we present the parallelization of the deposition method for the shape function, which has been developed in the scope of this thesis.

For the localization we use a Cartesian background mesh (CBM) as shown in figure 3.12. The CBM is laid over the unstructured mesh. The localization of the particle on the CBM is straightforward. Assuming an equidistant element size in each spatial direction, which is defined as $\Delta x_i|_{i=1,3}$, we can determine the index vector $\boldsymbol{Elem}_{(x,y,z)}$ of the element where the i-th particle at the position $\mathbf{x}_{p,i} = [x_{p,i}, y_{p,i}, z_{p,i}]^T$ is located in, by

$$\boldsymbol{Elem}_{(x,y,z)} = \begin{pmatrix} (x_{p,i} - x_{min})/\Delta x_1 \\ (y_{p,i} - y_{min})/\Delta x_2 \\ (z_{p,i} - z_{min})/\Delta z_3 \end{pmatrix}, \tag{3.94}$$

where $\mathbf{x}_{min} = [x_{min}, y_{min}, z_{min}]^T$ denotes the minimum position in the computational grid. In the initialization phase the unstructured grid is mapped to the

CBM, i.e. every CBM cell has a list of unstructured elements located within it. The size Δ_{CBM} is chosen by the cutoff radius R_s of the shape function. When $R_s = \Delta_{CBM}$, we can guarantee that the entire shape function always fits into the eight surrounding CBM cells of the one CBM cell the particle is located in, i.e. a 3×3 square of CBM cells. For the deposition, in all nine CBM elements, the shape function is evaluated on the interpolation point of the unstructured elements located within these CBM cells. Due to overlapping unstructured elements between the CBM cells, the interpolation points having already been evaluated are marked, in order to avoid a doubled deposition.
This technique has proven to be very efficient and was introduced in [Stindl et al., 2011].

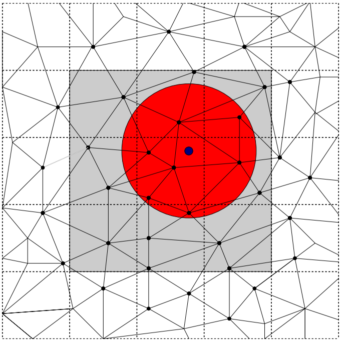


Fig. 3.12.: Unstructured grid overlaid by a Cartesian background mesh (dotted grid) to localize the particle and all elements within its shape function radius R_s. The grey shaded area is the smallest area where the entire shape function can have potentially influence on. The figure is from [Pirrung, 2010].

3.6.2. Parallelization Approach for the Shape Function Deposition

Having introduced the principles of the localization and deposition with the shape function on unstructured grids, now the parallelization approach will be derived.

Figure 3.13 shows the easiest situation where the shape function sphere of influence is reaching over the local MPI domain boundary. The particle is located in domain 1. Domain 2 is the MPI neighboring domain, where the shape function is reaching into. Figure 3.14 shows a more complex situation, where the sphere of influence is extended into several neighboring domains.

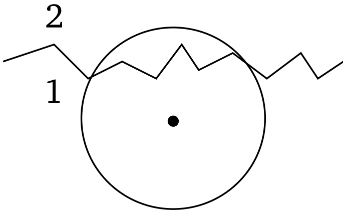


Fig. 3.13.: Shape function sphere of influence reaching over a MPI boundary. The figure is from [Pirrung, 2010].

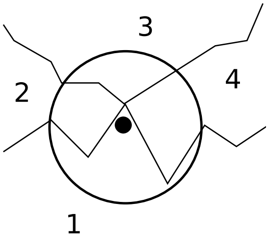


Fig. 3.14.: Shape function sphere of influence reaching over several MPI boundaries. The figure is from [Pirrung, 2010].

The problem of the shape function parallelization is that a particle not necessarily needs to pass the MPI boundary, but still has to be communicated in order to allow a deposition of it in the neighboring domain. Therefore, the parallelization of the shape function needs to be separated from the parallelization due to boundary passing of particles. To decide if the shape function reaches into neighboring MPI domains it is necessary to track the distance between the particle and the boundary. Doing this explicitly for every particle in the domain would lead to huge computational efforts. Instead we make use of the CBM introduced in section 3.6.1.

Figure 3.15 shows the two MPI domains and their boundary with an overlaid CBM. Three particles in domain 1 are located near to the boundary, but only particle one is that near to the boundary that its shape function intersects the MPI boundary with domain 2. Particle 1 and 2 are located in the same CBM cell. Potentially every particle located in this CBM cell can have a shape function sphere intersecting with the MPI boundary, but it does not have to. For particle 3 the situation is different. Definitely all particles located in this

CBM cell cannot reach the MPI boundary with its shape function. This is similar to the situation for the deposition. We consider the eight neighboring CBM cells of the one, the particle is located in. If within these CBM cells no MPI boundary is located, the shape function cannot intersect with a MPI boundary. Figure 3.16 shows all CBM cells grey shaded that potentially can host particles reaching the MPI boundary with the shape function. Note, this is from the perspective of domain 1. Of course this is a very conservative guess, but it is a safe guess.

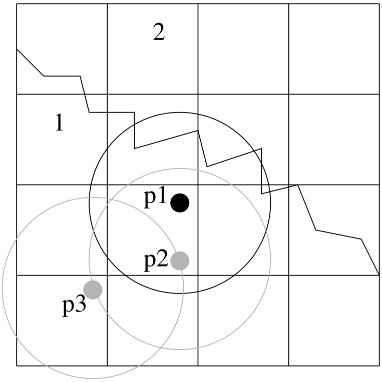


Fig. 3.15.: Three particles near a MPI boundary. Parts of the figure are from [Pirrung, 2010].

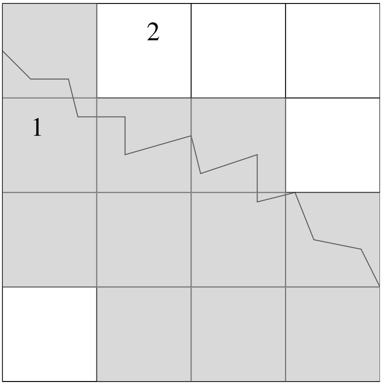


Fig. 3.16.: Grey shaded CBM cells can contain particles where the shape function intersects the MPI boundary. Parts of the figure are from [Pirrung, 2010].

The computationally expensive part of this concept is the identification of the relevant CBM cells that can contain so called shape-function-particles, i.e. the particles potentially intersecting the MPI boundary. This identification can be done during the initialization. Furthermore, we have to store the ID of the neighboring domains in each CBM cell. Reviewing figure 3.14, we can find that several neighboring MPI domains can be affected by a particle shape function. In the following section, we describe the initialization algorithm for the shape function parallelization.

3.6.3. Initialization of the Shape Function Deposition Parallelization

In a first step a global CBM is introduced. Every MPI domain communicates its minimum and maximum x, y and z position. We use the MPI_ALLREDUCE function with the minimum and maximum option. Each MPI domain $i|_{1,\text{nprocs}}$ calculates its minimum and maximum indices in the CBM by

$$CBM_{i,max} = (\mathbf{x}_{i,max} - \mathbf{x}_{minglob})/\Delta_{CBM} + 1, \tag{3.95}$$

$$CBM_{i,min} = (\mathbf{x}_{i,min} - \mathbf{x}_{minglob})/\Delta_{CBM} + 1. \tag{3.96}$$

Algorithm 2 Communication of global CBM cells. From [Pirrung, 2010]

```
for all CBM cells do
  if CBM cell i,j,k contains at least 1 element then
    save CBM cell in CBMCellsArray(i,j,k)
    nCBMCells = nCBMCells +1
  end if
end for
MPI_ALLGATHER (nCBMCells, nGlobalCBMCells(0:nProcs-1))
MPI_ALLGATHERV (CBMCellsArray(1:nCBMCells*3), GlobalCBMCellsArray)
```

Since every CBM cell is unique in the global CBM, the neighboring relation can be derived. Algorithm 2 sums up the idea how this can be done. First every CBM cell is checked for the presence of an element of the unstructured mesh. The CBM is a Cartesian grid that has its origin at the minimum position of the unstructured grid and goes up to the maximum coordinate. It does not consider the unstructured shape. Therefore, it might cover regions where the unstructured grid is not present. If there is at least one unstructured element inside a CBM cell, the index is stored in CBMCellsArray(i,j,k). Additionally, we save the number of CBM cells that contain elements. We then gather all indices of all processes into the GlobalCBMCellsArray buffer. The number nCBMCells is also gathered into one array, i.e. nGlobalCBMCells. In GlobalCBMCellsArray the indices are stored in the order of the processes from 0 to

$n_{procs}-1$. Together with the information stored in the array nGlobalCBMCells, we can reconstruct which CBM cells are owned by which process. The following example shall clarify this issue:

$$\begin{aligned} GlobalBGMCells &= (3,2,5,10,...)\\ GlobalBGMCellsArray &= (\underbrace{3,1,2,5,7,8,3,9,1,}_{\text{process 0 (3 cells)}}\ \underbrace{9,5,3,5,6,8}_{\text{process 1 (2 cells)}}\ ,6,7,...)\end{aligned}$$

Obviously, the GlobalCBMCellsArray contains a huge amount of data that is not required for every MPI domain, since only the neighboring data are required. In fact, only the CBM cells that are not further away from my MPI domain boundary than the cutoff radius of the shape function, which equals Δ_{CBM}, are required. Thus, we can reduce the data and store them in the ReducedCBMCellsArray. Algorithm 3 lines up the reduction operation. Again we save the number of cells of each process saved in nReducedCBMCellsArray.

Algorithm 3 Reduction of GlobalBGMCellsArray. From [Pirrung, 2010]

```
iProc = 0
for cell from 1 to SUM(nGlobalCBMCells(0:nProcs-1) do
   if cell is greater than SUM(GlobalCBMCells(0:iProc)) then
      iProc = iProc+1
   end if
   if cell (i,j,k) is not more than one $\Delta_{CBM}$ away from myProc domain
   and iProc $\neq$ myProc then
      save cell in ReducedCBMCellsArray
      nReducedCBMCells(iProc) = nReducedCBMCells(iProc) + 1
   end if
end for
```

Figure 3.17 shows the ReducedCBMCellsArray buffer for process 1, i.e. the grey shaded CBM cells. Note, that only CBM cells are included that are shared with other MPI processes. In advance the size of the ReducedCBMArray is not known. An option is to perform the identification twice. In the first loop the number of reduced CBM cells are counted. In the second sweep the information is stored. Since memory is the bottleneck on future HPC clusters this proceeding is desirable. This operation is only done in the initialization and the computational demand is negligible. Another important situation is covered by this proceeding, shown in figure 3.18, where the shape function reaches over several domains.

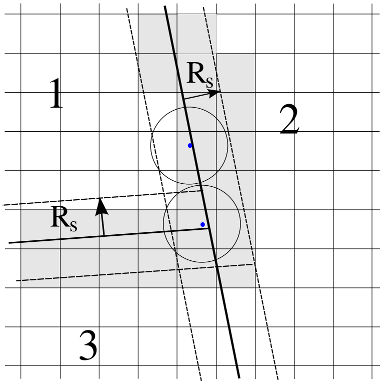


Fig. 3.17.: Grey shaded CBM cells show the ReducedCBMCellsArray for process 1.

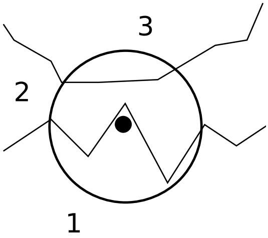


Fig. 3.18.: Shape function reaching over two domains. The figure is from [Pirrung, 2010].

In the next step, we have to use the ReducedCBMCellArray to determine to which neighboring processes the particles have to be communicated to. Figure 3.19 shows two CMB cells. Cell I has to communicate the particle data to process 2 only and cell II has to communicate it to process 2 and 3. Each CBM cell stores the information to which process it has to send the particle data. The mapping between the local CBM and the ReducedCBMCellArray is shown in algorithm 4.

For each CBM cell of the local process, we check if the surrounding 8 CBM cells and itself are part of the ReducedCBMCellsArray. Figure 3.20 shows the stencil around CBM cell II. Only the shaded cells contain data in the ReducedCBMArray. Since every CBM cell knows to which process it belongs to, we can tell to which process CBM cell II has to communicate the particle data.

Algorithm 2-4 are the main ingredients for the initialization of the parallel shape function deposition based on a CBM. This framework has to be linked to the particle communication method used within the PIC code. Figure 3.21

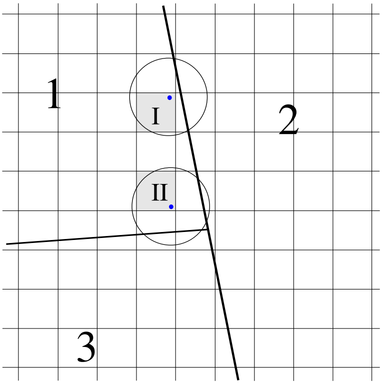


Fig. 3.19.: The particle in CBM cell I has to be sent to process 2 only. The one in CBM cell II has to be sent to process 2 and 3.

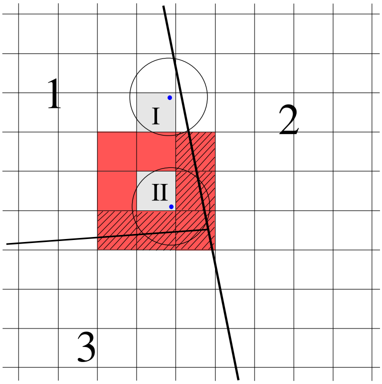


Fig. 3.20.: Mapping of ReducedCBMArray to CBMArray with stencil.

shows a particle that is leaving a MPI domain. The shape function still has an influence on the domain that the particle left. In this case, the question is how to handle the MPI communication. The following two possibilities shall be reviewed in detail:

I First, the particle gets communicated to process 2. Afterwards, in another communication, it is treated as shape function particle and is communicated back to the original process, where it started its journey. In this case, we split the communication in two steps. First, all particles that physically leave the domain are communicated. In a second step all shape-function-particles are communicated. This proceeding requires two MPI communication and separates the deposition procedure from the particle pushing over MPI boundaries.

II The local CBM is extended to the maximum possible range that a particle can step within one time step. In advance to the communication of the physically leaving particle, the processes that are affected at the final

Algorithm 4 Mapping of ReducedCBMCellArray to local CBM. From [Pirrung, 2010]

```
for iCell from 0 to nCBMCells-1 do
    for $CBM_x$ from CBMCellsArray(iCell*3+1)-1 to CBMCellsArray(iCell*3+1)+1 do
        for $CBM_y$ from CBMCellsArray(iCell*3+2)-1 to CBMCellsArray(iCell*3+2)+1 do
            for $CBM_z$ from CBMCellsArray(iCell*3+3)-1 to CBMCellsArray(iCell*3+3)+1 do
                if ($CBM_x$, $CBM_y$, $CBM_z$) has not been compared then
                    for externalCell from 0 to SUM(ReducedCBMCells)-1 do
                        iProc = 0
                        if externalcell is greater than SUM(nReducedCBMCells(0:Proc)) then
                            iProc = iProc+1
                            repeat
                        end if
                        if $CBM_x$ = ReducedCBMCellsArray(externalCell*3+1) and $CBM_y$ and $CBM_z$ also match then
                            save iProc in CellProcList($CBM_x$, $CBM_y$, $CBM_z$) and TempProcList
                        end if
                    end for
                    save that ($CBM_x$, $CBM_y$, $CBM_z$) has been compared
                else
                    save Procs from CellProcList($CBM_x$, $CBM_y$, $CBM_z$) in TempProcList
                end if
            end for
        end for
    end for
    save Contents of TempProcList in CBMCellsArray($CBM_x$, $CBM_y$, $CBM_z$)%ShapeProcs
    TempProcList = 0
end for
```

position of the particle, are already known. The communication message is divided in two sections. In the first section only the physically leaving particles are listed and in the second section the shape function particles are listed. Everything can be communicated in one step. This reduces the number of required communications to one message, lowering the latencies generated by each communication. If a particle has to be communicated back to the process it was leaving from, it would not be deleted but kept as shape-function-particle. The price to pay for this advantage is a more cumbersome initiation. The CBM has to be extended by 2 or more cells reaching over the own MPI domain to cover all possible destinations of leaving particles, e.g. shown in figure 3.22. This would require more memory storage, which is expected to be a minor problem due to the simple data structure (only integers). A larger CBM coverage would also increase the computational demand for the initiation procedure, which is not expected to be a major drawback because the initiation is only done once.

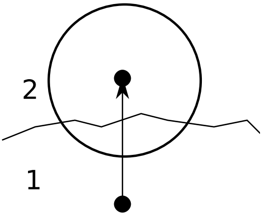


Fig. 3.21.: A particle moving from process 1 to 2 has to be communicated as shape function particle back to its origin process. The figure is from [Pirrung, 2010].

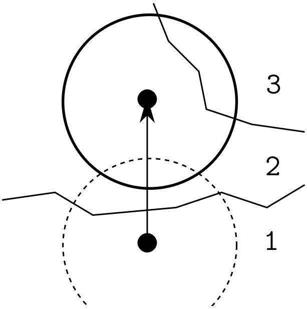


Fig. 3.22.: A more complex situation for a shape function particle when leaving its origin domain. The figure is from [Pirrung, 2010].

The two ideas presented above, still have to be embedded into the communication framework of the specific PIC code. This aspect is not described in detail since it would exceed the scope of this thesis. In [Pirrung, 2010] the embedding of the presented shape function parallelization, using the first idea is described in detail. In [Neudorfer, 2012] a communication approach following the second idea is presented in detail. Clearly, the second approach is the favorable one, since it requires less communication than the first one. Nevertheless, it is the more cumbersome one w.r.t. the implementation.

3.6.4. Scaling Benchmark

In this section, we benchmark the scaling of the proposed shape function parallelization, which has been presented in [Stock et al., 2011a].

As benchmark test case, we use the 30 GHz resonator, investigated in detail in section 4.2.1. We use the same setup parameter; only the grid differs. For the gyrotron resonator simulation a computational grid with approximately 14 300 tetrahedrons and 3 million DOF for a 4^{th}-order DG method was used. The particle hollow beam was modeled with approximately 100 000 pseudo electrons with a macro particle factor of 10^6.

Since the graph partitioning algorithm of the domain decomposition does not take the number of particles in each element into account, load imbalances occur for such particle distributions. Most of the domain is not affected by the particles due to the very inhomogeneous particle distribution. All particles are focused in the hollow beam which takes just a very tiny amount of the domain. This limits the parallel scalability of the problem. In order to quantify the parallelization capabilities of the PIC code, scaling tests were conducted. All scaling tests were performed on the "Nehalem" cluster of the High Performance Computing Centre Stuttgart (HLRS) where each compute node features two quad-core Intel® Xeon X5560 CPUs and 12 GB memory with an Infiniband node-node interconnect.

3.6.4.1. Strong Scaling

The parallel efficiency (PE) for the strong scaling is defined as

$$PE_{strong}(n_{procs}) = \frac{1}{n_{procs}} \frac{\Delta t_{serial}}{\Delta t(n_{procs})}, \tag{3.97}$$

where Δt_{serial} is the computational time required to run the problem with the serial code, n_{procs} is the number of processes and $\Delta t(n_{procs})$ is the time used to compute the problem with n_{procs} in parallel. Figure 3.23 shows the strong scaling PE. For 2 and 4 processes due to cache effects the PE is above one. From 8 processes on, the plot shows a nearly linear decay in the PE.

Figure 3.24 shows the strong scaling speed up (SU). The black dotted line shows the ideal SU. The largest SU is achieved with 64 processes. The maximum SU is approximately 20. Considering that the PE at 64 processes is ≈ 0.35, the strong scaling is not very good, which can be explained by the load imbalances caused by the particles. For increasing numbers of processes the number of processes without particles increases. These processes have idle times while other processes are working on the particle routines.

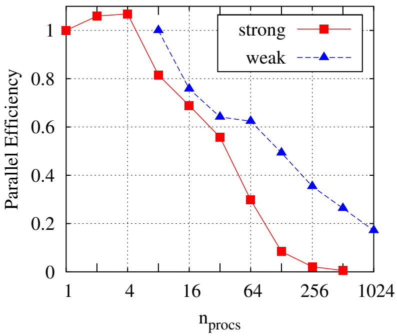


Fig. 3.23.: Parallel efficiency (PE) for strong and weak scaling.

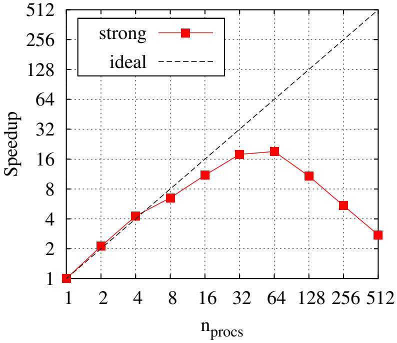


Fig. 3.24.: Speed up (SU) for strong scaling.

3.6.4.2. Weak Scaling

For the weak scaling the problem size and the number of processes has to grow by the same factor, i.e. $f_{WS} = n_{procs,2}/n_{procs,1}$ with $n_{procs,1} < n_{procs,2}$. Usually the problem size is depending on the number of degrees of freedom n_{DOF}. Here, also the number of particles have to be considered, since they affect the computational time as well. Thus, also the number of particles is increased by f_{WS}. For different n_{DOF} the grid is built with different element sizes, where the main parameter is the edge length of a tetrahedron. Since the grid is generated by a grid generator, sometimes it is impossible to match f_{WS} exactly to n_{DOF}. The resulting discrepancy is recognized by the constant $c_{DOF}(n_{procs,2}) = \frac{n_{DOF,2}}{n_{DOF,1}} \frac{n_{procs,1}}{n_{procs,2}}$. Since the timestep decreases for increasing number of elements, [1] we additionally normalize the PE by the number of iterations to get the PE for a unity timestep. This is recognized by the constant $c_{iter}(n_{procs,2}) = \frac{n_{iter,2}}{n_{iter,1}}$. All constants are considered for the weak scaling PE formula, yielding

[1] due to the CFL-condition

$$PE_{weak}(n_{procs,2}) = \frac{\Delta t(n_{procs,1})}{\Delta t(n_{procs,2})}\ c_{DOF}(n_{procs,2})\ c_{iter}(n_{procs,2}). \tag{3.98}$$

In our results always $n_{procs,1} = 8$ and $n_{procs,2}$ is the running variable for the number of processes. Table 3.3 lists up the setup parameter for each run with n_{procs}. For more than 256 processes the number of particles could not be increased due to stack size problems. It is assumed that this limitation has a minor effect on the weak scaling. Figure 3.23 shows the weak scaling PE as well. Already in the first step, the weak scaling efficiency is below 0.8. For increasing process numbers it decreases nearly linearly. This can be explained by the very inhomogeneous particle distribution. With increasing process and element numbers the inhomogeneity even grows since more elements are located in non-particle affected areas of the domain. Thus the load imbalances grow even more, resulting in a poor efficiency for the weak scaling.

$n_{procs,2}$	n_{Elems}	n_{DOF}	c_{DOF}	c_{iter}	Edge length	n_{parts}	MPF
8	20 661	413 220	1.0000	1.0000	0.003000	1E6	1E6
16	41 641	832 820	1.0077	1.5044	0.002400	2E6	0.5E6
32	90 664	1 813 280	1.0970	1.9102	0.001900	4E6	0.25E6
64	168 932	3 378 640	1.0220	2.2397	0.001450	8E6	0.125E6
128	327 269	6 545 380	0.9899	3.0750	0.001225	16E6	0.0625E6
256	683 707	13 674 140	1.0341	3.7750	0.000950	32E6	0.03125E6
512	1 351 672	27 033 440	1.0222	4.9323	0.000750	32E6	0.03125E6
1 024	2 778 757	55 575 140	1.0507	7.2161	0.000580	32E6	0.03125E6

Tab. 3.3.: Computational parameters for the weak scaling test with a 4^{th}-order DG-PIC simulation.

3.7. Implementation Framework: The HALO Code Framework

In this thesis a DG-PIC code is used for the computations. The algorithms developed in sections 3.5 and 3.6 have been implemented within this framework. The code has been developed since 2006 in the group of Prof. Munz at the Institute of Aerodynamics and Gas Dynamics, University of Stuttgart. The initiators were Dr. Lörcher [Loercher, 2009] and Dr. Gassner [Gassner, 2009]. Since then the code experienced a lot of changes, which all have been documented with a concurrent versioning system (CVS). A CVS is essential when developing a code framework with many developers. As CVS the Apache Subversion tool has been used [SVN, 2012].

The code framework is named Highly-Adaptive-Local-Operator (HALO) according to the fact that it is written as an adaptive framework and can be applied to any PDE for which a DG formulation can be derived. The locality of the DG operator is its main advantage w.r.t. parallelization. The code is written in the imperative programming language Fortran following the 2003 standard [FORTRAN, 2004]. The output of the field and particle data is done with the CFD General Notation System (cgns) format [CGNS, 2012, Rumsey et al., 2001] and in a later code version with the follow up HDF5 standard [HDF5, 2012]. Parallelization is based on the message passing interface standard, version 2.2 [MPI-Forum, 2009].

The main operations performed in the code are loops over all elements or sides. We use derived data types, and elements are stored in a pointer list, which is the best way to store topologically complex data structures as appearing for unstructured grids [Chapman, 2007]. Algorithm 5 gives an overview to the calculation of the residual, i.e. everything except for $\frac{\partial \hat{\boldsymbol{u}}_h}{\partial t}$ in (D.8).

All grid-based operations (deposition, Riemann-Solver, physical flux evaluation) are done with nodal DOF $\tilde{\boldsymbol{u}}$ (3.13). The transformation into modal DOF is done with the DG operators (D.3) for the spatial derivative and (D.4) for the surface integral. The particles are incorporated in the source term calculation with the operator (D.5) when the deposition is performed.

