Fakultät für Informatik der Technischen Universität München

Development of a high-performance gyrokinetic turbulence code for the edge and scrape-off layer of magnetic confinement fusion devices

Dominik Michels

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften (Dr. rer. nat.) genehmigten Dissertation.

Vorsitzende/-r: Prof. Dr. Martin Schulz

Prüfende/-r der Dissertation:

1. Hon.-Prof. Dr. Frank Jenko

2. Prof. Dr. Eric Sonnendrücker

Die Dissertation wurde am 21.09.2021 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 21.12.2021 angenommen.

Development of a high-performance gyrokinetic turbulence code for the edge and scrape-off layer of magnetic confinement fusion devices

Dominik Michels

September 2021

Abstract

Understanding the role of turbulence in the edge and scrape-off layer (SOL) of magnetic confinement fusion devices is key to predict their heat exhaust and energy confinement time. Turbulence is commonly studied with computer simulations based on the gyrokinetic equations, a five-dimensional, nonlinear, integro-partial differential system that describes the evolution of the distribution of particles in phase space.

Performing gyrokinetic simulations of edge and SOL plasmas is particularly difficult. First, the plasma density and temperature is low and the turbulent fluctuations are of the same size as the background plasma. This requires a so-called full-f treatment of the gyrokinetic equations. Second, the magnetic geometry of the edge and SOL is complex. It contains regions of open magnetic flux surfaces, X-points and, in stellarator fusion experiments, even regions where field lines are oriented stochastically. This renders flux-coordinates, commonly used in gyrokinetic turbulence simulations of the plasma core, ineffective.

To tackle these issues, we present a new gyrokinetic turbulence code, `GENE-X`, in this work. `GENE-X` implements a full-f version of the gyrokinetic system that includes electromagnetic effects and can handle turbulence with large fluctuation amplitudes. Furthermore, `GENE-X` can perform simulations in realistic magnetic geometries by implementing the so-called flux-coordinate independent approach. In this work, we detail the equations implemented in the code, the coordinate system, the discretisation scheme and careful numerical and physical verifications. In addition, we present the first application of `GENE-X` to ASDEX Upgrade (AUG). The simulations of AUG are performed with realistic and reduced electron-to-ion mass ratio, with and without collisions over a time of 1ms, studying profile evolution. We compare the results to previous simulations with the Braginskii fluid code `GRILLIX` and experimental measurements. Furthermore, we discuss the influence of the electron-to-ion mass ratio and collisions on the profiles and the heat exhaust.

Publications

Parts of this work have been published in

[1] D. Michels, A. Stegmeir, P. Ulbl *et al.* GENE-X: A full-f gyrokinetic turbulence code based on the flux-coordinate independent approach. *Computer Physics Communications*, **264**, 107986 (2021).

Contents

1. Introduction

1.1. Motivation

The climate of our planet is changing – with dramatic effects. Glaciers melt, sea levels rise and, as a consequence, the natural habitat of many species is destroyed [1]. It is scientific consensus that this trend cannot be explained by natural climate variability and is caused by the anthropogenic emission of greenhouse gases [2]. To limit global warming and its tremendous effects, it is therefore vital to reduce the emission of greenhouse gases by using and developing new, environmentally friendly, energy sources.
Inspired by the energy production of stars, fusion energy offers a unique pathway to achieve that goal. Energy produced by fusion power plants would be CO_2-free, safe and sustainable for generations to come. Consequently, developing net energy-producing fusion experiments, as a first step towards a fusion power plant, has been a dream since the 1950s.

1.1.1. Nuclear fusion

Many stars, like our sun, produce energy by fusing hydrogen via the *Bethe-Weizäcker cycle* or the *proton-proton chain* [3]. These reactions have a low cross section at temperatures and densities producible in the laboratory. Therefore, they are not suited for fusion reactors on earth. In the most promising fusion experiments today, hydrogen is replaced by its isotopes deuterium and tritium (D-T). The D-T fusion process has the highest known cross section at experimentally relevant parameters and is therefore the easiest to access [4]. The products of the fusion reaction have a slightly lower mass than the inputs. This difference in mass is converted into kinetic energy by the phenomenon known as *mass defect*. During the D-T fusion process, visualised in Fig. 1.1, a neutron with 14.1 MeV energy and a helium atom with 3.5 MeV energy is released.

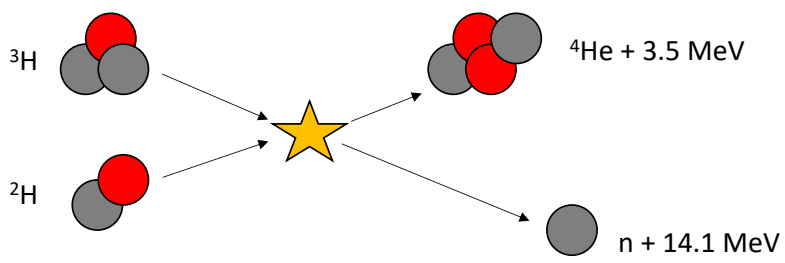

Figure 1.1.: Figure displaying the fusion reaction of deuterium and tritium to helium and a neutron. The reaction releases a total of 17.6 MeV energy.

However, making atoms fuse is not simple. Due to the Coulomb force, nuclei repel each other. To make fusion reactions possible, the two nuclei need to get close enough such that quantum tunnelling through the Coulomb barrier is likely to happen. This can either be achieved by high temperatures or high pressures. In these conditions, the atoms are ionised and form the state of a plasma. Part of the energy, produced in the fusion process, stays within the plasma and makes, in theory, a self-sustained fusion reaction possible. To obtain such an energy producing, also called burning, plasma, the product of its temperature T, density n and energy confinement time τ_E must exceed the threshold [5]

$$nT\tau_E > 3 \cdot 10^{21} \mathrm{KeVsm}^{-3} \,. \tag{1.1}$$

1.1.2. Confinement and plasma stability

In stars the plasma is confined with the help of gravity. Temperatures and densities are large and energy confinement times long. As a result, the fusion reaction in stars sustains itself and produces energy. While different confinement strategies exist on earth, the most promising experiments use magnetic fields to confine the plasma. Due to the Lorentz force, charged particles are trapped to a spiral like motion around magnetic field lines. To avoid particle losses at the end of field lines, they are bend into a toroidal shape. The two most promising concepts for fusion power plants are the *tokamak* and the *stellarator* [6]. In tokamaks densities around $10^{19}/\mathrm{m}^3 - 10^{20}/\mathrm{m}^3$, Temperatures of $1\mathrm{KeV} - 40\mathrm{KeV}$ and energy confinement times of $\tau_E = 0.1s - 1s$ can be achieved [5, 7].
In first approximation, the plasma and its interaction with the magnetic field $\boldsymbol{B}$, in these fusion devices, can be described as a fluid within *magnetohydrodynamics* (MHD). The equilibrium state of the plasma is classified by the force balance [8]

$$\boldsymbol{j} \times \boldsymbol{B} = c\boldsymbol{\nabla} p, \tag{1.2}$$

between the Lorentz force $\boldsymbol{j} \times \boldsymbol{B}$, caused by the plasma current $\boldsymbol{j}$, and the plasma pressure p. The variable c denotes the speed of light. In axisymmetric devices, like the tokamak, every magnetic field line generates a magnetic flux surface. On a magnetic flux surface the pressure is constant and particles stream freely along magnetic field lines. An MHD equilibrium for the tokamak ASDEX Upgrade (AUG) is visualised in Fig. 1.2.

1.1.3. Transport and turbulence

The confinement of the plasma is limited by radial transport perpendicular to magnetic flux surfaces, leading to particle and energy loss. The first component of radial transport is caused by diffusive processes due to collisions in the plasma. However, the radial transport measured in experiments cannot be explained by collisional transport alone. The second component is given by transport due to turbulent fluctuations within the plasma and commonly referred to as *anomalous* transport [9, 7]. The turbulence is driven by microinstabilities that are caused by density and temperature fluctuations. They grow by extracting energy from gradients

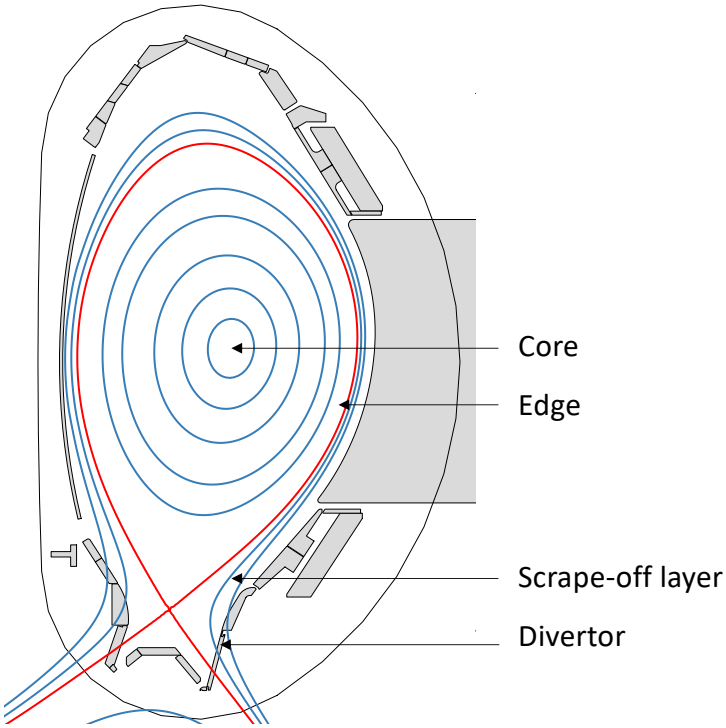

Figure 1.2.: Figure visualising the magnetic equilibrium of a tokamak at the example of AUG [10]. Magnetic flux surfaces are drawn in blue and the last closed flux surface, also known as separatrix, is drawn in red. This plot has been generated with the tool `diaggeom` from the AUG experiment.

within the plasma. The turbulence transports the energy from the microstabilities, in a cascade like way, to different spatial scales, where it is dissipated.

1.1.4. The edge and scrape-off layer

Inevitably, the plasma comes into contact with the main wall in the boundary region of the fusion device. To avoid damage, the plasma needs to be cooled by orders of magnitude before it hits the wall. Consequently, the boundary region of magnetic confinement fusion devices is characterised by steep density and temperature gradients, driving microinstabilites and turbulence.

In the fusion community, the boundary region is conventionally divided into two parts. The first is the so-called scrape-off layer (SOL). In the SOL magnetic field lines are open and the plasma is in contact with the device wall. The second is the so-called edge of the device. It represents the region just before the last closed flux surface. For visualisation, the different regions are shown at the example of AUG in Fig. 1.2.

In recent years, it has become apparent that understanding turbulent transport in the edge and SOL is important for the energy confinement time and the feasibility of fusion energy. The reason is given in the following two subsections.

1.1.5. Understanding the L-H transition

To achieve long energy confinement times, most fusion experiments operate in high confinement regimes, like the H-mode [11]. These operating modes are characterised by a transport barrier in the edge that limits turbulence and, as a consequence, increases the energy confinement time. The goal of the upcoming fusion experiment ITER [12], to achieve a burning plasma and produce more than ten times the energy injected, is based on the H-mode [13].

Measurements at different experiments have shown that the transport barrier is accompanied by a drop in the radial electric field just before the separatrix [14]. This drop induces a poloidal rotation of the plasma through the $E \times B$ drift that shears turbulent vortices and, consequently, suppresses turbulent transport [14]. As a result of the transport barrier, the temperature and density in plasma core increase and the gradients, close to the edge of the device, are steepened.

Over the last decades, tremendous efforts have been undertaken to understand the formation of the transport barrier and the shift into the high confinement regime, also referred to as the L-H transition [15]. Experimentally, the transition into H-Mode is triggered when the plasma heating power exceeds a critical value [11]. There exist many different models that try to explain the L-H transition. A popular choice, that can explain some of the experimentally measured features of the L-H transition, are predator-prey models [16, 17, 18]. We explain them at the example of [17]. The turbulence takes the position of the prey. With increasing heating power the turbulence increases. After it has reached a certain threshold, it excites zonal flows, the predator, that regulate the turbulence. As soon as the drive of the zonal flows becomes stronger than their damping, they grow exponentially and annihilate the turbulence. At the same time a steep pressure profile with a poloidal flow builds that takes over the role of the zonal flow in damping the turbulence. The result is a quiescent H-mode with a steep pressure profile, a large poloidal flow, no zonal flow and no turbulence.

Some of these results have been reproduced with fluid turbulence simulations [19, 20, 21]. Additionally, simulations with the gyrokinetic turbulence code `XGC1` showed that effects due to ion-orbit losses, that are not resolved within fluid codes, are important in the L-H transition [22]. The scientific evidence suggests that turbulence plays an integral part in the L-H transition and predictive, first principle modelling is needed to understand it.

1.1.6. Limiting the particle and heat exhaust of fusion devices

To avoid damage to the fusion device by hot particles hitting the wall, limiting the particle and heat exhaust poses an important problem. Particles are transported over the separatrix and stream along magnetic field lines onto the wall. This generates heat fluxes up to $500\mathrm{MW/m^2}$ in present tokamaks and is expected to reach $1\mathrm{GW/m^2}$ in ITER [23]. To avoid damage to the wall, the heat flux must be reduced to $10\mathrm{MW/m^2}$ normal to the plates for actively cooled materials in steady-state operation [23]. The heat flux onto the divertor is commonly reduced by using shallow incident angles between the magnetic field lines and the divertor plates [24, 25], which effectively distributes the heat flux onto a wider area. The mechanism

that describes the width of the scrape-off layer heat flux, commonly called λ_q, has not been understood and predictions of λ_q yield vastly different results for different plasma models. At the example of ITER, predictions from scaling laws constructed out of measurements of present experiments suggest $\lambda_q \approx 1$mm [26]. Calculations performed with the transport code `SOLPS` yield $\lambda_q = 3.6$mm [27] and recent studies with the gyrokinetic turbulence code `XGC1` showed that electron turbulence can lead to a broadening of λ_q to 5.9mm [28]. The heat flux onto the divertor is also reduced by inserting a neutral gas into the SOL. The neutral gas gets ionised from the particles in the plasma. Free electrons recombine with the ions and emit gamma radiation. This provides an efficient mechanism to transport energy out of the plasma before it hits the wall. It is even possible to radiate all the excess energy of the plasma before it comes into contact with the wall. This state is also known as detachment [29]. In order to understand and accurately describe the particle and heat exhaust of a fusion experiment, all processes involved and their interrelation need to be understood. This includes the turbulent transport of heat from the core over the separatrix into the SOL, the dynamics of the plasma in the SOL itself and its interaction with the wall.

1.1.7. Turbulence simulation

As we have established so far, turbulence plays integral part in the feasibility of fusion energy. Due to the limited number of analytical results, numerical methods are employed to study turbulence in fusion experiments.

High fidelity simulations of turbulent fusion plasmas are based on the gyrokinetic equations, a set of nonlinear, integro-partial differential equations that describe the evolution of the distribution of particles in a five-dimensional phase space [30]. The first gyrokinetic simulations were performed in the 80s and 90s based on many simplifying assumptions [31, 32, 33, 34, 35]. Around the turn of the millennium, the simulations became more realistic and were able to provide quantitative insights into the properties of plasma turbulence in the core of fusion devices [36, 37, 38, 39, 40, 41, 42, 43, 44, 45].

Expanding the simulations from the core to the edge and SOL is difficult. In the boundary region the gradients are steep and the temperature and density is low. As a result, the amplitude of the turbulent fluctuations is comparable to the plasma background and a so-called full-f treatment of the equations is necessary. Furthermore, the geometry of the magnetic equilibrium is complex. We explain this challenge in more detail in the next subsection.

Due to the high computational demand of gyrokinetic simulations and the increased collsionality in the edge and SOL, fluid and gyrofluid models are often used [46, 47, 48, 49, 50, 51]. They assume that collisions move the velocity distribution of particles close to a Maxwellian that can be described with a few moments like density and temperature. They are either based on the drift reduced Braginskii equations [52, 53] or on gyrofluid models [54]. However, kinetic effects are known to be important in the edge and SOL [55, 56, 57]. Consequently, several different projects have been launched to simulate gyrokinetic turbulence in the edge and SOL, namely `XGC` [58], `COGENT` [59], `GKEYLL` [60], `GENE` [61], `GYSELA` [62, 63] and `PICLS` [64]. The codes are based on different numerical approaches with their respective advantages and

disadvantages. The different approaches are split into three categories.

The particle-in-cell approach

The first gyrokinetic plasma simulations were based on the so-called particle-in-cell (PIC) approach [31, 32, 33]. Therein, a sample of so-called markers are integrated along the characteristics of the gyrokinetic Vlasov equation, and the electromagnetic fields are solved on a three-dimensional grid in position space. Particle-in-cell codes are straightforward to implement and have shown good scalability on modern supercomputers [28]. Due to the probabilistic nature of these algorithms, the results contain statistical noise which has shown to be problematic for certain applications [65, 66, 67]. From the gyrokinetic edge and SOL codes presented, `XGC` and `PICLS` implement the PIC method. To resolve the full distribution function in SOL plasmas and keep the numerical noise at a manageable level, many markers are needed which may require larger computational resources than comparable continuum codes.

The Eulerian approach

In continuum or Eulerian schemes the gyrokinetic Vlasov-Maxwell system is discretised on a five-dimensional position-velocity space grid. They were first introduced around the year 2000 [68, 39, 38, 56, 40] and use a variety of numerical methods available for partial differential equations. This includes finite difference, finite volume, finite element and discontinuous Galerkin methods with explicit and implicit time stepping schemes. Compared to PIC codes, continuum codes are free of numerical noise at the expense of introducing a velocity space grid. From the gyrokinetic edge and SOL codes mentioned above, `COGENT`, `GKEYLL` and `GENE` implement a continuum method. `GKEYLL` uses an energy conservative discontinuous Galerkin discretisation, `COGENT` discretises the gyrokinetic system with a high order finite volume method and `GENE` features a mixed shock stable finite-volume, finite difference discretisation scheme.

The semi-Lagrangian approach

In the semi-Lagrangian approach the gyrokinetic Vlasov-Maxwell system is, similar to Eulerian schemes, discretised on a five-dimensional position-velocity space grid. Therefore, they are also free of numerical noise. Semi-Lagrangian schemes are commonly divided into three categories: (i) The (standard) backward [69], (ii) the forward [70] and (iii) the conservative [71] semi-Lagrangian method. We explain them at the example of the backward method. The distribution function is evolved by following the characteristics of the gyrokinetic Vlasov equation from a given grid point, on the five-dimensional position-velocity space grid, backwards in time for one timestep. The new value of the distribution function at the grid point is the interpolated value at the foot of that characteristic curve, using the values of the distribution function at the previous timestep [69]. Semi-Lagrangian schemes can allow for larger timesteps than comparable Eulerian schemes, which makes them, next to fusion applications, also attractive for numerical weather predictions and atmospheric models [72]. The gyrokinetic edge and SOL code `GYSELA` implements the semi-Lagrangian approach.

1.1.8. Handling X-point geometries

Especially challenging for turbulence simulations in the edge and SOL is the complex magnetic geometry. There are regions of open and closed magnetic field lines, and, in stellarator fusion experiments, even regions, where field lines are oriented stochastically. Many gyrokinetic codes present in the community implement so-called flux-coordinate systems [73]. While these allow for efficient simulations in regions with regular magnetic flux surfaces, they become ill-defined at the separatrix or in stochastic regions [74]. To allow for predictive edge and SOL simulations, coordinate systems free of singularities that can treat realistic magnetic geometries must be used. `XGC1` has demonstrated that PIC codes can perform simulations in X-point geometries by using a cylindrical coordinate system. PIC codes follow the trajectories of marker particles in continuous real space and therefore parallel structures in the plasma are well represented. However, coordinate systems agnostic of the magnetic field are suboptimal for continuum codes. They use a five-dimensional phase space grid and a fine toroidal resolution is needed to represent structures that are aligned to magnetic field lines. This increases the cost of the simulation significantly because more grid points and, in case of explicit time stepping, a smaller timestep is needed. To allow for efficient simulations in X-point geometry with continuum codes, locally field-aligned methods based on the flux-coordinate independent approach (FCI) [75, 76] have been developed and tested in fluid codes [48, 77, 51, 78]. Within the fluid community, they have shown to be effective for studying edge and SOL turbulence. Recently, the use of FCI has enabled the simulation of turbulence in the diverted geometry of AUG [79, 80]. In the gyrokinetic community, locally field-aligned methods have been implemented in `COGENT` [81] and used to perform five-dimensional simulations in a geometry with one X-point. Further, field-aligned interpolation has been used to accelerate semi-Lagrangian codes [82, 83].