Algorithm 6 sums up the time update loop for the entire DG-PIC framework. The particle treatment is a nearly independent module, which can be plugged into any other field solver. It is linked to the DG framework at the following points:

- initialization: building of a Cartesian background mesh for the particle localization and deposition (strongly depending on the data structure and

Algorithm 5 DG weak form derivative.

```
for iElem = 1:nElements do
    nullify elem-residual
    calc volume integral: add to elem-residual $\boldsymbol{Res}(iElem)$
    for i = 1:nIPs (nodal IP) do
        get physical flux at IP $\boldsymbol{x}_i$: $\underline{\tilde{F}}_\beta(\mathbf{x}_i) = \mathbb{K}_\beta \tilde{\mathbf{u}}_h(\mathbf{x}_i)$
    end for
    spatial derivative and transformation to modal DOF: $\sum_{\beta=1}^{3} \mathcal{S}_\beta \; \underline{\tilde{F}}_\beta(\mathbf{x}_i)$
    calc source: add to elem-residual $\boldsymbol{Res}(iElem)$
    for i = 1:nIPs (nodal IP) do
        PIC deposition at $\mathbf{x}_i$, calc $\tilde{\mathbf{g}}_h(\mathbf{x}_i) \to (\rho, \boldsymbol{j})$
    end for
    transform to modal DOF: $\mathcal{G}\tilde{\mathbf{g}}_h(\mathbf{x}_i)$
end for
for iSide from 1 to nSides do
    calc surface integral: add to to elem-residual $\boldsymbol{Res}(iElem(iSide))$
    for i = 1:nIPs (nodal IP on surface) do
        if boundary then
            apply boundary condition
        end if
        calculate Riemann problem on every face $f$ : $\underline{\tilde{F}}_{h,f}(\mathbf{x}_i)$
    end for
    integrate and transform to modal DOF: $\mathcal{E}_f \; \underline{\tilde{F}}_{h,f}(\mathbf{x}_i)$
end for
$\frac{\partial \hat{\mathbf{u}}_h}{\partial t} = \boldsymbol{Res}$
```

Algorithm 6 time update loop

```
while $t < t_{end}$ do
    for i = 1:5 (nRK-Stages) do
        EM-field: transformation modal to nodal DOF, $\mathcal{V}\hat{\mathbf{u}} = \tilde{\mathbf{u}}$
        calc *DG weak form derivative* (see Algorithm 5): elem-residual
        for p = 1:nParticles do
            interpolate EM field at particle position: $\boldsymbol{E}(\boldsymbol{x}_p), \boldsymbol{B}(\boldsymbol{x}_p)$
            calc Lorentz force, $\boldsymbol{F}_{L,p}$, eq.(2.2)
        end for
        RK-Stage-Update: eq. (3.21)
        EM-field: $f(t_i, \boldsymbol{u}_{i-1})$ = elem-residual
        particles: $f(t_i, \boldsymbol{u}_{i-1})$ = Lorentz force
        particle boundary treatment
    end for
    particle inserting due to emmission
    analysis and I/O
end while
```

handling)

- interpolation of the EM field to the particle position for evaluating the Lorentz force
- deposition: assignment of particles to the computational mesh to compute ρ and $\boldsymbol{j}$

For the deposition, the nodal IPs on the grid are essential. The distribution of the nodal IPs for our method is not optimal for the PIC method. For a low-order discretization no IPs are located in the volume of an element, but only on the surface. This results in a bad sampling of the particles shape when using a shape function. A more volume-based distribution of IPs is desired in the PIC context, such as used in [Jacobs and Hesthaven, 2006]. The IP location $\mathbf{x}_i$ is optimized to the basis function used for the polynomial describing the state in the element, but not for the shape-function. However, the effect is not studied in detail within this thesis.

4. Calculations

This chapter presents the verification calculations and the gyrotron resonator calculations.

4.1. Particle-in-Cell Verification Test Cases

A complex framework like a PIC scheme cannot be used without extensive validation and verification. The objective of validation is to check if the correct equations are being solved. For verification the objective is to check if the equations are being solved correctly. Since the PIC framework is well proved for the target application, i.e. the gyrotron resonator, a validation is not necessary, but we still have to verify our implementation of the PIC framework. Thus, in this section test cases are presented to verify our DG-PIC code implementation. In section 4.1.1 a 3D plasma-wave test case shows the ability to reproduce the fundamental plasma frequency ω_p. This is followed by the two-stream instability test case in section 4.1.2. Finally, in section 4.1.3 a diode test case demonstrates the features and capabilities of the hyperbolic divergence cleaning method.

4.1.1. 3D Plasma Wave

The plasma wave test case describes the plasma oscillation frequency, or Langmuir oscillation, named after [Tonks and Langmuir, 1929] who first discovered these oscillations. This test case is a basic benchmark to verify the entire PIC code and is widely used [Birdsall and Langdon, 1991, Hockney and Eastwood, 1981, Jacobs and Hesthaven, 2006]. The plasma oscillation frequency $\omega_{p,i}$ for a certain particle species i is given by

$$\omega_{p,i} = \sqrt{\frac{n_i q_i^2}{m_i \varepsilon_0}}, \tag{4.1}$$

where n_i is the number density, q_i is the charge of the species, m_i is the particle mass, and ε_0 is the permittivity of free space.

To demonstrate the ability of our high-order DG-PIC code to reproduce the plasma frequency, a setup proposed in [Jacobs and Hesthaven, 2006] has been used. A plasma wave can be created by a particle distribution function that deviates a set of equidistant particles $\boldsymbol{x}_{eq}$ by a sine function, yielding

$$\mathbf{x} = \mathbf{x}_{eq} + A \cdot \sin(k\ \mathbf{x}_{eq}), \tag{4.2}$$

where the amplitude of the deviation is A and k is the wave number. In this case, $A = 0.001\,\mathrm{m}$ and $k = 2\,\mathrm{m}$ have been chosen. In y- and z-direction the one-dimensional distribution function is copied equidistantly, so that two-dimensional slices in the yz-plane are created. We use 100 particles in x-, and 15 particles in y- and z-direction. The particles within the zy-plane have to be equidistant, in order to prevent a potential difference in this plane. The potential difference occurs only in the x-direction. According to the particle distribution, the computational domain is chosen $2\pi\,\mathrm{m}$ in x- and $1\,\mathrm{m}$ in y- and z-direction discretized by a Cartesian mesh with $60 \times 10 \times 10$ elements. Using a 2^{nd}-order space discretization results in 156 000 DOF. We use periodic boundaries.

To achieve neutrality, two species are used - negatively charged electrons and positively charged protons - 45 000 in total. The protons have no deviation, but an equidistant distribution in all dimensions, i.e. $A = 0\,\mathrm{m}$ and $k = 0\,\mathrm{m}$ according to (4.2). The mass and charge of each particle is multiplied by a macro particle factor of 10^{10}, i.e. each simulated particle represents 10^{10} real particles.

For the calculation of ρ and $\boldsymbol{j}$ we use the polynomial shape function (3.4) with $R_s = 0.2\,\mathrm{m}$ and $\alpha_s = 2$. The periodicity of the particles is considered and the shape function is also evaluated over the periodic boundaries.

For the given setup a plasma frequency of $\omega_{p,e} = 3.38 \times 10^8\ \frac{\mathrm{rad}}{\mathrm{s}}$ with a period of $T_e = 1.86 \times 10^{-8}\,\mathrm{s}$ is expected. Table 4.1 summarizes the physical- and setup-parameters.

Symbol	[unit]	Value
Domain:		
V	$[\mathrm{m}^3]$	2π
Electrons:		
q_e	[C]	$-1.60217653 \times 10^{-9}$
m_e	[kg]	$9.1093826 \times 10^{-21}$
N_e	$[-]$	$100 \times 15 \times 15 = 22\,500$
n_e	$[\mathrm{m}^{-3}]$	$3\,581$
$\omega_{p,e}$	[rad/s]	3.38×10^8
T_e	[s]	1.86×10^{-8}
A_e	[m]	0.001
k_e	$[-]$	2
Protons:		
q_p	[C]	$1.60217653 \times 10^{-9}$
m_p	[kg]	$1.672621637 \times 10^{-17}$
N_p	$[-]$	$100 \times 15 \times 15 = 22\,500$
n_p	$[\mathrm{m}^{-3}]$	$3\,581$
$\omega_{p,p}$	[rad/s]	7.88×10^6
T_p	[s]	7.98×10^{-7}
A_p	[m]	0
k_p	$[-]$	0

Tab. 4.1.: Setup parameters for the plasma wave test case.

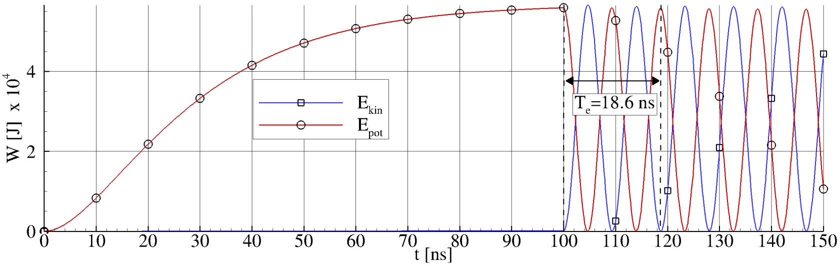


Fig. 4.1.: Potential field energy and kinetic energy for the plasma wave test case.

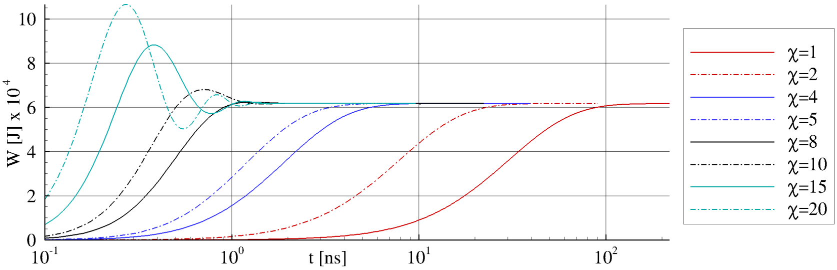


Fig. 4.2.: Potential field energy in the start-up phase of the plasma wave test case for different χ.

Figure 4.1 shows the kinetic energy of the electrons,

$$E_{kin} = \sum_{i=1}^{N_{parts}} c_0^2 \, m_i(\gamma_i - 1), \tag{4.3}$$

and the potential field energy E_{pot} (see eq. (3.93)) over the time. The first 100 ns only E_{pot} is growing. This is due to the evolution of the static electric field. During this period the particles are fixed and the PHM system can evolve a static electric field, which is required to initiate the desired plasma oscillation. The simulation time to evolve the static electric field only depends on the divergence cleaning speed, χc_0. In this case $\chi = 1$ during the entire simulation. It is possible to speed up the startup by increasing χ. Figure 4.2 shows the evolution of the potential field energy for different χ. For increasing χ the field evolves faster. This does not automatically mean that the computation time is faster, since the timestep decreases by the same factor χ as the eigenvalue grows. For too large χ an oscillation appears in the beginning, which is damped in the course of the simulation. In this sense the time-integration of the PHM system can be regarded as an iterative solution of the divergence constraint (2.6), or even as a Poisson solver where the time is the relaxation parameter.
After the static electric field has fully evolved and does not change any more, the particles can be released. They now will experience the force of the static electric field and start to move. This finally results in the oscillation known as the plasma oscillation. The period of the simulated setup is $\approx 1.86 \cdot 10^{-8} s$,

which corresponds to the analytic calculation and indicates that the code is able to reproduce basic plasma phenomena accurately.
Note, the frequency doubling phenomenon in figure 4.1. The particles oscillate with ω_e, which can be described by a sine function, $x(t) = \sin(t)$. The velocity is the time derivative, $v(t) = \cos(t)$. The kinetic energy is given as

$$E_{kin} = \frac{1}{2}mv^2. \tag{4.4}$$

There, the velocity is included with the square. Considering the trigonometric identity

$$cos^2(t) = \frac{1 + \cos(2t)}{2}, \tag{4.5}$$

we find that $\cos^2(t)$ doubles the frequency. This is the reason why we see the doubled frequency for the potential and kinetic energy in figure 4.1.

4.1.2. 1D Two-Stream Instability

The two-stream instability (TSI) is a classical verification example for PIC codes [Birdsall and Langdon, 1991, Jacobs and Hesthaven, 2006]. The TSI appears when two particle beams of equal density ($n_1 = n_2 = n_0$) with equal but opposite velocities ($v_1 = -v_2 = v_0$) interact with each other. The analysis for the dispersion relation and the resulting growth rate is given in [Gurnett and Bhattacharjee, 2005, p. 287-288].
We choose a special case of the TSI, i.e. the two-electron-stream instability with an ion beam at rest giving a quasi neutral configuration. Figure 4.3 shows a schematic drawing of the particle density function (PDF) of the two beams in velocity phase space. For quasi neutrality, in the centre a positively charged particle distribution can be found. According to [Krall and Trivelpiece, 1986, (p. 449-457)] the velocity v_0 has to be chosen as

$$v_0 = \frac{3}{k\sqrt{2}}\omega_p, \tag{4.6}$$

to achieve the fastest growth rate for the instability. Here, ω_{pe} is the plasma frequency of one single beam and k is the wave number.
We use a quasi 1D domain with $120 \times 1 \times 1$ 2^{nd}-order elements with the total size of $4\pi \times 0.01 \times 0.01$ m^3. The beams are initiated with the sinusoidal distribution

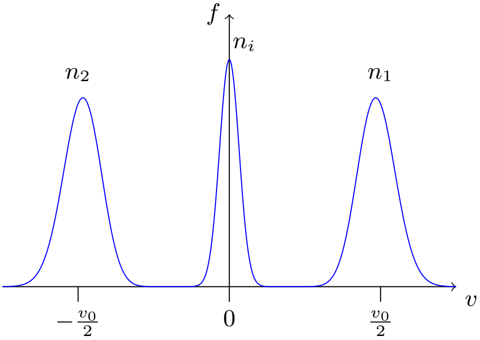


Fig. 4.3.: Schematic drawing of the particle density function leading to a two-stream-instability.

function (4.2). Each species is initialized with 1 000 particles in a line. Table 4.2 summarizes the parameters of the setup. As deposition method we use the volume-weighting method.

Symbol	[unit]	Value
General Parameters:		
k	[−]	0.5
$\omega_{pe}\|_{1,2}$	[rad/s]	$2.135122757 \times 10^{8}$
$\frac{V_0}{2}$	[m/s]	$1.0606601717798211 \times 10^{8}$
Species 1:		
q_1	[C]	$-1.60217653 \times 10^{-13}$
m_1	[kg]	$9.1093826 \times 10^{-25}$
n_1	[−]	$5.032532601 \times 10^{7}$
MPF		$\times 10^{6}$
v_1	[m/s]	$\frac{V_0}{2}$
α		0.001
k		0.5
$N_{p,1}$	[−]	1 000
Species 2:		
q_2	[C]	$-1.60217653 \times 10^{-13}$
m_2	[kg]	$9.1093826 \times 10^{-25}$
n_2	[−]	$5.032532601 \times 10^{7}$
MPF		$\times 10^{6}$
v_2	[m/s]	$-\frac{V_0}{2}$
α		0.001
k		0.5
$N_{p,2}$	[−]	1 000
Species 3:		
q_3	[C]	$3.20435306 \times 10^{-13}$
m_3	[kg]	$9.1093826 \times 10^{-23}$
n_3	[−]	$5.032532601 \times 10^{7}$
MPF		$\times 10^{6}$
v_3	[m/s]	0
α		0
k		0
$N_{p,3}$	[−]	1 000

Tab. 4.2.: Setup parameters of the two-stream instability.

Figure 4.4 shows the kinetic and potential field energies over the time. In the beginning E_{kin} is constant. At approximately 1.5×10^{-7} s a drop in the kinetic energy and a growth in the potential energy appears. This is a typical phenomenon for a starting TSI. After the drop appears the energies oscillate chaotically. For a better analysis of this part, figure 4.5 shows the drift velocity

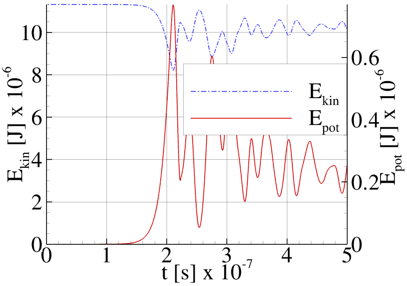


Fig. 4.4.: Kinetic and potential field energy versus time.

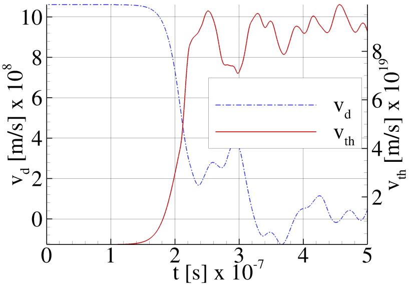


Fig. 4.5.: Thermal and drift velocity (v_{th} and v_d) of the particles versus time.

in x-direction,

$$v_d = \frac{1}{N_p}\sum_{i=1}^{N_p} v_i, \tag{4.7}$$

and the thermal velocity in x-direction,

$$v_{th} = \frac{1}{N_p - 1}\sum_{i=1}^{N_p}(v_i - v_d)^2, \tag{4.8}$$

over the time. While the drift velocity experiences a drop, the thermal velocity grows, which is in accordance with [Krall and Trivelpiece, 1986, Birdsall and Langdon, 1991].

Although the two discussed figures already strongly support the appearance of a TSI, particle density function (PDF) plots in the v_x-x phase space plane are used to identify a TSI. Figure 4.6 shows the PDF in v_x-x phase space at different timesteps. Initially both beams do not interfere with each other and the PDF shows three separated lines. After a while, the TSI grows and a bump of particles that are decelerating can be found in the upper beam. In the course of the simulation the typical holes in the PDF appear while the PDF of the two electron beams are mixing up with each other. Note, that the centre beam (i.e. the positive charged particles) do not move or change its velocity, since its particles are too inert to act on the observed timescales.

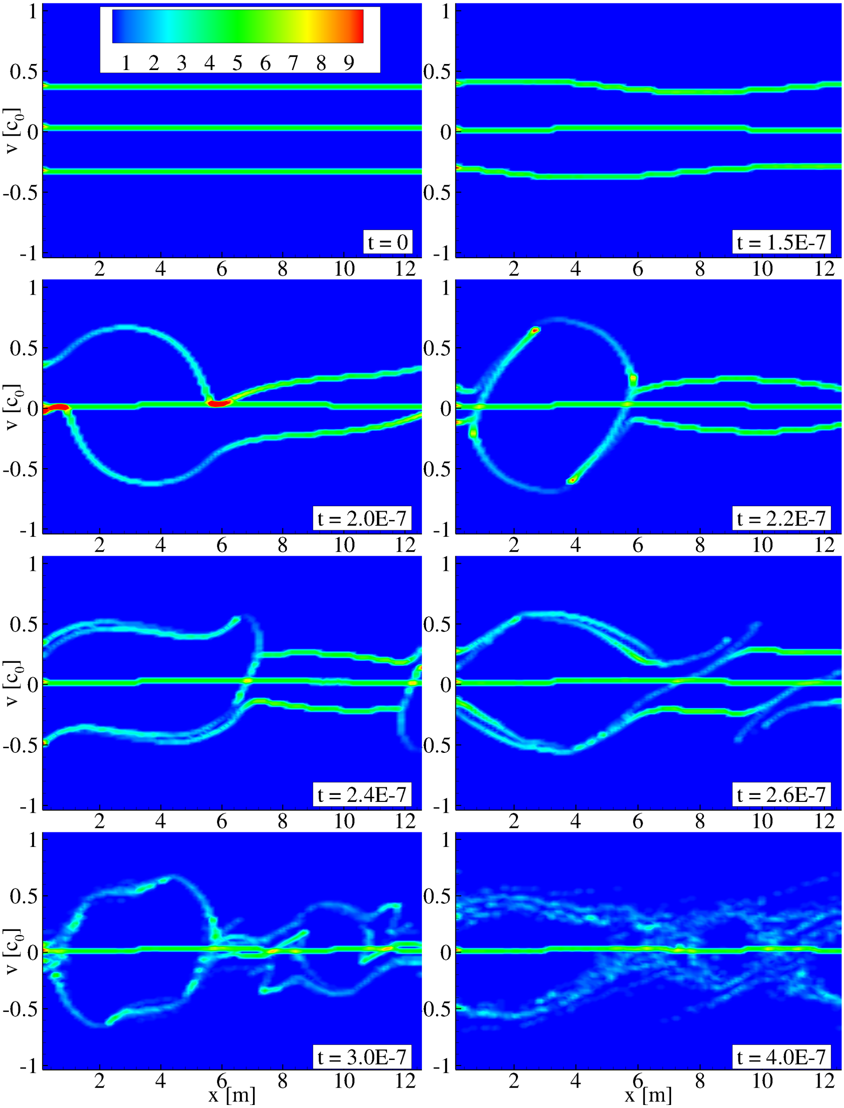


Fig. 4.6.: PDF in v_x (x-axis) and x (y-axis) phase space at different timesteps.

4.1.3. Verification of the Hyperbolic Divergence Cleaning: 2D Diode

To show the impact of the hyperbolic divergence cleaning, we chose a simple but very typical situation occurring in the context of PIC simulations, i.e. a diode test case as proposed in [Illy, 1997]. Figure 4.7 shows a schematically setup of the test case. The domain ($0.1 \times 0.1 \times 0.0015\,\mathrm{m}^3$) is considered to be a plane diode with cathode and anode situated on the left and on the right side, respectively. Figures 4.8 and 4.9 show the domain in the xy-plane for a Cartesian and a tetrahedral unstructured mesh, which have been used in the following computations. The spatial setup is a three-dimensional grid, set up to imitate a plane two-dimensional diode. In x-direction we have implemented perfectly conducting, in y-direction open and in z-direction periodic boundaries.

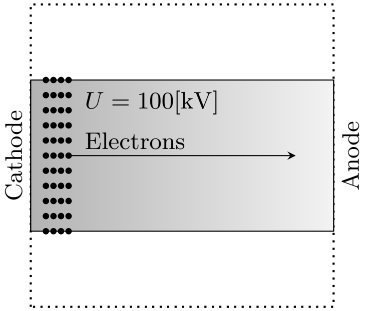


Fig. 4.7.: Schematic setup of 2D diode test case.

In the center of the cathode (left side) we inject an electron beam with a current density that has 75% of the space charge limiting current density. The length of the emitter is 37.5 mm, the voltage is 100 kV and the current density is $\approx 5\,536\,\mathrm{Am}^{-2}$. The current density is simulated by an electron beam where 9 000 charges per ns are injected with a macro particle factor of 2.180367×10^5, which are accelerated in x-direction by an external electric field of $-10^5\,\mathrm{Vm}^{-1}$. Figure 4.10 shows the situation for the Cartesian mesh at $t = 50\,\mathrm{ns}$ without divergence cleaning, only using the hyperbolic evolution equations (2.3) and (2.4). Strong gradients of the E_x-field in the region of the electron beam indicate large charge conservation errors. Figure 4.11 shows the corresponding position of the particles. A weak bunching of the beam in the left part of the

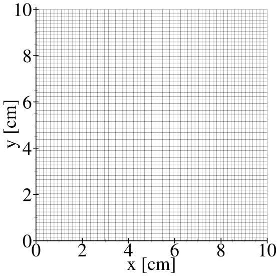


Fig. 4.8.: Cartesian grid of 64×64×1 second-order hexahedrons with ≈ 33 000 DOF.

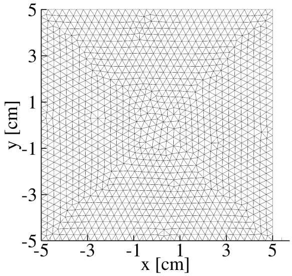


Fig. 4.9.: Unstructured grid of 7452 second-order tetrahedrons (edge length 0.0001) with ≈ 30 000 DOF.

figure can be observed. Filaments of particles, especially on the edges of the beam, can be found in the right part of the figure. These filaments are a typical sign for inconsistencies between the electric field and the charge density, i.e. a violation of Gauss' law (2.6).

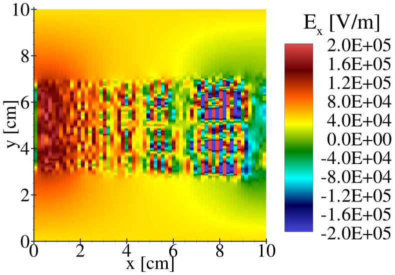


Fig. 4.10.: E_x-field at 50 ns of y-x-plane slice at 50% z-thickness for $\chi = 0$ with Cartesian mesh.

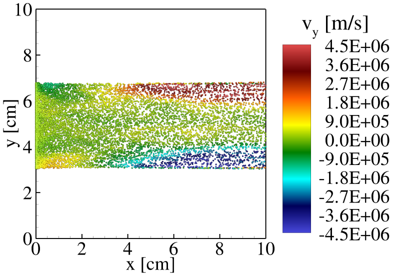


Fig. 4.11.: Particles for Cartesian mesh at 50 ns with $\chi = 0$.

In figure 4.12 and 4.13 we find the situation for $t = 50\,\text{ns}$ with $\chi = 1$. Here, we

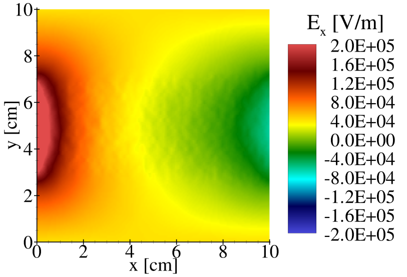


Fig. 4.12.: E_x-field at 50 ns of y-x-plane slice at 50% z-thickness for $\chi = 1$ with Cartesian mesh.

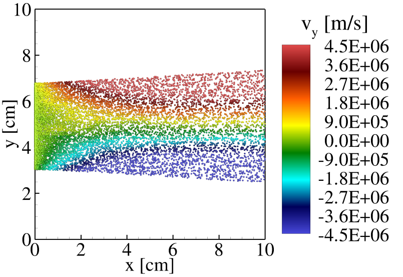


Fig. 4.13.: Particles for Cartesian mesh at 50 ns with $\chi = 1$.

can see a smooth electric field without strong gradients and no significant divergence errors. The particle beam is diverging and strong filaments, especially on the edges, are not occurring. Moreover, a homogeneous particle distribution can be found.

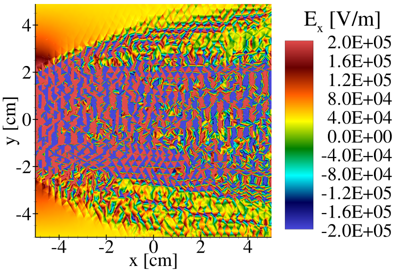


Fig. 4.14.: E_x-field at 50 ns of y-x-plane slice at 50% z-thickness for $\chi = 0$ with unstructured mesh.

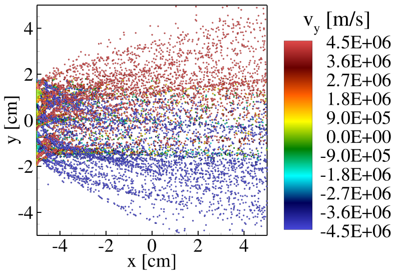


Fig. 4.15.: Particles for unstructured mesh at 50 ns with $\chi = 0$.

The results for the unstructured mesh without divergence cleaning can be found in figure 4.14 showing the E_x field and figure 4.15 showing the corresponding particles at t=50 ns. Strong scattering in the E_x-field indicates large charge conservation errors. The particles are totally scattered in a strong diverging fashion along the diode, which can be explained by the violation of the charge conservation. For $\chi = 1$ the results for the unstructured mesh are very similar to the results of the Cartesian mesh shown in figure 4.12 and 4.13. Therefore, we skip to show them explicitly.

For analysis we consider the L_2-norm of the time derivative of the Lagrange multiplier, yielding

$$\frac{1}{\chi}\|\frac{\partial\Phi}{\partial t}\|_{L_2} := \frac{1}{\chi}\sqrt{\int_\Omega \left|\frac{\partial\Phi}{\partial t}\right|^2 dV} = \sqrt{\int_\Omega \left|\nabla\cdot\boldsymbol{E} - \frac{\rho}{\varepsilon_0}\right|^2 dV}, \tag{4.9}$$

i.e. the charge conservation error. Figure 4.16 shows the charge conservation error for $\chi = [0, 1]$ in logarithmic scale due to a strong growth in the error with $\chi = 0$. We can find that for the unstructured grid the error grows significantly stronger than for the Cartesian grid. This can be explained by the propagation direction of the particles and the electromagnetic waves. While the Cartesian mesh is aligned with the propagation direction, the unstructured mesh is not aligned, leading to stronger errors.

Figure 4.17 shows the charge conservation error for both grids with $\chi = [1, 2, 5]$. The errors evolve about linearly in the beginning phase and remain constant after 1.5 ns. Electrons are initially emitted from the left, traveling about 1 ns to the right. After the initial phase and another 0.5 ns the current is fully developed and the errors stay constant. For increasing χ the error decreases as expected. Again the unstructured mesh has larger errors than the Cartesian mesh.

The necessity for divergence cleaning of Maxwell's equations has been demonstrated. A major influence on the errors is caused by the grid. In both examples the unstructured grid led to larger errors than the Cartesian grid. When the grid is uniform with the wave and source propagation directions, errors are smaller than for a non-uniform grid. With $\chi = 1$ satisfactory results are achieved for both Cartesian and unstructured grids. However, a proper study of the wave propagation directions seems to be an important issue. A grid designed for these specific demands can improve the charge conservation.

The study of this test case was conducted in [Stock et al., 2011b], where also the relevance of the enforcing of the magnetic monopole constraint (2.5) is discussed in detail.

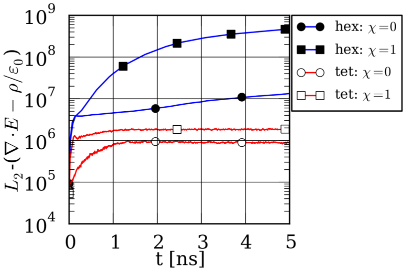


Fig. 4.16.: L_2-norm of the charge conservation error for the Cartesian and unstructured mesh using hexahedral and tetrahedral elements, respectively.

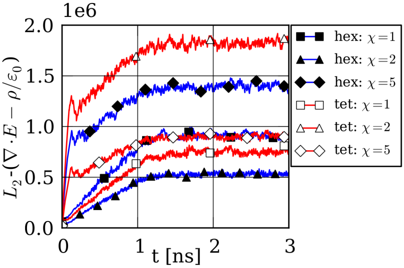


Fig. 4.17.: L_2-norm of the charge conservation error for the Cartesian and unstructured mesh using hexahedral and tetrahedral elements, respectively.

4.2. Gyrotron Resonator Simulations

A major objective of the development of a high-order DG-PIC code is the demonstration of the ability to simulate complex large-scale plasma devices. In the 2008 DFG proposal “Numerical Modeling and Simulation of Highly Rarefied Plasma Flows”, the gyrotron resonator was explicitly mentioned to be a good candidate for this purpose. The physics of a gyrotron were introduced in section 2.2. In this section we present the results of the gyrotron resonator computations performed with the DG-PIC code. In section 4.2.1 we investigate a 30 GHz gyrotron resonator with a setup suggested in [Kern, 1996, Illy, 1997]. Together with the benchmark study of the 30 GHz resonator a grid refinement study was conducted. The aim of this study was to find a convergence criteria for the resolution of the grid. In section 4.2.3 we investigate the 140 GHz resonator of the gyrotron aimed for electron resonance heating of the Wendelstein (W7-X) Stellarator [Dammertz et al., 2002]. We will compare the results to the SELFT code used within the design of this cavity. SELFT is a fast simulation code for gyrotron interaction processes, based on a slow variables formalism [Kern, 1996]. It calculates the electron trajectories and the field profiles of selected relevant modes self-consistently over time, and is thus capable of simulating complete startup sequences of gyrotrons. For a detailed description of the SELFT code and its advantages and disadvantages over the DG-PIC method see section 1.1.6.

4.2.1. 30 GHz Resonator

As benchmark we reinvestigate a 30 GHz gyrotron resonator with a setup suggested in [Kern, 1996, Illy, 1997], for which no experimental data are available. However, this example could serve as benchmark problem for state-of-the-art PIC codes.
Note, that this section is based on the journal article [Stock et al., 2012a]. Furthermore the project thesis by [Riedlinger, 2011] also deals with the simulation of the presented 30 GHz resonant cavity.

4.2.1.1. Computational Setup

Figure 4.18 shows the geometry of the gyrotron resonator with an electron hollow beam near the center of the device.

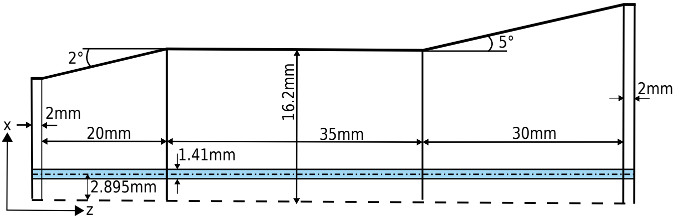


Fig. 4.18.: Resonator geometry with electron hollow beam near the center.

The electron hollow beam has a kinetic energy of 79 keV per electron. Instead of using single electrons, we make use of macro-particles imitating the presence of 10^6 electrons per macro-particle. Consequently, the mass, charge and the kinetic energy of the macro-particle increase by this factor, leading to 93 622 macro-particles, being emitted per nanosecond at the entry (z = 0) of the resonator, to achieve the designated current of $I = 15\,\mathrm{A}$. We set $\alpha = 1.5$ for the beam (see eq. (2.14)). From these values and the external static magnetic flux $B_z = 1.161030826\,\mathrm{T}$, we can compute the cyclotron frequency $\Omega_c = 28.14\,\mathrm{GHz}$. Table 4.3 summarizes all relevant parameters.

Symbol	[unit]	Value
Elementary Parameters:		
q_p	[C]	$-1.60217653 \times 10^{-13}$
m_p	[kg]	$9.1093826 \times 10^{-25}$
Velocity & Energy Parameters:		
E_p	[keV]	79×10^{6}
α		1.5
γ	[-]	1.1545991
$v_{\perp}$	[m/s]	$1.2468833876 \times 10^{8}$
$v_{\|\|}$	[m/s]	$0.8312555917 \times 10^{8}$
$\|\mathbf{v}\|$	[m/s]	$1.4985673295 \times 10^{8}$
Gyration Parameters:		
B_z	[T]	1.161030826
r_L	[mm]	0.705
R_e	[mm]	2.895
Ω_c	[GHz]	28.1486
Emission Parameters:		
U	[kV]	79
I	[A]	15
Emission rate	[parts/ns]	93 622

Tab. 4.3.: Parameters for the 30 GHz resonator aimed to excite a $\mathrm{TE}_{2,3}$ mode.

The hollow beam is initiated on the entire length of the resonator, with 10^5 macro particles at $t = 0\,\mathrm{s}$.

Figure 4.19 shows the principle variables considered for the positioning of the particles on the hollow beam. While the radius $R_e = 2.895\,\mathrm{mm}$ defines the size of the hollow beam, $r_L = 0.705\,\mathrm{mm}$ defines the Larmor radius. The position is depending on the two angles θ_1 and θ_2 that are chosen randomly by a uniform distribution forming a hollow cylinder within the bounds $|R_e \pm r_L|$. Since particles are only inserted on discrete time levels, the distance in z-direction covered by each particle in one time step also has to be considered to achieve a continuous three-dimensional beam. Thus, the z-position within the section $\Delta z = v_{||}/\Delta t$ is chosen randomly as well.

The grid size is determined by the minimum number of DOF required to resolve the expected wavelength. For a nodal DG scheme with Maxwell's equations, a resolution of 12 points per wavelength is known to be sufficient enough [Jacobs et al., 2006]. Since we expect a $\mathrm{TE}_{0,3}$ mode, due to the results in [Kern, 1996,Illy, 1997], the wavelength is approximately the diameter of the resonator

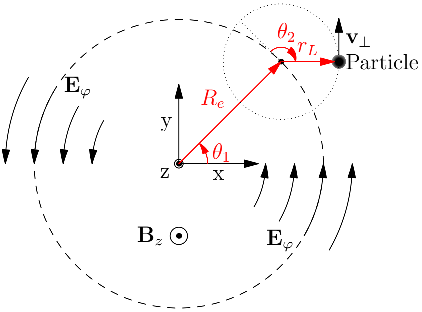


Fig. 4.19.: Electron hollow beam setup.

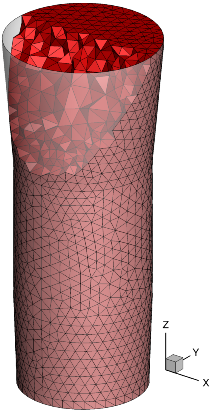


Fig. 4.20.: Cut-open view of tetrahedral grid with 30609 elements.

divided by the number of waves in it, leading to a wavelength of $\lambda \approx 0.0108\,\mathrm{m}$. According to the dimensions of the resonator, this means we need ≈ 100 DOF in z-direction and ≈ 20 in radial direction, resulting in ≈ 105000 DOF in total. The results presented below were obtained by a 4^{th}-order space discretization using a grid with 30609 tetrahedrons having a mean edge length of $\approx 2.6\,\mathrm{mm}$ and 612180 DOF shown in figure 4.20. We use six times more DOF as required due to better particle resolution by the deposition and safety margins. The spatial resolution is not only related to the resolution of the wavelength but also to the resolution of the particle-field coupling. The grid refinement study in section 4.2.2 showed that the consistency of the energy transfer between the beam and the EM field is only achieved for this resolution.

The bottom and the top have open boundaries for the EM field. Here, we use a Silver-Müller absorbing condition. On the surface shell, we use a perfectly conducting boundary condition. All field boundary conditions are derived from characteristics theory and further described in the appendix section B.3.

For the particles treatment, the bottom and the top are also open boundaries. The surface shell is never crossed by the particles. Thus, a particle boundary

is not required here.
The charge and current density is deposited to the grid by the 4^{th}-order polynomial shape function (3.4) with $\alpha_s = 2$ and $R_s = 1\,\mathrm{mm}$. The open boundary condition for the particles does not support high-order representation of the particles provided by the shape-function. Rather, it cuts the particle and its sphere of influence out of the domain as soon as the particles leaves the domain, even if the sphere would still intersect the domain. Therefore, results within $R_s = 1\,\mathrm{mm}$ of the boundary are not to be overrated.

4.2.1.2. Results

First we analyse the integral energy quantities, i.e. the potential field energy, E_{pot} (see eq. (3.93)), and the kinetic energy of the particles, E_{kin} (see eq. (4.3)). The sum of both parts yields the total energy $E_{tot} = E_{pot} + E_{kin}$ of the system. Figure 4.21 shows the integral energies over a period of 50 ns.
Within the first 20 ns, the EM field evolves until it reaches a steady state. We initiate the electric field with $\boldsymbol{E}_0 = 0$, which is somewhat unphysical since the already fully inserted particle beam generates a static electric field. But, in the PHM equations (3.5)-(3.8), the static field evolves due to the scalar potential $\chi c^2 \nabla \Phi$ in Ampère's law (3.5). This evolution can be seen in figure 4.22 where E_{El} and E_{Mag} are shown separately focused on the initial phase. While E_{Mag} stays nearly zero, E_{El} grows up to $\approx 0.6 \times 10^{-4}$ J at 2 ns. Then, E_{Mag} starts to grow since the RF field begins to evolve and the difference of $\approx 0.5 \times 10^{-4}$ J between E_{El} and E_{Mag} stays constant. The theory of RF fields says that the same amount of energy is stored in the magnetic and electric field altering with ω_{RF}. In figure 4.22, we can observe this equality between the two fields holding for the entire simulation except a difference of $\approx 0.5 \times 10^{-4}$ J, which is the static energy of the electric field due to the presence of the particle beam. The right part of figure 4.22 zooms in to a 0.1 ns section to clarify the reason why the energies oscillate within a range of 0.1×10^{-4} J. The oscillation can be explained by partially leaving energy packages when modes are propagating out of the domain. The dotted lines show the averaged values of the energies revealing the energy difference of $\approx 0.5 \times 10^{-4}$ J between the magnetic and electric field due to the static electric field.
Also, the kinetic energy is altering within the first 20 ns as a direct response to the static electric field until it reaches a steady state which is a bit lower than the initial energy. The loss of energy can be explained by the voltage depression caused by the static electric field.
From 20 ns, the system energy reaches a steady state, while in [Kern, 1996, Illy,

1997], a startup period of more than 50 ns was observed. The difference in the startup time can be attributed by different startup strategies for the current. While in [Kern, 1996, Illy, 1997], the current is slowly raised over several ns, we introduce the full current from the beginning. However, both startup strategies are unphysical. In reality, the current rises to its final value in the range of 150 μs.

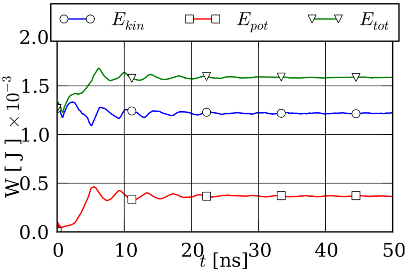


Fig. 4.21.: Integral system energies.

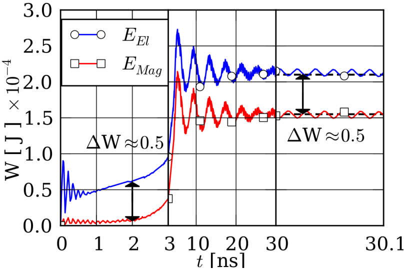


Fig. 4.22.: Integral potential field energies in the initial and steady state.

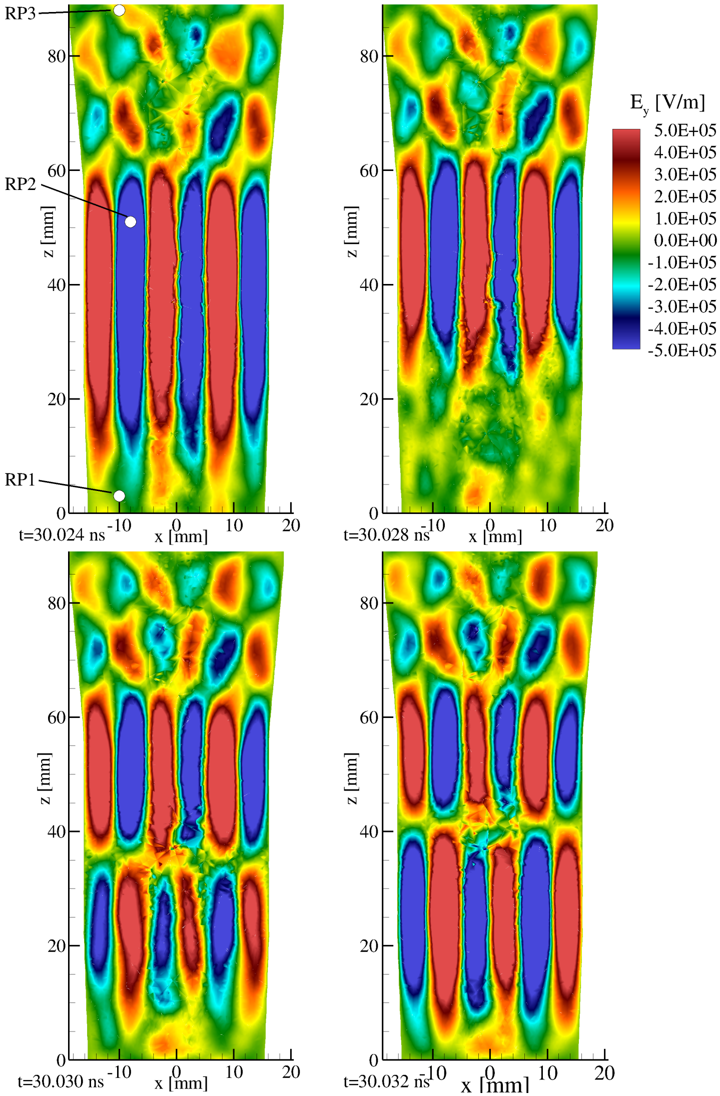


Fig. 4.23.: Sequence of E_y-field slices in the xz-plane at y = 0 mm and the position of the record points (RPs).

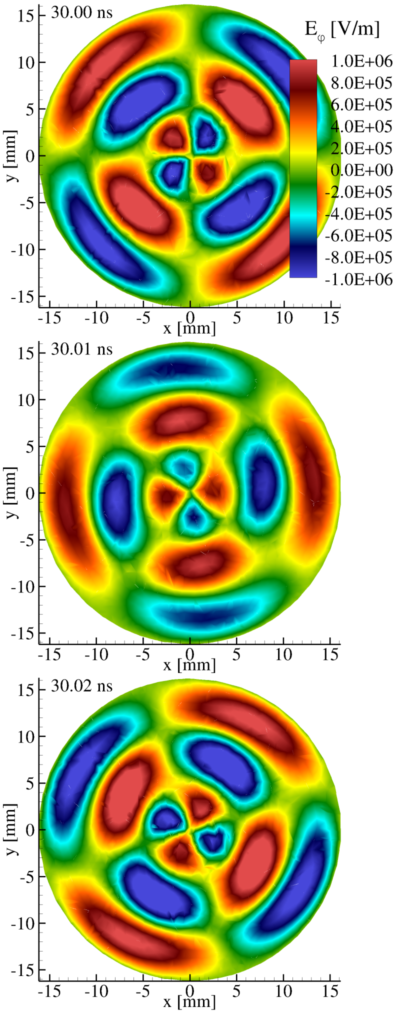


Fig. 4.24.: Sequence of E_φ-field slices in the xy-plane at z = 50 mm. The field rotates counter-clockwise due to a positive static magnetic field.

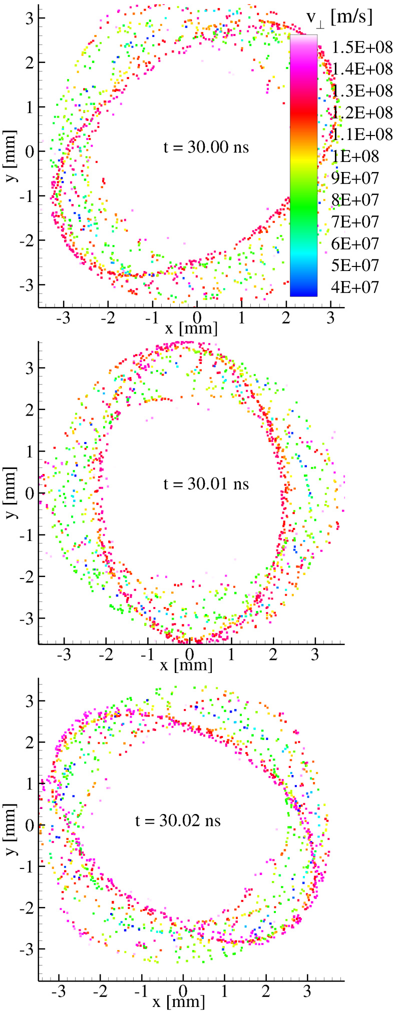


Fig. 4.25.: Sequence of slices showing the tangential velocity $v_\perp$ of particles in the xy-plane at z = 50 mm.

Figure 4.23 shows a sequence of four figures of the E_y-field in the xz-plane at y = 0 m. It presents a qualitative assessment of the mode pattern and an understanding of the temporal evolution of the EM field. The temporal resolution is chosen by a visualization time step of 2 ps (picoseconds) precluding temporal aliasing showing that the mode pattern is moving upwards. Clearly, a patch pattern of the EM field is visible. A wavelength consists of two patches resulting in three wavelengths in radial direction, indicating a $\mathrm{TE}_{X,3}$ mode. The azimuthal index X can be derived from figure 4.24 consisting of a sequence of plots showing the E_φ-field in the xy-plane at z = 50 mm, with $E_\varphi = -E_x \sin(\varphi) + E_y \cos(\varphi)$ and $\varphi = \tan^{-1}(y/x)$. In circumferential direction, we have four maxima covering two wavelengths, resulting in $X = 2$ for the first subscript, yielding a $\mathrm{TE}_{2,3}$ mode. This observation is in contrast to [Kern, 1996, Illy, 1997], where for the same setup, as used here, a $\mathrm{TE}_{0,3}$ mode was found, which can be explained by the fact that only rotationally symmetric modes were allowed. However, very recent computations performed with the commercial three-dimensional simulation code CST confirm our results [Private communication with G. Latsas, 2011].

Note that the field solution in Fig. 4.23 and 4.24 under-represents the high-order numerical solution since the visualization does not evaluate the polynomials directly, but recognizes a piecewise constant distribution only.

Figure 4.26 shows the temporal evolution of the E_φ-field on a circle at r = 8.1 mm - the second radial EM field maximum - and z = 50 mm over a period of 70 ps. The EM field was recorded with 72 record points (RPs) on the circle. Two wavelengths can be seen over the diameter moving in counter-clockwise direction with a rotational frequency of ≈ 14.3 GHz.

Additionally, we introduce in the left plot of figure 4.23 three RPs. RP1 is located at the bottom directly in the entry at $(-10, 1)$ mm in the downtaper, RP2 is located in the first radial extremum of the $\mathrm{TE}_{2,3}$ mode in the resonator at $(-8, 51)$ mm, and RP3 is located at the exit at $(-10, 88)$ mm in the uptaper. Figure 4.27 shows the E_x-field strength at RP2 evolving over a transient startup phase between 1 and 20 ns until a steady state of $\approx 1.3 \times 10^6$ V/m is reached. For further analysis of the frequency of the EM field signal, we computed the dominant frequency by measuring the time between two changes in the sign of the signal, i.e. a wavelength. To reduce noise, the signal then is post-processed with a one-dimensional digital filter, which averages the frequency values computed over a certain number of preceding points, i.e. the filter window size. We use a filter window size of 20 . Figure 4.28 shows the frequencies of the E_x-field at all three RPs. While RP1 has a very noisy frequency signal, RP2 and RP3 show a distinct frequency after a transitional startup phase until ≈ 10 ns. For

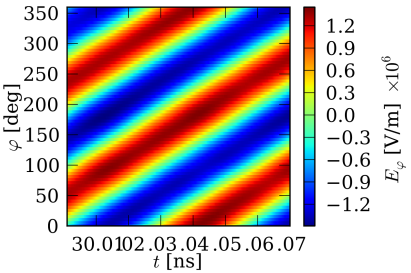


Fig. 4.26.: E_φ on a circle of 72 record points (RP) in the xy-plane at the second radial maximum (r = 8.1 mm, z = 50 mm).

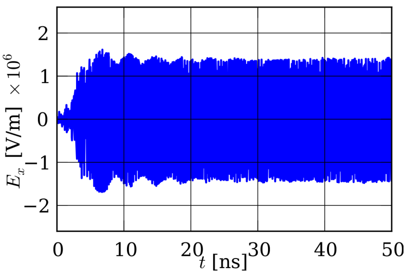


Fig. 4.27.: E_x-field strength at record point 2 (RP2).