1.2. Overview

Predictive, first principle modelling of turbulence is essential to understand the plasma properties in the boundary of fusion devices. To this end, we present a new gyrokinetic turbulence code that is tailored towards edge an SOL simulations in this work. The code is based on the continuum approach to resolve the full distribution in regions in the SOL where the plasma density is low. We put a special focus on handling the geometric complexities. For this reason, we employ a locally field-aligned coordinate system based on the FCI approach. By using the FCI approach, we are not limited by the magnetic geometry and are able to perform simulations in geometries with a single and even multiple X-points. This will enable the investigation of advanced divertor concepts [84, 85] within gyrokinetic theory in the future. Furthermore, the coordinate system is well-defined in stochastic regions, and a generalisation of the code to stellarator geometries is straightforward. The implementation follows the example of the Braginskii fluid code `GRILLIX` [74, 48, 86] and uses unstructured, locally Cartesian grids. These grids allow for a high flexibility regarding the geometry while ensuring good numerical properties and a high computational efficiency [87]. The discretisation follows

the example of the GENE code [38, 45] and uses explicit time stepping with a mixed order finite difference scheme. The numerical solution of the gyrokinetic Vlasov-Maxwell system on modern supercomputers is resource intensive. To maximize the performance of the code and prepare it for the heterogeneous architectures of future exascale machines, we put a special emphasis on software engineering and performance. Based on its scientific ancestor, GENE, we call the code GENE-X.

The major scientific contributions of this work are threefold. First, next to GENE-X, only XGC1 and COGENT can perform gyrokinetic turbulence simulations in X-point geometry. To ensure good scientific practice, it is vital to have multiple codes, using different numerical techniques, that can reproduce each others findings. Second, we demonstrate that the FCI approach together with the presented numerical scheme works well with a gyrokinetic model. Third, we present gyrokinetic turbulence simulations of the edge and SOL of ASDEX Upgrade that include a magnetic X-Point.

This work is structured as follows. In Chapter 2, we give an introduction into gyrokinetic theory and derive the full-f, electromagnetic model implemented in the code. We continue with the rigorous derivation of the FCI coordinate system and the metric coefficients in Chapter 3. In Chapter 4, we discuss the normalisation of the equations and present the finite difference discretisation scheme implemented. In Chapter 5, we detail the software engineering of the GENE-X code and demonstrate the performance of the software design. We present thorough verification benchmarks in different magnetic geometries, including a full order analysis with the method of manufactured solutions, in Chapter 6. We finish this work with presenting electromagnetic, gyrokinetic simulations of the edge and SOL of ASDEX Upgrade in Chapter 7.

2. Gyrokinetic theory

We start the discussion of gyrokinetic theory at the example of a single particle with mass m and charge q that moves in a homogeneous magnetic field with field strength B. We denote the unit vector of the magnetic field with $\boldsymbol{b} = \boldsymbol{B}/B$. Due to the Lorentz force, the particle gyrates around a magnetic field line with the cyclotron frequency

$$\Omega = \frac{qB}{mc} \,. \tag{2.1}$$

The speed of the particle on the gyroorbit, $v_\perp = |\boldsymbol{v} - (\boldsymbol{b} \cdot \boldsymbol{v})\boldsymbol{b}|$, is constant and the magnetic moment of the particle,

$$\mu = \frac{m v_\perp^2}{2B} \,, \tag{2.2}$$

is conserved. Therefore, the constant gyromotion can be neglected. In a fusion device the conservation of the magnetic moment is violated. First, the magnetic field is non-uniform. Second, the plasma produces density and current fluctuations which produce electromagnetic fields themselves. In the following, we denote the perturbed electrostatic potential with ϕ_1 and the perturbed electromagnetic vector potential with $\boldsymbol{A}_1$. These fields alter the trajectory of particles via the Lorentz force, which ultimately leads to turbulence.

The plasma in a fusion device is a collection of particles of different species, like electrons, ions or impurities. The phase space distribution of all particles of a given species, denoted by σ, is described with their distribution function f_σ. The time evolution of the distribution function is given by the Vlasov equation that is coupled to Maxwell's equations to provide the electromagnetic fields ϕ_1 and $\boldsymbol{A}_1$. This system contains a multitude of physical effects, including the full gyroorbits of particles, Debye shielding and Langmuir waves. While these are interesting phenomena, they act on significantly smaller time scales than turbulence. Numerical solutions of the full Vlasov-Maxwell system at the scale of medium size fusion experiments and on turbulence time scales are not even possible on the largest supercomputers available today. Consequently, it is necessary to derive a simplified model that removes the fast timescales from the system and makes the simulation possible. This is the goal of gyrokinetic theory.

In gyrokinetic theory the goal is achieved by introducing a new set of phase space coordinates, called *gyrocentre coordinates*. They include a generalised magnetic moment μ_{gy} that is an exact invariant of the motion of the particle. In addition, the dynamics associated to the fast gyromotion of the particle, called θ_{gy}, are decoupled from the other gyrocentre coordinates [88]. This removes the fast timescales from the system and reduces the dimensionality from six to five.

The reduction from the full kinetic system to the gyrokinetic subset is controlled with so-called ordering parameters. They express the spatial and temporal scales of different phenomena and are used to control their size. In the following, we define the ordering parameters used in standard gyrokinetic theory [88].

1. We call the length scale of perturbations in the magnetic field $L_B = B/|\boldsymbol{\nabla} B|$ and define the ordering parameter $\epsilon_B = \rho_T / L_B$, where

$$\rho_T = \frac{m c v_{T,\perp}}{|q| B} \tag{2.3}$$

   is the gyroradius and $v_{T,\perp}$ is the perpendicular velocity of a thermal ion. We assume that the length scale of the magnetic fluctuations is much larger than the gyroradius of a thermal ion, i.e. $\epsilon_B \ll 1$.

2. We define ordering parameter for electrostatic perturbations as $\epsilon_\delta = k_\perp \rho_T q \phi_1 / T_i$, where $k_\perp$ represents the wavenumber of the electrostatic fluctuation perpendicular to the magnetic field and T_i the ion temperature. We assume that $\epsilon_\delta \ll 1$ and that fluctuations of the magnetic vector potential $\boldsymbol{A}_1$ are of the order $\phi_1 c / v_T$.

3. We call the wavevector of fluctuations parallel to the background magnetic field $k_\parallel$. In strongly magnetised plasmas parallel perturbations act on larger scales than perpendicular ones. We reflect this property by assuming that $\epsilon_\parallel = k_\parallel / k_\perp \ll 1$.

4. We call the characteristic frequency of fluctuations in the plasma ω and assume that the frequency is much smaller than the ion cyclotron frequency Ω_i, i.e. $\epsilon_\omega = \omega / \Omega_i \ll 1$.

For the derivation of the gyrokinetic model, different orderings between the four different parameters can be used. In this work, we employ the standard ordering used by gyrokinetic codes in the community [36, 37, 38, 40, 41, 42, 43, 44, 45]. It is assumed that fluctuations in the background magnetic field are smaller than fluctuations in the electrostatic potential, i.e. $\epsilon_B \sim \epsilon_\delta^2$. Further, it is assumed that $\epsilon_\parallel \sim \epsilon_\delta$ and $\epsilon_\omega \sim \epsilon_\delta$. The projects mentioned above study plasma turbulence in the core of fusion devices. At the point of writing it is an area of active research whether this ordering and the resulting gyrokinetic theory is also applicable in the edge and SOL. Different studies with hybrid and fully kinetic models indicate that the standard gyrokinetic model is still useful [89, 90, 91, 92].

In the remainder of this chapter, we present the gyrokinetic Vlasov-Maxwell system that is implemented in the `GENE-X` code. As gyrokinetic theory is fundamentally linked to coordinate transformations on manifolds, we start by discussing the basics of Hamiltonian mechanics. We proceed by giving an overview of the gyrokinetic reduction and present the gyrokinetic Vlasov-Maxwell system. Finally, we simplify the system to a self-consistent subset, suitable for the first implementation in the `GENE-X` code and discuss conservation laws.

2.1. Hamiltonian mechanics

The plasma in a fusion device is formally described by a manifold which we call $\mathcal{M}$. This manifold carries a differentiable structure inherited from differentiable coordinate charts (φ, U) or a collection of coordinate charts $(\varphi_\alpha, U_\alpha)$, called an atlas[†]. Charts assign a coordinate $\boldsymbol{q} \in V = \varphi(U)$ to every point on the manifold $\mathcal{M}$. What is commonly known as a coordinate map is the inverse of the chart $\varphi^{-1} : V \to \mathcal{M}$, assigning a point on the manifold to every coordinate. As the charts are diffeomorphic, the coordinate map is always well-defined. In the application of fusion physics the manifold is embedded in $\mathbb{R}^3$ and a natural Cartesian coordinate system exists. We call the Cartesian coordinates of the three dimensional manifold $\boldsymbol{q} = (x_c, y_c, z_c) \in V$. Particles in the plasma have a velocity. The velocity space at position $\boldsymbol{q}$ is modelled by the tangent space of the manifold $T_{\boldsymbol{q}}\mathcal{M}$. The inverse of the coordinate chart φ^{-1} provides a natural basis for the tangent space $T_{\boldsymbol{q}}\mathcal{M}$ by

$$\boldsymbol{e}_i := \frac{\partial \varphi^{-1}}{\partial q_i} \,. \tag{2.4}$$

The basis of the cotangent space $T^*_{\boldsymbol{q}}\mathcal{M}$, denoted by $\boldsymbol{e}^i = \mathrm{d}q_i$, is defined by the condition

$$\boldsymbol{e}^i \cdot \boldsymbol{e}_j = \delta^i_j \,. \tag{2.5}$$

We denote the Cartesian coordinates in velocity space, induced by the Cartesian coordinates in real space, with (v_x, v_y, v_z). In the Lagrangian formulation of classical mechanics the state of a particle is described by its position on the manifold $\boldsymbol{q} \in \mathcal{M}$ and its generalised velocity $\dot{\boldsymbol{q}} \in T_{\boldsymbol{q}}\mathcal{M}$. The equations of motion are determined by the so-called Lagrangian

$$L : (\boldsymbol{q}, \dot{\boldsymbol{q}}, t) \mapsto L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) \tag{2.6}$$

through the Euler-Lagrange equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_i} = \frac{\partial L}{\partial q_i} \,. \tag{2.7}$$

In the Lagrange formalism arbitrary coordinate transformations of the coordinates $\boldsymbol{q}$ are possible with a transformation to a different coordinate chart. In Hamiltonian mechanics the generalised velocity $\dot{\boldsymbol{q}}$ is replaced by the generalised momentum

$$p_i := \left(\frac{\partial L}{\partial \dot{q}_i}\right)(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) \,. \tag{2.8}$$

The generalised momentum $\boldsymbol{p}$ is a cotangent vector and contained in the cotangent space $T^*_{\boldsymbol{q}}\mathcal{M}$. The vector bundle of the cotangent spaces at all the points in the manifold $\mathcal{M}$ defines the phase space. We denote it with $\mathcal{P} := T^*\mathcal{M}$. The phase space is a six-dimensional, differentiable manifold and different charts can be used to define coordinate systems. The phase space $\mathcal{P}$ carries a symplectic structure given by the closed, non-degenerate differential

[†]We construct an atlas of $\mathcal{M}$ in Chapter 3 when we introduce a locally field-aligned coordinate system.

2-form

$$\omega^2 = \mathrm{d}\boldsymbol{p} \wedge \mathrm{d}\boldsymbol{q} = \mathrm{d}p_1 \wedge \mathrm{d}q_1 + \cdots + \mathrm{d}p_3 \wedge \mathrm{d}q_3 \,. \tag{2.9}$$

The 2-form ω^2 can be interpreted as a skew-symmetric scalar product on $T\mathcal{P}$. Like a normal scalar product, Eq. (2.9) introduces a natural isomorphism between differential forms and vector fields on the manifold. A vector $\xi \in T\mathcal{P}$ is mapped to a differential form via [93, pp. 203–204]

$$I : \xi \mapsto \omega^1 = \left(\eta \mapsto \omega^2(\eta, \xi)\right) \,. \tag{2.10}$$

In the coordinate system $(\boldsymbol{p}, \boldsymbol{q})$ the isomorphism I can be represented as a simple matrix multiplication with [93, p. 203]

$$I = \begin{pmatrix} 0 & \mathbb{1}_3 \\ -\mathbb{1}_3 & 0 \end{pmatrix} \,. \tag{2.11}$$

A coordinate transformation τ on the phase space $\mathcal{P}$ is called canonical if it preserves the 2-form ω^2 [93, p. 206]. The volume element of the phase space is given by $\omega^2 \wedge \omega^2 \wedge \omega^2$ and therefore canonical transformations preserve the volume element in phase space. The Hamiltonian H is a function on the phase space. With the inverse of the isomorphism I, a vector field can be associated to the Hamiltonian via $I^{-1}\,\mathrm{d}H$. In the coordinates $(\boldsymbol{p}, \boldsymbol{q})$ the Hamiltonian vector field reads [93, p. 203]

$$I^{-1}\,\mathrm{d}H = \begin{pmatrix} \frac{\partial H}{\partial \boldsymbol{p}} \\ -\frac{\partial H}{\partial \boldsymbol{q}} \end{pmatrix} \,. \tag{2.12}$$

The Hamiltonian flow $g_H^\lambda : \mathcal{P} \to \mathcal{P}$ is defined as a 1-parameter group of diffeomorphisms on the manifold $\mathcal{P}$ such that the rate of change of the flow, contained in the tangent space $T\mathcal{P}$, is given by the Hamiltonian vector field, i.e. [93, p. 204]

$$\left.\frac{\mathrm{d}}{\mathrm{d}\lambda}\right|_{\lambda=0} g_H^\lambda = I^{-1}\,\mathrm{d}H \,. \tag{2.13}$$

In the coordinates $(\boldsymbol{p}, \boldsymbol{q})$, Eq. (2.13) gives rise to Hamilton's equations [93, p. 204]

$$\begin{aligned} \dot{q}_i &= \frac{\partial H}{\partial p_i} \\ \dot{p}_i &= -\frac{\partial H}{\partial q_i} \,. \end{aligned} \tag{2.14}$$

They describe the trajectories of the Hamiltonian flow in phase space. The Hamiltonian orbit parameter λ can be interpreted as the time. The Poisson bracket $\{f, h\}$ of two functions on the manifold $\mathcal{P}$ is defined as the derivative of the function f in the direction of the Hamiltonian

flow of h [93, pp. 214–215]

$$\begin{aligned}\{f,h\} &:= \left.\frac{\mathrm{d}}{\mathrm{d}\lambda}\right|_{\lambda=0} f(g_h^\lambda)\\ &= \mathrm{d}f(I^{-1}\,\mathrm{d}h)\\ &= \omega^2(I^{-1}\,\mathrm{d}h, I^{-1}\,\mathrm{d}f)\end{aligned} \tag{2.15}$$

and yields the well-known form in the coordinates $(\boldsymbol{p},\boldsymbol{q})$ [93, p. 215]

$$\{f,h\} = \sum_{i=1}^{3} \frac{\partial f}{\partial q_i}\frac{\partial h}{\partial p_i} - \frac{\partial f}{\partial p_i}\frac{\partial h}{\partial q_i}\,. \tag{2.16}$$

In simple words, Eq. (2.15) shows that the rate of change of a function along the Hamiltonian flow is given by the Poisson bracket. It can be shown that a transformation is canonical if and only if it preserves the structure of the Poisson bracket [93, p. 216]. After this formal discussion of Hamiltonian mechanics on symplectic manifolds, we introduce the Lagrangian 1-form that connects Hamiltonian to Lagrangian mechanics and is used in the presentation of the gyrokinetic model in the next section. In the following, the phase space is extended by the Hamiltonian orbit parameter λ [30]. The Lagrangian 1-form reads

$$\omega^1 := \boldsymbol{p}\,\mathrm{d}\boldsymbol{q} - H\,\mathrm{d}\lambda\,. \tag{2.17}$$

The part $\boldsymbol{p}\,\mathrm{d}\boldsymbol{q}$ is often referred to as the symplectic part and $H\,\mathrm{d}\lambda$ as the Hamiltonian part of the 1-form. The symplectic 2-form ω^2 is obtained by taking the exterior derivative $\omega^2 = \mathrm{d}(\boldsymbol{p}\,\mathrm{d}\boldsymbol{q})$ of the symplectic part of ω^1. Using Eq. (2.15), ω_1 gives rise to the Poisson bracket and therefore Hamiltonian mechanics. Further, the 1-form ω^1 allows for a straightforward connection to Lagrangian mechanics. The integral over ω^1 along a trajectory $\boldsymbol{\gamma}$ in the extended phase space starting from $(\boldsymbol{p}_0,\boldsymbol{q}_0,\lambda_0)$ and going to $(\boldsymbol{p}_1,\boldsymbol{q}_1,\lambda_1)$ has an extremal under variations of $\boldsymbol{\gamma}$. The extremum is given by the integral curves of Hamilton's equations [93, pp. 243–244]. The connection to the Lagrangian, as the Legendre transformation of the Hamiltonian, is given by

$$\int_{\boldsymbol{\gamma}} \boldsymbol{p}\,\mathrm{d}q - H\,\mathrm{d}\lambda = \int_{\lambda_0}^{\lambda_1} (\boldsymbol{p}\dot{\boldsymbol{q}} - H)\,\mathrm{d}\lambda = \int_{\lambda_0}^{\lambda_1} L\,\mathrm{d}\lambda\,. \tag{2.18}$$

In the following sections, we give an overview of the derivation of the gyrokinetic equations. The starting point is the form ω^1 of a charged particle in an electromagnetic field. The coordinate transformation into gyrocentre coordinates is non-canonical and therefore results in a non-canonical expression for the symplectic part of ω^1 and the Poisson bracket [30].

2.2. Common coordinate systems for fusion plasmas

Before we start with the derivation of the gyrokinetic Vlasov-Maxwell system, we introduce different coordinate systems that are more suited to describe fusion plasmas than the

$(x_c, y_c, z_c, v_x, v_y, v_z)$ coordinate system introduced above. In magnetic confinement fusion devices the shape of the plasma is formed by the magnetic field. Therefore, it is convenient to use a cylindrical and cylindrical-toroidal coordinate system that reflects its structure. The cylindrical coordinate system is given by

$$R = \sqrt{x_c^2 + y_c^2} \tag{2.19}$$
$$\varphi = \operatorname{atan2}(y_c, x_c) \tag{2.20}$$
$$Z = z_c \tag{2.21}$$

and the cylindrical-toroidal coordinate system by [73]

$$r = \sqrt{(\sqrt{x_c^2 + y_c^2} - R_0)^2 + z_c^2} \tag{2.22}$$
$$\theta = \operatorname{atan2}(z_c, \sqrt{x_c^2 + y_c^2} - R_0) \tag{2.23}$$
$$\varphi = \operatorname{atan2}(y_c, x_c)\,, \tag{2.24}$$

where R_0 denotes the major radius of the torus. As the plasma is magnetised, charged particles follow magnetic field lines. Therefore, it is useful to also introduce different coordinates in velocity space. Given an orthonormal basis $(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b})$ in the tangent space $T_{\boldsymbol{q}}\mathcal{M}$ at the point $\boldsymbol{q} \in \mathcal{M}$ and the vector $\boldsymbol{v} \in T_{\boldsymbol{q}}\mathcal{M}$, we define the coordinates $(v_1, v_2, v_\parallel)$ by the coordinate transformation

$$v_1 = \boldsymbol{b}_1 \cdot \boldsymbol{v} \tag{2.25}$$
$$v_2 = \boldsymbol{b}_2 \cdot \boldsymbol{v} \tag{2.26}$$
$$v_\parallel = \boldsymbol{b} \cdot \boldsymbol{v} \tag{2.27}$$

and the coordinates $\left(v_\parallel, \mu, \theta\right)$ by the coordinate transformation

$$\mu = \frac{m_\sigma \left(v_1^2 + v_2^2\right)}{2B} \tag{2.28}$$
$$\theta = \operatorname{atan2}(v_2, v_1)\,. \tag{2.29}$$

The new coordinates in velocity space can be interpreted in a simple way. The coordinate $v_\parallel$ denotes the velocity of a charged particle along the field line, μ the magnetic moment and θ the angle of the gyromotion. The kinetic energy of the particle is given by $E = \mu B + \frac{1}{2} m_\sigma v_\parallel^2$. Further, we define the gyromotion vectors

$$\boldsymbol{\rho}(\theta) = \boldsymbol{b}_1 \cos\theta - \boldsymbol{b}_2 \sin\theta \tag{2.30}$$
$$\perp(\theta) = -\boldsymbol{b}_1 \sin\theta - \boldsymbol{b}_2 \cos\theta \tag{2.31}$$

forming a second orthonormal basis set $(\perp(\theta), \boldsymbol{\rho}(\theta), \boldsymbol{b})$[†]. As the magnetic field in general changes for every point $\boldsymbol{q} \in \mathcal{M}$, so do the coordinates in velocity spaces.