RP2 and RP3, we can identify a frequency of ≈ 30 GHz for the steady state. At 4 ns, we could identify a $TE_{1,3}$ mode in the downtaper at z = 3 mm which switches to a $TE_{2,3}$ mode suddenly.

We processed the signal of RP2 with a discrete fast Fourier transform shown in figure 4.29 for two different time slots. During the startup phase, the dominant frequency is 29.7 GHz but shifts towards 29.75 GHz for the steady state.

Figure 4.30 gives a compact assessment of the energy transfer mechanism between the EM field and the electron beam. The left y-axis of figure 4.30 shows the cyclotron frequency Ω_c of the beam along the z-axis at 30 ns. For this purpose, we subdivided the beam into 50 equidistant sections and calculated the mean value of Ω_c in each section. The cyclotron frequency Ω_c increases from 28.25 GHz at the entry until it reaches ω_{RF} - dotted line - at ≈ 35 mm. After reaching ω_{RF}, the cyclotron frequency Ω_c alleviates again down to ≈ 28.5 GHz at the exit of the geometry. Additionally, we show the mean value of the kinetic energy normalized to the value at the entry, on the right y-axis of figure 4.30. From the normalized kinetic energy we can read that, in total, only ≈ 5 % of the beam energy is transferred into the EM field. At most, ≈ 40 % of the beam energy is transferred into the EM field at the point where Ω_c exceeds ω_{RF}. The ECM instability only can transfer energy from the electron beam to the EM field if Ω_c is slightly lower than ω_{RF}. Thus, an over bunching occurs around z=38 mm, and the beam takes back energy out of the field corresponding to

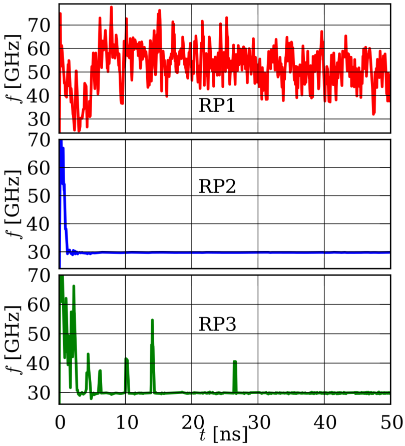


Fig. 4.28.: Dominant frequency of E_x-field at all three record points.

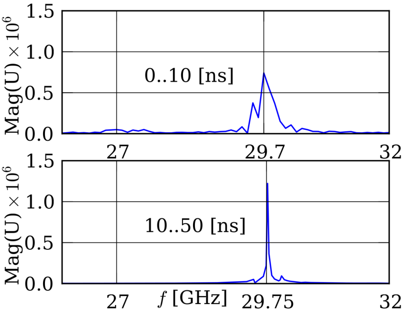


Fig. 4.29.: Frequency spectra at record point 2 (RP2) for two different time intervals.

the increasing beam energy after ≈ 40 mm.

Figure 4.31 shows the kinetic energy in keV of the electron beam at 30 ns split into transversal $E_\perp$, parallel $E_{||}$ and total components, according to $v_\perp$, and $v_{||}$. The decrease by ≈ 10 keV for $E_{k,tot}$ is caused by the voltage depression. The voltage depression is induced by the static electric field generated by the electron beam and only affects $E_{||}$. Obviously the part of the beam which is relevant for the energy transfer to the EM field only has an energy of ≈ 69 keV. Figure 4.32 shows a scatter plot of the particle beam at 30 ns where the color of the particles indicates v_z. The x-axis is scaled up by a factor of 8 w.r.t. the z-axis.

From the color pattern, the higher v_z at the entry and exit and also the deceleration in the middle part caused by the voltage depression can be observed. The deceleration leads to an increasing of the total number of particles by ≈ 15 % compared to the initially calculated number of particles in the beam. The density patterns of the particles in figure 4.32 show bunching effects on the beam.

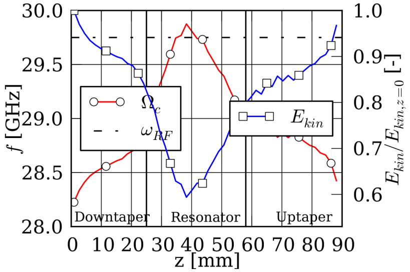


Fig. 4.30.: Cyclotron frequency Ω_c (left y-axis) and the normalized kinetic energy (right y-axis) of the electron beam along the z-axis at $t = 30$ ns.

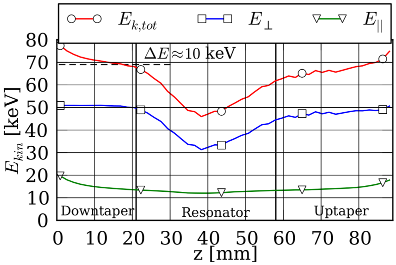


Fig. 4.31.: Energy profile of the electron beam in z-direction at t=30 ns. $E_{||}$ and $E_\perp$ are the relativistic kinetic energies corresponding to the v_z and $v_\perp$ components of the electron beam.

To get a better qualitative understanding of the bunching effect of the beam, figure 4.25 shows a sequence of particles scatter plots of slices in the xy-plane at z = 50 mm where the colors indicate the magnitude of $v_\perp$. The sequence corresponds to the sequence of E_φ-field slices shown in the above figure 4.24. The particles bunch along elliptically shaped orbits. On these orbits, the particles have the largest velocities, whereas the particles not located on the orbits are slower.

The angle φ_L between the particle vector from the gyration guiding-centre and the electric field vector $\boldsymbol{E} = [E_x, E_y]$ is schematically shown in figure 4.33. We assumed that $\boldsymbol{E}$ is uniform within $|r_g|$ since r_g is small compared to the wavelength. Azimuthal particle bunching occurs due to the relativistic deceleration of the particles in a φ_L section between 90 ° and 180 °. If bunching appears, the particles have to accumulate in this region. Figure 4.34 shows the relative particle density plotted over φ_L for 10 °-sections along the z-axis at 30 ns. For the electric field vector, we used the values at the particle position. Shortly after the resonator at z=22 mm starts, bunching in $\Delta\varphi_L = [100°, 150°]$ occurs. Approximately 20 % of the particles at z=25 mm can be found at ≈ 130 °. After z=30 mm, the strong bunching disappears, and the particles weakly

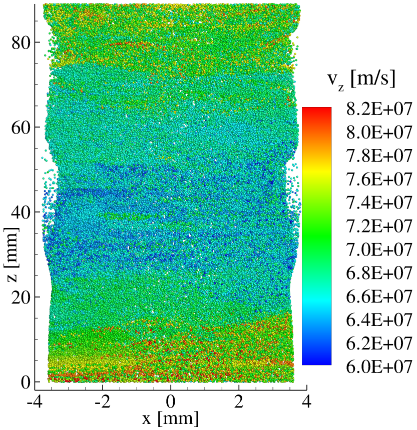


Fig. 4.32.: Scatter plot of electron beam at 30 ns. x-axis scaled up by a factor of six compared to z-axis.

accumulate in φ_L regions where they get accelerated due to over bunching. In the uptaper no specific bunching can be observed, which also is true for the donwtaper.

A further quantity of interest for gyrotron design is the output power gained by different geometries. To compute the power in the EM field, we use a surface integral of the Poynting vector over a defined slice C in the xy-plane at a certain z-coordinate, yielding

$$P(z,t) = \int_C \boldsymbol{n} \cdot \boldsymbol{S} \, da = \frac{1}{\mu_0} \int_C \left(E_x \, B_y - E_y \, B_x \right) \, da. \tag{4.10}$$

Figure 4.35 shows the power of the EM field calculated at different z-positions of the geometry. Due to interferences with the boundary conditions, the values at the ends of the geometry are not to be interpreted as physically reliable values but as numerical-method-dependent artefact. Physically relevant is the z = 87 mm position where a power of ≈ 65 kW at the output port of the resonator is gained.

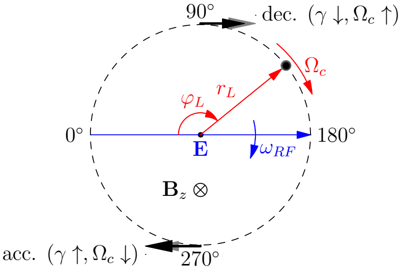

90°
dec. $(\gamma\downarrow, \Omega_c\uparrow)$
Ω_c
r_L
φ_L
0°
180°
E
ω_{RF}
$\mathbf{B}_z \otimes$
acc. $(\gamma\uparrow, \Omega_c\downarrow)$
270°

Fig. 4.33.: Relativistic particle-EM-field interaction.

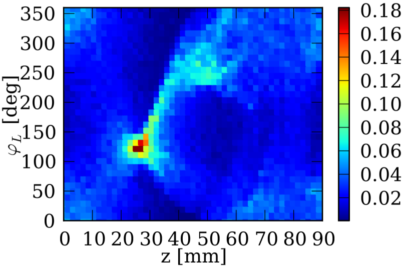


Fig. 4.34.: Relative particle density at different φ_L-angle sections (resolution 10 °-steps) vs. z-axis at 30 ns.

To proof the consistency of the power transfer between the particles and the EM field, figure 4.36 shows the power budget of the particle beam, which has been calculated by adding up the kinetic energies of the particles entering and leaving the domain in each timestep. The difference between the entry and exit gives the power the beam loses when passing the resonator. For the steady state, the beam loses ≈ 65 kW.

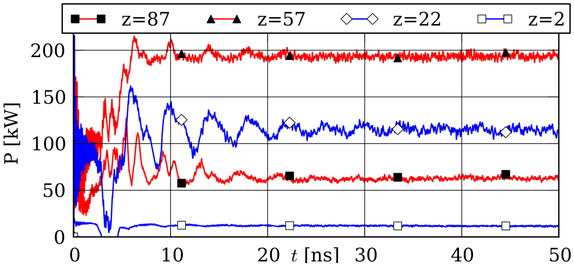
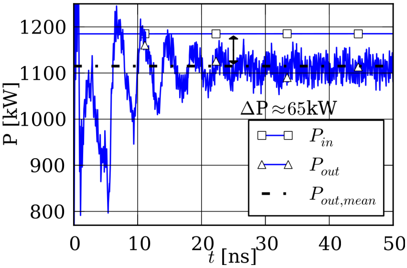


Fig. 4.35.: Power by Poynting vector integral in the EM field.



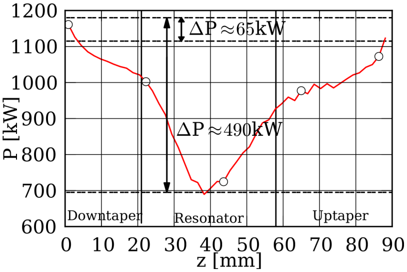


Fig. 4.36.: Particles power budget for in- and output power.

Fig. 4.37.: Particle beam power profile along z-axis at t = 30 ns.

For a more detailed view on the power drain of the beam, figure 4.37 shows the power profile of the beam along the z-axis. Again, we subdivided the beam into 50 equidistant sections, computed the mean kinetic energy in each section, and multiplied it by the particle emission rate. There, we can also see ≈ 65 kW overall power loss in the beam as computed in the particles power budget in figure 4.36. In the peak, the beam loses 490 kW power, but due to the over bunching, it gets most of the power back. Computations with less elements showed an inconsistency between the power drain from the particles and the power transfered to the EM field. Much less power than the beam lost was transfered into the EM field. With increasing spatial resolution the difference diminished until it fully disappeared for the presented setup. In section 4.2.2 this issue is investigated in detail with a grid refinement study.

4.2.1.3. Parallel-Computing Aspects

The computations were performed on 128 processes on the HPC-FF (High Performance Computing For Fusion) cluster of the Jülich Supercomputing Center (JSC), where each compute node features two quad-core Intel® Xeon X55670 CPUs and 24 GB memory with an Infiniband node-node interconnect. The number of 128 processes was chosen due to a detailed analysis of the parallelization capabilities, i.e. strong and weak scaling, for this setup presented in [Stock et al., 2011a].

One nanosecond simulation time took ≈ 10^4 s wall clock time, yielding for 50 ns simulation time ≈ 140 h wall clock time and ≈ 18 000 CPU hours in total. Thus even longer simulation times of 1 000 ns can be reached within a reasonable amount of time, which might be required for more detailed analysis.

4.2.1.4. Conclusion

In [Stock et al., 2012a] for the first time a three-dimensional high-order discontinuous Galerkin approach-based Particle-in-Cell simulation of a 30 GHz gyrotron resonator was performed. The results describe the entire particle-field interaction dynamics, including the electron beam bunching.

The performed computations were conducted in parallel on 128 processes on a LINUX cluster within 6 days of computation time, classifying them as demonstration computations with main output for research, but not as an online design tool for resonators.

Further conclusions and the implications for gyrotron design and research is given in the outlook and conclusion chapter 5.

4.2.2. Grid Convergence Study for a 30 GHz Resonator

In section 4.2.1 we investigated a 30 GHz resonator in detail. The chosen grid resolution was the result of a grid refinement study. Table 4.4 summarizes the different grids used in the study, which is the topic of this section. We investigated pure tetrahedra and pure hexahedra grids. While the tetrahedra grids are totally unstructured and have nearly the same edge length in every element, the hexahedra grids utilize a nearly structured mesh. Figure 4.40 shows the H3 grid which can be divided in two parts, i.e. the inner part which is a Cartesian grid block and an outer part which is a structured grid in circumferential direction.

Name	Elements	DOF
tetrahedra:		
T1	9 965	199 300
T2	30 609	612 180
T3	62 087	1 241 700
T4	89 682	1 793 640
hexahedra:		
H1	8 036	450 016
H2	2 2161	1 241 016
H3	55 536	3 110 016

Tab. 4.4.: Different computational grids used for the grid refinement study. The number of DOF is for a 4^{th}-order space discretization.

We conducted for all grids a computation to at least 30 ns simulation time. The setup is the same as in section 4.2.1. For all grids a $\mathrm{TE}_{2,3}$ mode at 30 GHz could be resolved. To resolve the expected wavelength the lowest resolution with the T1 grid was satisfactory. But further investigations with the different analysis methods showed significant differences between the grids, i.e. different power budgets or oscillations. Obviously the grid resolution has an influence on the result. The most relevant difference in the results can be pointed out by analysing the potential field energy over the simulation time. Figure 4.38 and 4.39 show the potential field energy E_{pot} for the tetrahedral and the hexahedral grids, respectively. We focus the attention on the oscillation appearing after the startup phase when the $\mathrm{TE}_{2,3}$ already is fully evolved at about 10 ns. For low

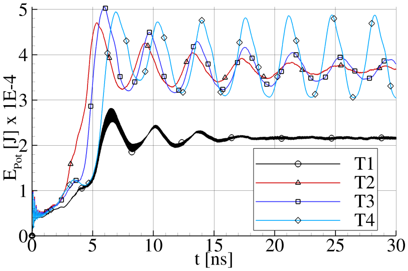


Fig. 4.38.: Potential field energy versus time for different tetrahedral grids according to table 4.4.

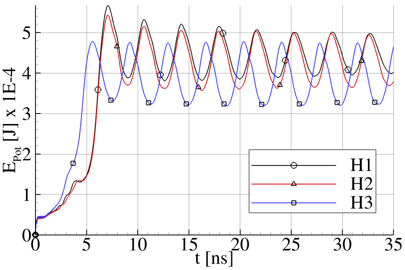


Fig. 4.39.: Potential field energy versus time for different hexahedral grids according to table 4.4.

resolutions of the tetrahedral grids the oscillations damp away. An increasing resolution leads to stable periodically oscillations at approximately 250 MHz. For hexahedral grids the oscillation also appears to be stable for low resolution grids. To explain the difference between the hexahedral and the tetrahedral grids a detailed analysis of the reason for the oscillations has to be made.

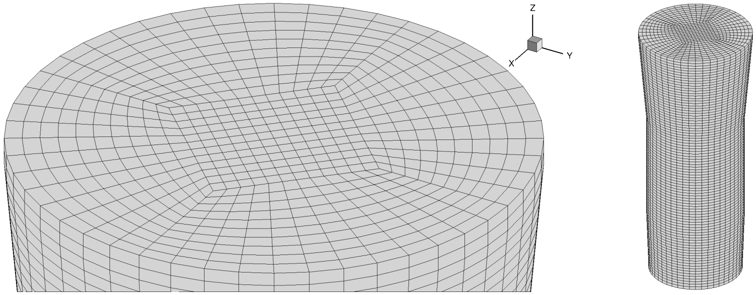


Fig. 4.40.: left: upper part of a hexahedral grid of a 30 GHz resonator; right: entire grid with 55536 elements

Further investigations in this field, mainly performed during the project thesis

of [Ortwein, 2012], revealed that an auto-modulation and a dynamic after-cavity interaction can be associated with the found oscillations.
According to [Private communication with S. Kern, 2012] an auto-modulation in a gyrotron resonator is an oscillatory effect which can be explained in the following steps:

1. After the start-up, the prevailing mode gets saturated into a state where it cannot remain stable. This often is caused by a de-tuning of the interaction frequency towards the optimal resonance frequency.

2. Due to the de-tuning the mode starts to break down, which brings the interaction frequency back into the stable zone.

3. In the stable zone, the mode starts to saturate, until it gets de-tuned, restarting the cycle.

To analyze the oscillation w.r.t. the above definition, we plot the mean gyro frequency Ω_c [1] of the particle beam and ω_{RF} of the EM field at a record point at z=70 mm over an oscillation period in figure 4.41. Additionally, the kinetic and potential field energy and the power at the xy-slice at z=70 mm are plotted. This mixture of quantities shall help to explain the oscillation phenomenon in detail. The point z=70 mm has been chosen, because we expect the phenomenon to be associated with the uptaper. While the cutoff-frequency $\omega_\perp$ for the $\mathrm{TE}_{2,3}$ mode in the resonator section is 29.36 GHz, it will decrease in the uptaper due to the increasing radius. At approximately 25.8 ns Ω_c exceeds $\omega_\perp$. Shortly after this event the power in the field reaches a maximum at 26 ns. After 26 ns ω_{RF} increases and the mode cannot remain stable. The gyro frequency Ω_c breaks down until it reaches at 27 ns a minimum of 28.5 GHz. Between 27 - 29.8 ns Ω_c increases again until it reaches a maximum and the cycle starts again. The power has an unexpected behavior because it grows and breaks down once again during one period, but Ω_c does not follow these changes. Also the integral energies do not follow this behavior. They support the auto-modulation theory very well. The potential energy E_{pot} has a maximum shortly after 26 ns. Similar to the power, a deferred reaction on the beam quantities (i.e. E_{kin} and Ω_c) can be observed. The kinetic energy follows Ω_c perfectly. For decreasing Ω_c the kinetic energy increases and vice versa. Even though the behavior of the power and ω_{RF} cannot fully be explained with the auto-modulation, their overall tendency fits into the phenomenon.

[1] We cut the beam into 40 equidistant slices along the z-axis and calculate the mean Ω_c in each slice.

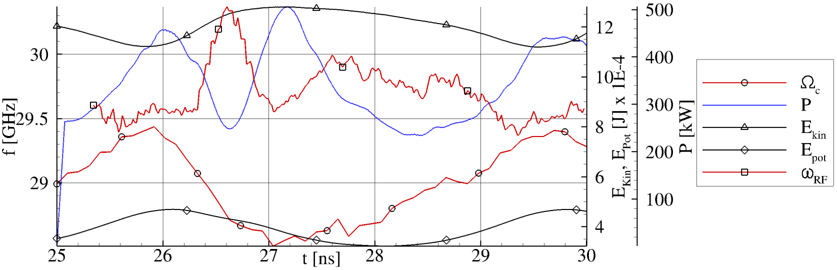


Fig. 4.41.: Different quantities versus time recorded over an auto-modulation period.

A reason for the auto-modulation can be found in the absorbing boundary condition at the top of the resonator, which is explained in detail in the appendix section B.3.2. They should act as an open ending of the resonator. In reality a wave guide is following the uptaper. However, the absorbing boundary condition cannot approximate this wave-guide. The biggest problem with the BC is the reflections that it generates. Only waves perpendicularly incoming to the boundary surface are absorbed perfectly. Incoming waves at angles below 90° are partially reflected. The smaller the angle gets, the more reflection appears. Figure 4.42 shows the reflection schematically. Unfortunately the waves in the resonator have angles close to 0° leading to strong reflections. A wave traveling through the resonator is shown in figure 4.43.
The same problem has been documented in [Illy, 1997]. Absorbing BCs for Maxwell's equations are still under research and a variety of different approaches exist in order to overcome the problem of reflections, e.g. perfectly matched layers [Taflove, 1998, Hagstrom et al., 2010].
The electromagnetic dipole verification example for our DG Maxwell solver presented in [Neudorfer et al., 2009] provides a good assessment on the reflection appearing with the absorbing boundary condition. The dipole is generated by a source term, yielding

$$\mathbf{g} = -\frac{1}{\epsilon_0}(0,\, 0,\, q\omega d\cos(\omega t),\, 0,\, 0,\, 0,\, -\chi q,\, 0)^T, \tag{4.11}$$

with $\omega = 6.284 \times 10^8\,\mathrm{s}^{-1}$, $q = 1\,\mathrm{C}$ and $d = 1\,\mathrm{m}$. The computational domain measures $1 \times 1 \times 1\,\mathrm{m}^3$ and has approximately 2.5 mio DOF. Figure 4.44 shows

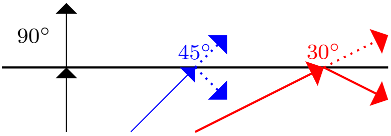


Fig. 4.42.: Reflection of waves arriving with different angles of impact. Waves incoming perpendicular to the boundary no reflection appears. For waves with an impact angle near to 0° the reflection is at maximum.

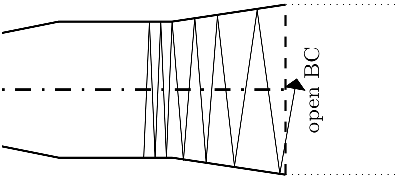


Fig. 4.43.: Trace of a wave traveling through the resonator of a gyrotron.

a three-slice of the $|\boldsymbol{E}|$-field of the dipole. The reflections can be found in figure 4.45, where a cut in the xy-plane at z=0 for the radial electric field E_r is shown. While the dipole in the centre has circular field lines, the outer regions towards the boundary have nearly straight field lines. This is due to reflections caused by the absorbing BC.

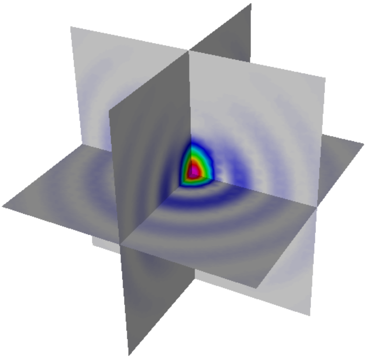


Fig. 4.44.: Three-slice showing $|\boldsymbol{E}|$ of the electromagnetic dipole example (4.11).

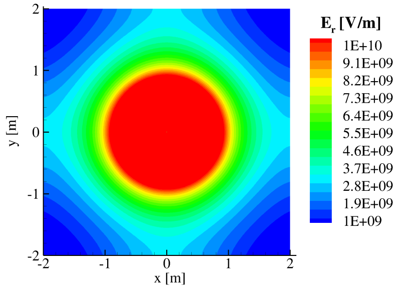


Fig. 4.45.: Radial electrical field E_r of an electromagnetic dipole in the x-y-plane.

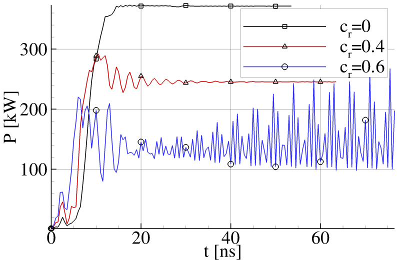


Fig. 4.46.: Power versus time for different reflection coefficient c_r calculated with SELFT.

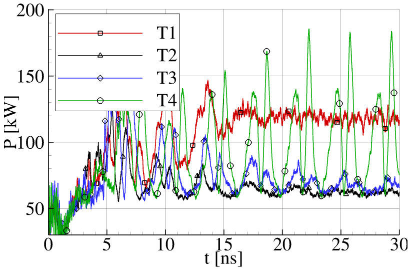


Fig. 4.47.: Power versus time at z=87 mm for different tetrahedral grids, according to table 4.4.

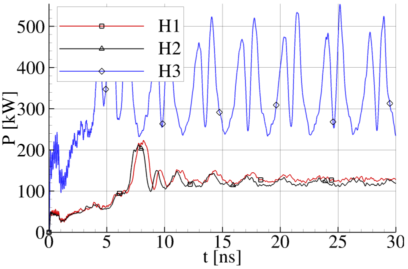


Fig. 4.48.: Power versus time at z=87 mm for different hexahedral grids, according to table 4.4.

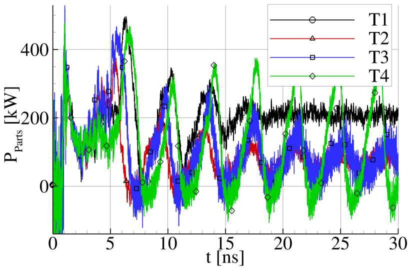


Fig. 4.49.: Particle power budget versus time for tetrahedral grids, according to table 4.4.

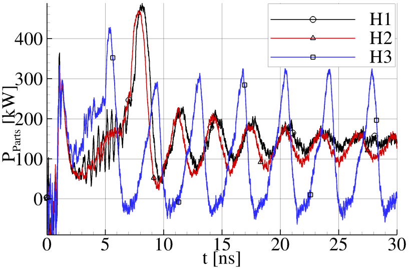


Fig. 4.50.: Particle power budget versus time for hexahedral grids, according to table 4.4.