[†]The set $(\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b})$ is also known as the fixed frame and $(\perp(\theta), \boldsymbol{\rho}(\theta), \boldsymbol{b})$ as the rotating frame [94].

2.3. Derivation of the gyrokinetic Vlasov-Maxwell system

The derivation of the gyrocentre coordinates starts with a charged particle moving in the background magnetic field $\boldsymbol{B}$ under the influence of time dependent perturbed electromagnetic fields $\boldsymbol{A}_1$ and ϕ_1. In the presence of time dependent electromagnetic fields, the energy k of the particle is not conserved and changes according to the additional equation of motion [30]

$$\frac{\mathrm{d}k}{\mathrm{d}t} = q_\sigma \left(\frac{\partial \phi_1}{\partial t} - \frac{\boldsymbol{v}}{c} \cdot \frac{\partial \boldsymbol{A}_1}{\partial t} \right) . \tag{2.32}$$

To account for the change of energy, the phase space is expanded by the canonically conjugate pair (t, k), where k is constrained to the conserved energy in the time independent case [30]. This allows the formulation of Hamilton's equations of motion with time dependent electromagnetic fields [30].

2.3.1. The gyrokinetic Lagrangian 1-form

The 1-form ω^1, introduced in Eq. (2.17), describing the trajectory of the charged particle in the extended phase space coordinates $(R, \varphi, Z, t, v_\parallel, \mu, \theta, k, \lambda)$ reads

$$\omega^1 = \left(\frac{q_\sigma}{c} \boldsymbol{A}(\boldsymbol{R}) + \epsilon_\delta \frac{q_\sigma}{c} \boldsymbol{A}_1(\boldsymbol{R}, t) + m_\sigma \boldsymbol{v} \right) \cdot \mathrm{d}\boldsymbol{R} - k \, \mathrm{d}t - H \, \mathrm{d}\lambda \, , \tag{2.33}$$

with the Hamiltonian

$$H = \frac{1}{2} m_\sigma \boldsymbol{v}^2 + \epsilon_\delta q_\sigma \phi_1(\boldsymbol{R}, t) - k \, . \tag{2.34}$$

The physical motion of the particle takes place on the subspace $H = 0$, and Hamilton's equation of motion for the time, $\dot{t} = 1$, can be used to replace the Hamiltonian orbit parameter λ with the time t [30].

Both the symplectic and Hamiltonian part of the Lagrangian 1-form depend on the velocity and therefore have a dependence on the gyroangle θ. The symplectic part of ω^1 depends on the perturbed magnetic vector potential $\boldsymbol{A}_1$ and the Hamiltonian part on the perturbed electrostatic potential ϕ_1. The derivation of the gyrokinetic 1-form comprises of two steps. First, the perturbed fields $\boldsymbol{A}_1$ and ϕ_1 are removed from ω_1 and a transformation into so-called guiding-centre coordinates $(R_{\mathrm{gc}}, \varphi_{\mathrm{gc}}, Z_{\mathrm{gc}}, v_{\parallel \mathrm{gc}}, \mu_{\mathrm{gc}}, \theta_{\mathrm{gc}})$ is performed. The guiding-centre transformation removes the fast gyromotion time scales that are associated with the magnetic background $\boldsymbol{A}$. In a second step, the perturbed fields $\boldsymbol{A}_1$ and ϕ_1 are added to the guiding-centre 1-form again and a transformation into gyrocentre coordinates $(R_{\mathrm{gy}}, \varphi_{\mathrm{gy}}, Z_{\mathrm{gy}}, v_{\parallel \mathrm{gy}}, \mu_{\mathrm{gy}}, \theta_{\mathrm{gy}})$ is performed. The gyrocentre transformation removes the fast gyromotion time scales caused by the perturbed electromagnetic fields. In this work, we employ the so-called symplectic gyrocentre model [95, 30]. The symplectic gyrocentre model retains explicit time derivatives in the Poisson bracket and has been shown to be well suited for gyrokinetic continuum simulations [96, 97]. In contrast, so-called Hamiltonian gyrocentre models [95, 30] are often used in particle-in-cell codes [43]. They have the advantage that they are free of explicit time derivatives of the perturbed electromagnetic vector po-

tential, but can lead to the so-called Ampère cancellation problem in gyrokinetic turbluence simulations [98, 99, 100].

We start with the discussion of the guiding-centre coordinates. For brevity, we state and interpret the result of the transformation. For an explicit calculation we refer to [101, 102, 103, 88]. In guiding-centre coordinates, the form ω^1 reads

$$\omega^1_{\mathrm{gc}} = \left(\frac{1}{\epsilon_B}\frac{q_\sigma}{c}\boldsymbol{A}(\boldsymbol{R}_{\mathrm{gc}}) + \epsilon_B m_\sigma v_{\parallel\mathrm{gc}}\boldsymbol{b}(\boldsymbol{R}_{\mathrm{gc}}) - \epsilon_B\frac{m_\sigma c}{q_\sigma}\mu_{\mathrm{gc}}\boldsymbol{R}^*\right)\cdot \mathrm{d}\boldsymbol{R}_{\mathrm{gc}} + \epsilon_B\frac{m_\sigma c}{q_\sigma}\mu_{\mathrm{gc}}\,\mathrm{d}\theta_{\mathrm{gc}} - k\,\mathrm{d}t - H_{\mathrm{gc}}\,\mathrm{d}\lambda + \mathcal{O}(\epsilon_B^2)\,, \tag{2.35}$$

where the expression $\boldsymbol{R}^* = \boldsymbol{\nabla}\boldsymbol{b}_1\cdot\boldsymbol{b}_2 + (\boldsymbol{b}\cdot\boldsymbol{\nabla}\times\boldsymbol{b})\,\boldsymbol{b}/2$ and the Hamiltonian

$$H_{\mathrm{gc}} = \frac{1}{2}m_\sigma v^2_{\parallel\mathrm{gc}} + \mu_{\mathrm{gc}}B(\boldsymbol{R}_{\mathrm{gc}}) - k \tag{2.36}$$

were used. As a result of the guiding-centre transformation, the symplectic and Hamiltonian part of ω^1 are independent of the gyroangle θ_{gc}. The Poisson bracket of the system is calculated from the symplectic part of ω^1 by deriving the matrix representation of the isomorphism I, its inverse I^{-1} and using Eq. (2.15). It reads [30, 88]

$$\begin{aligned}\{f,g\}_{\mathrm{gc}} = \frac{1}{\epsilon_B}\frac{q_\sigma}{m_\sigma c}\left(\frac{\partial f}{\partial\theta_{\mathrm{gc}}}\frac{\partial g}{\partial\mu_{\mathrm{gc}}} - \frac{\partial f}{\partial\mu_{\mathrm{gc}}}\frac{\partial g}{\partial\theta_{\mathrm{gc}}}\right) + \frac{\boldsymbol{B}^*_{\mathrm{gc}}}{m_\sigma B^*_{\parallel\mathrm{gc}}}\cdot\left(\boldsymbol{\nabla}^*_{\mathrm{gc}}f\frac{\partial g}{\partial v_{\parallel\mathrm{gc}}} - \frac{\partial f}{\partial v_{\parallel\mathrm{gc}}}\boldsymbol{\nabla}^*_{\mathrm{gc}}g\right)\\ - \epsilon_B\frac{c\boldsymbol{b}}{q_\sigma B^*_{\parallel\mathrm{gc}}}\cdot\left(\boldsymbol{\nabla}^*_{\mathrm{gc}}f\times\boldsymbol{\nabla}^*_{\mathrm{gc}}g\right)\\ + \left(\frac{\partial f}{\partial k}\frac{\partial g}{\partial t} - \frac{\partial f}{\partial t}\frac{\partial g}{\partial k}\right)\,,\end{aligned} \tag{2.37}$$

where the definitions

$$\boldsymbol{\nabla}^*_{\mathrm{gc}} = \boldsymbol{\nabla} + \boldsymbol{R}^*\frac{\partial}{\partial\theta_{\mathrm{gc}}} \tag{2.38}$$

$$\boldsymbol{B}^*_{\mathrm{gc}} = \boldsymbol{B} + \epsilon_B\frac{m_\sigma c}{q_\sigma}v_{\parallel\mathrm{gc}}\boldsymbol{\nabla}\times\boldsymbol{b} - \epsilon_B^2\frac{m_\sigma c^2}{q_\sigma^2}\mu_{\mathrm{gc}}\boldsymbol{\nabla}\times\boldsymbol{R}^* \tag{2.39}$$

were used. The parallel component of $\boldsymbol{B}^*_{\mathrm{gc}}$ is called $B^*_{\parallel\mathrm{gc}} = \boldsymbol{b}\cdot B^*_{\parallel\mathrm{gc}}$. The Jacobian of the transformation from laboratory coordinates $(R,\varphi,\mu,v_\parallel,\mu,\theta)$ into guiding-centre coordinates $(R_{\mathrm{gc}},\varphi_{\mathrm{gc}},Z_{\mathrm{gc}},v_{\parallel\mathrm{gc}},\mu_{\mathrm{gc}},\theta_{\mathrm{gc}})$ is given by $\mathcal{J}_{\mathrm{gc}} = m_\sigma^2 B^{*\dagger}_\parallel$. Although the guiding-centre coordinates are already quite involved, they have a straightforward interpretation: The relation between the particle and its guiding-centre position is given by $\boldsymbol{R} = \boldsymbol{R}_{\mathrm{gc}} + \boldsymbol{\rho}_{\mathrm{gc}}(\boldsymbol{R}_{\mathrm{gc}},\mu_{\mathrm{gc}},\theta_{\mathrm{gc}})$ with the guiding-centre displacement

$$\boldsymbol{\rho}_{\mathrm{gc}}(\boldsymbol{R}_{\mathrm{gc}},\mu_{\mathrm{gc}},\theta_{\mathrm{gc}}) = \frac{m_\sigma c}{q_\sigma B(\boldsymbol{R}_{\mathrm{gc}})}\sqrt{\frac{2\mu_{\mathrm{gy}}B(\boldsymbol{R}_{\mathrm{gy}})}{m}}\boldsymbol{\rho}(\theta_{\mathrm{gc}},\boldsymbol{R}_{\mathrm{gc}}) + \mathcal{O}(\epsilon_B)\,. \tag{2.40}$$

We only present the guiding-centre displacement $\boldsymbol{\rho}_{\mathrm{gc}}$ up to order ϵ_B. For the full expression

[†]The guiding-centre Jacobian is often also given as $\mathcal{J}_{\mathrm{gc}} = m_\sigma B^*_\parallel$. The difference originates from the use of $p_\parallel$ instead of $v_\parallel$ in the guiding-centre coordinates.

we refer to [88, Appendix A]. The guiding-centre position can be interpreted as the centre of the gyromotion of the particle given the non-uniform magnetic field. We continue with the discussion of gyrocentre coordinates. To perform the transformation, the perturbed electromagnetic fields are added to ω^1 again. The 1-form is transformed into

$$\omega^1_{\mathrm{gc}} \to \omega^1_{\mathrm{gc}} + \frac{q_\sigma}{c}\boldsymbol{A}_1\left(\boldsymbol{R}_{\mathrm{gc}} + \boldsymbol{\rho}_{\mathrm{gc}}, t\right) \cdot \mathrm{d}\left(\boldsymbol{R}_{\mathrm{gc}} + \boldsymbol{\rho}_{\mathrm{gc}}\right) - q_\sigma \phi_1\left(\boldsymbol{R}_{\mathrm{gc}} + \boldsymbol{\rho}_{\mathrm{gc}}, t\right) \mathrm{d}\lambda \tag{2.41}$$

and regains an explicit dependence on the fast gyroangle θ_{gc} through the guiding-centre displacement vector $\boldsymbol{\rho}_{\mathrm{gc}}$. Before we present the 1-form in gyrocentre coordinates, we introduce further notation that is used throughout this chapter. We define the well-known gyroaverages by

$$G : \psi \mapsto \left((\boldsymbol{R}, v_\parallel, \mu) \mapsto \frac{1}{2\pi} \int_0^{2\pi} \psi(\boldsymbol{R}, v_\parallel, \mu, \theta) d\theta \right) \tag{2.42}$$

and the fluctuating part of a function g, called $\tilde{g}$, by $\tilde{g} := g - G[g]$. We define the parallel and orthogonal components of the electromagnetic vector potential by

$$A_{1\parallel}(\boldsymbol{R}_{\mathrm{gy}}, t) := \boldsymbol{b}(\boldsymbol{R}_{\mathrm{gy}}) \cdot \boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}, t) \tag{2.43}$$

$$\boldsymbol{A}_{1\perp}(\boldsymbol{R}_{\mathrm{gy}}, t) := \boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}, t) - A_{1\parallel}(\boldsymbol{R}_{\mathrm{gy}}, t)\boldsymbol{b}\,, \tag{2.44}$$

and we introduce the effective electrostatic potential

$$\begin{aligned}\psi_1(\boldsymbol{R}_{\mathrm{gy}}, v_{\parallel\mathrm{gy}}, \mu_{\mathrm{gy}}, \theta_{\mathrm{gy}}, t) = \phi_1(\boldsymbol{R}_{\mathrm{gy}} + \boldsymbol{\rho}_{\mathrm{gc}}, t) \\ - \sqrt{\frac{2\mu_{\mathrm{gy}} B(\boldsymbol{R}_{\mathrm{gy}})}{mc^2}} \perp(\theta_{\mathrm{gy}}, \boldsymbol{R}_{\mathrm{gy}} + \boldsymbol{\rho}_{\mathrm{gc}}) \cdot \boldsymbol{A}_{1\perp}(\boldsymbol{R}_{\mathrm{gy}} + \boldsymbol{\rho}_{\mathrm{gc}}, t)\,.\end{aligned} \tag{2.45}$$

The transformation to gyrocentre coordinates and, in the process, the elimination of the fast gyroangle in the electromagnetic fields is done with the help of Lie transformations [104]. Methods free of Lie transformations based on variational averaging have been used for the derivation as well [105]. Similar to the gyrocentre reduction, we state the results. For a detailed derivation we refer to [106, 30, 88]. In gyrocentre coordinates, the 1-form reads

$$\begin{aligned}\omega^1_{\mathrm{gy}} = \left(\frac{q_\sigma}{c}\left(\boldsymbol{A}(\boldsymbol{R}_{\mathrm{gy}}) + \epsilon_\delta \boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}} + \boldsymbol{\rho}_{\mathrm{gy}})\right) + m_\sigma v_{\parallel\mathrm{gy}} \boldsymbol{b}(\boldsymbol{R}_{\mathrm{gy}}) - \frac{m_\sigma c}{q_\sigma}\mu \boldsymbol{R}^* \right) \cdot \mathrm{d}\boldsymbol{R}_{\mathrm{gy}} \\ + \frac{m_\sigma c}{q_\sigma}\mu_{\mathrm{gy}}\,\mathrm{d}\theta_{\mathrm{gy}} - k\,\mathrm{d}t - H_{\mathrm{gy}}\,\mathrm{d}\lambda + \mathcal{O}(\epsilon_\delta^3)\,,\end{aligned} \tag{2.46}$$

with the gyrocentre Hamiltonian [30]

$$\begin{aligned}H_{\mathrm{gy}} = H_{\mathrm{gc}} + \epsilon_\delta q_\sigma G[\psi_1] + \epsilon_\delta^2 \Bigg(\frac{q_\sigma^2}{2mc^2}\left(G\left[|\boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}} + \boldsymbol{\rho}_{\mathrm{gy}}, t)|^2\right] - G\left[A_{1\parallel}(\boldsymbol{R}_{\mathrm{gy}} + \boldsymbol{\rho}_{\mathrm{gy}}, t)\right]^2 \right) \\ - \frac{q_\sigma^2}{2B(\boldsymbol{R}_{\mathrm{gy}})} \frac{\partial G\left[\widetilde{\psi_1}^2\right]}{\partial \mu_{\mathrm{gy}}} - \frac{mc^2}{2B(\boldsymbol{R}_{\mathrm{gy}})^2} \boldsymbol{b}(\boldsymbol{R}_{\mathrm{gy}}) \cdot G\left[\boldsymbol{\nabla}\widetilde{\psi_1} \times \int \boldsymbol{\nabla}\widetilde{\psi_1}\,\mathrm{d}\theta_{\mathrm{gy}} \right] \\ + G\left[\boldsymbol{A}_{1\perp}(\boldsymbol{R}_{\mathrm{gy}} + \boldsymbol{\rho}_{\mathrm{gy}}, t)\right] \cdot \frac{q_\sigma}{B}\boldsymbol{b} \times \boldsymbol{\nabla} G\left[\psi_1\right] \Bigg) + \mathcal{O}(\epsilon_\delta^3)\,.\end{aligned} \tag{2.47}$$

Analogous to Eq. (2.37), the Poisson bracket in gyrocentre coordinates is obtained by deriving the matrix representation of the isomorphism I, calculating I^{-1} and using Eq. (2.15). It reads [30]

$$\begin{aligned}\{f,g\}_{\mathrm{gy}} = \frac{q_\sigma}{m_\sigma c}\left(\frac{\partial f}{\partial\theta_{\mathrm{gy}}}\frac{\partial g}{\partial\mu^*_{\mathrm{gy}}} - \frac{\partial f}{\partial\mu^*_{\mathrm{gy}}}\frac{\partial g}{\partial\theta_{\mathrm{gy}}}\right) + \frac{\boldsymbol{B}^*}{m_\sigma B^*_\parallel}\cdot\left(\boldsymbol{\nabla}^*_{\mathrm{gy}} f\frac{\partial g}{\partial v_{\parallel\mathrm{gy}}} - \frac{\partial f}{\partial v_{\parallel\mathrm{gy}}}\boldsymbol{\nabla}^*_{\mathrm{gy}} g\right)\\ - \frac{c\boldsymbol{b}}{q_\sigma B^*_\parallel}\cdot\left(\boldsymbol{\nabla}^*_{\mathrm{gy}} f\times\boldsymbol{\nabla}^*_{\mathrm{gy}} g\right)\\ + \left(\frac{\partial f}{\partial k}\frac{\partial g}{\partial t} - \frac{\partial f}{\partial t}\frac{\partial g}{\partial k}\right),\end{aligned} \tag{2.48}$$

where the definitions

$$\boldsymbol{\nabla}^*_{\mathrm{gy}} f = \boldsymbol{\nabla}^*_{\mathrm{gc}} f - \epsilon_\delta\frac{q_\sigma}{c}\left(\frac{\partial G[\boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}+\boldsymbol{\rho}_{\mathrm{gc}},t)]}{\partial t}\frac{\partial f}{\partial k} - \frac{q_\sigma}{m_\sigma c}\frac{\partial G[\boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}+\boldsymbol{\rho}_{\mathrm{gc}},t)]}{\partial\mu_{\mathrm{gy}}}\frac{\partial f}{\partial\theta_{\mathrm{gy}}}\right) \tag{2.49}$$

$$\frac{\partial f}{\partial\mu^*_{\mathrm{gy}}} = \frac{\partial f}{\partial\mu_{\mathrm{gy}}} - \epsilon_\delta^2\left(\frac{q_\sigma}{c}\frac{\partial G[\boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}+\boldsymbol{\rho}_{\mathrm{gc}},t)]}{\partial t}\cdot\left(\frac{\partial G[\boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}+\boldsymbol{\rho}_{\mathrm{gc}},t)]}{\partial\mu_{\mathrm{gy}}}\times\boldsymbol{b}\right)\right)\frac{\partial f}{\partial k} \tag{2.50}$$

and

$$\boldsymbol{B}^* = \boldsymbol{B}^*_{\mathrm{gc}} + \epsilon_\delta G[\boldsymbol{\nabla}\times\boldsymbol{A}_1] \tag{2.51}$$

were used. With Eqs. (2.47) and (2.48), Hamilton's equations of motion in gyrocentre coordinates read [30]

$$\dot{\boldsymbol{R}}_{\mathrm{gy}} = \frac{\boldsymbol{B}^*}{m_\sigma B^*_\parallel}\frac{\partial H_{\mathrm{gy}}}{\partial v_{\parallel\mathrm{gy}}} + \frac{c}{q_\sigma B^*_\parallel}\boldsymbol{b}\times\left(\boldsymbol{\nabla}H_{\mathrm{gy}} + \epsilon_\delta\frac{q_\sigma}{c}\frac{\partial G[\boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}+\boldsymbol{\rho}_{\mathrm{gc}},t)]}{\partial t}\right) \tag{2.52}$$

$$\dot{v}_{\parallel\mathrm{gy}} = -\frac{\boldsymbol{B}^*}{m_\sigma B^*_\parallel}\cdot\left(\boldsymbol{\nabla}H_{\mathrm{gy}} + \epsilon_\delta\frac{q_\sigma}{c}\frac{\partial G[\boldsymbol{A}_1(\boldsymbol{R}_{\mathrm{gy}}+\boldsymbol{\rho}_{\mathrm{gc}},t)]}{\partial t}\right) \tag{2.53}$$

$$\dot{\mu}_{\mathrm{gy}} = 0 \tag{2.54}$$

$$\dot{t} \quad = 1\,. \tag{2.55}$$

For brevity, we omitted the equations of motion for θ and k. As required at the beginning of the section, the equations of motion for $\boldsymbol{R}_{\mathrm{gy}}, v_{\parallel\mathrm{gy}}$ and μ_{gy} are independent of the fast gyroangle θ_{gy}. With the gyrokinetic Lagrangian 1-form the dynamics of charged particles in the plasma is specified. We proceed by deriving the gyrokinetic Vlasov-Maxwell system to obtain equations for the distribution function as well as the perturbed electromagnetic fields. From now on, we drop the $_{\mathrm{gy}}$ annotation on the gyrocentre coordinates and assume that we work in gyrocentre coordinates unless explicitly specified otherwise. In addition, we omit the ordering parameters ϵ_δ and ϵ_B.

2.3.2. The Vlasov equation in gyrocentre coordinates

The particle follows a trajectory through the eight-dimensional extended phase space. The path is denoted with $\boldsymbol{\gamma} : \lambda \mapsto \boldsymbol{\gamma}(\lambda) = (\boldsymbol{R}(\lambda), t(\lambda), v_\parallel(\lambda), \mu(\lambda), \theta(\lambda), k(\lambda))$ such that $\boldsymbol{\gamma}(0)$ represents the initial state of the particle. According to Liouville's theorem in Hamiltonian

mechanics, the distribution function in extended phase space, denoted by $\mathcal{F}_\sigma$, is constant along the trajectories $\boldsymbol{\gamma}$ of the system, i.e.

$$\mathcal{F}_\sigma(\boldsymbol{\gamma}(\lambda)) = \mathcal{F}_\sigma(\boldsymbol{\gamma}(0)). \tag{2.56}$$

Differentiating Eq. (2.56) with respect to the orbit parameter yields the Vlasov equation in extended phase space

$$\frac{\mathrm{d}\mathcal{F}_\sigma}{\mathrm{d}t} = \dot{\boldsymbol{R}} \cdot \boldsymbol{\nabla}\mathcal{F}_\sigma + \dot{t}\frac{\partial \mathcal{F}_\sigma}{\partial t} + \dot{v}_\parallel \frac{\partial \mathcal{F}_\sigma}{\partial v_\parallel} + \dot{\mu}\frac{\partial \mathcal{F}_\sigma}{\partial \mu} + \dot{\theta}\frac{\partial \mathcal{F}_\sigma}{\partial \theta} + \dot{k}\frac{\partial \mathcal{F}_\sigma}{\partial k} = 0\,. \tag{2.57}$$

As mentioned in the beginning, the physical motion of the particle takes place on the hypersurface $H = 0$. The extended phase space is reduced to six dimensions by projecting it onto the hypersurface defined by $H = 0$ and using Hamilton's equations of motion for t to replace the orbit parameter λ. To meet these physical requirements, the extended distribution function $\mathcal{F}_\sigma$ is related to the six-dimensional, physical distribution function f_σ via $\mathcal{F}_\sigma = f_\sigma \delta(H(k))$, where δ is the Dirac delta function [30]. The Vlasov equation on the six-dimensional hypersurface is obtained by integrating Eq. (2.57) over k. It reads

$$\frac{\mathrm{d}f_\sigma}{\mathrm{d}t} = \frac{\partial f_\sigma}{\partial t} + \dot{\boldsymbol{R}} \cdot \boldsymbol{\nabla} f_\sigma + \dot{v}_\parallel \frac{\partial f_\sigma}{\partial v_\parallel} + \dot{\mu}\frac{\partial f_\sigma}{\partial \mu} + \dot{\theta}\frac{\partial f_\sigma}{\partial \theta} = 0\,. \tag{2.58}$$

Similar to the last section, the particle distribution function f_σ can be split into a gyroaveraged part $G[f_\sigma]$ and a fluctuating part $\widetilde{f_\sigma}$ such that $f_\sigma = G[f_\sigma] + \widetilde{f_\sigma}$. By construction of the gyrocentre coordinates, both parts evolve independently in time according to [105]

$$\frac{\partial G[f_\sigma]}{\partial t} + \dot{\boldsymbol{R}} \cdot \boldsymbol{\nabla} G[f_\sigma] + \dot{v}_\parallel \frac{\partial G[f_\sigma]}{\partial v_\parallel} + \dot{\mu}\frac{\partial G[f_\sigma]}{\partial \mu} = 0 \tag{2.59}$$

$$\frac{\partial \widetilde{f_\sigma}}{\partial t} + \dot{\boldsymbol{R}} \cdot \boldsymbol{\nabla} \widetilde{f_\sigma} + \dot{v}_\parallel \frac{\partial \widetilde{f_\sigma}}{\partial v_\parallel} + \dot{\mu}\frac{\partial \widetilde{f_\sigma}}{\partial \mu} + \dot{\theta}\frac{\partial \widetilde{f_\sigma}}{\partial \theta} = 0\,. \tag{2.60}$$

In gyrokinetic theory the fluctuating part $\widetilde{f_\sigma}$ is discarded and only the gyroaveraged part $G[f_\sigma]$ is retained. This removes the fast time scales associated with the gyroangle θ from the system and, therefore, reduces the phase space from six to five dimensions. The equation of motion for the gyroaveraged distribution function is known as the gyrokinetic Vlasov equation. In slight abuse of notation we denote the gyroaveraged distribution function $G[f_\sigma]$ with f_σ and the gyroaveraged extended distribution function $G[\mathcal{F}_\sigma]$ with $\mathcal{F}_\sigma$ for the remainder of this work. The gyrokinetic Vlasov equation reads

$$\frac{\partial f_\sigma}{\partial t} + \dot{\boldsymbol{R}} \cdot \boldsymbol{\nabla} f_\sigma + \dot{v}_\parallel \frac{\partial f_\sigma}{\partial v_\parallel} = 0\,, \tag{2.61}$$

where the time derivative of the gyrocentre trajectories $\dot{\boldsymbol{\gamma}}$ are given by Hamilton's equation of motion presented in Eqs. (2.52) and (2.53).

2.3.3. The gyrokinetic action functional

To solve the dynamics of the plasma self consistently, additional equations for the perturbed electromagnetic fields ϕ_1 and $\boldsymbol{A}_1$ are needed. They are commonly derived via variational methods from the total gyrokinetic action functional of the plasma. We start with the particle Lagrangian, which is obtained from ω^1 via Eq. (2.18). Expressed in gyrocentre coordinates, the particle Lagrangian reads

$$L(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \lambda) = \left(\frac{q_\sigma}{c} \left(\boldsymbol{A} + \boldsymbol{A}_1 \right) + m_\sigma v_\parallel \boldsymbol{b} - \frac{m_\sigma c}{q_\sigma} \mu \boldsymbol{R}^* \right) \cdot \dot{\boldsymbol{R}} + \frac{m_\sigma c}{q_\sigma} \mu \dot{\theta} - k \dot{t} - H \,. \tag{2.62}$$

From Eq. (2.62) the total gyrokinetic Lagrangian can be derived [107, 106]. It yields

$$\mathcal{L}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \phi_1, \boldsymbol{A}_1, \lambda) = \sum_\sigma \int \mathcal{F}_\sigma(\boldsymbol{\gamma}(0), t) L(\boldsymbol{\gamma}(0), \dot{\boldsymbol{\gamma}}(0), t) \, \mathrm{d}\Omega + \int \frac{\boldsymbol{E}^2 - \boldsymbol{B}_{\mathrm{tot}}^2}{8\pi} \, \mathrm{d}V \, \mathrm{d}t \,, \tag{2.63}$$

where we used the volume element of the extended phase space $\mathrm{d}\Omega = \mathrm{d}W \, \mathrm{d}k \, \mathrm{d}V \, \mathrm{d}t$, the volume element of velocity space $\mathrm{d}W = 2\pi \mathcal{J}_{\mathrm{gy}} / m_\sigma^3 \, \mathrm{d}v_\parallel \, \mathrm{d}\mu$, the gyrocentre Jacobian $\mathcal{J}_{\mathrm{gy}} = \mathcal{J}_{\mathrm{gc}} + m_\sigma^2 G[\boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{A}_1]$, the volume element of real space $\mathrm{d}V$, the electric field $\boldsymbol{E} = -\boldsymbol{\nabla} \phi_1$ and the magnetic field $\boldsymbol{B}_{\mathrm{tot}} = \boldsymbol{\nabla} \times (\boldsymbol{A} + \boldsymbol{A}_1)$. In the following, we use cylindrical coordinates (R, φ, Z) in real space such that $\mathrm{d}V = R \, \mathrm{d}R \, \mathrm{d}\varphi \, \mathrm{d}Z$. The gyrokinetic action functional is defined as the time integral of the total gyrokinetic Lagrangian

$$I(\boldsymbol{\gamma}, \phi_1, \boldsymbol{A}_1, \lambda) := \int_0^\lambda \mathcal{L}(\boldsymbol{\gamma}(\tilde{\lambda}), \dot{\boldsymbol{\gamma}}(\tilde{\lambda}), \phi_1(\boldsymbol{\gamma}(\tilde{\lambda}), \tilde{\lambda}), \boldsymbol{A}_1(\boldsymbol{\gamma}(\tilde{\lambda}), \tilde{\lambda}), \tilde{\lambda}) \, \mathrm{d}\tilde{\lambda} \,. \tag{2.64}$$

The equations of motion for the particle in gyrocentre coordinates as well as the gyrokinetic field equations can be derived from the action functional with variational calculus [106, 88, 30]. To perform the calculation, we define the functional derivative. Let $I : V \to \mathbb{R}$ be a functional defined on a suitable vector space V. The functional derivative $\delta I : V \to \mathbb{R}$ for $\zeta \in V$ is the functional defined by

$$\delta I(\psi; \zeta) := \lim_{\epsilon \to 0} \frac{I(\psi + \epsilon \zeta) - I(\psi)}{\epsilon} \,. \tag{2.65}$$

By requiring $\delta I(\phi_1; \delta\phi_1) = 0$ the gyrokinetic Poisson equation is derived, by requiring $\delta I(\boldsymbol{A}_1; \delta \boldsymbol{A}_1) = 0$ the gyrokinetic Ampère's law is derived and by requiring $\delta I(\boldsymbol{\gamma}; \delta\boldsymbol{\gamma}) = 0$ the Euler-Lagrange equations for the particle Lagrangian

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial L}{\partial \dot{\boldsymbol{\gamma}}_i} = \frac{\partial L}{\partial \boldsymbol{\gamma}_i} \tag{2.66}$$

are derived, which are equivalent to Hamilton's equation of motion presented in Eqs. (2.52), (2.53), (2.54) and (2.55). For simplicity, we perform the variation with respect to ϕ_1 and $\boldsymbol{A}_1$ after we have simplified the model. For a derivation of the field equations of the full system, represented by Eq. (2.64), we refer to [30, 88].

2.3.4. Simplification of the gyrokinetic Vlasov-Maxwell system

In order to focus on the development of a gyrokinetic turbulence code targeted for the edge and SOL, it is beneficial to further simplify the gyrokinetic Vlasov-Maxwell system. The physics in the edge and SOL is not only described by the gyrokinetic interaction of the plasma with the electromagnetic fields, but also by collisions and neutral physics, which are not covered by gyrokinetic theory. If needed, the approximations made can be lifted step by step in the future. We think that it is very important to understand the nature and origin of the approximations made. Therefore, we explain and justify the simplifications in this section in detail. In order to satisfy precise conservation laws with self-consistent equations, we simplify the system such that the Hamiltonian structure of the equations is retained. We start by neglecting all orthogonal fluctuations of the electromagnetic vector potential $\boldsymbol{A}_1$, i.e.

$$\boldsymbol{A}_{1\perp} = 0\,. \tag{2.67}$$

Orthogonal fluctuations in the electromagnetic vector potential give rise to parallel fluctuations in the magnetic field. They have shown to be important in the simulations of kinetic ballooning modes in the core of fusion devices [108]. We focus on the implementation of parallel electromagnetic fluctuations that already model finite β effects and decrease the frequency of kinetic shear Alfv́en waves, which has a positive effect on the timestep of the simulation and is discussed further in Section 6.4. With this simplification the electromagnetic vector potential is solely given by its parallel component

$$\boldsymbol{A}_1 = A_{1\parallel}\boldsymbol{b}\,. \tag{2.68}$$

We neglect the $\mathcal{O}(\epsilon_B)$ term $\boldsymbol{R}^*$. Further, we employ a long wavelength limit $k_\perp \rho \ll 1$ [109] simplifying the gyrocentre displacement

$$g(\boldsymbol{R}+\boldsymbol{\rho}) \approx g(\boldsymbol{R}) + \boldsymbol{\rho}\cdot(\boldsymbol{\nabla}_\perp g)(\boldsymbol{R})\,. \tag{2.69}$$

With this simplification gyroaverages of functions of the form $G[g(\boldsymbol{R}+\boldsymbol{\rho})]$ reduce to the function g itself. Terms of the form $G[\widetilde{g}]$ vanish. With the long wavelength approximation and the vanishing of $\boldsymbol{A}_\perp$, the second order part of the Hamiltonian, H_2, is greatly simplified and reads [109]

$$\begin{aligned} H_2 &\approx -\frac{q_\sigma^2}{2B(\boldsymbol{R})}\frac{\partial G\left[\widetilde{\psi_1}^2\right]}{\partial \mu} \\ &\approx -\frac{m_\sigma c^2}{2B^2}|\boldsymbol{\nabla}_\perp \phi_1|^2. \end{aligned} \tag{2.70}$$

The long wavelength limit is often referred to as the drift-kinetic limit of the gyrokinetic equations [57, 59, 60]. Although most gyroaverage operations vanish, the second order, finite-Larmor radius effect in Eq. (2.70) is retained. It gives rise to the polarisation that is used to calculate the electrostatic potential ϕ_1 in the gyrokinetic Poisson equation. The

simplified Hamiltonian reads $H = H_0 + H_1 + H_2$ with

$$H_0 = \frac{1}{2} m_\sigma v_\parallel^2 + \mu B - k \tag{2.71}$$

$$H_1 = q_\sigma \phi_1 \tag{2.72}$$

$$H_2 = -\frac{m_\sigma c^2}{2B^2} |\boldsymbol{\nabla}_\perp \phi_1|^2 \tag{2.73}$$

and the simplified Poisson bracket in gyrocentre coordinates is given by

$$\begin{aligned} \{f, g\} = \frac{q_\sigma}{m_\sigma c} \left(\frac{\partial f}{\partial \theta} \frac{\partial g}{\partial \mu} - \frac{\partial f}{\partial \mu} \frac{\partial g}{\partial \theta} \right) + \frac{\boldsymbol{B}^*}{m_\sigma B_\parallel^*} \cdot \left(\boldsymbol{\nabla} f \frac{\partial g}{\partial v_\parallel} - \frac{\partial f}{\partial v_\parallel} \boldsymbol{\nabla} g \right) \\ - \frac{q_\sigma}{m_\sigma c} \frac{\partial A_{1\parallel}}{\partial t} \left(\frac{\partial f}{\partial k} \frac{\partial g}{\partial v_\parallel} - \frac{\partial f}{\partial v_\parallel} \frac{\partial g}{\partial k} \right) \\ - \frac{c \boldsymbol{b}}{q_\sigma B_\parallel^*} \cdot (\boldsymbol{\nabla} f \times \boldsymbol{\nabla} g) + \frac{\partial f}{\partial t} \frac{\partial g}{\partial k} - \frac{\partial f}{\partial k} \frac{\partial g}{\partial t} \,, \end{aligned} \tag{2.74}$$

with the guiding-centre magnetic field

$$\boldsymbol{B}^* = \boldsymbol{B} + \frac{m_\sigma c}{q_\sigma} v_\parallel \boldsymbol{\nabla} \times \boldsymbol{b} + \boldsymbol{\nabla} \times \left(A_{1\parallel} \boldsymbol{b} \right) . \tag{2.75}$$

In this form $\boldsymbol{B}^*$ depends on $A_{1\parallel}$. Next, we simplify the free field part of the Lagrangian. We employ the quasi-neutrality approximation and neglect the $\boldsymbol{E}^2$ term in Eq. (2.63) [106]. We linearise the polarisation by assuming that H_2 acts on the initial distribution $\mathcal{F}_{0\sigma}$ [110, 106]. The gyrokinetic Lagrangian with the described simplifications is given by

$$\begin{aligned} \mathcal{L} = \sum_\sigma \int \Bigg(\Big(\frac{q_\sigma}{c} \left(\boldsymbol{A} + A_{1\parallel} \boldsymbol{b} \right) + m_\sigma v_\parallel \boldsymbol{b} \Big) \cdot \dot{\boldsymbol{R}} \\ + \frac{m_\sigma c}{q_\sigma} \mu \dot{\theta} - k \dot{t} - \frac{1}{2} m_\sigma v_\parallel^2 - \mu B - q_\sigma \phi_1 + k \Bigg) \mathcal{F}_\sigma \, \mathrm{d}\Omega \\ + \int \left(\sum_\sigma \frac{m_\sigma c^2}{2B^2} |\boldsymbol{\nabla}_\perp \phi_1|^2 \mathcal{F}_{0\sigma} - \frac{1}{8\pi} |\boldsymbol{\nabla} \times \left(A_{1\parallel} \boldsymbol{b} \right)|^2 \right) \mathrm{d}V \, \mathrm{d}t \,. \end{aligned} \tag{2.76}$$

Hamilton's equations of motion read

$$\dot{\boldsymbol{R}} = v_\parallel \frac{\boldsymbol{B}^*}{B_\parallel^*} + \frac{c}{q_\sigma B_\parallel^*} \boldsymbol{b} \times (\mu \boldsymbol{\nabla} B + q_\sigma \boldsymbol{\nabla} \phi_1) \tag{2.77}$$

$$\dot{v}_\parallel = -\frac{\boldsymbol{B}^*}{m_\sigma B_\parallel^*} \cdot (\mu \boldsymbol{\nabla} B + q_\sigma \boldsymbol{\nabla} \phi_1) - \frac{q_\sigma}{m_\sigma c} \frac{\partial A_{1\parallel}}{\partial t} \,, \tag{2.78}$$

where we have omitted the second order term, generated by H_2, to ensure energetic consistency due to the linearisation of the quasi-neutrality equation [110]. Finally, we assume that

the magnetic field created by $A_{1\parallel}$ is orthogonal to $\boldsymbol{b}$ by simplifying

$$\begin{aligned}\boldsymbol{\nabla} \times \left(A_{1\parallel}\boldsymbol{b}\right) &= \boldsymbol{\nabla} A_{\parallel} \times \boldsymbol{b} + A_{1\parallel}\boldsymbol{\nabla} \times \boldsymbol{b} \\ &\approx \boldsymbol{\nabla} A_{1\parallel} \times \boldsymbol{b}\,. \end{aligned} \tag{2.79}$$

This approximation cannot be made on the Lagrangian entirely. The field part of the Lagrangian can be simplified $1/(8\pi)|\boldsymbol{\nabla} \times \left(A_{1\parallel}\boldsymbol{b}\right)|^2 \approx 1/(8\pi)|\boldsymbol{\nabla}_\perp A_{1\parallel}|^2$. Nevertheless, the modification of the symplectic part is not straightforward. Therefore, we perform the simplification on the Poisson bracket of the system, where $\boldsymbol{B}^*$, presented in Eq. (2.75), is modified to

$$\boldsymbol{B}^* \approx \boldsymbol{B} + m_\sigma v_\parallel \frac{c}{q_\sigma}\boldsymbol{\nabla} \times \boldsymbol{b} + \boldsymbol{\nabla} A_{1\parallel} \times \boldsymbol{b}\,. \tag{2.80}$$

This simplification removes the $A_{1\parallel}$ dependence of $B^*_\parallel$ because $\boldsymbol{b} \cdot \left(\boldsymbol{\nabla} A_{1\parallel} \times \boldsymbol{b}\right) = 0$. Consequently, $J_{\mathrm{gy}} = J_{\mathrm{gc}}$ and the velocity space volume element, $\mathrm{d}W$, is independent of $A_{1\parallel}$, which decouples the gyrokinetic field equations.