Tests performed by S. Kern with the SELFT code for the same setup using different reflection coefficients for the output port revealed that an auto-modulation phenomenon appeared. Figure 4.46 shows the power computed for different reflection coefficients c_r versus time. With increasing reflection an auto-modulation can be found. Additionally, the power decreases with increasing reflection coefficients. This phenomenon could also be observed in the DG-PIC calculations. With increasing resolution the power calculated at the output port was decreasing and the oscillation was growing. Figures 4.47 and 4.48 show the power for the different tetrahedral and hexahedral grids, respectively. From T1 to T2 the power was decreasing. For the grid T3 the power is rising again and grid T4 has the fully evolved oscillation. The mean value of the power for grid T4 is 100 kW, which is even more than for grid T3. For the hexahedral girds H1 and H2 a decreasing of the power can be observed with a finer resolution. The mean power is decreasing from 130 to 120 kW. The highly resolved H3 grid has a jump in the mean power to 345 kW which does not fit into the convergence tendencies of H1 and H2. A clear convergence, both for the hexahedral and tetrahedral grids, cannot be concluded from these results.

Grid	$\Delta\mathrm{P}_{\mathrm{particles}}$	P_{out}	$\Delta\mathrm{P}_{\mathrm{loss}}$
tetrahedra:			
T1	200	125	75
T2	65	65	0
T3	70	80	10
T4	125	100	25
hexahedra:			
H1	145	130	15
H2	130	120	10
H3	93	345	-252

Tab. 4.5.: Power budget and consistency for different grids used to simulate a 30 GHz resontor.

As counterpart to the power recorded in the EM field, in figures 4.49 and 4.50 the particle power budget, $\Delta P_{Parts} = P_{in} - P_{out}$, for the different grids is shown. For many grids a mean value has to be calculated since the oscillation is too strong. While for H1 and H2 the power in the field has a stable amplitude the particle power budget gets damped. For the tetrahedral grids the particle power budget is qualitatively very similar to the field power.

This brings us back to the question raised in section 4.2.1 on the convergence criteria for the consistency between beam power drain and power measured in the field by the Poynting vector integral. Table 4.5 shows the power consistency for the different tetrahedral grids, i.e. ΔP_{loss}. For the hexahedral grids and the T4 grid, the mean values can be taken as reference. For the T2 grid, a consistency was found, which is the grid used for a detailed analysis of the 30 GHz resonator in section 4.2.1. All other grids have inconsistent budgets, especially the power budget for H3 seems to be unphysical. However, the thesis, that a consistency of the power budget might be a good criteria for the grid convergence seems not to hold when considering the reflections as major influence of the results.

Explicitly said: The gird T2 resolves the particle-field-interaction accurately enough in order to excite a $\mathrm{TE}_{2,3}$ mode and has at the same time enough dissipation to damp the reflections. If the resolution increases the dissipation is not big enough to damp the reflections.

The inconsistencies in the power budget of the particles can be explained by an interaction between the particles and the EM field appearing in the uptaper, i.e. a dynamic after-cavity interaction (ACI). This interaction leads to an overbunching giving energy back from the field to the beam. Details on this effect are given in [Ortwein, 2012]. The effect varies w.r.t. the grid resolution.

Another interesting aspect is the difference between the hexahedral and the tetrahedral meshes. While the tetrahedral meshes only evolve a stable oscillation for very high resolution, the hexahedral grids already fully evolve a stable auto-modulation for the lowest resolution. This can be explained by the different discretization in z-direction. While the hexahedral grid has a quasi-Cartesian grid in z-direction the tetrahedral grid is fully unstructured in z-direction. For the tetrahedral grid, waves are dissipated in all directions, while the hexahedral grid propagates the waves much better in z-direction. This can be the reason why a hexahedral grid is much more vulnerable for reflections than a tetrahedral grid. In other words: The numerical dissipation is much lower for the hexahedral grids. Hence, the reclections are not damped away.

Five major conclusions can be drawn from this grid refinement study:

1. the absorbing BC leads to reflection that causes auto-modulation effects,
2. an increasing grid resolution does not lead to a convergence of the power-budget,
3. the power calculated in the field and for the beam is grid-resolution-dependent,

4. a proper BC without reflections has to be implemented,

5. if a non-reflecting BC is available, the grid resolution should be only as high as required to resolve the desired wavelength, resulting in much less elements and a much shorter simulation time.

The result of this investigation also casts a shadow of doubt on the verification of the Maxwell solver. Without having tested the absorbing BC properly it is a dangerous enterprise to start such complex computations as the resonator simulations. However, though it has been missed to understand the influence of the reflections in advance of the computation campaign, this investigation helps to understand the results better. Furthermore, we could show that the auto-modulation effect could be physically correctly reproduced by our DG-PIC code. This is important as these transient effects are the reason why simulations with a full-wave DG-PIC solver are desired. They cannot be reproduced and analysed in the same quality and depth with SOTA design codes.

A consequence of these findings would be the implementation of non-reflective BCs. In [Illy, 1997] an absorbing BC proposed by Higdon [Higdon, 1986, Higdon, 1987] is used for resonator calculations within a FDTD framework. In a next step it should be investigated if these BCs are possible to be used within the DG method and if they solve the problem of reflections. Another approach to lower the influence of the absorbing boundary condition would be the introduction of a damping zone, as proposed in the high-order DG simulations of the launcher antenna of a 170 GHz gyrotron by [Neudorfer et al., 2012]. Such a damping zone consists of a coarsened grid at the output port, dissipating the outgoing waves by successively lowering the resolution. Furthermore, it should be investigated if a perfectly matched layer BC can be applied within the DG-PIC framework. Here, the approach proposed in [Hagstrom et al., 2010] is an interesting startingpoint, since it has been developed for the specific needs of DG methods.

4.2.3. W7-X 140 GHz Resonator

In this section we simulate the 140 GHz 1 MW continuous wave gyrotron aimed for the W7-X Stellarator [Dammertz et al., 2002] with our high-order DG-PIC code and compare the results to those achieved with SELFT. The section is based on [Stock et al., 2012b].

4.2.3.1. Computational Setup

Due to the high computational effort of the DG-PIC simulations we have to use SELFT in advance to get a reasonable setup. Initially we start with a soft excitation (SE) setup in order to excite a $\mathrm{TE}_{28,8}$ mode by exceeding its starting current. After a short startup phase we switch to a hard excitation (HE) setup, where the mode's starting current exceeds the actual operating current, in order to raise the efficiency and the output power to a level that is desired in continuous operation mode of the 140 GHz gyrotron. Table 4.6 summarizes the chosen parameters for the different setups where E_p is the electron's energy, α the ratio of perpendicular to parallel electron velocity, γ the relativistic factor, B_z the static magnetic field, r_L the Larmor radius, R_e the radius of the hollow electron beam and I the beam current.

Symbol	[unit]	Soft Excitation	Hard Excitation
Elementary Parameters:			
q_p	[C]	$-1.60217653 \times 10^{-19}$	-
m_p	[kg]	$9.1093826 \times 10^{-31}$	-
MPF		$\times 10^6$	-
Velocity & Energy Parameters:			
E_p	[keV]	81.25	-
α		1.3	-
γ	[-]	1.159	-
$v_\perp$	[m/s]	1.201245755×10^8	-
$v_{\|\|}$	[m/s]	0.924035194×10^8	-
$\|\mathbf{v}\|$	[m/s]	1.515530401×10^8	-
$\|\mathbf{v}\|$	[c]	0.505526527	-
Gyration Parameters:			
B_z	[T]	5.587	5.537 (↓)
r_L	[mm]	0.14168	0.14296 (↑)
R_e	[mm]	10.1	-
Ω_c	[GHz]	134.9385	133.7309 (↓)
Emission Parameters:			
U	[kV]	81.25	-
I	[A]	44	40 (↓)
Emission rate	[parts/ns]	270 455	249 660 (↓)

Tab. 4.6.: Setup parameters for the 140 GHz gyrotron simulation for soft and hard excitation.

The simulation domain covers the complete resonant cavity including uptaper and downtaper. In total a 91 mm long section around the resonant cavity has been discretized with 163 436 hexahedral elements, shown in figure 4.51. A 5^{th}-order DG space discretization leads to approximately 16 million degrees of freedom.

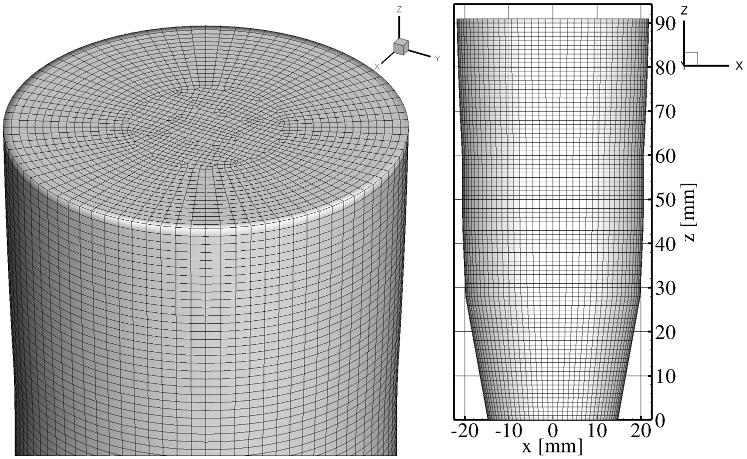
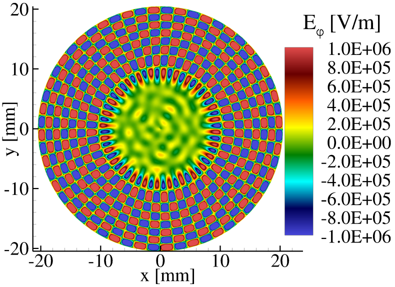


Fig. 4.51.: Computational grid with 163 436 hexahedral elements.

4.2.3.2. DG-PIC Results

The computations were performed on the CRAY-Hermit cluster of the high-performance computing centre in Stuttgart (HLRS) and on the JUROPA-HPC-FF cluster of the Jülich supercomputing centre (JSC) (both in Germany) with 512 MPI-parallel processes. One nanosecond simulation time requires approximately 4.5 hours of computational time leading to a total computation time of 10 days (for 50 ns).
Introductory we discuss the integral energy quantities, shown in figure 4.53 over a period of 50 ns. At 30 ns the operation parameters are switched from SE to HE. After a short initial phase until 15 ns a $TE_{28,8}$ mode prevails and a steady operation point is reached in the SE setup. After switching to the HE setup at 30 ns the kinetic energy drops drastically while the potential field energy



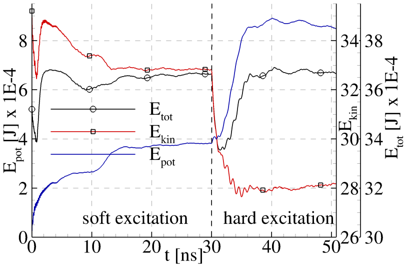
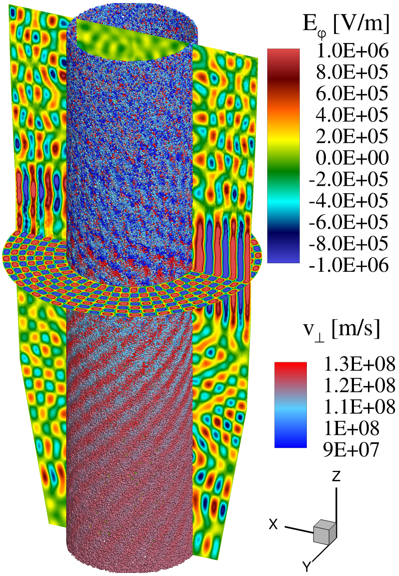


Fig. 4.52.: E_φ in xy-plane at z=51 mm (t=50 ns).

Fig. 4.53.: Integral system energies over time. Left scale: E_{pot}=Energy stored in the RF field, right scale: E_{kin}=energy of the electron beam, E_{tot}=total stored energy.

rises in the magnitude. This effect was desired for the HE setup. Figure 4.52 shows a slice in the xy-plane at z=51 mm with the E_φ-field at t=50 ns. Here, a $\mathrm{TE}_{28,8}$ mode pattern is clearly visible. Figure 4.54 shows an isometric view on the E_φ-field and the hollow beam for t=50 ns. There as well a $\mathrm{TE}_{28,8}$ mode pattern is visible. The particle beam has a helical pattern matching to the TE field pattern in the xy-slice. Below the xy-slice in the pattern is very uniform. Above, a disturbed pattern is visible, indicating the main interaction is over and the resonance condition is not matched any more. Furthermore the color code of the beam, showing the magnitude of the $v_\perp$ component, tells us that the energy below the resonator is higher than above, which is in accordance with the fact that energy is transfered from the beam to the EM field in the resonator. In figure 4.55 a slice in the xz-plane at y=0 for t=50 ns shows the E_y-field. The eight wavelengths across the diameter can be counted; four on each half of the slice. The pattern is not symmetric w.r.t. the z-axis, which can be explained by the fact that the mode has not fully stabilized yet.

In contrast to experimental measurements we are able to record the EM fieldstrength at arbitrary points in the domain to perform a frequency spectrum analysis. We introduce a record point at $(x, y, z) = (11.5, 0, 50)$ mm in the res-



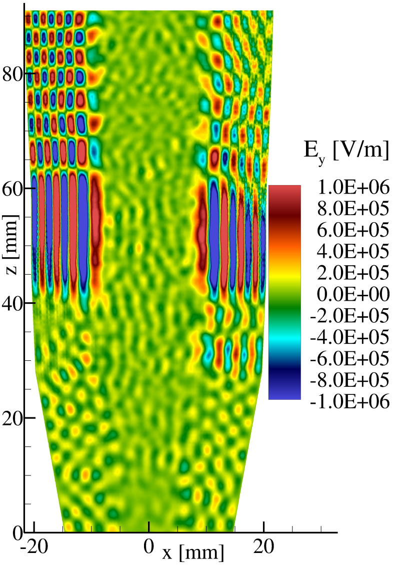


Fig. 4.54.: E_φ iso view with particle hollow beam at t=50 ns.

Fig. 4.55.: E_y in x-z plane at y=0 mm. at t=50 ns.

onator. Figure 4.56 shows the frequency spectrum by a discrete FFT at the record point for a time window of 20 ns in HE setup. Clearly, the 140 GHz frequency peak associated with the TE$_{28,8}$ mode is visible. The second harmonic of this oscillation at 280 GHz can also be found. Some lower peaks at other frequencies indicate that we have parasitic oscillations at lower powers.

To compute the output power in the EM field, we use a surface integral of the Poynting vector over a defined slice C in the xy-plane at a certain z-coordinate (see eq. (4.10)). Figure 4.57 shows the temporal evolution of the power in the EM field at four different slices in the xy-plane versus time. A similar qualitative behavior as for the integral energies shown in figure 4.53 is visible. At the output port (z=80 mm) a power of approximately 500 kW can be calculated.

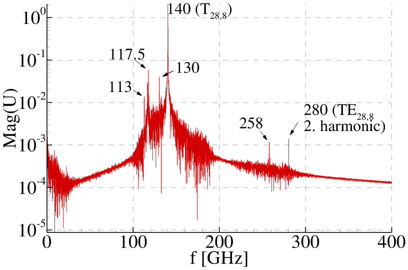


Fig. 4.56.: Frequency spectrum in the resonator recorded at (x,y,z)=(11.5,0,50) mm for t=30 − 50 ns.

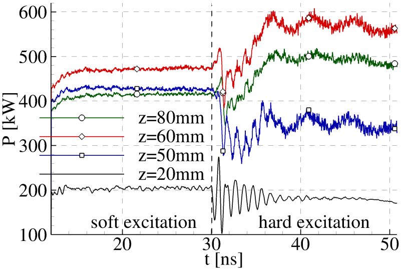


Fig. 4.57.: Power versus time by Poynting vector integral over xy-plane slices at different z-axis points.

4.2.3.3. SELFT-Results

The following results were kindly provided by A. Schlaich and S. Kern of the gyrotron research group at the IHM (KIT).

The 140 GHz resonator was simulated with the same parameter setup with the SELFT code. In contrast to the full-wave DG-PIC code, SELFT uses a preselected mode spectrum of 25 different TE-modes. Figure 4.59 shows the output power of the most relevant modes over time. Here, the switch from HE to SE is shifted from 30 ns to 55 ns because the simulation reaches stationarity only after 30 ns. The $TE_{28,8}$ mode prevails much later as dominant frequency compared to the DG-PIC approach. This is probably caused by a difference in the output boundary condition: While SELFT imposes a continuous waveguide boundary condition, HALO implements an irradiating open end, which actually introduces a non-zero output reflection factor (see section 4.2.2). This additional reflection usually enhances and accelerates the oscillation processes. In the startup phase for a short period a $TE_{27,8}$ mode is dominant and vanishes later. The output port has a power of 1 000 kW. The presented DG-PIC method shows a good agreement in the predicted operating parameters for the soft and hard excitation region and for the dominating line in the predicted oscillation frequency spectrum (shown in figure 4.58 for SELFT), while some spurious oscillations at 113 and 117.5 GHz may again be enhanced in the DG-PIC simulation by the artificial output reflections. Especially, the 130 GHz peak could be found with both codes. It is associated with the $TE_{27,8}$ mode, which is expected to play an important role in a mode competition with the $TE_{28,8}$ mode.

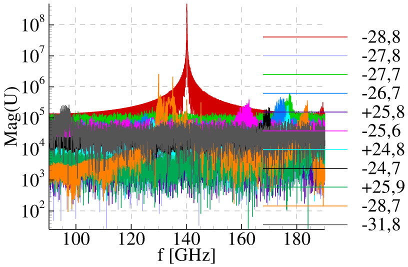


Fig. 4.58.: SELFT power spectrum at t=90 ns of the relevant TE modes included in the simulation.

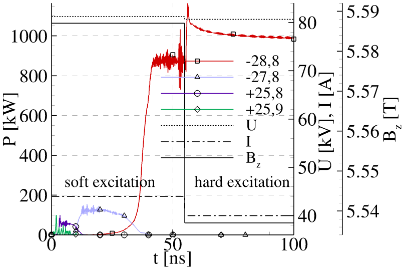


Fig. 4.59.: Power in selected modes versus time calculated with SELFT. U, I and B_z are the beam parameters.

4.2.3.4. Conclusions

Through first simulations, the feasibility of simulating highly oversized gyrotron resonators over physically relevant time frames with a high-order DG-PIC code was demonstrated. While special questions like the appropriate implementation of dedicated boundary conditions still have to be solved, the way towards a simulation tool free of the otherwise necessary approximations and model limitations is shown.[2]

Even if 10 days of computational time was required to simulate 50 ns simulation time, the extra costs of this type of simulation is worth it. We can study beam misalignment, mode competition, parasitic modes, dynamic after cavity interaction (as expected behind the 130 GHz peak in the frequency spectrum) and gain unique insights into driving mechanisms.

[2] by S. Kern, cited from [Stock et al., 2012b]

5. Conclusions and Outlook

5.1. Conclusions

The aim of this thesis was to show the ability of a full-wave, transient, high-order DG-PIC code to simulate state-of-the-art gyrotron resonators for design issues. An essential feature for the DG-PIC method is the parallelization of the high-order shape function deposition. In section 3.6.1 a new parallelization concept has been developed, verified and used for the high-performance computations.

In section 4.2.1 a benchmark of a 30 GHz resonator has been presented, showing the ability of our DG-PIC code to compute real plasma devices. An analysis of the physical properties provides new insights to the resonator physics, e.g. a new beam statistics showing the azimuthal bunching and the effect of over-bunching in detail.

Section 4.2.2 revealed the problems and effects caused by reflections in the absorbing boundary conditions for the resonator simulations. Especially the power quantities are critically affected by these reflections. However, the grid resolution for a proper boundary condition without reflection should be chosen by the wave-resolution criteria only.

In section 4.2.3 a 140 GHz gyrotron resonator was simulated, which is aimed for the W7-X Stellarator fusion reactor. The results have been compared to those achieved with the design tool SELFT. They showed good qualitatively agreement and could verify an oscillation frequency, which is expected to be a competitive mode. The computational time of 10 days on a SOTA HPC cluster still is not compatible with the needs of fast design processes.

However, the ability of a high-order, full-wave, transient DG-PIC code to simulate a SOTA gyrotron in a reasonable amount of time was demonstrated. In the outlook section 5.2 the potentials of our code for future research activities on gyrotrons will be discussed.

The development of multi-rate Adams-Bashforth (MRAB) methods for the PIC method presented in [Stock, 2009] has been pursued, and a MRAB method for the hyperbolic divergence cleaning technique has been derived in section 3.5. Even if the MRAB-PHM approach could not speed-up the computation, the concept of a time domain decomposition for the PHM system has been demonstrated. An alternative time-stepping method could maybe solve this problem. The new multi-rate RK approach by [Grote et al., 2012] is an interesting candidate for this problem. Furthermore a stability region optimization as presented in [Niegemann et al., 2012] could be considered to make the approach more efficient.

5.2. Outlook

State-of-the-art fast gyrotron design tools like SELFT have limitations in the range of parameters and operation setups due to the simplification used within them. A high-order, transient, full-wave DG-PIC code, as used in this thesis, is able to go beyond these limitations and to give insights to problems that cannot be simulated with the fast codes. It is expected "*...that computations like the presented resonator simulation are essentially needed if beam misalignments or transient problems such as mode competition or parasitic modes are to be investigated. In future work ...*" our DG-PIC code can help to verify "*...a recent theory which questions coaxial super power gyrotron feasibility [Dumbrajs and Nusinovich, 2004], concluding that maximal values of the azimuthal index exist, beyond which stationary single mode operation of a gyrotron is not possible, due to the onset of stochastic oscillations of the radio-frequency fields.*"[1]

Furthermore, the presented DG method was applied for the simulation of the quasi-optical antenna (launcher) of the 2 MW, 170 GHz gyrotron in [Neudorfer et al., 2012]. A coupled simulation of the resonant cavity and the launcher would provide new insights into the physics of electromagnetic wave propagation under influence of a particle beam reaching into the antenna and the effects of the antenna on the resonant cavity, so called after cavity interactions [Zapevalov and Moiseev, 2004]. The presented code framework is expected to be capable of performing such a large simulation at the edge of todays HPC capabilities. In [Stock et al., 2012c] the first coupled resonator and launcher simulation of the 1 MW, 140 GHz gyrotron was presented. Figure 5.1 shows an isometric view on the electric field and the electron hollow beam of the coupled simulation.

Of course, further code improvements have to raise the efficiency of our DG-PIC code. Improvements on the parallelization in [Neudorfer et al., 2013] demonstrated the ability of our high-order DG-PIC scheme to simulate a 2

[1] R. Schneider, EFDA proposal for a HPC-FF project at the Juelich Supercomputing Centre, 2012

MW, 170 GHz gyrotron resonator with over 16 000 processes within one day, which is a reasonable computation time for design purposes. The achieved results provide new insights to the mode excitation, mode competition and power budget aspects. Since the 170 GHz gyrotron plays a crucial role in the development of future fusion reactors like ITER we expect a high relevance of this work for the research in the field of gyrotron design.
Also new developments in the area of spatial discretization methods have to be considered. Very recently, a discontinuous Galerkin Spectral Element method (DG-SEM) from [Kopriva, 2009] was coupled with the PIC module described in this thesis. The higher efficiency of the DG-SEM approach allows a speed-up of nearly 70 % for the entire DG-SEM-PIC scheme.
The thesis of [Neudorfer, 2012] and this thesis provide the building blocks and the proof that a high-order DG-PIC code is capable of simulating gyrotrons with high mode-indices, and moreover can provide reasonable input to the analysis of them. Considering further code improvements and the growing HPC capacities, it is expected that our high-order DG-PIC code is fully capable for design tasks in the development of gyrotrons within the scope of the near future.

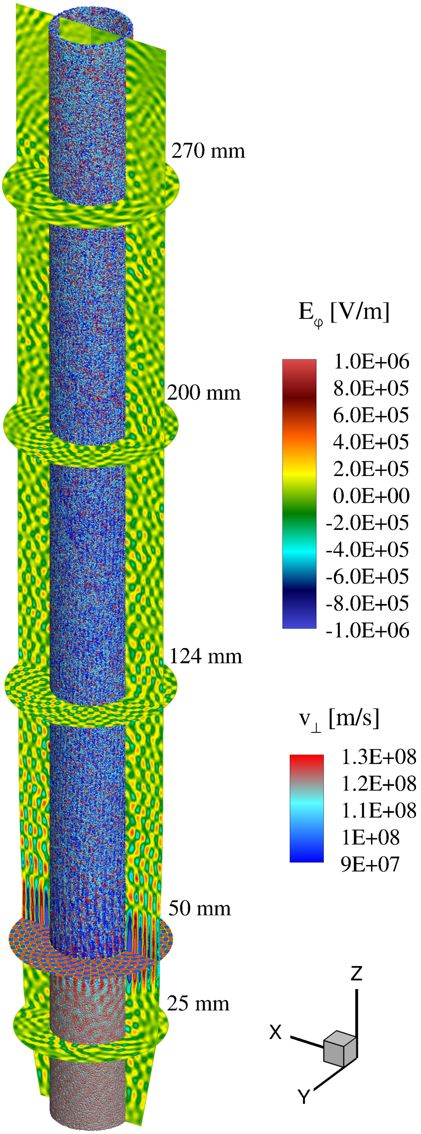


Fig. 5.1.: E_φ-field and particle beam with $v_\perp$ at 17 ns simulation time of a 140 GHz gyrotron (resonator and launcher). Simulated with the DG-PIC code on 1536 processes in approximately 2.5 weeks.

Bibliography

[Andrus, 1993] Andrus, J. (1993). Stability of a multi-rate method for numerical integration of ODE's. Computers & Mathematics with Applications, 25(2):3–14.

[Andrus, 1979] Andrus, J. F. (1979). Numerical solution of systems of ordinary differential equations separated into subsystems. SIAM Journal on Numerical Analysis, 16(4):605–611.

[Armstrong, 1967] Armstrong, T. P. (1967). Numerical studies of the nonlinear Vlasov equation. Phys. Fluids, 10(6):1269–1281.

[Ashutosh and Jain, 2012] Ashutosh, B. R. C. and Jain, P. K. (2012). Multimode behavior of a 42 GHz, 200 kW gyrotron. Progress In Electromagnetics Research B, 24:75–91.

[Assous et al., 1993] Assous, F., Degond, P., Heintze, E., Raviart, P., and Segre, J. (1993). On a finite-element method for solving the three-dimensional Maxwell equations. J. Comput. Phys., 109:222–237.

[Avramides et al., 2012] Avramides, K. A., Pagonakis, I. G., Iatrou, C. T., and Vomvoridis, J. L. (2012). EURIDICE: a code-package for gyrotron interaction simulations and cavity design. In European Physical Journal Web of Conferences, volume 32, page 04016.

[Barroso et al., 1999] Barroso, J. J., Kostov, K. G., and Correa, R. A. (1999). Electromagnetic simulation of a 32-GHz, TE021 gyrotron. IEEE Transactions on Plasma Science, 27(2):384–390.

[Bashforth, 1883] Bashforth, F. (1883). An attempt to test the theories of capillary action by comparing the theoretical and measured forms of drops of fluid. With an explanation of the method of integration employed in constructing the tables which give the theoretical form of such drops, by J.C. Admas. Cambridge Univ. Press [III.1].

[Bettencourt and Greenwood, 2008] Bettencourt, M. T. and Greenwood, A. D. (2008). Performance improvements for efficient electromagnetic Particle-In-Cell computation on 1000s of CPUs. IEEE Transactions on Antennas and Propagation, 56(8):2178–2186.

[Birdsall and Langdon, 1991] Birdsall, C. and Langdon, A. (1991). Plasma Physics via Computer Simulation. Adam Hilger, Bristol, Philadelphia, New York.

[Blahovec et al., 2000] Blahovec, J., Bowers, L., Luginsland, J., Sasser, G., and Watrous, J. (2000). 3-D ICEPIC simulations of the relativistic klystron oscillator. Plasma Science, IEEE Transactions on, 28(3):821 – 829.

[Borie and Jödicke, 1992] Borie, E. and Jödicke, B. (1992). Self-consistent theory of mode competition for gyrotrons. International Journal of Electronics, 72:721–744.

[Boris, 1970] Boris, J. P. (1970). Relativistic plasma simulations – Optimization of a hybrid code. In Proc. 4th Conf. on Num. Sim. of Plasmas, pages 3–67, NRL Washington, Washington DC.

[Botton et al., 1998] Botton, M., Antonsen, T. M., Levush, B., Nguyen, K. T., and Vlasov, A. N. (1998). MAGY: a time-dependent code for simulation of slow and fast microwave sources. IEEE Transactions on Plasma Science, 26:882–892.

[Bouzarth and Minion, 2010] Bouzarth, E. L. and Minion, M. L. (2010). A multirate time integrator for regularized stokeslets. J. Comput. Phys., 229:4208–4224.

[Bui et al., 2005] Bui, T., Ives, L., Verboncoeur, J., and Birdsall, C. (2005). Code development of a 3D finite element Particle-In-Cell code with adaptive meshing. In IEEE International Conference on Plasma Science, 2005. ICOPS '05. IEEE Conference Record - Abstracts, pages 269–269. IEEE.

[Burau et al., 2010] Burau, H., Widera, R., Hönig, W., Juckeland, G., Debus, A., Kluge, T., Schramm, U., Cowan, T., Sauerbrey, R., and Bussmann, M. (2010). PIConGPU: A fully relativistic Particle-in-Cell code for a GPU cluster. Plasma Science, IEEE Transactions on, 38(10):2831 –2839.

[Cai et al., 1992] Cai, S. Y., Antonsen, T. M., Saraph, G., and Levush, B. (1992). Multifrequency theory of high power gyrotron oscillators. International Journal of Electronics, 72:759–777.

[Campos Pinto et al., 2009] Campos Pinto, M., Jund, S., Salmon, S., and Sonnendrücker, E. (2009). Charge conserving FEM-PIC schemes on general grids. only published as INRIA internal report; available at http://hal.inria.fr/hal-00311429/.

[Candel et al., 2009] Candel, A., Kabel, A., Lee, L., Li, Z., Limborg, C., Ng, C., Prudencio, E., Schussman, G., Uplenchwar, R., and Ko, K. (2009). Parallel high-order finite element method for accurate field computations in wakefield and PIC simulations. Technical report. SLAC-PUB-13667, SLAC National Accelerator Laboratory.

[Carpenter et al., 1993] Carpenter, M. H., Gottlieb, D., Abarbanel, S., and Don, W.-S. (1993). The theoretical accuracy of Runge-Kutta time discretizations for the initial boundary value problem: A careful study of the boundary error. Technical report, Langley Research Center.

[Carpenter and Kennedy, 1994] Carpenter, M. H. and Kennedy, C. A. (1994). Fourth-order 2n-storage Runge-Kutta schemes. Technical report, Langley Research Center.

[CGNS, 2012] CGNS (2012). CFD general notation system. http://cgns.sourceforge.net/.

[Chapman et al., 2007] Chapman, B., Jost, G., and Van Der Pas, R. (2007). Using OpenMP: portable shared memory parallel programming, volume 10. The MIT Press.

[Chapman, 2007] Chapman, S. J. (2007). Fortran 95/2003 for Scientists and Engineers: 1995-2003. Mcgraw-Hill Higher Education.

[Chen et al., 2011] Chen, G., Chacón, L., and Barnes, D. (2011). An energy- and charge-conserving, implicit, electrostatic particle-in-cell algorithm. J. Comput. Phys., 230(18):7018–7036.

[Cockburn et al., 2000] Cockburn, B., Karniadakis, G., and (Eds), C. S. (2000). Discontinuous Galerkin Methods: Theory, Computation and Application. Springer, Berlin, Heidelberg.

[Cockburn and Shu, 2001] Cockburn, B. and Shu, C.-W. (2001). Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. Journal of Scientific Computing, 16:173–261. 10.1023/A:1012873910884.

[Courant et al., 1928] Courant, R., Friedrichs, K., and Lewy, H. (1928). Über die partiellen Differenzengleichungen der mathematischen Physik. Mathematische Annalen, 100(1):32–74.

[Crow and Chen, 1994] Crow, M. and Chen, J. (1994). The multirate method for simulation of power system dynamics. Power Systems, IEEE Transactions on, 9(3):1684 –1690.

[CST, 2012] CST (2012). CST - Computer Simulation Technologies. http://www.cst.com/.

[Dammertz et al., 2002] Dammertz, G., Alberti, S., Arnold, A., Borie, E., Erckmann, V., Gantenbein, G., Giguet, E., Heidinger, R., Hogge, J., Illy, S., Kasparek, W., Koppenburg, K., Kuntze, M., Laqua, H., LeCloarec, G., LeGoff, Y., Leonhardt, W., Lievin, C., Magne, R., Michel, G., Muller, G., Neffe, G., Piosczyk, B., Schmid, M., Schworer, K., Thumm, M., and Tran, M. (2002). Development of a 140-GHz 1-MW continuous wave gyrotron for the W7-X stellarator. Plasma Science, IEEE Transactions on, 30(3):808 – 818.

[Dawson, 1962] Dawson, J. (1962). One-dimensional plasma model. Phys. Fluids, 5(4):445–460.

[Dumbrajs and Nusinovich, 2004] Dumbrajs, O. and Nusinovich, G. S. (2004). Coaxial gyrotrons: past, present, and future (review). IEEE Transactions on Plasma Science, 32(3):934– 946.

[Eastwood, 1991] Eastwood, J. W. (1991). The virtual particle electromagnetic particle-mesh method. Computer Physics Communications, 64(2):252–266.

[Engstler and Lubich, 1997] Engstler, C. and Lubich, C. (1997). Multirate extrapolation methods for differential equations with different time scales. Computing, 58:173–185. 10.1007/BF02684438.

[Flamm et al., 2011] Flamm, J., Jin, J., and Thumm, M. (2011). Wave propagation in advanced gytrotron output couplers. Journal of Infrared, Millimeter, and Terahertz Waves, 32:887–896. doi:10.1007/s10762-011-9800-y.

[Fliflet et al., 1982] Fliflet, A. W., Read, M. E., Chu, K. R., and Seeley, R. (1982). A self-consistent field theory for gyrotron oscillators: application to a low Q gyromonotron. International Journal of Electronics, 53:505–521.

[FORTRAN, 2004] FORTRAN (2004). ISO/IEC 1539-1:2004. Information technology - Programming languages - Fortran - Part 1: Base language. Informally known as Fortran 2003.

[Gassner, 2009] Gassner, G. (2009). Discontinuous Galerkin Methods for the Unsteady Compressible Navier-Stokes Equations. PhD thesis, Faculty of Aerospace Engineering and Geodesy, University of Stuttgart, Germany.

[Gassner and Kopriva, 2011] Gassner, G. and Kopriva, D. (2011). A comparison of the dispersion and dissipation errors of Gauss and Gauss–Lobatto discontinuous Galerkin spectral element methods. SIAM Journal on Scientific Computing, 33(5):2560–2579.

[Gassner et al., 2009] Gassner, G., Lörcher, F., C.-D. Munz, and J. S. Hesthaven (2009). Polymorphic nodal elements and their application in discontinuous Galerkin methods. J. Comput. Phys., 228(5):1573 – 1590.

[Gear, 1974] Gear, C. W. (1974). Multirate methods for ordinary differential equations. Technical report. Illinois Univ., Urbana (USA). Dept. of Computer Science.

[Gear and Wells, 1984] Gear, C. W. and Wells, D. R. (1984). Multirate linear multistep methods. BIT Numerical Mathematics, 24(4):484–502.

[Gomm, 1981] Gomm, W. (1981). Stability analysis of explicit multirate methods. Mathematics and Computers in Simulation, 23(1):34–50.

[Goplen et al., 1995] Goplen, B., Ludeking, L., Smith, D., and Warren, G. (1995). User-configurable MAGIC for electromagnetic PIC calculations. Computer Physics Communications, 87(1-2):54–86.

[Grote et al., 2012] Grote, M., Mehlin, M., and Mitkova, T. (2012). High-order local time-stepping with explicit Runge-Kutta methods. Published in proceedings of SCEE: Conference on Scientific Computing in Electrical Engineering, Zurich, Switzerland.

[Grote and Mitkova, 2011] Grote, M. and Mitkova, T. (2011). High-order explicit local time-stepping methods for damped wave equations. eprint arXiv:1109.4480.

[Grote and Mitkova, 2010] Grote, M. J. and Mitkova, T. (2010). Explicit local time-stepping methods for Maxwell's equations. Journal of Computational and Applied Mathematics, 234(12):3283–3302.

[Gurnett and Bhattacharjee, 2005] Gurnett, D. and Bhattacharjee, A. (2005). Introduction to Plasma Physics: With Space and Laboratory Applications. Cambridge University Press.

[Günther et al., 2001] Günther, M., Kværnø, A., and Rentrop, P. (2001). Multirate partitioned runge-kutta methods. BIT Numerical Mathematics, 41(3):504–514.

[Hagstrom et al., 2010] Hagstrom, T., Warburton, T., and Givoli, D. (2010). Radiation boundary conditions for time-dependent waves based on complete plane wave expansions. Journal of Computational and Applied Mathematics, 234:1988–1995.

[Hairer et al., 2009] Hairer, E., Norsett, S. P., and Wanner, G. (2009). Solving Ordinary Differential Equations I. Springer Verlag, Berlin/Heidelberg/New York.

[Harlow, 1955] Harlow, F. H. (1955). A machine calculation method for hydrodynamic problems. Technical report, Los Alamos.

[HDF5, 2012] HDF5 (2012). HDF5. http://www.hdfgroup.org/HDF5/.

[Hesthaven et al., 2007] Hesthaven, J., Gottlieb, S., and Gottlieb, D. (2007). Spectral Methods for Time-Dependent Problems. Cambridge University Press, Cambridge.

[Hesthaven and Warburton, 2002] Hesthaven, J. and Warburton, T. (2002). Nodal high-order methods on unstructured grids: I. Time-domain solution of Maxwell's equations. J. Comput. Phys., 181(1):186–221.

[Hesthaven and Warburton, 2008] Hesthaven, J. and Warburton, T. (2008). Nodal Discontinuous Galerkin Methods. Springer, New York.

[Higdon, 1986] Higdon, R. (1986). Absorbing boundary conditions for difference approximations to the multi-dimensional wave equation. Mathematics of Computation, 47(176):437–459.

[Higdon, 1987] Higdon, R. (1987). Numerical absorbing boundary conditions for the wave equation. Mathematics of Computation, 49(179):65–90.

[Hindenlang, 2008] Hindenlang, F. (2008). Untersuchung der Zeitschrittbestimmung von DG-Verfahren auf mehrdimensionalen unstrukturierten Gittern. Technical report, Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, Germany.

[Hockney and Eastwood, 1981] Hockney, R. and Eastwood, J. (1981). Computer Simulation using Particles. McGraw-Hill, New York.

[Holtkamp, 2007] Holtkamp, N. (2007). An overview of the ITER project. Fusion Engineering and Design, 82(5–14):427 – 434. Proceedings of the 24th Symposium on Fusion Technology SOFT-24.

[Hopkins et al., 2012] Hopkins, M. M., Boerner, J. J., Barnat, E. V., Crozier, P. S., Bettencourt, M. T., Musson, L. C., Meyer, H. E., Hooper, R., and Moore, C. H. (2012). 3D vacuum arc breakdown simulation: Many challenges and some solutions. In 39th IEEE International Conference on Plasma Science (ICOPS), Edinburgh, Scotland.

[Illy, 1997] Illy, S. (1997). Untersuchungen von Strahlinstabilitäten in der Kompressionszone von Gyrotron-Oszillatoren mit Hilfe der kinetischen Theorie und zeitab- hängiger Particle-in-Cell-Simulationen. Technical report, Karlsruhe, FZKA 6037.

[Issautier et al., 1995] Issautier, D., Poupaud, F., Cioni, J., and Frezoui, L. (1995). A 2D Vlasov-Maxwell solver on unstructured meshes. In Cohen, G., Joly, P., Becache, E., and Roberts, J., editors, Proc. 3rd Intl. Conf. on Math. & Num. Aspects on Wave Prop., 1995, pages 355–371.

[Jackson, 1999] Jackson, J. D. (1999). Classical Electrodynamics. Wiley, New York, 3. ed. edition.

[Jacobs and Hesthaven, 2006] Jacobs, G. and Hesthaven, J. (2006). High-order nodal discontinuous Galerkin particle-in-cell method on unstructured grids. J. Comput. Phys., 214:96 – 121.

[Jacobs and Hesthaven, 2009] Jacobs, G. and Hesthaven, J. (2009). Implicit–explicit time integration of a high-order particle-in-cell method with hyperbolic divergence cleaning. Computer Physics Communications, 180(10):1760 – 1767.

[Jacobs et al., 2006] Jacobs, G., Hesthaven, J., and Lapenta, G. (2006). Simulations of the Weibel instability with a high-order discontinuous Galerkin Particle-In-Cell solver. 44th AIAA Aerospace Sciences Meeting and Exhibit.

[Kern, 1996] Kern, S. (1996). Numerische Simulation der Gyrotron-Wechselwirkung in koaxialen Resonatoren. Technical report, Karlsruhe, FZKA 5837.

[Klöckner, 2010] Klöckner, A. (2010). High-Performance and High-Order Simulation of Wave and Plasma Phenomena. PhD thesis, Brown University, Providence, RI, USA.

[Kopriva, 2009] Kopriva, D. (2009). Implementing Spectral Methods for Partial Differential Equations. Springer, New York.

[Krall and Trivelpiece, 1986] Krall, N. A. and Trivelpiece, A. W. (1986). Principles of Plasma Physics. San Francisco Press, San Francisco.

[Kumar et al., 2010] Kumar, N., Singh, U., Kumar, A., Khatun, H., Singh, T. P., and Sinha, A. K. (2010). Numerical analysis of interaction cavity for 1.5 MW/127.5 GHz gyrotron. Journal of Fusion Energy, 30(1):1–6.

[Kværnø and Rentrop, 1999] Kværnø, A. and Rentrop, P. (1999). Low order multirate Runge-Kutta methods in electric circuit simulation. published at http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.9.3084.

[Langdon and Birdsall, 1970] Langdon, A. B. and Birdsall, C. K. (1970). Theory of plasma simulation using finite-size particles. Phys. Fluids, 13(8):2115–2123.

[Lapenta, 2012] Lapenta, G. (2012). Particle simulations of space weather. J. Comput. Phys., 231(3):795–821.

[Lee et al., 2004] Lee, Y., Munz, C.-D., and Schneider, R. (2004). Lagrangian and symmetry structure of the divergence cleaning model based on generalized Lagrange multipliers. Int. J. of Mod. Phys. C, 15:59–114.

[Lin and Smithe, 2012] Lin, M. and Smithe, D. (2012). CFDTD PIC simulation study on low-frequency oscillations in a gyrotron. In 39th IEEE International Conference on Plasma Science (ICOPS), Edinburgh, Scotland.

[Loercher, 2009] Loercher, F. (2009). Predictor-Corrector Discontinuous Galerkin Schemes for the Numerical Solution fo the Compressible Navier-Stokes Equations in Clomplex Domains. PhD thesis, Faculty of Aerospace Engineering and Geodesy, University of Stuttgart, Germany.

[Lörcher et al., 2007] Lörcher, F., Gassner, G., and C.-D.Munz (2007). A discontinuous Galerkin scheme based on a space-time expansion I. Inviscid compressible flow in one space dimension. J. Sci. Comput., 32(2):175–199.

[Marder, 1987] Marder, B. (1987). A method for incorporating Gauss' law into electromagnetic PIC codes. J. Comput. Phys., 68(1):48–55.

[Melrose, 1989] Melrose, D. B. (1989). Instabilities in Space and Laboratory Plasmas. Cambridge University Press.

[Moore, 1965] Moore, G. (1965). Cramming more components onto integrated circuits. Electronics, 38:114 ff.

[Morse and Nielson, 1969] Morse, R. L. and Nielson, C. W. (1969). One-, two-, and three-dimensional numerical simulation of two-beam plasmas. Phys. Rev. Lett., 23:1087–1090.

[Morse and Nielson, 1971] Morse, R. L. and Nielson, C. W. (1971). Numerical simulation of the Weibel instability in one and two dimensions. Physics of Fluids, 14(4):830–840.

[MPI-Forum, 2009] MPI-Forum (2009). MPI: A message-passing interface standard, version 2.2. Specification.

[Munz et al., 2000a] Munz, C.-D., Omnes, P., and Schneider, R. (2000a). A three-dimensional finite-volume solver for the Maxwell equations with divergence cleaning on unstructured meshes. Computer Physics Communications, 130:83 – 117.

[Munz et al., 2000b] Munz, C.-D., Omnes, P., Schneider, R., Sonnendrücker, E., and Voß, U. (2000b). Divergence correction techniques for Maxwell solvers based on a hyperbolic model. J. Comput. Phys., 161:484 – 511.

[Munz et al., 1999] Munz, C.-D., Schneider, R., Sonnendrücker, E., Stein, E., Voss, U., and Westermann, T. (1999). A finite-volume particle-in-cell method for the numerical treatment of Maxwell-Lorentz equations on boundary-fitted meshes. International Journal for Numerical Methods in Engineering, 44(4):461–487.

[Müller, 1969] Müller, C. (1969). Foundation of the Mathematical Theory of Electromagnetic Waves. Springer, Berlin.

[Neudorfer, 2012] Neudorfer, J. (2012). Numerical Methods and Computational Strategies for Particle-In-Cell Simulations of Large Scale Engineering Applications. PhD thesis, Fakultät für Maschinenwesen der Rheinisch-Westfälischen Technischen Hochschule Aachen, Germany.

[Neudorfer et al., 2009] Neudorfer, J., Munz, C.-D., Stindl, T., Fertig, M., Roller, S., Schneider, R., and Auweter-Kurtz, M. (2009). Particle In Cell Simulation Using a Discontinuous Galerkin Method. IEPC-2009-237, Proceedings of the 31th International Electric Propulsion Conference, Ann Arbor, MI, USA.

[Neudorfer et al., 2011] Neudorfer, J., Stindl, T., Stock, A., Schneider, R., Petkow, D., Roller, S., Munz, C.-D., and Auweter-Kurtz, M. (2011). Three-dimensional simulation of rarefied plasma flows using a high order particle in cell method. In Nagel, W. E., Kröner, D. B., and Resch, M. M., editors, High Performance Computing in Science and Engineering '10, pages 593–604. Springer Berlin Heidelberg. 10.1007/978-3-642-15748-6_43.

[Neudorfer et al., 2012] Neudorfer, J., Stock, A., Flamm, J., Hindenlang, F., Gassner, G., Munz, C.-D., Schneider, R., and Roller, S. (2012). Numerical investigation of high-order gyrotron mode propagation in launchers at 170 GHz. IEEE Transactions on Plasma Science, 40(6):1512–21.

[Neudorfer et al., 2013] Neudorfer, J., Stock, A., Schneider, R., Roller, S., and Munz, C.-D. (2013). Efficient parallelization of a three-dimensional high-order particle-in-cell method for the simulation of a 170 GHz gyrotron resonator. Plasma Science, IEEE Transactions on, 41(1):87 –98.

[Niegemann et al., 2012] Niegemann, J., Diehl, R., and Busch, K. (2012). Efficient low-storage Runge–Kutta schemes with optimized stability regions. J. Comput. Phys., 231(2):364–372.

[Ortwein, 2012] Ortwein, P. (2012). 3D numerical simulation of an automodulation phenomenon in a 30 GHz gyrotron resonator with an explicit high-order discontinuous Galerkin approach-based particle-in-cell method. Technical Report 2012, Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, Germany.

[Peterkin Jr. et al., 2012] Peterkin Jr., R. E., Fleming, T. P., French, D. M., Gensheimer, P. D., Greenwood, A. D., Hoff, B. W., Karrels, D. R., Lambrecht, M., Lockwood, N. P., Mardahl, P. J., and W., T. W. (2012). Predictive, optimized numerical simulation of microwave generation in complex geometries with real materials. In 39th IEEE International Conference on Plasma Science (ICOPS), Edinburgh, Scotland.

[Peterson, 1988] Peterson, A. F. (1988). Absorbing boundary conditions for the vector wave equation. Microwave and Opt. Techn. Lett., 1:62–64.

[Pirrung, 2010] Pirrung, G. (2010). Simulation eines Gyrotron-Resonators mit einem Discontinuous Galerkin basierten 3D Particle-In-Cell Verfahren und die Parallelisierung der Shape-Function. Technical Report 2010, Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, Germany.

[Private communication with G. Latsas, 2011] Private communication with G. Latsas (2011). rivate communications with G. Latsas, Sep. 2011.

[Private communication with S. Kern, 2012] Private communication with S. Kern (2012). private communications with S. Kern, Mar. 2012.

[Quandt, 2010] Quandt, M. (2010). High Order Particle Transport for PIC Simulations of Plasma Flows. PhD thesis, Fakultät Luft- und Raumfahrttechnik und Geodäsie, Universität Stuttgart, Germany.

[Riedlinger, 2011] Riedlinger, M. (2011). 3D numerical simulation of a 30 GHz gyrotron resonator with an explicit high-order discontinuous Galerkin approach based particle-in-cell method. Technical report, Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, Germany.

[Riemann, 1860] Riemann, B. (1860). Über die Fortpflanzung ebener Luftwellen von endlicher Schwingungsweite. Abhandlungen der Königlichen Gesellschaft der Wissenschaften in Göttingen, 8.

[Rumsey et al., 2001] Rumsey, C. L., Poirier, D. M. A., Bush, R. H., and Towne, C. E. (2001). CFD General notation system a user's guide to CGNS. Technical report, Langley Research Center.

[Savcenco et al., 2007] Savcenco, V., Hundsdorfer, W., and Verwer, J. G. (2007). A multirate time stepping strategy for stiff ordinary differential equations. BIT Numerical Mathematics, 47(1):137–155.

[Schlegel et al., 2009] Schlegel, M., Knoth, O., Arnold, M., and Wolke, R. (2009). Multirate Runge-Kutta schemes for advection equations. J. Comput. and Appl. Math., 226:345–657.

[Sonnendrücker, 2011] Sonnendrücker, E. (2011). Numerical methods for the Vlasov-Maxwell equations - lecture notes ETH Zürich. Technical report, Strasbourg, France.

[Stindl et al., 2011] Stindl, T., Neudorfer, J., Stock, A., Auweter-Kurtz, M., Munz, C.-D., Roller, S., and Schneider, R. (2011). Comparison of coupling techniques in a high-order discontinuous Galerkin based particle in cell solver. J. Phys. D: Applied Physics, 44:194004.

[Stock, 2009] Stock, A. (2009). Development and application of a multirate multistep AB method to a discontinuous Galerkin method based particle-in-cell scheme. Technical Report 2009-34, Scientific Computing Group, Brown University, Providence, RI, USA.

[Stock et al., 2012a] Stock, A., Neudorfer, J., Riedlinger, M., Pirrung, G., Gassner, G., Schneider, R., Roller, S., and Munz, C.-D. (2012a). Three-dimensional numerical simulation of a 30-GHz gyrotron resonator with an explicit high-order discontinuous-Galerkin-based parallel particle-in-cell method. IEEE Transactions on Plasma Science, 40(7):1860–70.