2.3.5. The gyrokinetic field equations

The quasi-neutrality equation is derived by requiring that the variation of the action functional with respect to ϕ_1 vanishes. The variation reads

$$\begin{aligned}\delta I(\phi_1; \delta\phi_1) &= -\sum_\sigma q_\sigma \int \mathcal{F}_\sigma \delta\phi_1 \,\mathrm{d}\Omega + \sum_\sigma \int \frac{m_\sigma c^2}{B^2}\mathcal{F}_{0\sigma}\boldsymbol{\nabla}_\perp\phi_1 \cdot \boldsymbol{\nabla}_\perp\delta\phi_1 \frac{2\pi}{m_\sigma} B^*_\parallel \,\mathrm{d}v_\parallel \,\mathrm{d}\mu \,\mathrm{d}k \,\mathrm{d}V \,\mathrm{d}t \\ &= 0 \,\forall \delta\phi_1\,. \end{aligned} \tag{2.81}$$

This is also known as the weak formulation of the quasi-neutrality equation and can be used directly in a finite element discretisation scheme [111]. We are interested in a finite difference discretisation and derive a strong formulation of the quasi-neutrality equation. We use the product rule to modify the integrand

$$\frac{\mathcal{F}_{0\sigma}}{B^2}B^*_\parallel \boldsymbol{\nabla}_\perp\phi_1 \cdot \boldsymbol{\nabla}_\perp\delta\phi_1 = \boldsymbol{\nabla} \cdot \left(\delta\phi_1 \frac{\mathcal{F}_{0\sigma}}{B^2}B^*_\parallel \boldsymbol{\nabla}_\perp\phi_1\right) - \boldsymbol{\nabla} \cdot \left(\frac{\mathcal{F}_{0\sigma}}{B^2}B^*_\parallel \boldsymbol{\nabla}_\perp\phi_1\right)\delta\phi_1 \tag{2.82}$$

and apply the divergence integral theorem, where we assume that the boundary integral vanishes

$$-\sum_\sigma q_\sigma \int \mathcal{F}_\sigma \delta\phi_1 \,\mathrm{d}\Omega - \sum_\sigma \int \frac{1}{B^*_\parallel}\boldsymbol{\nabla} \cdot \left(\frac{m_\sigma c^2}{B^2}\mathcal{F}_{0\sigma}B^*_\parallel \boldsymbol{\nabla}_\perp\phi_1\right)\delta\phi_1 \,\mathrm{d}\Omega = 0 \,\forall \delta\phi_1\,. \tag{2.83}$$

By applying the fundamental lemma of variational calculus, we obtain the strong form of the quasi-neutrality equation

$$-\boldsymbol{\nabla} \cdot \left(\sum_\sigma \frac{m_\sigma c^2}{B^2} n_{0\sigma}\boldsymbol{\nabla}_\perp\phi_1\right) = \sum_\sigma q_\sigma \int f_\sigma \,\mathrm{d}W\,, \tag{2.84}$$

with the initial density $n_{0\sigma} = \int f_{0\sigma}\,\mathrm{d}W$. Similar to the quasi-neutrality equation, we derive Ampère's law by requiring that the variation of the action functional with respect to $A_{1\parallel}$ vanishes

$$\delta I(A_{1\parallel};\delta A_{1\parallel}) = \sum_\sigma \frac{q_\sigma}{c}\int v_\parallel \mathcal{F}_\sigma \delta A_{1\parallel}\,\mathrm{d}\Omega - \frac{1}{4\pi}\int \boldsymbol{\nabla}_\perp A_{1\parallel}\cdot\boldsymbol{\nabla}_\perp \delta A_{1\parallel}\,\mathrm{d}V\,\mathrm{d}t = 0\,\forall \delta A_\parallel\,. \tag{2.85}$$

After applying the divergence theorem and requiring that boundary terms vanish, we obtain Ampère's law

$$-\Delta_\perp A_{1\parallel} = 4\pi\sum_\sigma \frac{q_\sigma}{c}\int f_\sigma v_\parallel\,\mathrm{d}W\,, \tag{2.86}$$

where we have used the definition $\Delta_\perp := \boldsymbol{\nabla}\cdot\boldsymbol{\nabla}_\perp$. In the symplectic formulation of the gyrokinetic Vlasov equation, presented above, explicit time derivatives of $A_{1\parallel}$ appear. To allow for a straightforward temporal discretisation in Section 4.2, we derive a generalised Ohm's law by taking the time derivative of Eq. (2.86) and solving for $\partial A_{1\parallel}/\partial t$ directly [98, 96, 97]

$$-\Delta_\perp \frac{\partial A_{1\parallel}}{\partial t} = 4\pi\sum_\sigma \frac{q_\sigma}{c}\int \frac{\partial f_\sigma}{\partial t} v_\parallel\,\mathrm{d}W\,. \tag{2.87}$$

We write the Vlasov equation as

$$\frac{\partial f_\sigma}{\partial t} = \left(\frac{\partial f_\sigma}{\partial t}\right)^\star + \frac{q_\sigma}{m_\sigma c}\frac{\partial f_\sigma}{\partial v_\parallel}\frac{\partial A_{1\parallel}}{\partial t}\,, \tag{2.88}$$

where $(\partial f_\sigma/\partial t)^\star$ denotes the part excluding explicit time derivatives. Inserting Eq. (2.88) into Eq. (2.87) yields the generalised Ohm's law

$$-\left(\Delta_\perp + 4\pi\sum_\sigma \frac{q_\sigma^2}{m_\sigma c^2}\int \frac{\partial f_\sigma}{\partial v_\parallel} v_\parallel\,\mathrm{d}W\right)\frac{\partial A_{1\parallel}}{\partial t} = 4\pi\sum_\sigma \frac{q_\sigma}{c}\int \left(\frac{\partial f_\sigma}{\partial t}\right)^\star v_\parallel\,\mathrm{d}W\,. \tag{2.89}$$

Eqs. (2.61), (2.77), (2.78), (2.80), (2.84), (2.86) and (2.89) describe the system implemented in the `GENE-X` code.

2.3.6. Conservation laws

The gyrokinetic Vlasov-Maxwell system, presented in the last section, fulfils multiple conservation laws. On the one hand, conservation laws are important to ensure numerical stability, on the other hand, they can serve as a method of verification of the numerical scheme. In this section, we discuss the conservation of the particle number, momentum and energy of the gyrokinetic system. As we performed the approximation in Eq. (2.75) on the Poisson bracket and not on the Lagrangian of the system, we proof the conservation by direct calculation. We start with the energy

$$E = E_H - E_\phi + E_A\,, \tag{2.90}$$

with the contributions

$$E_H = \sum_\sigma \int H_\sigma f_\sigma \,\mathrm{d}V\,\mathrm{d}W = \sum_\sigma \int \left(\frac{1}{2} m_\sigma v_\parallel^2 + \mu B + q_\sigma \phi\right) f_\sigma \,\mathrm{d}V\,\mathrm{d}W \tag{2.91}$$

$$E_A = \int \frac{1}{8\pi} |\boldsymbol{\nabla}_\perp A_{1\parallel}|^2 \mathrm{d}V \tag{2.92}$$

$$E_\phi = \int \frac{m_\sigma c^2}{2B^2} n_{0\sigma} |\boldsymbol{\nabla}_\perp \phi_1|^2 \mathrm{d}V\,. \tag{2.93}$$

We assume that the domain is periodic and boundary terms vanish. The time derivative of the Hamiltonian part of the energy reads

$$\begin{aligned}
\frac{\partial E_H}{\partial t} &= \sum_\sigma \int \left(\frac{\partial H_\sigma}{\partial t} f_\sigma + H_\sigma \frac{\partial f_\sigma}{\partial t}\right) \mathrm{d}V\,\mathrm{d}W \\
&= \sum_\sigma \int \left[q_\sigma \frac{\partial \phi_1}{\partial t} f_\sigma + H_\sigma \left(\{f_\sigma, H_\sigma\}^\star + \frac{q_\sigma}{m_\sigma c} \frac{\partial A_\parallel}{\partial t} \frac{\partial f_\sigma}{\partial v_\parallel}\right)\right] \mathrm{d}V\,\mathrm{d}W \\
&= \sum_\sigma \int \left[q_\sigma \frac{\partial \phi_1}{\partial t} f_\sigma - \frac{q_\sigma}{c} \frac{\partial A_\parallel}{\partial t} v_\parallel f_\sigma\right] \mathrm{d}V\,\mathrm{d}W\,,
\end{aligned} \tag{2.94}$$

where we used the integral property of the Poisson bracket

$$\int H_\sigma \{f_\sigma, H_\sigma\}^\star \,\mathrm{d}V\,\mathrm{d}W = 0\,, \tag{2.95}$$

proved in Appendix A.4, and partial integration in $v_\parallel$. The Poisson bracket $\{\,,\}^\star$ is given by Eq. (2.74) without the $\partial A_{1\parallel}/\partial t$ term. We continue with the energy contribution of the electromagnetic vector potential. The rate of change of the electromagnetic energy is simplified with Ampère's law to

$$\begin{aligned}
\frac{\partial E_A}{\partial t} &= \int \frac{1}{8\pi} \frac{\partial}{\partial t} |\boldsymbol{\nabla}_\perp A_\parallel|^2 \mathrm{d}V = \int \frac{1}{4\pi} \boldsymbol{\nabla}_\perp A_\parallel \cdot \boldsymbol{\nabla}_\perp \frac{\partial A_\parallel}{\partial t} \,\mathrm{d}V \\
&= -\int \frac{1}{4\pi} \Delta_\perp A_\parallel \frac{\partial A_\parallel}{\partial t} \,\mathrm{d}V = \sum_\sigma \frac{q_\sigma}{c} \int \frac{\partial A_\parallel}{\partial t} v_\parallel f_\sigma \,\mathrm{d}V\,\mathrm{d}W\,,
\end{aligned} \tag{2.96}$$

and the rate of change of the electrostatic energy is simplified with the quasi-neutrality equation to

$$\begin{aligned}
\frac{\partial E_\phi}{\partial t} &= \sum_\sigma \int \frac{m_\sigma c^2}{2B^2} n_{0\sigma} \frac{\partial}{\partial t} |\boldsymbol{\nabla}_\perp \phi_1|^2 \mathrm{d}V = \sum_\sigma \int \frac{m_\sigma c^2}{B^2} n_{0\sigma} \boldsymbol{\nabla}_\perp \phi_1 \cdot \boldsymbol{\nabla}_\perp \frac{\partial}{\partial t} \phi_1 \,\mathrm{d}V \\
&= -\sum_\sigma \int \boldsymbol{\nabla} \cdot \left(\frac{m_\sigma c^2}{B^2} n_{0\sigma} \boldsymbol{\nabla}_\perp\right) \frac{\partial \phi_1}{\partial t} \,\mathrm{d}V = \sum_\sigma \int q_\sigma \frac{\partial \phi_1}{\partial t} f_\sigma \,\mathrm{d}V\,\mathrm{d}W\,.
\end{aligned} \tag{2.97}$$

Combining Eqs. (2.94), (2.96) and (2.97) yields

$$\frac{\partial E}{\partial t} = \frac{\partial E_H}{\partial t} - \frac{\partial E_\phi}{\partial t} + \frac{\partial E_A}{\partial t} = 0. \tag{2.98}$$

The expression for the energy is further simplified by using the weak form of the quasi-neutrality equation, presented in Eq. (2.81), with $\delta\phi_1 = \phi_1$

$$E = \sum_\sigma \int \left(\frac{1}{2} m_\sigma v_\parallel^2 + \mu B + \frac{1}{2} q_\sigma \phi_1 \right) f_\sigma \, \mathrm{d}V \, \mathrm{d}W + \int \frac{1}{8\pi} |\boldsymbol{\nabla}_\perp A_{1\parallel}|^2 \mathrm{d}V \,. \tag{2.99}$$

Liouville's theorem states that the characteristics of the system satisfy the conservation law [30, 42]

$$\boldsymbol{\nabla} \cdot \left(\mathcal{J}_{\mathrm{gy}} \dot{\boldsymbol{R}} \right) + \frac{\partial}{\partial v_\parallel} \left(\mathcal{J}_{\mathrm{gy}} \dot{v}_\parallel \right) = 0 \,. \tag{2.100}$$

This property can be used to write the gyrokinetic Vlasov equation in the form of a conservation law

$$\frac{\partial \left(\mathcal{J}_{\mathrm{gy}} f_\sigma \right)}{\partial t} + \boldsymbol{\nabla} \cdot \left(\mathcal{J}_{\mathrm{gy}} f_\sigma \dot{\boldsymbol{R}} \right) + \frac{\partial}{\partial v_\parallel} \left(\mathcal{J}_{\mathrm{gy}} f_\sigma \dot{v}_\parallel \right) = 0 \,. \tag{2.101}$$

Integrating Eq. (2.101) over phase space and assuming that boundary terms vanish yields the conservation of the number of particles

$$n_\sigma = \int f_\sigma \, \mathrm{d}V \, \mathrm{d}W \,. \tag{2.102}$$

Finally, we discuss the conservation of canonical toroidal momentum. For a complete discussion of momentum conservation of the gyrokinetic Vlasov-Maxwell system we refer to [110, 112]. The canonical toroidal momentum of a particle is defined as

$$p_\varphi := \frac{\partial L}{\partial \dot{\varphi}} \,, \tag{2.103}$$

where φ denotes the toroidal angle in a cylindrical coordinate system. We use the canonical toroidal momentum in Section 4.5 to construct an initial condition for the simulations. The canonical toroidal momentum of a particle is conserved under the assumption of an axisymmetric equilibrium $\boldsymbol{B} = \boldsymbol{B}(R, Z)$ and a time independent electromagnetic vectorpotential $\boldsymbol{A}_{1\parallel}$. In the following, we assume that $\boldsymbol{A}_{1\parallel} = 0$. The conservation of the canonical toroidal momentum follows directly from the Euler-Lagrange equations and the independence of the particle Lagrangian L of φ, i.e.

$$\frac{\partial L}{\partial \varphi} = 0 \,. \tag{2.104}$$

Using $\boldsymbol{R} = R\boldsymbol{e}_R + Z\boldsymbol{e}_Z$ and $\dot{\boldsymbol{R}} = \dot{R}\boldsymbol{e}_R + R\dot{\varphi}\boldsymbol{e}_\varphi + \dot{Z}\boldsymbol{e}_Z$, the canonical toroidal momentum reads

$$\begin{aligned} p_\varphi &= \left(\frac{q_\sigma}{c} \boldsymbol{A} + m_\sigma v_\parallel \boldsymbol{b} \right) R \boldsymbol{e}_\varphi \\ &= \frac{q_\sigma R}{c} A_\varphi + m_\sigma v_\parallel b_\varphi R \,. \end{aligned} \tag{2.105}$$

We simplify the expression by using the poloidal flux function defined as[†]

$$\psi_p := \frac{1}{2\pi} \int\limits_{D_R(0)} \boldsymbol{B} \cdot \mathrm{d}\boldsymbol{F} = \frac{1}{2\pi} \int\limits_{D_R(0)} \boldsymbol{\nabla} \times \boldsymbol{A} \cdot \mathrm{d}\boldsymbol{F} = \frac{1}{2\pi} \int\limits_{\partial D_R(0)} \boldsymbol{A} \cdot \mathrm{d}\boldsymbol{l} = R A_\varphi \,, \tag{2.106}$$

where the axisymmetry of the electromagnetic vector potential and Stokes theorem were used. The set $D_R(0)$ denotes a disk of radius R with the origin at the magnetic axis in the R, Z plane. With the poloidal flux function, the canonical toroidal momentum simplifies to

$$p_\varphi = \frac{q_\sigma}{c} \psi_p + m_\sigma v_\parallel b_\varphi R \,. \tag{2.107}$$

The conserved canonical toroidal momentum of the total gyrokinetic system reads [112]

$$P_\varphi := \sum_\sigma \int \left(\frac{q_\sigma}{c} \psi_p + m_\sigma v_\parallel b_\varphi R \right) f_\sigma \, \mathrm{d}V \, \mathrm{d}W \,. \tag{2.108}$$

2.4. Conclusion and outlook

In this chapter, we presented an overview of gyrokinetic theory and introduced the gyrokinetic model implemented in the `GENE-X` code. The model presented is consistent, evolves the full distribution function and fulfils multiple exact conservation laws. We put a special emphasis on a precise presentation of the gyrokinetic theory and a rigorous simplification from the full second order theory to the model implemented. This makes the generalisation of the model straightforward in the future.

The model implemented uses a long wavelength limit, a linearisation of the polarisation density and neglects perpendicular electromagnetic fluctuations.

For the application of `GENE-X` to the edge and SOL, the linearisation of the polarisation density should be lifted first. Next to a nonlinear quasi-neutrality equation, this generates additional terms in the gyrokinetic Vlasov equation as the second order part of the Hamiltonian H_2, generating the polarisation, acts on the full distribution function f_σ and not the linearisation $f_{0\sigma}$ [110, 106]. Furthermore, the implementation of gyroaverages and, consequently, further second order terms would be interesting.

Next to the gyrokinetic plasma model, other physical effects like collisions and neutral physics are important in the edge and SOL. In Chapter 7, we present first simulations that use a simplified Bhatnagar-Gross-Krook (BGK) collision operator [113]. The implementation of more sophisticated collision operators, like a full-f Lennard-Bernstein/Dogherty operator [114, 115], is work in progress [116]. To study the effect of neutrals on gyrokinetic edge and SOL turbulence, implementing the simple fluid neutral model used in the `GRILLIX` code [80] is a good starting point for future developments.

[†]Note that the poloidal flux function defined here contains a factor of $1/(2\pi)$, as it appears in the Grad-Shafranov equation. It is also common to define the poloidal flux without the factor of $1/(2\pi)$, e.g. in [8].

3. Coordinate systems

Due to the strong magnetic field in fusion devices, structures in the plasma are homogeneous along and heterogeneous perpendicular to magnetic field lines. This property is also referred to as the flute-like character of the plasma and produces small gradients parallel to magnetic field lines. It can be exploited in turbulence simulations by choosing a coordinate system in which parallel gradients resolve to a partial derivative in one of the coordinate variables. This enables the use of a reduced resolution in the corresponding direction, sufficient to resolve the small gradients, and spares computational resources.

For predictive edge and SOL simulations it is further important that the chosen coordinate system is well-defined in realistic magnetic geometries, including the separatrix and one or even multiple X-points.

In this chapter, we discuss coordinate systems suitable for turbulence simulations in magnetic confinement fusion devices[†]. We discuss the advantages and disadvantages of established flux-coordinate systems and introduce the locally field-aligned coordinate system based on the flux-coordinate independent approach [76], implemented in the `GENE-X` code. This coordinate system is free of coordinate singularities while still being aligned to the magnetic field. It can handle general fusion device geometries including multiple X-points and stochastic regions.

3.1. Flux coordinate systems

In the gyrokinetic community so-called flux-coordinate systems are common. In these magnetic field lines are straight and one coordinate, we call it ζ, can be chosen such that the desired property

$$\boldsymbol{b} \cdot \boldsymbol{\nabla} f \propto \frac{\partial f}{\partial \zeta} \tag{3.1}$$

is fulfilled. We start the discussion of flux-coordinate systems from the cylindrical-toroidal coordinates introduced in Chapter 2. For simplicity, we assume that the magnetic field is axisymmetric and given as a function of (r, θ, φ). The first coordinate ψ of the flux-coordinate system is a label for the flux surfaces. It can for example be realised with the toroidal flux function, the poloidal flux function, the volume of a flux surface or the pressure on a flux surface. [73]. Consider a magnetic field line on a given flux surface. The rotational transform angle ι of the magnetic field line is defined as the poloidal angle θ travelled by a field line

[†]The coordinate transformations presented in this chapter affect the gyrocentre position $\boldsymbol{R}$ and not the velocity space.

after one poloidal transit [73]. It can be expressed as [73]

$$\iota := \frac{\iota}{2\pi} := \frac{\mathrm{d}\psi_p}{\mathrm{d}\psi_t}\,, \tag{3.2}$$

where ψ_p represents the poloidal flux function, defined in Eq. (2.106), and

$$\psi_t := \frac{1}{2\pi}\int_{A_\mathrm{t}} \boldsymbol{B}\cdot\mathrm{d}\boldsymbol{F} \tag{3.3}$$

the toroidal flux function. The set A_t is given by the part of a $\varphi = \text{const}$ surface that lies within the magnetic flux surface under consideration [73]. The q factor is defined as the reciprocal of ι, i.e.

$$q := \frac{1}{\iota}\,. \tag{3.4}$$

The two remaining coordinates of the flux-coordinate system are chosen such that magnetic field lines are described by straight lines on a flux surface. Such a coordinate system can be constructed by replacing either the toroidal angle or the poloidal angle with a new flux-coordinate, or both at the same time [73]

$$\varphi \to \varphi_f \tag{3.5}$$

$$\theta \to \theta_f\,. \tag{3.6}$$

The choice of φ_f and θ_f is not unique and many different flux coordinate systems can be constructed. Often used are for example the coordinates implemented in the `PEST` code [117] and the coordinate systems described by Hamada [118] and Boozer [119]. For the further discussion, we assume generic flux angles θ_f and φ_f. The flux-coordinate system can be modified such that the magnetic field is represented as a Clebsch form

$$\boldsymbol{B} = \boldsymbol{\nabla}\psi \times \boldsymbol{\nabla}\nu\,, \tag{3.7}$$

with the new coordinate [73]

$$\nu = \frac{1}{2\pi}\frac{\partial\psi_t}{\partial\psi}\theta_f - \frac{1}{2\pi}\frac{\partial\psi_p}{\partial\psi}\varphi_f\,. \tag{3.8}$$

In the Clebsch form ψ and ν identify a magnetic field line on a magnetic flux surface. The position on a magnetic field line is then provided by the additional coordinate ζ. As $\boldsymbol{\nabla}\psi\cdot\boldsymbol{b} = 0$ and $\boldsymbol{\nabla}\nu\cdot\boldsymbol{b} = 0$, the magnetic field vector $\boldsymbol{b}$ is proportional to $\boldsymbol{\nabla}\zeta$ and parallel derivatives resolve to partial derivatives in ζ. In tokamaks it is common to use the poloidal flux function as a flux surface label such that [73]

$$\psi = \frac{\psi_p}{2\pi} \tag{3.9}$$

$$\nu = q\theta_f - \varphi_f \tag{3.10}$$

$$\zeta = \theta_f\,, \tag{3.11}$$

and in stellarators it is common to use the toroidal flux function as a flux surface label such that [73]

$$\psi = \frac{\psi_t}{2\pi} \tag{3.12}$$

$$\nu = \theta_f - \iota\varphi_f \tag{3.13}$$

$$\zeta = \varphi_f \,. \tag{3.14}$$

Nevertheless, these coordinate systems have two major drawbacks making them unsuitable for the simulation of edge and SOL turbulence. We explain the first problem at the example of the coordinates given in Eqs. (3.9), (3.10) and (3.11). The term $q\theta_f$ in the coordinate ν increases with the poloidal angle θ_f. This introduces a distortion in the grid for large poloidal angles and large magnetic shear [120]. This problem can be partially solved by using the so-called shifted-metric approach [120]. The manifold $\mathcal{M}$ of the plasma, introduced in Chapter 2, is divided over the variable θ_f into a family of overlapping sectors, labelled by $\{\theta_k\}$. Within each sector the coordinate transformation is given by

$$\psi = \frac{\psi_p}{2\pi} \tag{3.15}$$

$$\nu = q(\theta_f - \theta_k) - \varphi_f \tag{3.16}$$

$$\zeta = (\theta_f - \theta_k) \,. \tag{3.17}$$

If the number of sectors is chosen equal to the number of grid points in θ_f, the grid in ψ, ν is orthogonal. Second, flux-coordinates are generally ill-defined at the separatrix. The reason is the following. In flux coordinates magnetic field lines are straight and their equation can be written as [73]

$$\theta_f = \iota\varphi_f + \theta_0 \,. \tag{3.18}$$

This implies that the flux-coordinates need to fulfil the relation

$$\frac{\mathrm{d}\theta_f}{\mathrm{d}\varphi_f} = \iota \,. \tag{3.19}$$

At the X-point the toroidal transform vanishes and the angle θ_f reduces to a constant θ_0

$$\theta_f = \theta_0 \,. \tag{3.20}$$

Hence, the coordinate transformation into flux-coordinates cannot be bijective [74] and is ill-defined. To visualise the behaviour of flux coordinates close to the separatrix, we present the lines of constant ψ and θ_f of the `PEST` coordinate system in Fig. 3.1. In summary, it is not possible to use flux coordinate systems to simulate turbulence across the separatrix.

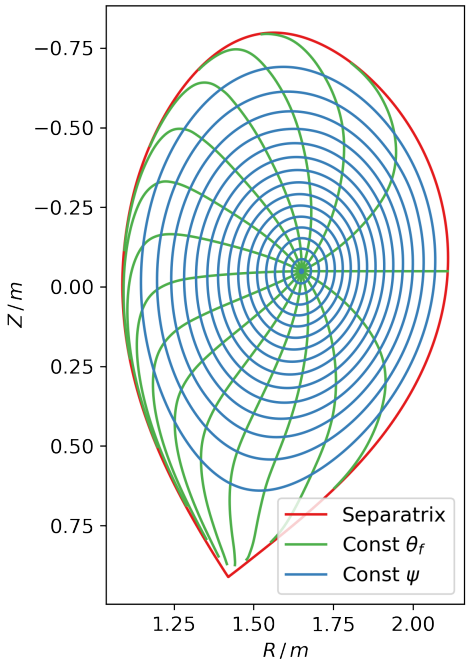

Figure 3.1.: Figure displaying the lines of constant ψ and θ_f for the `PEST` coordinate system in the magnetic geometry of AUG, used for simulations in Chapter 7. For $\psi \to 1$ and $\theta \neq 0$, the symmetry angle θ_f converges to the poloidal angle of the X-point. This demonstrates visually that the `PEST` coordinate system is ill-defined on the separatrix.

3.2. A locally field-aligned coordinate system

Including the separatrix and handling large magnetic shear is important for edge and SOL plasmas. Therefore, we introduce a locally field-aligned coordinate system following the flux-coordinate independent (FCI) approach [76] in this section. This coordinate system retains the property that parallel gradients can be expressed as a partial derivative in one of the coordinate variables while being able to handle the separatrix and large shear. The coordinate system is similar to the one used in the `GRILLIX` code [86]. We introduce the coordinate transformation and express the gyrokinetic Vlasov-Maxwell system in the new coordinates.

3.2.1. Coordinate transformation

The results presented in this subsection have been published in [87]. For the completeness of the presentation the inverse coordinate transformation, presented in Eq. (3.27), has been added.

Consider a family of poloidal planes located at the angles φ_k. We define the locally field-aligned coordinate system in a region $[\varphi_k - \Delta\varphi, \varphi_k + \Delta\varphi]$ around the poloidal plane. For linear fusion devices or calculations with a large aspect ratio assumption, the toroidal angle φ is interpreted as a Cartesian coordinate labelling poloidal planes. For every k we define a different coordinate system. Hence, the coordinate system is local to every poloidal plane.

We choose $\Delta\varphi$ large enough that the coordinate systems for different poloidal planes overlap and the collection of all these coordinate systems, or charts respectively, form an atlas of the manifold $\mathcal{M}$. In the following, we assume without loss of generality that the poloidal plane is located at $\varphi_k = 0$. It is always possible to rotate the cylindrical (R, φ, Z) coordinate system such that $\varphi_k = 0$ is fulfilled. To derive the coordinate transformation, we use a Cartesian reference frame (x_c, y_c, z_c) on the poloidal plane such that x_c points in the direction of the cylindrical R coordinate, z_c points in the direction of the cylindrical Z coordinate and y_c out of the poloidal plane. The magnetic field line is a curve $\boldsymbol{\gamma} : [0, 1] \to \mathcal{M}$ on the configuration space manifold $\mathcal{M}$ and defined by the differential equation

$$\frac{\mathrm{d}\boldsymbol{\gamma}}{\mathrm{d}\lambda} = \boldsymbol{b} \circ \boldsymbol{\gamma} \,. \tag{3.21}$$

Let $\boldsymbol{\gamma}_{\mathrm{cart}} = (\gamma_{x_c}, \gamma_{y_c}, \gamma_{z_c})$ be the curve of the magnetic field line expressed in the Cartesian reference frame. Eq. (3.21) reads

$$\frac{\mathrm{d}\gamma_{x_c}}{\mathrm{d}\lambda} = b_{x_c} \circ \boldsymbol{\gamma}_{\mathrm{cart}} \tag{3.22}$$

$$\frac{\mathrm{d}\gamma_{y_c}}{\mathrm{d}\lambda} = b_{y_c} \circ \boldsymbol{\gamma}_{\mathrm{cart}} \tag{3.23}$$

$$\frac{\mathrm{d}\gamma_{z_c}}{\mathrm{d}\lambda} = b_{z_c} \circ \boldsymbol{\gamma}_{\mathrm{cart}} \,. \tag{3.24}$$

We define the locally field-aligned coordinate system (x, y, z) by giving the coordinate transformation τ from the locally field-aligned coordinate system to the Cartesian reference frame

$$\tau : \begin{pmatrix} x \\ y \\ z \end{pmatrix} \mapsto \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \begin{pmatrix} x + \int_0^y b_{x_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(u))\,\mathrm{d}u \\ \int_0^y b_{y_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(u))\,\mathrm{d}u \\ z + \int_0^y \ b_{z_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(u))\,\mathrm{d}u \end{pmatrix} . \tag{3.25}$$

Next, we construct the inverse τ^{-1}. The function γ_{y_c} assigns the y_c coordinate of the magnetic field line to its orbit parameter λ and is invertible in a region around the poloidal plane. In this region the coordinate transformation can be inverted by substituting the orbit parameter u, in the integral, with the function $\gamma_{y_c}(u)$ and using the derivative rule of the inverse function

$$\frac{\mathrm{d}\gamma_{y_c}^{-1}}{\mathrm{d}u}(u) = \frac{1}{\frac{\mathrm{d}\gamma_{y_c}}{\mathrm{d}u}(\gamma_{y_c}^{-1}(u))} = \frac{1}{b_{y_c}(\gamma_{y_c}^{-1}(u))} \,. \tag{3.26}$$

The inverse reads

$$\tau^{-1} : \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \mapsto \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x_c - \int_0^{y_c} \frac{b_{x_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(\gamma_{y_c}^{-1}(u)))}{b_{y_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(\gamma_{y_c}^{-1}(u)))}\,\mathrm{d}u \\ \int_0^{y_c} \frac{1}{b_{y_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(\gamma_{y_c}^{-1}(u)))}\,\mathrm{d}u \\ z_c - \int_0^{y_c} \frac{b_{z_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(\gamma_{y_c}^{-1}(u)))}{b_{y_c}(\boldsymbol{\gamma}_{\mathrm{cart}}(\gamma_{y_c}^{-1}(u)))}\,\mathrm{d}u \end{pmatrix} . \tag{3.27}$$

The new coordinate system can be interpreted in a simple way. The y coordinate corresponds to the length of the magnetic field line starting at the point $(x_c, y_c = 0, z_c)$. The x and z

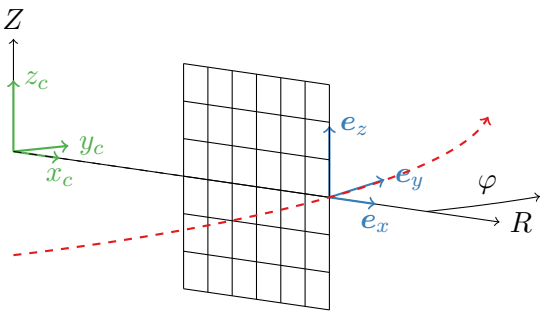

Figure 3.2.: Figure displaying the cylindrical (R, φ, Z) coordinate system, the Cartesian reference frame (x_c, y_c, z_c) and the unit vectors $(\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)$ of the locally field-aligned (x, y, z) coordinate system. The locally field-aligned coordinate system is non-orthogonal and coincides with the cartesian reference frame on the poloidal plane. This figure has been published in [87].

coordinates correspond to the x_c and y_c coordinates of the point the field line is originating from. On the poloidal plane, i.e. $y_c = \varphi = y = 0$, the locally field-aligned coordinates coincide with the Cartesian reference frame. The Jacobian matrix of the coordinate transformation evaluated at $y = 0$ reads

$$\begin{aligned} J(x,0,z) &= \left(\left(\frac{\partial \tau}{\partial x}\right)(x,0,z), \left(\frac{\partial \tau}{\partial y}\right)(x,0,z), \left(\frac{\partial \tau}{\partial z}\right)(x,0,z)\right) \\ &= \begin{pmatrix} 1 & b_{x_c} & 0 \\ 0 & b_{y_c} & 0 \\ 0 & b_{z_c} & 1 \end{pmatrix}, \end{aligned} \tag{3.28}$$

and its inverse is given by

$$\begin{aligned} J^{-1}(x_c,0,z_c) &= \left(\left(\frac{\partial \tau^{-1}}{\partial x_c}\right)(x_c,0,z_c), \left(\frac{\partial \tau^{-1}}{\partial y_c}\right)(x_c,0,z_c), \left(\frac{\partial \tau^{-1}}{\partial z_c}\right)(x_c,0,z_c)\right) \\ &= \begin{pmatrix} 1 & -\frac{b_{x_c}}{b_{y_c}} & 0 \\ 0 & \frac{1}{b_{y_c}} & 0 \\ 0 & -\frac{b_{z_c}}{b_{y_c}} & 1 \end{pmatrix}. \end{aligned} \tag{3.29}$$

For clarity, we omit the dependence of b_{x_c}, b_{y_c} and b_{z_c} on x and z. The basis of the tangent space at $y_c = 0$ is given by the columns of the Jacobian matrix

$$\boldsymbol{e}_x = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \qquad \boldsymbol{e}_y = \begin{pmatrix} b_{x_c} \\ b_{y_c} \\ b_{z_c} \end{pmatrix} \qquad \boldsymbol{e}_z = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{3.30}$$

The basis is non-orthogonal. The basis vector $\boldsymbol{e}_y$ is equal to the unit vector of the magnetic field. The coordinate system is depicted in Fig. 3.2. We continue by introducing the metric tensor and the dual basis of the coordinate system to set the stage for the derivation of the differential operators.

3.2.2. Metric coefficients

The results presented in this subsection have been published in [87]. For the completeness of the presentation the inverse metric, presented in Eq. (3.33), and a discussion about the orthogonal projection operator, presented in Eqs. (3.41)–(3.44), have been added.

The components of the metric tensor read

$$g_{ij} = J^T J \tag{3.31}$$

$$= \begin{pmatrix} 1 & b_{x_c} & 0 \\ b_{x_c} & 1 & b_{z_c} \\ 0 & b_{z_c} & 1 \end{pmatrix} , \tag{3.32}$$

the inverse reads

$$g^{ij} = \frac{1}{b_{y_c}^2} \begin{pmatrix} b_{x_c}^2 + b_{y_c}^2 & -b_{x_c} & b_{x_c} b_{z_c} \\ -b_{x_c} & 1 & -b_{z_c} \\ b_{x_c} b_{z_c} & -b_{z_c} & b_{z_c}^2 + b_{y_c}^2 \end{pmatrix} \tag{3.33}$$

and the Jacobian determinant is given by

$$\mathcal{J}_{\text{fci}} := \sqrt{\det\left(g_{ij}\right)} = b_{y_c} . \tag{3.34}$$

As gradients correspond to 1-forms, they are contained in the dual of the tangent space. The basis of the dual space can be expressed as

$$\boldsymbol{e}^i = g^{ij} \boldsymbol{e}_j , \tag{3.35}$$

where the basis vectors $\boldsymbol{e}_i$ of the tangent space, presented in Eq. (3.30), were used. Evaluating Eq. (3.35) yields

$$\boldsymbol{e}^x = \left(1, -\frac{b_{x_c}}{b_{y_c}}, 0\right) \tag{3.36}$$

$$\boldsymbol{e}^y = \left(0, \quad \frac{1}{b_{y_c}}, 0\right) \tag{3.37}$$

$$\boldsymbol{e}^z = \left(0, -\frac{b_{z_c}}{b_{y_c}}, 1\right) . \tag{3.38}$$

We derive the basis expansion of the covariant $\boldsymbol{b}$ vector

$$\boldsymbol{b} = b_i \boldsymbol{e}^i .$$

Remembering that $\boldsymbol{b} = \boldsymbol{e}_y$, due to the alignment of the y coordinate line with the magnetic field line, and using the relation between reciprocal sets of basis vectors $\boldsymbol{e}_i = g_{ij} \boldsymbol{e}^j$, we find that

$$\boldsymbol{b} = \boldsymbol{e}_y = g_{2j} \boldsymbol{e}^j = b_{x_c} \boldsymbol{e}^x + \boldsymbol{e}^y + b_{z_c} \boldsymbol{e}^z . \tag{3.39}$$

This expression provides us with covariant components of the magnetic unit vector in the chosen coordinate system

$$(b_i) = (b_{x_c}, 1, b_{z_c})\,. \tag{3.40}$$

Finally, we derive two new basis vectors $\boldsymbol{e}_{1\perp}$ and $\boldsymbol{e}_{2\perp}$ that form an orthogonal basis system $(\boldsymbol{e}_y, \boldsymbol{e}_{1\perp}, \boldsymbol{e}_{2\perp})$ with the magnetic field unit vector $\boldsymbol{e}_y$. By using the Gram-Schmidt process on the basis vectors $(\boldsymbol{e}_y, \boldsymbol{e}_z, \boldsymbol{e}_x)$ and rescaling the third output vector of the Gram-Schmidt process, we obtain

$$\boldsymbol{e}_{1\perp} = \begin{pmatrix} -b_{x_c} b_{z_c} \\ -b_{z_c} b_{y_c} \\ 1 - b_{z_c}^2 \end{pmatrix} \qquad \boldsymbol{e}_{2\perp} = \begin{pmatrix} 1 \\ -\frac{b_{x_c}}{b_{y_c}} \\ 0 \end{pmatrix}\,. \tag{3.41}$$

We use Eq. (3.41) to construct a projection operator into the two-dimensional plane orthogonal to $\boldsymbol{e}_y$. Given a vector $\boldsymbol{v} = v^x \boldsymbol{e}_x + v^y \boldsymbol{e}_y + v^z \boldsymbol{e}_z$ in the locally field-aligned coordinate system. We perform a change of basis into $(\boldsymbol{e}_y, \boldsymbol{e}_{1\perp}, \boldsymbol{e}_{2\perp})$

$$\begin{aligned} \boldsymbol{v} = &\left(v^y + b_{x_c} v^x + b_{y_c} v^z\right) \boldsymbol{e}_y \\ &+ \left(v^z + \frac{b_{x_c} b_{z_c}}{1 - b_{z_c}^2} v^x\right) \boldsymbol{e}_{1\perp} \\ &+ \left(v^x - \frac{b_{x_c}^2}{1 - b_{z_c}^2} v^x\right) \boldsymbol{e}_{2\perp} \end{aligned} \tag{3.42}$$

and calculate the orthogonal projection $\boldsymbol{v}_\perp$ by setting the component parallel to $\boldsymbol{e}_y$ to zero. Reverting the change of basis to $(\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)$ yields $\boldsymbol{v}_\perp$ expressed in the locally field-aligned coordinate system

$$\boldsymbol{v}_\perp = v^x \boldsymbol{e}_x - \left(b_{x_c} v^x + b_{z_c} v^z\right) \boldsymbol{e}_y + v^z \boldsymbol{e}_z\,. \tag{3.43}$$

Formally, we define the orthogonal projection operator via

$$\mathcal{P}_\perp : \boldsymbol{v} \to v^x \boldsymbol{e}_x - \left(b_{x_c} v^x + b_{z_c} v^z\right) \boldsymbol{e}_y + v^z \boldsymbol{e}_z\,. \tag{3.44}$$

With these we are set to derive differential operators in the new coordinate system.

3.2.3. Differential operators

The results presented in this subsection have been published in [87]. For the completeness of the presentation a discussion about the operator $\boldsymbol{\nabla} \cdot (h \boldsymbol{\nabla}_\perp f)$, presented in Eqs. (3.56)–(3.60), has been added.