[Stock et al., 2012b] Stock, A., Neudorfer, J., Schlaich, A., Kern, S., Schneider, R., and Munz, C.-D. (2012b). Simulation of the W7-X 140 GHz gyrotron resonator with an explicit 3D discontinuous Galerkin method based particle-in-cell scheme. In EPS/ICPP Conference on Plasma Physics: Stockholm, Sweden, 2-6 July 2012.

[Stock et al., 2012c] Stock, A., Neudorfer, J., Schneider, R., Roller, S., and Munz, C.-D. (2012c). A full-wave 3D high-order discontinuous-Galerkin-method-based particle-in-cell code for the simulations of high-power gyrotron resonators. In EAPPC 2012 / BEAMS 2012 Karlsuhe, Germany, September 30 - October 4.

[Stock et al., 2011a] Stock, A., Neudorfer, J., Steinbusch, B., Stindl, T., Schneider, R., Roller, S., Munz, C.-D., and Auweter-Kurtz, M. (2011a). Three-dimensional gyrotron simulation using a high-order particle-in-cell method. In Nagel, W. E., Kröner, D. B., and Resch, M. M., editors, High Performance Computing in Science and Engineering '11, pages 637–649. Springer Berlin Heidelberg.

[Stock et al., 2011b] Stock, A., Neudorfer, J. Schneider, R., Altmann, C., and Munz, C.-D. (2011b). Investigation of the purely hyperbolic Maxwell system for divergence cleaning in discontinuous Galerkin based particle-in-cell methods. In Coupled Problems 2011 Conference, Kos Island, Greece.

[SVN, 2012] SVN (2012). Apache subversion (svn).

[Taflove, 1998] Taflove, A. (1998). Advances in Computational Electrodynamics. Artech House, Boston.

[Taube et al., 2009] Taube, A., Dumbser, M., Munz, C.-D., and Schneider, R. (2009). 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.

[Thumm, 2010] Thumm, M. (2010). Recent developments on High-Power Gyrotrons – Introduction to this special issue. Journal of Infrared, Millimeter, and Terahertz Waves, 32(3):241–252.

[Thumm, 2011] Thumm, M. (2011). Progress on gyrotrons for ITER and future thermonuclear fusion reactors. IEEE, Transaction on Plasma Science, 39(4):971–979.

[Tonks and Langmuir, 1929] Tonks, L. and Langmuir, I. (1929). Oscillations in ionized gases. Physical Review, 33(2):195.

[Toro, 1997] Toro, E. F. (1997). Riemann Solvers and Numerical Methods for Fluid Dynamics. Springer, Berlin – Heidelberg.

[Villasenor and Buneman, 1992] Villasenor, J. and Buneman, O. (1992). Rigorous charge conservation for local electromagnetic field solvers. Computer Physics Communications, 69(2-3):306–316.

[Vlasov, 1945] Vlasov, A. A. (1945). On the kinetic theory of an assembly of particles with collective interaction. J. Physics (USSR), 9:25–40.

[Vlasov et al., 1969] Vlasov, S. N., Zhislin, G. M., Orlova, I. M., Petelin, M. I., and Rogacheva, G. G. (1969). Irregular waveguides as open resonators. Radiophysics and Quantum Electronics, 12:972–978.

[VORPAL, 2012] VORPAL (2012). VORPAL User Guide, Release 5.2.

[Wells, 1982] Wells, D. R. (1982). Multirate linear multistep methods for the solution of systems of ordinary differential equations. Technical report. Dept. of Computer Science, University of Illinois at Urbana-Champaign (Urbana, Ill. 1304 W. Springfield Ave., Urbana 61801-2987).

[Yee, 1966] Yee, K. (1966). Numerical solution of the initial boundary value problems involving Maxwell's equation in isotropic media. IEEE Trans. Antennas and Propagation, AP-14:302–307.

[Zapevalov and Moiseev, 2004] Zapevalov, V. E. and Moiseev, M. A. (2004). Influence of aftercavity interaction on gyrotron efficiency. Radiophysics and Quantum Electronics, 47(7):520–27.

A. The Charge Conservation Equation

Equation (2.3) and (2.6) reveals an important link between ρ and $\boldsymbol{j}$. Applying the divergence operator „$\nabla\cdot$“to (2.3), yields

$$\nabla \cdot \left(\frac{\partial \boldsymbol{E}}{\partial t} \right) = c^2 \nabla \cdot (\nabla \times \boldsymbol{B}) - \nabla \cdot \frac{\boldsymbol{j}}{\varepsilon_0}. \tag{A.1}$$

Based on the fact that the divergence of the curl of any differentiable vector field $\boldsymbol{u} \in \mathbb{R}^n$ is zero, $\nabla \cdot (\nabla \times \boldsymbol{u}) = 0$, the middle part vanishes, yielding

$$\nabla \cdot \left(\frac{\partial \boldsymbol{E}}{\partial t} \right) = -\nabla \cdot \frac{\boldsymbol{j}}{\varepsilon_0}. \tag{A.2}$$

Putting the temporal derivative operator from the left side in front of the divergence operator we can write

$$\frac{\partial}{\partial t} (\nabla \cdot \boldsymbol{E}) = -\nabla \cdot \frac{\boldsymbol{j}}{\varepsilon_0}. \tag{A.3}$$

Inserting (2.6) on the left side, yields rearranged and canceled down the charge conservation equation,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \boldsymbol{j} = 0. \tag{A.4}$$

B. The Purely Hyperbolic Maxwell Equations

B.1. Compact Form of the PHM Equations

In (3.16) the solution vector is $\boldsymbol{u} = (E_x, E_y, E_z, B_x, B_y, B_z, \Phi, \Phi)^T$, the source vector is $\boldsymbol{g} = (j_x, j_y, j_z, 0, 0, 0, 0, \rho)^T$ and the physical flux matrices $\mathbb{K}_\beta|_{\beta=1,3}$ are given as,

$$\mathbb{K}_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_0^2\chi \\ 0 & 0 & 0 & 0 & 0 & c_0^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & -c_0^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \kappa & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c_0^2\kappa & 0 & 0 & 0 & 0 \\ \chi & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{B.1}$$

$$\mathbb{K}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -c_0^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_0^2\chi \\ 0 & 0 & 0 & c_0^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \kappa & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_0^2\kappa & 0 & 0 & 0 \\ 0 & \chi & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{B.2}$$

$$\mathbb{K}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & c_0^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -c_0^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_0^2\chi \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \kappa & 0 \\ 0 & 0 & 0 & 0 & 0 & c_0^2\kappa & 0 & 0 \\ 0 & 0 & \chi & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{B.3}$$

Note, due to the linearity of the PHM equations we can express them with the help of the above matrices.

B.2. Riemann Problem for the Purely Hyperbolic Maxwell Equations

The Riemann problem (RP) describes the situation appearing at the border, i.e. the edge or surface, of the elements used to discretize the computational domain. Figure B.1 shows the situation. Between two elements we have two different states, i.e. u_L and u_R.

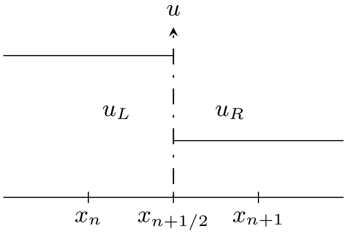


Fig. B.1.: Riemann problem between two elements with different states u_L and u_R.

If the state is not equal we have two different physical states at the same point, $x_{n+1/2}$. This type of problem is called Riemann problem, due to Bernhard Riemann's mathematical analysis on hyperbolic conservation laws and his discovery of discontinuities in the regime of the Euler equations in [Riemann, 1860]. Today we know that these discontinuities are shock-waves. They could be first visualized by Ernst Mach in 1886 on supersonic projectiles. The theory of the mathematical treatment of the discontinuities is in the context of the DG method important, since the DG approach couples two elements with a flux over the boundary. There also two different states appear at the interpolation points on the surface of an element. For each of these points we have to solve the RP. The solution tells us the state at the boundary between two elements. As Maxwell's equations are rotational invariant we can reduce the flux calculation to a one-dimensional problem depending on the normal vector $\boldsymbol{n}$ of the face.

For the solution of the RP we use a flux-vector splitting approach proposed in [Munz et al., 2000a] which we will explain detailed in the following. This is necessary since for the DG approach a slightly different geometrically interpretation appears. While in [Munz et al., 2000a] the flux is derived for a second-order finite volume scheme at the barycentre of the face, in the DG scheme we have several IPs on the surface and we have to calculate the solution of the RP at every IP. The surface flux is the integral of the solution of the RP over the surface. For the flux vector splitting approach the solution of the RP at a certain IP $\boldsymbol{x}_i$ on the surface of an element is given by

$$\boldsymbol{u}(\boldsymbol{x}_i) = (\mathbb{A}^+ \boldsymbol{u}_L(\boldsymbol{x}_i) + \mathbb{A}^- \boldsymbol{u}_R(\boldsymbol{x}_i)), \tag{B.4}$$

where

$$\mathbb{A}^\pm = \frac{1}{2}\left(\mathbb{A} \pm |\mathbb{A}|\right). \tag{B.5}$$

We split the flux into a flux coming from the “left”, having positive eigenvalues only, and a flux coming from the “right”, only possessing negative eigenvalues. They are associated with $\mathbb{A}^+$ and $\mathbb{A}^-$, respectively. To build $\mathbb{A}^\pm$ we consider the linear combination of the physical flux matrices and the components of the face normal vector,

$$\mathbb{A} = \sum_{\beta=1}^{D} n_\beta \mathbb{K}_\beta, \quad \text{with} \sum_{\beta=1}^{D} n_\beta^2 = 1, \tag{B.6}$$

where D is the spatial dimension, which is in our case three.

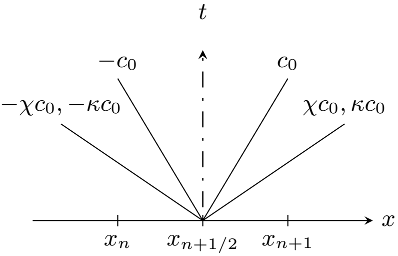


Fig. B.2.: Characteristics for the PHM system in the x-t plane where the slop indicates the speed.

The matrix $\mathbb{A}$ posses eight real eigenvalues λ_i,

$$\Lambda = diag(-c_0, -c_0, c_0, c_0, -\kappa c_0, \kappa c_0, -\chi c_0, \chi c_0), \tag{B.7}$$

according to the characteristics shown in figure B.2 and a complete set of right eigenvectors $\mathcal{R} = (\boldsymbol{r}_1, ..., \boldsymbol{r}_8)$, i.e. for the 3D case

$$\mathcal{R} = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & \frac{n2}{n1} & \frac{n2}{n1} \\ -\frac{n1}{n3} & -\frac{n2}{n3} & -\frac{n1}{n3} & -\frac{n2}{n3} & 0 & 0 & \frac{n3}{n1} & \frac{n3}{n1} \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ -\frac{n1\,n2}{c_0\,n3} & -\frac{n3^2+n2^2}{c_0\,n3} & \frac{n1\,n2}{c_0\,n3} & \frac{n3^2+n2^2}{c_0\,n3} & \frac{n1}{c_0} & -\frac{n1}{c_0} & 0 & 0 \\ \frac{n3^2+n1^2}{c_0\,n3} & \frac{n1\,n2}{c_0\,n3} & -\frac{n3^2+n1^2}{c_0\,n3} & -\frac{n1\,n2}{c_0\,n3} & \frac{n2}{c_0} & -\frac{n2}{c_0} & 0 & 0 \\ -\frac{n2}{c_0} & \frac{n1}{c_0} & \frac{n2}{c_0} & -\frac{n1}{c_0} & \frac{n3}{c_0} & -\frac{n3}{c_0} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{c_0\,n1} & -\frac{1}{c_0\,n1} \end{pmatrix}. \tag{B.8}$$

$|\mathbb{A}|$ is given by

$$|\mathbb{A}| = \mathcal{R}|\Lambda|\mathcal{R}^{-1}, \tag{B.9}$$

where

$$\mathcal{R}^{-1} = \begin{pmatrix} \frac{n3^2+n2^2}{2} & -\frac{n1\,n2}{2} & -\frac{n1\,n3}{2} & 0 & 0 & \frac{c_0\,n3}{2} & -\frac{c_0\,n2}{2} & 0 \\ -\frac{n1\,n2}{2} & \frac{n3^2+n1^2}{2} & -\frac{n2\,n3}{2} & 0 & -\frac{c_0\,n3}{2} & 0 & \frac{c_0\,n1}{2} & 0 \\ \frac{n3^2+n2^2}{2} & -\frac{n1\,n2}{2} & -\frac{n1\,n3}{2} & 0 & 0 & -\frac{c_0\,n3}{2} & \frac{c_0\,n2}{2} & 0 \\ -\frac{n1\,n2}{2} & \frac{n3^2+n1^2}{2} & -\frac{n2\,n3}{2} & 0 & \frac{c_0\,n3}{2} & 0 & -\frac{c_0\,n1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{c_0\,n1}{2} & \frac{c_0\,n2}{2} & \frac{c_0\,n3}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & -\frac{c_0\,n1}{2} & -\frac{c_0\,n2}{2} & -\frac{c_0\,n3}{2} & 0 \\ \frac{n1^2}{2} & \frac{n1\,n2}{2} & \frac{n1\,n3}{2} & 0 & 0 & 0 & 0 & \frac{c_0\,n1}{2} \\ \frac{n1^2}{2} & \frac{n1\,n2}{2} & \frac{n1\,n3}{2} & 0 & 0 & 0 & 0 & -\frac{c_0\,n1}{2} \end{pmatrix}. \tag{B.10}$$

According to the state vector $\boldsymbol{u} = [E_1, E_2, E_3, B_1, B_2, B_3, \Psi, \Phi]^T$ the assembled matrices $\mathbb{A}^{\pm}$ reads

$$\mathbb{A}^{\pm} = \begin{pmatrix} & & & & & & 0 & c_0^2 n_1 \chi \\ & \pm\mathcal{D}(\chi) & & & c^2\mathcal{E} & & 0 & c_0^2 n_2 \chi \\ & & & & & & 0 & c_0^2 n_3 \chi \\ & & & & & & n_1\kappa & 0 \\ & \mathcal{E}^T & & & \pm\mathcal{D}(\kappa) & & n_2\kappa & 0 \\ & & & & & & n_3\kappa & 0 \\ 0 & 0 & 0 & c_0^2 n_1 \kappa & c_0^2 n_2 \kappa & c_0^2 n_3 \kappa & \pm\kappa c_0 & 0 \\ n_1\chi & n_2\chi & n_3\chi & 0 & 0 & 0 & 0 & \pm\chi c_0 \end{pmatrix}, \tag{B.11}$$

with

$$\mathcal{D}(x) = \begin{pmatrix} [1 + n_1^2(x-1)]c_0 & n_1 n_2 (x-1) c_0 & n_1 n_3 (x-1) c_0 \\ n_1 n_2 (x-1) c_0 & [1 + n_2^2(x-1)]c_0 & n_2 n_3 (x-1) c_0 \\ n_1 n_3 (x-1) c_0 & n_2 n_3 (x-1) c_0 & [1 + n_3^2(x-1)]c_0 \end{pmatrix}, \tag{B.12}$$

and

$$\mathcal{E} = \begin{pmatrix} 0 & n_3 & -n_2 \\ -n_3 & 0 & n_1 \\ n_2 & -n_1 & 0 \end{pmatrix} \tag{B.13}$$

.

B.3. Boundary Conditions for the Purely Hyperbolic Maxwell Equations

The boundary conditions describe the state on the domain boundaries occurring due to physical boundaries or numerical boundaries. The typical physical boundaries in the electromagnetic regime are perfectly conducting or absorbing boundary conditions. Numerically induced boundary conditions are periodic and MPI-domain boundary conditions. The theory of constructing boundary conditions is described in detail in [Munz et al., 2000a]. Thus we will only briefly recall the most important equations and ideas. Although a boundary condition in general is an initial-boundary value problem it can be recast to an initial value problem in the sense of a RP solution procedure. The RP at $t = t_n$ is an initial-value problem of the form

$$\frac{\partial \boldsymbol{u}}{\partial \tau} + \mathbb{A}\frac{\partial \boldsymbol{u}}{\partial \xi} = 0; \quad \boldsymbol{u}(\xi, 0) = \begin{cases} \boldsymbol{u}_L & \text{for } \xi < 0, \\ \boldsymbol{u}_R & \text{for } \xi > 0, \end{cases} \tag{B.14}$$

where ξ is oriented by $\boldsymbol{n}$ with the origin $\xi = 0$, as shown in figure B.3, and τ denotes scaled time $\tau = t - t_n$ within $t_n < t < t_n + 1$. $\boldsymbol{u}_L$ and $\boldsymbol{u}_R$ are states on the left and right side of the boundary. One of these two quantities is known while the other one has to be chosen in such a way that the boundary condition is met. In order to calculate a flux over the boundary we have to provide data from both sides of the boundary. Thus we introduce an artificial state on the outer side of the boundary. This is a common method in numerical boundary condition treatment.

We use the theory of characteristics to solve the RP for the PHM equations. Thus we introduce the characteristic variables

$$\boldsymbol{v} = \mathcal{R}^{-1}\boldsymbol{u} = \frac{1}{2}\begin{pmatrix} -(n_1^2+1)E_x - n_1n_2E_y - n_1n_3E_z - c_0n_3B_y + c_0n_2B_z \\ -n_1n_2E_x - (n_2^2+1)E_y - n_2n_3E_z + c_0n_3B_x - c_0n_1B_z \\ -(n_1^2+1)E_x - n_1n_2E_y - n_1n_3E_z + c_0n_3B_y - c_0n_2B_z \\ -n_1n_2E_x - (n_2^2+1)E_y - n_2n_3E_z - c_0n_3B_x + c_0n_1B_z \\ n_1^2E_x + n_1n_2E_y + n_1n_3E_z - c_0n_1\Phi \\ n_1^2E_x + n_1n_2E_y + n_1n_3E_z + c_0n_1\Phi \\ n_1^2B_x + n_1n_2B_y + n_1n_3B_z - \frac{n_1}{c_0}\Psi \\ n_1^2B_x + n_1n_2B_y + n_1n_3B_z + \frac{n_1}{c_0}\Psi \end{pmatrix}, \tag{B.15}$$

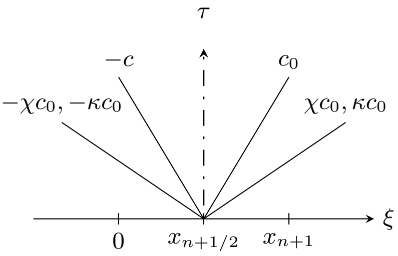


Fig. B.3.: Characteristics for the PHM system in the ξ-τ plane where the slope indicates the speed.

where $\mathcal{R}^{-1}$ is given by (B.10). Using this relation we can recast (B.14) into a set of eight uncoupled linear advection equations having the

$$v_k(\xi,\tau) = v_k|_{\tau=0}(\xi - \lambda_k \tau), \quad \text{with } k = 1, 8, \tag{B.16}$$

where $\mathbf{v}|_{\tau=0} = \mathcal{R}^{-1}\boldsymbol{u}(\xi, 0)$ and λ_k are the eigenvalues of $\boldsymbol{A}$ according to (B.7). We conclude that the result for the characteristic variables at $\xi = 0$ is

$$\mathbf{v}(0,\tau) = \begin{cases} v_{k,R} & \text{if } \lambda_k < 0, \\ v_{k,L} & \text{if } \lambda_k > 0, \end{cases}, \tag{B.17}$$

where $\mathbf{v}|_L = \mathcal{R}^{-1}\boldsymbol{u}_L$ and $\mathbf{v}|_R = \mathcal{R}^{-1}\boldsymbol{u}_R$. Since Maxwell's equations are linear all eigenvalues are constant. Characteristics of positive eigenvalues are propagating in positive direction. In figure B.3 these are the characteristics with a positive slope propagating information from left to right. The negative eigenvalues lead to negative sloped characteristics propagating the information from right to left. Using condition (B.17) for (B.15) gives the solution of the RP in characteristic variables,

$$\mathbf{v}(0,\tau) = \frac{1}{2}\begin{pmatrix} -(n_1^2+1)E_{x,R} - n_1 n_2 E_{y,R} - n_1 n_3 E_{z,R} - c_0 n_3 B_{y,R} + c_0 n_2 B_{z,R} \\ -n_1 n_2 E_{x,R} - (n_2^2+1)E_{y,R} - n_2 n_3 E_{z,R} + c_0 n_3 B_{x,R} - c_0 n_1 B_{z,R} \\ -(n_1^2+1)E_{x,R} - n_1 n_2 E_{y,R} - n_1 n_3 E_{z,R} + c_0 n_3 B_{y,R} - c_0 n_2 B_{z,R} \\ -n_1 n_2 E_{x,R} - (n_2^2+1)E_{y,R} - n_2 n_3 E_{z,R} - c_0 n_3 B_{x,R} + c_0 n_1 B_{z,R} \\ n_1^2 E_{x,R} + n_1 n_2 E_{y,R} + n_1 n_3 E_{z,R} - c_0 n_1 \Phi_R \\ n_1^2 E_{x,L} + n_1 n_2 E_{y,L} + n_1 n_3 E_{z,L} + c_0 n_1 \Phi_L \\ n_1^2 B_{x,R} + n_1 n_2 B_{y,R} + n_1 n_3 B_{z,R} - \frac{n_1}{c_0}\Psi_R \\ n_1^2 B_{x,R} + n_1 n_2 B_{y,L} + n_1 n_3 B_{z,L} + \frac{n_1}{c_0}\Psi_L \end{pmatrix}. \tag{B.18}$$

To get the solution at $\xi = 0$ in physical state variables we have to transform the solution back according to $\boldsymbol{u}(0,\tau) = \mathcal{R}\boldsymbol{v}(0,\tau)$, yielding

$$\boldsymbol{u}_0 = \frac{1}{2}\begin{pmatrix} (E_{1,L}+E_{1,R}) + c_0\left[n_3(B_{2,L}-B_{2,R}) - n_2(B_{3,L}-B_{3,R})\right] + c_0\ n_1(\Phi_L-\Phi_R) \\ (E_{2,L}+E_{2,R}) + c_0\left[n_1(B_{3,L}-B_{3,R}) - n_3(B_{1,L}-B_{1,R})\right] + c_0\ n_2(\Phi_L-\Phi_R) \\ (E_{3,L}+E_{3,R}) + c_0\left[n_2(B_{1,L}-B_{1,R}) - n_1(B_{2,L}-B_{2,R})\right] + c_0\ n_3(\Phi_L-\Phi_R) \\ (B_{1,L}+B_{1,R}) - \frac{1}{c_0}\left[n_3(E_{2,L}-E_{2,R}) - n_2(E_{3,L}-E_{3,R})\right] + c_0\ n_1(\Psi_L-\Psi_R) \\ (B_{2,L}+B_{2,R}) - \frac{1}{c_0}\left[n_1(E_{3,L}-E_{3,R}) - n_3(E_{1,L}-E_{1,R})\right] + c_0\ n_2(\Psi_L-\Psi_R) \\ (B_{3,L}+B_{3,R}) - \frac{1}{c_0}\left[n_2(E_{1,L}-E_{1,R}) - n_1(E_{2,L}-E_{2,R})\right] + c_0\ n_3(\Psi_L-\Psi_R) \\ (\Psi_L+\Psi_R) + c_0\left[(B_{1,L}-B_{1,R})n_1 + (B_{2,L}-B_{2,R})n_2 + (B_{3,L}-B_{3,R})n_3\right] \\ (\Phi_L+\Phi_R) + \frac{1}{c_0}\left[(E_{1,L}-E_{1,R})n_1 + (E_{2,L}-E_{2,R})n_2 + (E_{3,L}-E_{3,R})n_3\right] \end{pmatrix}. \tag{B.19}$$

This can also be written in vector form,

$$\boldsymbol{E}_0 = \frac{1}{2}\left[(\boldsymbol{E}_L + \boldsymbol{E}_R) + c_0(\boldsymbol{B}_L - \boldsymbol{B}_R)\times\boldsymbol{n} + c_0(\Phi_L - \Phi_R)\boldsymbol{n}\right], \tag{B.20}$$

$$\boldsymbol{B}_0 = \frac{1}{2c_0}\left[c_0(\boldsymbol{B}_L + \boldsymbol{B}_R) - (\boldsymbol{E}_L - \boldsymbol{E}_R)\times\boldsymbol{n} + c_0(\Psi_L - \Psi_R)\boldsymbol{n}\right], \tag{B.21}$$

$$\Psi_0 = \frac{1}{2}\left[(\Psi_L + \Psi_R) + c_0(\boldsymbol{B}_L - \boldsymbol{B}_R)\cdot\boldsymbol{n}\right], \tag{B.22}$$

$$\Phi_0 = \frac{1}{2c_0}\left[c(\Phi_L + \Phi_R) + c_0(\boldsymbol{E}_L - \boldsymbol{E}_R)\cdot\boldsymbol{n}\right]. \tag{B.23}$$

While the interior state is known, the outer state has to be determined in such a way that the boundary condition is met. In the following we assume that $\boldsymbol{u}_L$ is inside the domain, while $\boldsymbol{u}_R$ is outside the domain.