There are four types of differential operators in the gyrokinetic Vlasov-Maxwell system which need to be expressed in the locally field-aligned coordinate system

$$\boldsymbol{b} \cdot \boldsymbol{\nabla} f, \qquad (\boldsymbol{\nabla} \times \boldsymbol{b}) \cdot \boldsymbol{\nabla} f, \qquad (\boldsymbol{b} \times \boldsymbol{\nabla} h) \cdot \boldsymbol{\nabla} f \qquad \text{and} \qquad \boldsymbol{\nabla} \cdot (h \boldsymbol{\nabla}_\perp f)\,, \tag{3.45}$$

where $\boldsymbol{b} = \boldsymbol{e}_y$ and f, h are arbitrary scalar functions. The first operation corresponds to the

parallel derivative and is given by

$$\boldsymbol{b}\cdot\boldsymbol{\nabla}f = \boldsymbol{e}_y\cdot\boldsymbol{e}^i\frac{\partial f}{\partial x^i} = \frac{\partial f}{\partial y}. \tag{3.46}$$

Eq. (3.46) shows that parallel derivatives are given by derivatives in the y coordinate. Due to the flute-like property of the plasma, mentioned in the first section of this chapter, it is, similar to flux-coordinates, possible to use a much coarser resolution in toroidal than in poloidal direction. This saves computational resources. For the second and third operator we express the curl and cross product in the new coordinate system. The curl of the magnetic field unit vector $\boldsymbol{b}$ is given by

$$\begin{aligned}\boldsymbol{\nabla}\times\boldsymbol{b} &= \frac{1}{\mathcal{J}_{\text{fci}}}\sum_{k=1}^{3}\epsilon_{ijk}\left(\frac{\partial b_j}{\partial x^i}-\frac{\partial b_i}{\partial x^j}\right)\boldsymbol{e}_k,\\ &= \frac{1}{b_{y_c}}\left[\frac{\partial b_{z_c}}{\partial y}\boldsymbol{e}_x-\frac{\partial b_{x_c}}{\partial y}\boldsymbol{e}_z+\left(\frac{\partial b_{x_c}}{\partial z}-\frac{\partial b_{z_c}}{\partial x}\right)\boldsymbol{e}_y\right],\end{aligned} \tag{3.47}$$

where the covariant components b_i, presented in Eq. (3.40), were used. The gradient of a function f in a curvilinear coordinate system can be expressed as

$$\boldsymbol{\nabla}f = \frac{\partial f}{\partial x^i}\boldsymbol{e}^i. \tag{3.48}$$

We obtain the second operation by combining Equations (3.47) and (3.48) and considering that $\boldsymbol{e}_i\cdot\boldsymbol{e}^j = \delta_i^j$

$$(\boldsymbol{\nabla}\times\boldsymbol{b})\cdot\boldsymbol{\nabla}f = \frac{1}{b_{y_c}}\left(\frac{\partial b_{z_c}}{\partial y}\frac{\partial f}{\partial x}-\frac{\partial b_{x_c}}{\partial y}\frac{\partial f}{\partial z}+\left(\frac{\partial b_{x_c}}{\partial z}-\frac{\partial b_{z_c}}{\partial x}\right)\frac{\partial f}{\partial y}\right). \tag{3.49}$$

The third operation is a triple vector product of the magnetic unit vector and two gradients of scalar functions. The gradients of the functions h and f are given by

$$\boldsymbol{\nabla}h = \frac{\partial h}{\partial x^i}\boldsymbol{e}^i \qquad \boldsymbol{\nabla}f = \frac{\partial f}{\partial x^j}\boldsymbol{e}^j. \tag{3.50}$$

The vector product in the curvilinear coordinate system takes the form

$$\boldsymbol{A}\times\boldsymbol{B} = \frac{1}{\mathcal{J}_{\text{fci}}}\sum_{k=1}^{3}\epsilon_{ijk}\left(A_iB_j-A_jB_i\right)\boldsymbol{e}_k \tag{3.51}$$

By combining Eqs. (3.39), (3.50) and (3.51), we obtain the vector product of the magnetic field unit vector with the gradient of h

$$\begin{aligned}\boldsymbol{b}\times\boldsymbol{\nabla}h = \frac{1}{b_{y_c}}\Bigg[&\left(\frac{\partial h}{\partial z}-b_{z_c}\frac{\partial h}{\partial y}\right)\boldsymbol{e}_x+\left(b_{x_c}\frac{\partial h}{\partial y}-\frac{\partial h}{\partial x}\right)\boldsymbol{e}_z\\ &+\left(b_{z_c}\frac{\partial h}{\partial x}-b_{x_c}\frac{\partial h}{\partial z}\right)\boldsymbol{e}_y\Bigg].\end{aligned} \tag{3.52}$$

Taking the dot product of the last expression with the gradient of the function f yields the third operation

$$(\boldsymbol{b} \times \boldsymbol{\nabla} h) \cdot \boldsymbol{\nabla} f = \frac{1}{b_{y_c}} \Bigg[\left(\frac{\partial h}{\partial z} - b_{z_c} \frac{\partial h}{\partial y} \right) \frac{\partial f}{\partial x} + \left(b_{x_c} \frac{\partial h}{\partial y} - \frac{\partial h}{\partial x} \right) \frac{\partial f}{\partial z} + \left(b_{z_c} \frac{\partial h}{\partial x} - b_{x_c} \frac{\partial h}{\partial z} \right) \frac{\partial f}{\partial y} \Bigg] . \tag{3.53}$$

We continue by rewriting Eq. (3.53) in terms of canonical Poisson brackets

$$\{h, f\}_{x,y} = \frac{\partial h}{\partial x} \frac{\partial f}{\partial y} - \frac{\partial h}{\partial y} \frac{\partial f}{\partial x}. \tag{3.54}$$

This rewriting is used in the discretisation, presented in Chapter 4, where we employ the Arakawa scheme for discretising Poisson brackets [121]

$$\begin{aligned} (\boldsymbol{b} \times \boldsymbol{\nabla} h) \cdot \boldsymbol{\nabla} f &= \frac{b_{x_c}}{b_{y_c}} \left(\frac{\partial h}{\partial y} \frac{\partial f}{\partial z} - \frac{\partial h}{\partial z} \frac{\partial f}{\partial y} \right) \\ &\quad + \frac{b_{z_c}}{b_{y_c}} \left(\frac{\partial h}{\partial x} \frac{\partial f}{\partial y} - \frac{\partial h}{\partial y} \frac{\partial f}{\partial x} \right) + \frac{1}{b_{y_c}} \left(\frac{\partial h}{\partial z} \frac{\partial f}{\partial y} - \frac{\partial h}{\partial x} \frac{\partial f}{\partial z} \right) \\ &= \frac{b_{x_c}}{b_{y_c}} \{h, f\}_{y,z} + \frac{b_{z_c}}{b_{y_c}} \{h, f\}_{x,y} + \frac{1}{b_{y_c}} \{h, f\}_{z,x} \,. \end{aligned} \tag{3.55}$$

The fourth operation is the divergence of a scalar function times the orthogonal gradient. The orthogonal gradient of the scalar function f is defined with the orthogonal projection operator $\mathcal{P}_\perp$, presented in Eq. (3.44), as

$$\boldsymbol{\nabla}_\perp f \cdot v := \boldsymbol{\nabla} f \cdot \mathcal{P}_\perp(v) \tag{3.56}$$

and yields in the locally field-aligned coordinate system

$$\boldsymbol{\nabla}_\perp = \boldsymbol{e}^x \left(\frac{\partial}{\partial x} - b_{x_c} \frac{\partial}{\partial y} \right) + \boldsymbol{e}^z \left(\frac{\partial}{\partial z} - b_{z_c} \frac{\partial}{\partial y} \right) . \tag{3.57}$$

The divergence of a vector field $\boldsymbol{v}$ in general curvilinear coordinate system reads

$$\boldsymbol{\nabla} \cdot \boldsymbol{v} = \frac{1}{\mathcal{J}_{\text{fci}}} \sum_{i=1}^{3} \sum_{j=1}^{3} \frac{\partial}{\partial x^i} \left(g^{ij} v_i \mathcal{J}_{\text{fci}} \right) . \tag{3.58}$$

Combining Eqs. (3.57) and (3.58) yields the fourth operation in locally field-aligned coordinates

$$\boldsymbol{\nabla} \cdot (h \boldsymbol{\nabla}_\perp f) = \frac{1}{\mathcal{J}_{\text{fci}}} \sum_{i=1}^{3} \sum_{j=1}^{3} \frac{\partial}{\partial x^i} \left(g^{ij} h (\boldsymbol{\nabla}_\perp f)_j \mathcal{J}_{\text{fci}} \right) . \tag{3.59}$$

In Eqs. (3.32) and (3.57) the metric tensor and the perpendicular gradient are evaluated on the poloidal plane at $y = 0$. As Eq. (3.59) contains y derivatives of the metric tensor and the perpendicular gradient, the full expressions of both are needed. As a result of the

y derivatives, the field equations on different poloidal planes are coupled and need to be solved at once. To save computational resources, we decouple the field equations on different poloidal planes by assuming that

$$\boldsymbol{\nabla} \cdot (h \boldsymbol{\nabla}_\perp f) \approx \left(\frac{1}{\mathcal{J}_{\text{cyl}}} \frac{\partial}{\partial x} \left(\mathcal{J}_{\text{cyl}} h \frac{\partial f}{\partial x} \right) + \frac{\partial}{\partial z} \left(h \frac{\partial f}{\partial z} \right) \right) . \tag{3.60}$$

In simulations of linear fusion devices or with a large aspect ratio assumption, the toroidal angle φ is interpreted as a Cartesian coordinate labelling poloidal planes and the Jacobian $\mathcal{J}_{\text{cyl}}$ is given by $\mathcal{J}_{\text{cyl}} = 1$. In toroidal geometries it reads $\mathcal{J}_{\text{cyl}} = x$. This approximation can be performed on the field part of the Lagrangian by replacing the orthogonal gradient with the gradient in cylindrical coordinates

$$\boldsymbol{\nabla}_\perp \approx \boldsymbol{\nabla}_{R,Z} := \boldsymbol{e}^R \frac{\partial}{\partial R} + \boldsymbol{e}^Z \frac{\partial}{\partial Z} . \tag{3.61}$$

By performing the approximation on the Lagrangian, the gyrokinetic Vlasov-Maxwell systems remains consistent. In summary, Eqs. (3.46), (3.49), (3.53) and (3.59) provide an explicit form of the operations in the gyrokinetic Vlasov-Maxwell system in the locally field aligned coordinate system. In `GENE-X` we replace Eq. (3.59) with Eq. (3.60).

3.2.4. Explicit form of the Vlasov-Maxwell system

The gyrokinetic Vlasov equation given by Eqs. (2.61), (2.77), (2.78) and (2.80) reads in the new coordinates

$$\begin{aligned} &\frac{\partial f}{\partial t} + \underbrace{v_\parallel \frac{\boldsymbol{B}^*}{B^*_\parallel} \cdot \boldsymbol{\nabla} f_\sigma}_{(a)} + \underbrace{\frac{c}{q_\sigma B^*_\parallel} \boldsymbol{b} \times (\mu \boldsymbol{\nabla} B + q_\sigma \boldsymbol{\nabla} \phi_1) \cdot \boldsymbol{\nabla} f_\sigma}_{(b)} - \underbrace{\frac{\boldsymbol{B}^*}{m_\sigma B^*_\parallel} \cdot (\mu \boldsymbol{\nabla} B + q_\sigma \boldsymbol{\nabla} \phi_1) \frac{\partial f_\sigma}{\partial v_\parallel}}_{(c)} \\ &\quad - \frac{q_\sigma}{m_\sigma c} \frac{\partial A_{1\parallel}}{\partial t} \frac{\partial f_\sigma}{\partial v_\parallel} = 0 \end{aligned} \tag{3.62}$$

with the components

$$\begin{aligned} (a) = v_\parallel &\frac{B}{B^*_\parallel} \frac{\partial f_\sigma}{\partial y} + m_\sigma v_\parallel^2 \frac{c}{q_\sigma B^*_\parallel} \frac{1}{b_{y_c}} \left(\frac{\partial b_{z_c}}{\partial y} \frac{\partial f_\sigma}{\partial x} - \frac{\partial b_{x_c}}{\partial y} \frac{\partial f_\sigma}{\partial z} + \left(\frac{\partial b_{x_c}}{\partial z} - \frac{\partial b_{z_c}}{\partial x} \right) \frac{\partial f_\sigma}{\partial y} \right) \\ &+ \frac{v_\parallel}{B^*_\parallel} \left(\frac{b_{x_c}}{b_{y_c}} \left(\frac{\partial A_{1\parallel}}{\partial z} \frac{\partial f_\sigma}{\partial y} - \frac{\partial A_{1\parallel}}{\partial y} \frac{\partial f_\sigma}{\partial z} \right) + \frac{b_{z_c}}{b_{y_c}} \left(\frac{\partial A_{1\parallel}}{\partial y} \frac{\partial f_\sigma}{\partial x} - \frac{\partial A_{1\parallel}}{\partial x} \frac{\partial f_\sigma}{\partial y} \right) \right. \\ &\left. + \frac{1}{b_{y_c}} \left(\frac{\partial A_{1\parallel}}{\partial x} \frac{\partial f_\sigma}{\partial z} - \frac{\partial A_{1\parallel}}{\partial z} \frac{\partial f_\sigma}{\partial x} \right) \right) , \end{aligned} \tag{3.63}$$

$$
\begin{aligned}
(b) = \frac{c}{q_\sigma B_\parallel^*}\Bigg[\quad & \mu\Bigg(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial B}{\partial y}\frac{\partial f_\sigma}{\partial z} - \frac{\partial B}{\partial z}\frac{\partial f_\sigma}{\partial y}\right) + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial B}{\partial x}\frac{\partial f_\sigma}{\partial y} - \frac{\partial B}{\partial y}\frac{\partial f_\sigma}{\partial x}\right) \\
& \quad + \frac{1}{b_{y_c}}\left(\frac{\partial B}{\partial z}\frac{\partial f_\sigma}{\partial x} - \frac{\partial B}{\partial x}\frac{\partial f_\sigma}{\partial z}\right)\Bigg) \\
& + q_\sigma\Bigg(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial \phi_1}{\partial y}\frac{\partial f_\sigma}{\partial z} - \frac{\partial \phi_1}{\partial z}\frac{\partial f_\sigma}{\partial y}\right) + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial \phi_1}{\partial x}\frac{\partial f_\sigma}{\partial y} - \frac{\partial \phi_1}{\partial y}\frac{\partial f_\sigma}{\partial x}\right) \\
& \quad + \frac{1}{b_{y_c}}\left(\frac{\partial \phi_1}{\partial z}\frac{\partial f_\sigma}{\partial x} - \frac{\partial \phi_1}{\partial x}\frac{\partial f_\sigma}{\partial z}\right)\Bigg)\Bigg] \quad \text{and}
\end{aligned} \tag{3.64}
$$

$$
\begin{aligned}
(c) = \frac{1}{m_\sigma B_\parallel^*}\Bigg[\quad & \mu\Bigg(B\frac{\partial B}{\partial y} + m_\sigma v_\parallel \frac{c}{q_\sigma}\frac{1}{b_{y_c}}\left(\frac{\partial b_{z_c}}{\partial y}\frac{\partial B}{\partial x} - \frac{\partial b_{x_c}}{\partial y}\frac{\partial B}{\partial z} + \left(\frac{\partial b_{x_c}}{\partial z} - \frac{\partial b_{z_c}}{\partial x}\right)\frac{\partial B}{\partial y}\right) \\
& \quad + \Bigg(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial A_{1\parallel}}{\partial z}\frac{\partial B}{\partial y} - \frac{\partial A_{1\parallel}}{\partial y}\frac{\partial B}{\partial z}\right) + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial A_{1\parallel}}{\partial y}\frac{\partial B}{\partial x} - \frac{\partial A_{1\parallel}}{\partial x}\frac{\partial B}{\partial y}\right) \\
& \quad + \frac{1}{b_{y_c}}\left(\frac{\partial A_{1\parallel}}{\partial x}\frac{\partial B}{\partial z} - \frac{\partial A_{1\parallel}}{\partial z}\frac{\partial B}{\partial x}\right)\Bigg)\Bigg) \\
& + q_\sigma\Bigg(B\frac{\partial \phi_1}{\partial y} + m_\sigma v_\parallel \frac{c}{q_\sigma}\frac{1}{b_{y_c}}\left(\frac{\partial b_{z_c}}{\partial y}\frac{\partial \phi_1}{\partial x} - \frac{\partial b_{x_c}}{\partial y}\frac{\partial \phi_1}{\partial z} + \left(\frac{\partial b_{x_c}}{\partial z} - \frac{\partial b_{z_c}}{\partial x}\right)\frac{\partial \phi_1}{\partial y}\right) \\
& \quad + \Bigg(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial A_{1\parallel}}{\partial z}\frac{\partial \phi_1}{\partial y} - \frac{\partial A_{1\parallel}}{\partial y}\frac{\partial \phi_1}{\partial z}\right) + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial A_{1\parallel}}{\partial y}\frac{\partial \phi_1}{\partial x} - \frac{\partial A_{1\parallel}}{\partial x}\frac{\partial \phi_1}{\partial y}\right) \\
& \quad + \frac{1}{b_{y_c}}\left(\frac{\partial A_{1\parallel}}{\partial x}\frac{\partial \phi_1}{\partial z} - \frac{\partial A_{1\parallel}}{\partial z}\frac{\partial \phi_1}{\partial x}\right)\Bigg)\Bigg]\frac{\partial f_\sigma}{\partial v_\parallel}\,.
\end{aligned} \tag{3.65}
$$

In the new coordinate system, employing Eq. (3.60), the quasi-neutrality equations reads

$$
-\left(\frac{1}{\mathcal{J}_{\text{cyl}}}\frac{\partial}{\partial x}\left(\mathcal{J}_{\text{cyl}}\sum_\sigma \frac{m_\sigma c^2 n_{0\sigma}}{B^2}\frac{\partial}{\partial x}\right) + \frac{\partial}{\partial z}\left(\sum_\sigma \frac{m_\sigma c^2 n_{0\sigma}}{B^2}\frac{\partial}{\partial z}\right)\right)\phi_1 = \sum_\sigma q_\sigma \int f_\sigma \,\mathrm{d}W\,, \tag{3.66}
$$

Ampère's law reads

$$
-\left(\frac{1}{\mathcal{J}_{\text{cyl}}}\frac{\partial}{\partial x}\left(\mathcal{J}_{\text{cyl}}\frac{\partial}{\partial x}\right) + \frac{\partial^2}{\partial z^2}\right)A_{1\parallel} = 4\pi\sum_\sigma \frac{q_\sigma}{c}\int f_\sigma v_\parallel \,\mathrm{d}W \tag{3.67}
$$

and the generalised Ohm's law reads

$$
\begin{aligned}
-\Bigg(\left(\frac{1}{\mathcal{J}_{\text{cyl}}}\frac{\partial}{\partial x}\left(\mathcal{J}_{\text{cyl}}\frac{\partial}{\partial x}\right) + \frac{\partial^2}{\partial z^2}\right) + 4\pi\sum_\sigma \frac{q_\sigma^2}{m_\sigma c^2}\int \frac{\partial f_\sigma}{\partial v_\parallel} v_\parallel \,\mathrm{d}W\Bigg)\frac{\partial A_{1\parallel}}{\partial t} \\
= 4\pi\sum_\sigma \frac{q_\sigma}{c}\int \left(\frac{\partial f_\sigma}{\partial t}\right)^\star v_\parallel \,\mathrm{d}W\,.
\end{aligned} \tag{3.68}
$$

3.2.5. Field line tracing

To calculate parallel y derivatives in practice, it is necessary to trace the position of magnetic field lines from one poloidal plane to another. In this subsection, we show how to perform

the field line tracing in the cylindrical coordinate system. We call the curve of the field line expressed in cylindrical coordinates $\boldsymbol{\gamma}_{\mathrm{cyl}} = (\gamma_R, \gamma_\varphi, \gamma_Z)$. In cylindrical coordinates, Eq. (3.21) reads

$$\frac{\mathrm{d}\gamma_R}{\mathrm{d}\lambda} = b_R \circ \boldsymbol{\gamma}_{\mathrm{cyl}} \tag{3.69}$$

$$\frac{\mathrm{d}\gamma_\varphi}{\mathrm{d}\lambda} = b_\varphi \circ \boldsymbol{\gamma}_{\mathrm{cyl}} \tag{3.70}$$

$$\frac{\mathrm{d}\gamma_Z}{\mathrm{d}\lambda} = b_Z \circ \boldsymbol{\gamma}_{\mathrm{cyl}}\,. \tag{3.71}$$