B.3.1. Perfectly Conducting Boundary Condition

A perfectly electrical conducting surface Γ_C, where the field cannot penetrate is specified by

$$\boldsymbol{E}(\mathbf{x},t)\times\boldsymbol{n} = 0, \quad \mathbf{x}\in\Gamma_C, \tag{B.24}$$

$$\boldsymbol{B}(\mathbf{x},t)\cdot\boldsymbol{n} = 0, \quad \mathbf{x}\in\Gamma_C. \tag{B.25}$$

For the Lagrange multipliers ϕ and ψ we prescribe the following conditions

$$\Phi(\mathbf{x},t) = 0, \quad \mathbf{x}\in\Gamma, \tag{B.26}$$

$$\Psi(\mathbf{x},t) = 0, \quad \mathbf{x}\in\Gamma_A, \tag{B.27}$$

$$\frac{\partial\Psi(\mathbf{x},t)}{\partial\boldsymbol{n}} = 0, \quad \mathbf{x}\in\Gamma_C, \tag{B.28}$$

where Γ_A is an absorbing boundary and Γ is any boundary. With these conditions and the PR solution (B.20)-(B.23) the following boundary condition can be derived,

$$\boldsymbol{E}_R = -\boldsymbol{E}_L + 2(\boldsymbol{E}_L \cdot \boldsymbol{n})\boldsymbol{n}, \tag{B.29}$$

$$\boldsymbol{B}_R = \boldsymbol{B}_L - 2(\boldsymbol{B}_L \cdot \boldsymbol{n})\boldsymbol{n}, \tag{B.30}$$

$$\Phi_R = \Phi_L, \Psi_R = -\Psi_R, \tag{B.31}$$

B.3.2. Absorbing Boundary Condition

Often a computational domain is artificially truncated to the part which is interesting for the simulation. To handle those artificially truncated domains we impose the Silver-Müller condition [Müller, 1969],

$$[\boldsymbol{E}_0 - c_0(\boldsymbol{B}_0 \times \boldsymbol{n})] \times \boldsymbol{n} = [\epsilon - c_0(\beta \times \boldsymbol{n})] \times \boldsymbol{n}, \tag{B.32}$$

$$\tag{B.33}$$

where ϵ and β are given electromagnetic fields along an absorbing boundary Γ_A. For the PHM system it was shown that the inserted solution of the Riemann problem, $\boldsymbol{E}_0, \boldsymbol{B}_0$, yields

$$[\boldsymbol{E}_R - c_0(\boldsymbol{B}_R \times \boldsymbol{n})] \times \boldsymbol{n} = [\epsilon - c_0(\beta \times \boldsymbol{n})] \times \boldsymbol{n}, \tag{B.34}$$

where $\boldsymbol{E}_R, \boldsymbol{B}_R$ is the exterior state. From this we can conclude, that for the PHM system

$$\boldsymbol{E}_R = \epsilon, \tag{B.35}$$

$$\boldsymbol{B}_R = \beta. \tag{B.36}$$

For $\epsilon = \beta = 0$ a first-order accurate absorbing BC is derived [Peterson, 1988]. This results in the following BC:

$$\boldsymbol{E}_R = 0, \tag{B.37}$$

$$\boldsymbol{B}_R = 0, \tag{B.38}$$

$$\Phi_R = -\Phi_L - c_0\boldsymbol{B} \cdot \boldsymbol{n}, \tag{B.39}$$

$$\Psi_R = -\Psi_L - \frac{1}{c_0}\boldsymbol{E} \cdot \boldsymbol{n}. \tag{B.40}$$

C. Interpolation Points for Different Element Types in the Discontinuous Galerkin Method

Figures C.1 - C.16 show the IPs for different element types and polynomial degrees p.

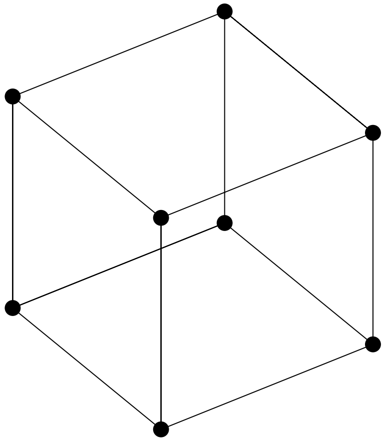


Fig. C.1.: $p = 1$ hexahedron.

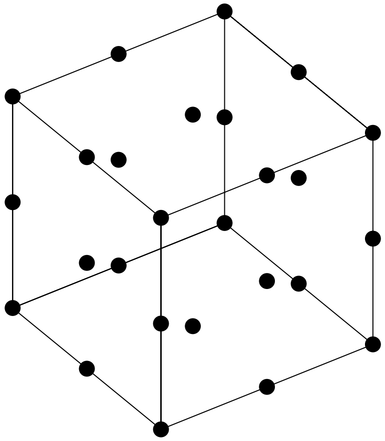


Fig. C.2.: $p = 2$ hexahedron.

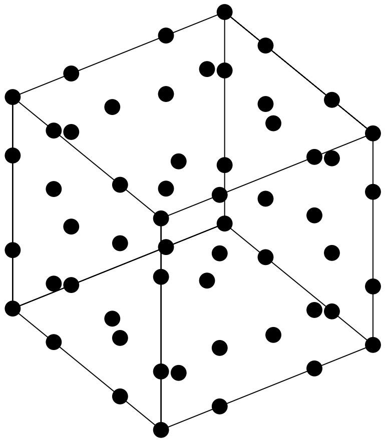


Fig. C.3.: $p = 3$ hexahedron.

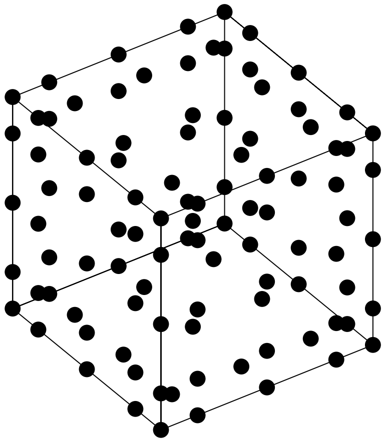


Fig. C.4.: $p = 4$ hexahedron.

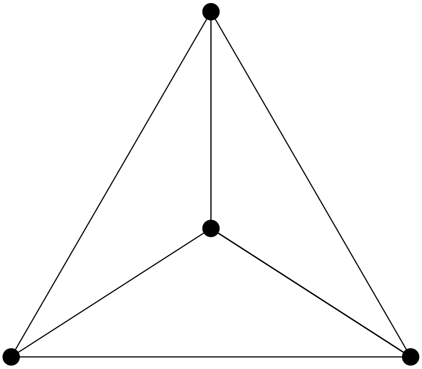


Fig. C.5.: $p = 1$ tetrahedron.

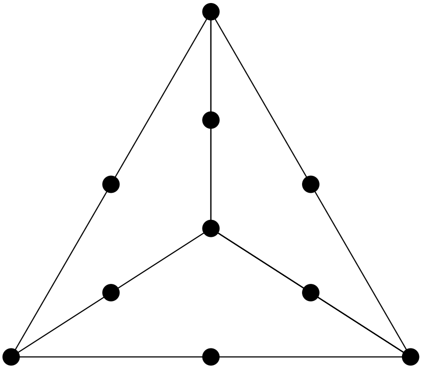


Fig. C.6.: $p = 2$ tetrahedron.

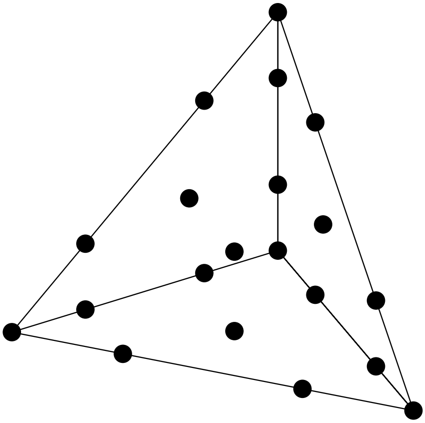


Fig. C.7.: $p = 3$ tetrahedron.

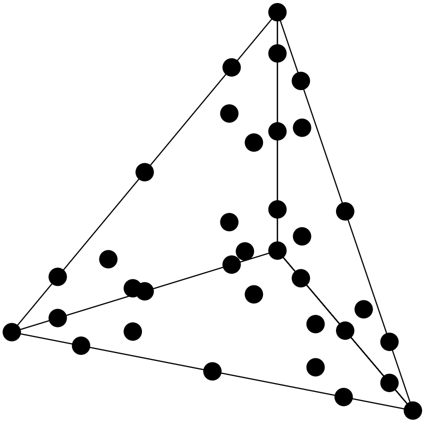


Fig. C.8.: $p = 4$ tetrahedron.

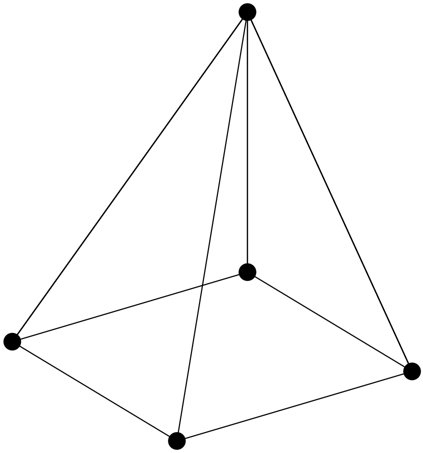


Fig. C.9.: $p = 1$ pyramid.

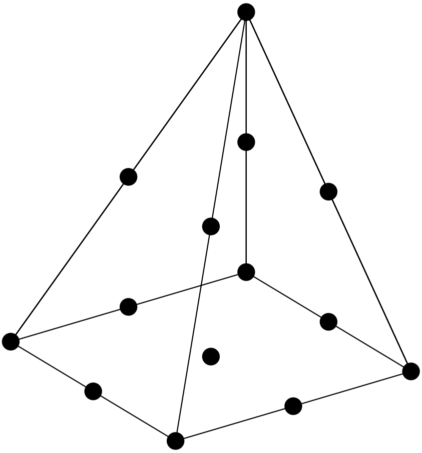


Fig. C.10.: $p = 2$ pyramid.

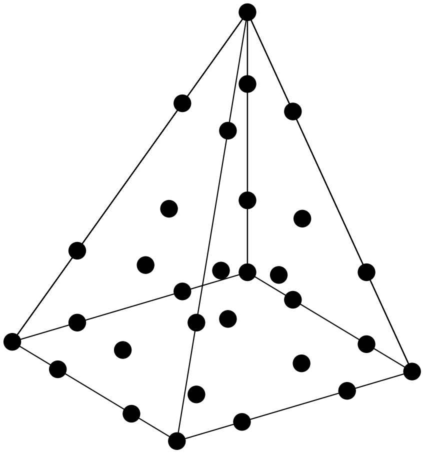


Fig. C.11.: $p = 3$ pyramid.

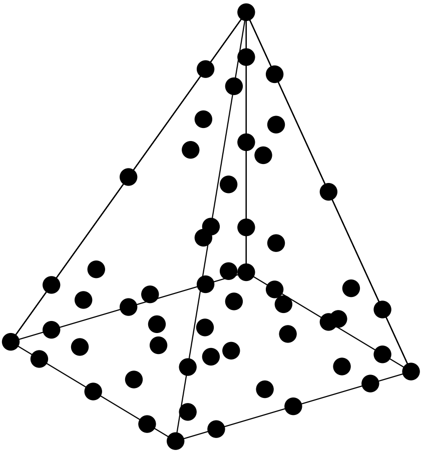


Fig. C.12.: $p = 4$ pyramid.

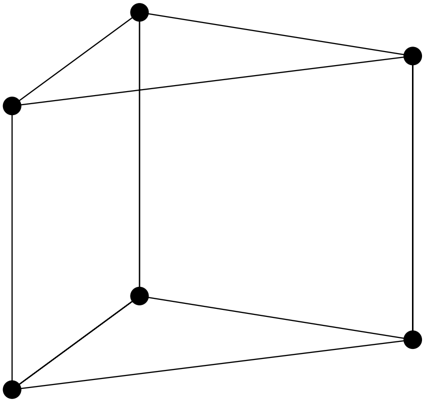


Fig. C.13.: $p = 1$ prism.

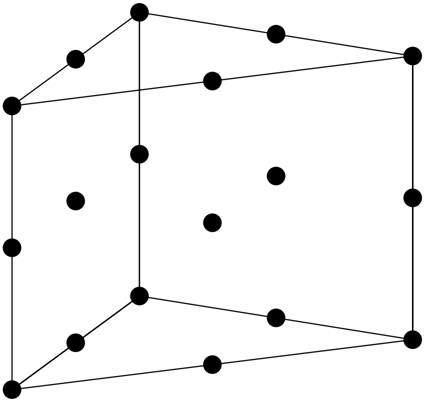


Fig. C.14.: $p = 2$ prism.

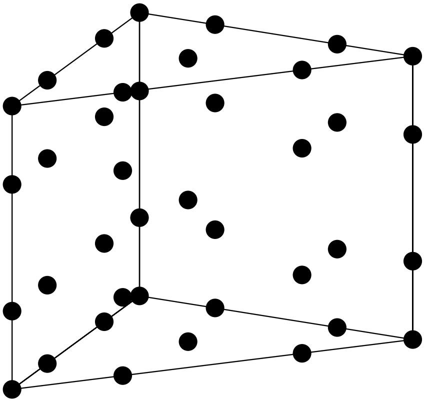


Fig. C.15.: $p = 3$ prism.

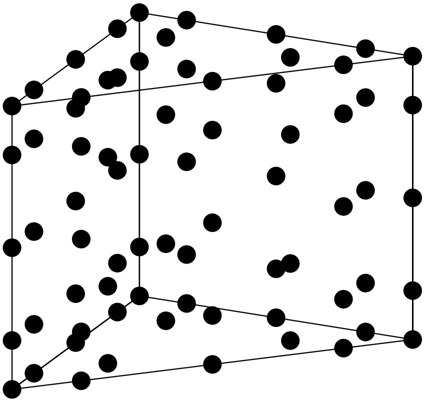


Fig. C.16.: $p = 4$ prism.

D. The Discrete Discontinuous Galerkin Scheme for the Purely Hyperbolic Maxwell Equations

Equation 3.20 gives the weak form of the variational formulation of Maxwell's equation. It uses a generally stated test function ψ_i and so far, nothing has been said about the choice of the modal or nodal basis functions introduced in 3.3.2. To get a discrete formulation, which can be implemented in a code, we have to introduce operators based on the different basis functions. In [Gassner et al., 2009] a detailed derivation of the mixed nodal and modal DG scheme that we use in this thesis is given. In the following we briefly review the building blocks and introduce the discrete operators.

Volume Integral of the Physical Flux:
First, we concentrate on the volume integral of the physical flux in the weak form (3.20),

$$\int_k \sum_{\beta=1}^{3} \mathbb{K}_\beta \boldsymbol{u}_h \frac{\partial \psi_i}{\partial x_\beta} \, d\boldsymbol{x}. \tag{D.1}$$

The physical flux is computed with nodal DOF, yielding $\underline{\tilde{F}}_\beta(\mathbf{x}_i) = \boldsymbol{K}_\beta \tilde{\mathbf{u}}_h(\mathbf{x}_i)$. In the mixed nodal an modal approach the above volume integral is expressed as

$$\int_k \sum_{\beta=1}^{3} \frac{\partial \underline{\phi}(\mathbf{x})}{\partial x_\beta} \underline{L}^T(\mathbf{x}) \; \underline{\tilde{F}}_\beta(\mathbf{x}_i) \; d\boldsymbol{x} = \sum_{\beta=1}^{3} \mathcal{S}_\beta \; \underline{\tilde{F}}_\beta(\mathbf{x}_i), \tag{D.2}$$

where $\underline{\phi}(\mathbf{x})$ is the modal basis function vector, $\underline{L}(\mathbf{x})$ is the nodal basis function vector and $\mathcal{S}_\beta$ is the stiffness matrix. The stiffness matrix is $N_m \times N_n$ matrix for each dimension β, yielding

$$\mathcal{S}_{\beta,ij} = \int_k \frac{\partial \phi_i(\mathbf{x})}{\partial x_\beta} \; L_j^T(\mathbf{x}) \; d\boldsymbol{x}. \tag{D.3}$$

It is pre-computed exactly with Gauss quadrature and stored in the initial phase. Hence it only has to be multiplied with the physical flux $\underline{\tilde{F}}_\beta$ during the computation.

Surface Integral:
Next we consider the surface integral term $\int_{\partial k} \boldsymbol{F}_h(\boldsymbol{u}_i, \boldsymbol{u}_e, \boldsymbol{n})\ \psi_i\ ds$. Here only the surface IPs are relevant. First the Riemann solver provides the solution of the RP $\underline{\tilde{F}}_{h,f}(\mathbf{x}_i)$ on the face f of the element. Note, that depending on the element type (hexahedron, tetrahedron, etc.) different numbers of faces exist. The surface integral is given as,

$$\sum_{f=1}^{\text{nFaces}} \int_{\partial f} \underline{\phi}(\mathbf{x})\ \underline{L}_f^T(\mathbf{x})\ \underline{\tilde{F}}_{h,f}(\mathbf{x}_i)\ ds = \sum_{f=1}^{\text{nFaces}} \mathcal{E}_f\ \underline{\tilde{F}}_{h,f}(\mathbf{x}_i), \tag{D.4}$$

where $\mathcal{E}_f$ is a $N_m \times N_{n,f}$ matrix. $N_{n,f}$ is the number of IPs and $\underline{L}_f^T$ are the nodal basis functions on the face f. Here, the integral is pre-computed as well. Thus, only the operator $\mathcal{E}_f$ has to be applied to $\underline{\tilde{F}}_{h,f}(\mathbf{x}_i)$ in every timestep.

Volume Integral for the Sources:
The volume integral of the source, $\int_k \mathbf{g}_h \psi_i\ d\mathbf{x}$, has to transform the sources, which are evaluated on a nodal basis at the IPs, into the modal DOFs. The corresponding operator is given as

$$\mathcal{G}_{ij} = \int_k \phi_i(\mathbf{x})\ L_j^T(\mathbf{x})\ \tilde{\mathbf{g}}_h(\mathbf{x}_i)\ d\mathbf{x} = \mathcal{G}\ \tilde{\mathbf{g}}_h(\mathbf{x}_i), \tag{D.5}$$

where $\tilde{\mathbf{g}}_h(\mathbf{x}_i)$ are the nodal values of the source polynomial. $\mathcal{G}$ is a $N_m \times N_n$ matrix, which has a similar function as the Vandermonde matrix (3.15) but also provides the integral over the element. Again the integral is pre-computed in the initial phase and only $\tilde{\mathbf{g}}_h(\mathbf{x}_i)$ has to be calculated in every timestep.

The Fully Discrete Form:
Finally we look at $\int_k \frac{\partial \boldsymbol{u}_h}{\partial t} \psi_i\ d\mathbf{x}$. Since no mesh interaction is required for the time derivative, we can store $\boldsymbol{u}_h$ in modal form as $\hat{\mathbf{u}}_t$. This is a key feature of the mixed nodal and modal approach, since it allows to reduce the DOF per element which have to be used within the time integration scheme to N_m. The integral is given by

$$\int_k \phi_i(\mathbf{x})\ \phi_j(\mathbf{x})\ \hat{\mathbf{u}}_t\ d\mathbf{x} = \mathcal{M}\ \hat{\mathbf{u}}_t, \tag{D.6}$$

where the mass matrix has the special property,

$$\mathcal{M}_{ij} = \int_k \phi_i \phi_j d\mathbf{x} = \delta_{ij}, \tag{D.7}$$

due to the choice of the orthonormalized modal basis functions ϕ.
After providing all necessary operators we now can assemble the fully discrete scheme, yielding

$$\frac{\partial \hat{\boldsymbol{u}}_h}{\partial t} + \sum_{f=1}^{\mathrm{nFaces}} \mathcal{E}_f\ \underline{\tilde{F}}_{h,f} - \sum_{\beta=1}^{3} \mathcal{S}_\beta\ \underline{\tilde{F}}_\beta = \mathcal{G}\tilde{\boldsymbol{g}}_h. \tag{D.8}$$

The time derivative $\hat{\boldsymbol{u}}_h$ is then used with an ordinary differential equation (ODE) solver to advance the state in time. The modal expression of the time derivative reduces the number of computational operations since N_m is smaller than N_n. Only in case of tetrahedrons $N_m = N_n$ for our choice of the interpolation points $\boldsymbol{x}_i$.
Note, that all matrix operators are precomputed and only need to be evaluated once. During the computation only the matrix vector product of the operator and the state has to be performed. Thus the DG scheme is very efficient and fast. The geometry is recognized via metric terms for the transformation from a unit element to the physical elements. Details to this transformation can be found in [Hesthaven and Warburton, 2008].

E. Multi-Rate Adams-Bashforth Coefficients Calculation

The idea of the AB approach is the extrapolation and integration of the RHS function $f(u,t)$ of an ODE or PDE, based on values at previous timesteps $t_i|_{i=0,\cdots,N_{AB}}$.
The interpolation of the value $f(t_a)$ at the point t_a for a given set of DOF $f(t_i)|_{i=0,\cdots,N_{AB}}$ is given by

$$f(t_a) = \sum_{i=0}^{N_{AB}} d_i \; f(t_i), \tag{E.1}$$

where a_i are interpolation coefficients.
A convenient way to compute these interpolation coefficients is the solution of the system

$$\begin{pmatrix} \phi_{0,1} & \cdots & \phi_{0,N_{AB}} \\ \phi_{1,1} & \cdots & \phi_{1,N_{AB}} \\ \vdots & \cdots & \vdots \\ \phi_{N_{AB},1} & \cdots & \phi_{N_{AB},N_{AB}} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{N_{AB}} \end{pmatrix} = \begin{pmatrix} \phi_0(t_a) \\ \phi_1(t_a) \\ \vdots \\ \phi_{N_{AB}}(t_a) \end{pmatrix},$$

where the basis function $\phi_{i,j} = x^i$ for the j^{th} DOF, a_i are the interpolation coefficients and $\phi_i(t_a)$ is the i^{th} basis function evaluated at t_a.
For the AB approach additionally the integral of $f(u,t)$ over the interval $[t_a, t_b]$ has to be computed, yielding

$$\int_{t_a}^{t_b} f(t)dt = \sum_{i=0}^{N_{AB}} b_i \; f(t_i). \tag{E.2}$$

The coefficients b_i can be found by the solution of the system

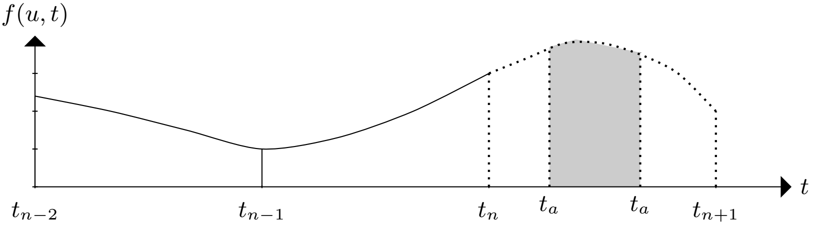


Fig. E.1.: Principle of extrapolation and integration of $f(u,t)$.

$$\begin{pmatrix} \phi_{0,1} & \cdots & \phi_{0,N_{AB}} \\ \phi_{1,1} & \cdots & \phi_{1,N_{AB}} \\ \vdots & \cdots & \vdots \\ \phi_{N_{AB},1} & \cdots & \phi_{N_{AB},N_{AB}} \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N_{AB}} \end{pmatrix} = \begin{pmatrix} \int_{t_a}^{t_b} \phi_0(t)dt \\ \int_{t_a}^{t_b} \phi_1(t)dt \\ \vdots \\ \int_{t_a}^{t_b} \phi_{N_{AB}}(t)dt \end{pmatrix}.$$

Figure E.1 shows the situation for an arbitrary interval $[t_a, t_b]$. For the interval $[t_n, t_{n+1}]$ b_i would be the classical AB coefficients.

F. Coefficients for Different Time Stepping Methods

o_1 = 0.0,	g_1 = 0.1496590219993,	s_1 = 0.0,
o_2 = -0.4178904745,	g_2 = 0.3792103129999,	s_2 = 0.1496590219993,
o_3 = -1.192151694643,	g_3 = 0.8229550293869,	s_3 = 0.3704009573644,
o_4 = -1.697784692471,	g_4 = 0.6994504559488,	s_4 = 0.6222557631345,
o_5 = -1.514183444257,	g_5 = 0.1530572479681,	s_5 = 0.9582821306748.

Tab. F.1.: Coefficients for the 4^{th}-order low-storage explicit Runge-Kutta (LSERK4) scheme from [Carpenter and Kennedy, 1994].

n	a_{j-6}	a_{j-5}	a_{j-4}	a_{j-3}	a_{j-2}	a_{j-1}	a_j
1						$-\frac{1}{2}$	$\frac{3}{2}$
2					$\frac{5}{2}$	$-\frac{16}{12}$	$\frac{23}{12}$
3				$-\frac{9}{24}$	$\frac{37}{24}$	$-\frac{59}{24}$	$\frac{55}{24}$
4			$\frac{251}{720}$	$-\frac{1247}{720}$	$\frac{2616}{720}$	$-\frac{2774}{720}$	$\frac{1901}{720}$
5		$-\frac{475}{1440}$	$\frac{2877}{1440}$	$-\frac{7298}{1440}$	$\frac{9982}{1440}$	$-\frac{7923}{1440}$	$\frac{4277}{1440}$
6	$\frac{19087}{60480}$	$-\frac{134472}{60480}$	$\frac{407139}{60480}$	$-\frac{688256}{60480}$	$\frac{705549}{60480}$	$-\frac{447288}{60480}$	$\frac{198721}{60480}$

Tab. F.2.: Coefficients for the classical Adams-Bashfort method of order n [Hairer et al., 2009, p. 356-359].

substep	b_3	b_2	b_1
1	$\frac{100}{12}$	$-\frac{35}{12}$	$\frac{85}{12}$
2	$\frac{29}{24}$	$-\frac{25}{24}$	$\frac{80}{24}$

Tab. F.3.: MRAB-PHM coefficients for $\chi = 2$ and $N_{AB} = 2$.