The length of the curve $\boldsymbol{\gamma}_{\mathrm{cyl}}$ starting at the point $\boldsymbol{\gamma}_{\mathrm{cyl}}(0) = \boldsymbol{R}_0 = (R_0, \varphi_0, Z_0)$ is given by

$$l : \boldsymbol{R}_0 \mapsto \int_0^a |\boldsymbol{b}(\boldsymbol{\gamma}_{\mathrm{cyl}}(\lambda))| \mathrm{d}\lambda = a\,, \tag{3.72}$$

which demonstrates that the orbit parameter λ of the curve labels the length of the field line. To follow the position of a magnetic field line, starting at $\boldsymbol{R}_0$, around the fusion device, we parametrise the curve of the magnetic field line with its poloidal angle instead of its length. We define the function that assigns the poloidal angle of the field line to its length as

$$s : \varphi \mapsto \int_0^a |\boldsymbol{b}(\boldsymbol{\gamma}_{\mathrm{cyl}}(\lambda)| \mathrm{d}\lambda = a = \gamma_\varphi^{-1}(\varphi)\,, \tag{3.73}$$

where a is chosen such that $\gamma_\varphi(a) = \varphi$. The function s can be simplified

$$\begin{aligned} s(\varphi) &= \int_0^a |\boldsymbol{b}(\boldsymbol{\gamma}_{\mathrm{cyl}}(\lambda)| \mathrm{d}\lambda \\ &= \int_0^{\gamma_\varphi^{-1}(\varphi)} 1 \,\mathrm{d}\lambda \\ &= \int_{\varphi_0}^{\varphi} \frac{\mathrm{d}\gamma_\varphi^{-1}}{\mathrm{d}u}(u)\,\mathrm{d}u \\ &= \int_{\varphi_0}^{\varphi} \frac{1}{b_\varphi(\boldsymbol{\gamma}_{\mathrm{cyl}}(\gamma_\varphi^{-1}(u))}\,\mathrm{d}u\,, \end{aligned} \tag{3.74}$$

where we have used Eq. (3.26). By taking the derivative of Eq. (3.74), we obtain an expression for the derivative of the field line length

$$\left(\frac{\mathrm{d}s}{\mathrm{d}\varphi}\right)(\varphi) = \frac{1}{b_\varphi(\boldsymbol{\gamma}_{\mathrm{cyl}}(\gamma_\varphi^{-1}(\varphi))}\,. \tag{3.75}$$

Similar to the function $s(\varphi)$, parametrising the field line length by the toroidal angle φ, it is possible to derive functions $Z(\varphi)$ and $R(\varphi)$. They parametrise the R and Z coordinates the

magnetic field line visits by the poloidal angle φ. Their derivatives read

$$\left(\frac{\mathrm{d}R}{\mathrm{d}\varphi}\right)(\varphi) = \frac{b_R(\boldsymbol{\gamma}_{\mathrm{cyl}}(\gamma_\varphi^{-1}(\varphi)))}{b_\varphi(\boldsymbol{\gamma}_{\mathrm{cyl}}(\gamma_\varphi^{-1}(\varphi)))} \tag{3.76}$$

$$\left(\frac{\mathrm{d}Z}{\mathrm{d}\varphi}\right)(\varphi) = \frac{b_Z(\boldsymbol{\gamma}_{\mathrm{cyl}}(\gamma_\varphi^{-1}(\varphi)))}{b_\varphi(\boldsymbol{\gamma}_{\mathrm{cyl}}(\gamma_\varphi^{-1}(\varphi)))} \tag{3.77}$$

$$\left(\frac{\mathrm{d}\varphi}{\mathrm{d}\varphi}\right)(\varphi) = 1\,. \tag{3.78}$$

Eqs. (3.75), (3.76), (3.77) and (3.78) provide the position of the magnetic field line and its length in cylindrical coordinates given a starting position of the field line $\boldsymbol{R}_0$ and a poloidal angle φ travelled. We solve the system numerically with an explicit Runge-Kutta method of order eight called `DOP853` [122, II. 5][†]. The integration is implemented in the library `PARALLAX`. For more details on `PARALLAX` we refer to Chapter 5.

3.3. Conclusion and outlook

When performing turbulence simulations of the edge and SOL, it is important to use a coordinate system that (i) is aligned to the magnetic field and (ii) is well-defined for realistic magnetic geometries, including the separatrix and multiple X-points. Flux-coordinate systems, which have been successfully used in fusion theory for decades, suffer from a coordinate singularity at the separatrix that renders them inapplicable for predictive edge and SOL simulations.

In this chapter, we presented a locally field-aligned coordinate system that is based on the FCI approach. This coordinate system is not restricted by the magnetic geometry and allows for the simulation in geometries with single and even multiple X-points. Furthermore, the coordinate system is still aligned to the magnetic field such that derivatives along the magnetic field line are represented as a derivative in the coordinate y. This respects the flute like structure of plasma turbulence and allows to reduce the toroidal resolution needed for simulations.

As the coordinate system is not limited by the magnetic geometry, it would be interesting to generalise the code to 3D geometries in the future. This would enable the exploration of turbulence in the SOL of stellarator fusion experiments as well as stochastic regions and magnetic islands.

[†]In the code, the toroidal component of the magnetic field is given with respect to the normalised basis vector $\boldsymbol{e}_{y_c} = \boldsymbol{e}_\varphi/|\boldsymbol{e}_\varphi| = \boldsymbol{e}_\varphi/\mathcal{J}_{\mathrm{cyl}}$. Therefore, the right-hand side of Eqs. (3.75), (3.76) and (3.77) is multiplied with and additional factor of $\mathcal{J}_{\mathrm{cyl}}$ and b_φ is replaced with b_{y_c}.

4. Numerical scheme

For a stable, long-time integration of the gyrokinetic Vlasov-Maxwell system it is important to select a numerical scheme that reflects the properties of the equations. In this chapter, we present the numerical scheme that is used in the `GENE-X` code. We start with the derivation of a dimensionless form of the gyrokinetic Vlasov-Maxwell system, appropriate for numerical discretisation. Next, we discuss which discretisation strategies are suited for the locally field aligned coordinate system, presented in the last chapter, and detail the discretisation scheme implemented in the `GENE-X` code. Finally, we discuss numerical diffusion and boundary as well as initial conditions for the simulations.

4.1. Normalisation

For the numerical implementation it is useful to renormalise the equations. With the normalisation different physical quantities can be scaled to numerical values of order $O(1)$. This is beneficial because floating point numbers are most dense around 1 and therefore smaller differences between numbers can be resolved. Furthermore, it is possible to choose different normalisations for different species. This is for example useful for the representation of the distribution function in velocity space. Due to their lower weight, electrons move faster than ions. By adjusting the normalisation of the velocity coordinate of the electron species, it is possible to use the same velocity space grid for ions and electrons.
The following paragraph has been published in [87]. As the gyrokinetic model has been generalised to include electromagnetic effects, the normalisation of the parallel component of the electromagnetic vector potential has been added.
We use the normalisation from the `GENE` code [45]. We denote reference quantities with a $_{\mathrm{ref}}$ subscript and renormalised quantities with a hat, i.e. for the mass $m_\sigma = \hat{m}_\sigma m_{\mathrm{ref}}$. The reference values, defining the scales of the simulation, are given in Tab. 4.1. With the reference quantities from Tab. 4.1, we normalise the gyrocentre coordinates as

$$\begin{aligned} x &= \hat{x} L_{\mathrm{ref}} & z &= \hat{z} L_{\mathrm{ref}} & \varphi &= \hat{\varphi} \\ v_\parallel &= \hat{v}_\parallel v_{T_\sigma}(\psi_0) & \mu &= \hat{\mu} \frac{T_{0\sigma}(\psi_0)}{B_{\mathrm{ref}}} & t &= \hat{t} \frac{L_{\mathrm{ref}}}{c_{\mathrm{ref}}}\,, \end{aligned} \tag{4.1}$$

where $T_{0\sigma}(\psi_0) = T_{\mathrm{ref}} \hat{T}_{0\sigma}(\psi_0)$ denotes the initial plasma temperature at a given flux surface ψ_0 and $v_{T_\sigma}(\psi_0) = \sqrt{2T_{0\sigma}(\psi_0)/m_\sigma} = c_{\mathrm{ref}} \hat{v}_{T_\sigma}(\psi_0)$ the corresponding thermal velocity. Similarly, $n_{0\sigma}(\psi_0) = n_{\mathrm{ref}} \hat{n}_{0\sigma}(\psi_0)$ denotes the initial plasma density. Finally, we normalise the distribution function, the electrostatic potential and the parallel component of the electromagnetic

Elementary charge	e	s.t. $q_\sigma = \hat{q}_\sigma e$
Reference mass	m_{ref}	s.t. $m_\sigma = \hat{m}_\sigma m_{\mathrm{ref}}$
Reference temperature	T_{ref}	
Reference length	L_{ref}	
Reference magnetic field strength	B_{ref}	
Reference density	n_{ref}	
Reference velocity	$c_{\mathrm{ref}} = \sqrt{T_{\mathrm{ref}}/m_{\mathrm{ref}}}$	
Reference gyrofrequency	$\Omega_{\mathrm{ref}} = eB_{\mathrm{ref}}/(m_{\mathrm{ref}}c)$	
Reference gyroradius	$\rho_{\mathrm{ref}} = c_{\mathrm{ref}}/\Omega_{\mathrm{ref}}$	
Reference β	$\beta_{\mathrm{ref}} = 8\pi n_{\mathrm{ref}} T_{\mathrm{ref}}/B_{\mathrm{ref}}^2$	

Table 4.1.: Elementary reference values for mass, temperature, length, magnetic field strength and density and derived reference values for velocity, gyrofrequency, gyroradius and thermal to magnetic pressure ratio. This table has been published in [87]. To account for the electromagnetic model, the reference β has been added.

vector potential according to

$$f_\sigma = \frac{n_{0\sigma}(\psi_0)}{v_{T_\sigma}(\psi_0)^3}\hat{f}_\sigma \tag{4.2}$$

$$\phi_1 = \frac{T_{\mathrm{ref}}}{e}\hat{\phi}_1 \tag{4.3}$$

$$A_{1\parallel} = \rho_{\mathrm{ref}} B_{\mathrm{ref}} \hat{A}_{1\parallel}\,. \tag{4.4}$$

4.1.1. Normalisation of the gyrokinetic Vlasov-Maxwell system

We normalise the gyrokinetic Vlasov-Maxwell system in parts. We start with the normalisation of $\boldsymbol{B}^*$

$$\begin{aligned}\boldsymbol{B}^* &= \boldsymbol{B} + \frac{m_\sigma v_\parallel c}{q_\sigma}\boldsymbol{\nabla}\times\boldsymbol{b} + \boldsymbol{\nabla}A_{1\parallel}\times\boldsymbol{b}\\ &= B_{\mathrm{ref}}\left(\hat{\boldsymbol{B}} + \sqrt{2\hat{m}_\sigma \hat{T}_{0\sigma}(\psi_0)}\frac{\hat{v}_\parallel}{\hat{q}_\sigma}\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\hat{\boldsymbol{\nabla}}\times\boldsymbol{b} + \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\hat{\boldsymbol{\nabla}}\hat{A}_{1\parallel}\times\boldsymbol{b}\right)\\ &= B_{\mathrm{ref}}\hat{\boldsymbol{B}}^*\,.\end{aligned} \tag{4.5}$$

As a corollary of Eq. (4.5), the parallel component of $\boldsymbol{B}^*$ is normalised in the same way, i.e. $B_\parallel^* = B_{\mathrm{ref}}\hat{B}_\parallel^*$. The partial derivatives of the distribution function f_σ are normalised as

$$\frac{\partial f_\sigma}{\partial t} = \frac{n_{0\sigma}(\psi_0)}{v_{T_\sigma}^3(\psi_0)}\frac{c_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{\partial \hat{f}_\sigma}{\partial \hat{t}} \tag{4.6}$$

$$\boldsymbol{\nabla} f_\sigma = \frac{n_{0\sigma}(\psi_0)}{v_{T_\sigma}^3(\psi_0)}\frac{1}{L_{\mathrm{ref}}}\hat{\boldsymbol{\nabla}}\hat{f}_\sigma \tag{4.7}$$

$$\frac{\partial f_\sigma}{\partial v_\parallel} = \frac{n_{0\sigma}(\psi_0)}{v_{T_\sigma}^3(\psi_0)}\frac{1}{v_{T_\sigma}(\psi_0)}\frac{\partial \hat{f}_\sigma}{\partial \hat{v}_\parallel}\,. \tag{4.8}$$

We proceed by normalising the equations of motion of the gyrocentres. The change of the gyrocentre position reads

$$\begin{aligned}\dot{\boldsymbol{R}} &= \frac{v_\parallel \boldsymbol{B}^*}{B_\parallel^*} + \frac{c\boldsymbol{b}}{q_\sigma B_\parallel^*} \times (\mu \boldsymbol{\nabla} B + q_\sigma \boldsymbol{\nabla} \phi_1) \\ &= c_{\mathrm{ref}} \left(\sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}} \hat{v}_\parallel \frac{\hat{\boldsymbol{B}}^*}{\hat{B}_\parallel^*} + \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}} \frac{\boldsymbol{b}}{\hat{q}_\sigma \hat{B}_\parallel^*} \times \left(\hat{T}_{0\sigma}(\psi_0) \hat{\mu} \hat{\boldsymbol{\nabla}} \hat{B} + \hat{q}_\sigma \hat{\boldsymbol{\nabla}} \hat{\phi}_1 \right) \right) , \end{aligned} \tag{4.9}$$

and the change of the parallel velocity reads

$$\dot{v}_\parallel = -\frac{\boldsymbol{B}^*}{m_\sigma B_\parallel^*} \cdot (\mu \boldsymbol{\nabla} B + q_\sigma \boldsymbol{\nabla} \phi_1) - \frac{q_\sigma}{m_\sigma c} \frac{\partial A_{1\parallel}}{\partial t} \tag{4.10}$$

$$= -\frac{c_{\mathrm{ref}}^2}{L_{\mathrm{ref}}} \frac{\hat{\boldsymbol{B}}^*}{\hat{m}_\sigma \hat{B}_\parallel^*} \cdot \left(\hat{T}_{0\sigma}(\psi_0) \hat{\mu} \hat{\boldsymbol{\nabla}} \hat{B} + \hat{q}_\sigma \hat{\boldsymbol{\nabla}} \hat{\phi}_1 \right) - \frac{c_{\mathrm{ref}}^2}{L_{\mathrm{ref}}} \frac{\hat{q}_\sigma}{\hat{m}_\sigma} \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} . \tag{4.11}$$

By collecting the normalised parts and dividing by the factor $c_{\mathrm{ref}}/L_{\mathrm{ref}}\, n_{0\sigma}(\psi_0)/v_{T_\sigma}^3(\psi_0)$ we obtain the normalised Vlasov equation

$$\begin{aligned}&\frac{\partial \hat{f}_\sigma}{\partial \hat{t}} + \sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}} \hat{v}_\parallel \frac{\hat{\boldsymbol{B}}^*}{\hat{B}_\parallel^*} \cdot \hat{\boldsymbol{\nabla}} \hat{f}_\sigma + \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}} \frac{\boldsymbol{b}}{\hat{q}_\sigma \hat{B}_\parallel^*} \times \left(\hat{T}_{0\sigma}(\psi_0) \hat{\mu} \hat{\boldsymbol{\nabla}} \hat{B} + \hat{q}_\sigma \hat{\boldsymbol{\nabla}} \hat{\phi}_1 \right) \hat{\boldsymbol{\nabla}} \hat{f}_\sigma \\ &\quad - \frac{\hat{\boldsymbol{B}}^*}{\sqrt{2\hat{m}_\sigma \hat{T}_{0\sigma}(\psi_0)} \hat{B}_\parallel^*} \cdot \left(\hat{T}_{0\sigma}(\psi_0) \hat{\mu} \hat{\boldsymbol{\nabla}} \hat{B} + \hat{q}_\sigma \hat{\boldsymbol{\nabla}} \hat{\phi}_1 \right) \frac{\partial \hat{f}_\sigma}{\partial \hat{v}_\parallel} - \frac{\hat{q}_\sigma}{\sqrt{2\hat{m}_\sigma \hat{T}_{0\sigma}(\psi_0)}} \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} \frac{\partial \hat{f}_\sigma}{\partial \hat{v}_\parallel} \\ &\quad = 0 . \end{aligned} \tag{4.12}$$

We continue with the normalisation of the quasi-neutrality equation, given in Eq. (2.84). We start with the right-hand side

$$\begin{aligned}\sum_\sigma q_\sigma \int f_\sigma \, \mathrm{d}W &= \sum_\sigma q_\sigma \int_0^\infty \int_{-\infty}^\infty 2\pi \frac{B_\parallel^*}{m_\sigma} f_\sigma \, \mathrm{d}v_\parallel \, \mathrm{d}\mu \\ &= e n_{\mathrm{ref}} \sum_\sigma \hat{n}_{0\sigma}(\psi_0) \hat{q}_\sigma \int_0^\infty \int_{-\infty}^\infty \pi \hat{B}_\parallel^* \hat{f}_\sigma \, \mathrm{d}\hat{v}_\parallel \, \mathrm{d}\hat{\mu} . \end{aligned} \tag{4.13}$$

The left-hand side reads

$$-\boldsymbol{\nabla} \cdot \left(\sum_\sigma \frac{m_\sigma c^2 n_{0\sigma}}{B^2} \boldsymbol{\nabla}_\perp \phi \right) = -e n_{\mathrm{ref}} \left(\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}} \right)^2 \hat{\boldsymbol{\nabla}} \cdot \left(\sum_\sigma \frac{\hat{m}_\sigma \hat{n}_{0\sigma}}{\hat{B}^2} \hat{\boldsymbol{\nabla}}_\perp \hat{\phi}_1 \right) . \tag{4.14}$$

Combining both sides and multiplying by $(n_{\mathrm{ref}} e)^{-1}$ yields the normalised quasi-neutrality equation

$$-\left(\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}} \right)^2 \hat{\boldsymbol{\nabla}} \cdot \left(\sum_\sigma \frac{\hat{m}_\sigma \hat{n}_{0\sigma}}{\hat{B}^2} \hat{\boldsymbol{\nabla}}_\perp \hat{\phi}_1 \right) = \sum_\sigma \hat{n}_{0\sigma}(\psi_0) \hat{q}_\sigma \int_0^\infty \int_{-\infty}^\infty \pi \hat{B}_\parallel^* \hat{f}_\sigma \, \mathrm{d}\hat{v}_\parallel \, \mathrm{d}\hat{\mu} . \tag{4.15}$$

The normalisation of Ampère's law is similar to the quasi-neutrality equation. We start with the right-hand side

$$\begin{aligned} 4\pi \sum_\sigma \frac{q_\sigma}{c} \int f_\sigma v_\parallel \,\mathrm{d}W &= 4\pi \sum_\sigma \frac{q_\sigma}{c} \int_0^\infty \int_{-\infty}^\infty 2\pi \frac{B_\parallel^*}{m_\sigma} f_\sigma v_\parallel \,\mathrm{d}v_\parallel \,\mathrm{d}\mu \\ &= \frac{B_\mathrm{ref}}{\rho_\mathrm{ref}} \frac{\beta_\mathrm{ref}}{2} \sum_\sigma \hat{n}_{0\sigma}(\psi_0) \hat{q}_\sigma \sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}} \int_0^\infty \int_{-\infty}^\infty \pi \hat{B}_\parallel^* \hat{f}_\sigma \hat{v}_\parallel \,\mathrm{d}\hat{v}_\parallel \,\mathrm{d}\hat{\mu} \,. \end{aligned} \tag{4.16}$$

The left-hand side reads

$$-\Delta_\perp A_{1\parallel} = -\frac{\rho_\mathrm{ref} B_\mathrm{ref}}{L_\mathrm{ref}^2} \hat{\Delta}_\perp \hat{A}_{1\parallel} \,. \tag{4.17}$$

Combining Eqs. (4.16), (4.17) and dividing by $B_\mathrm{ref}/\rho_\mathrm{ref}$ yields the normalised Ampère's law

$$-\left(\frac{\rho_\mathrm{ref}}{L_\mathrm{ref}}\right)^2 \hat{\Delta}_\perp \hat{A}_{1\parallel} = \frac{\beta_\mathrm{ref}}{2} \sum_\sigma \hat{n}_{0\sigma}(\psi_0) \hat{q}_\sigma \sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}} \int_0^\infty \int_{-\infty}^\infty \pi \hat{B}_\parallel^* \hat{f}_\sigma \hat{v}_\parallel \,\mathrm{d}\hat{v}_\parallel \,\mathrm{d}\hat{\mu} \,. \tag{4.18}$$

The normalised form of the generalised Ohm's law can either be derived explicitly or by taking the normalised time derivative of Eq. (4.18). It reads

$$\begin{aligned} &-\left(\left(\frac{\rho_\mathrm{ref}}{L_\mathrm{ref}}\right)^2 \hat{\Delta}_\perp + \frac{\beta_\mathrm{ref}}{2} \sum_\sigma \hat{n}_{0\sigma}(\psi_0) \frac{\hat{q}_\sigma^2}{\hat{m}_\sigma} \int_0^\infty \int_{-\infty}^\infty \pi \hat{B}_\parallel^* \frac{\partial \hat{f}_\sigma}{\partial \hat{v}_\parallel} \hat{v}_\parallel \,\mathrm{d}\hat{v}_\parallel \,\mathrm{d}\hat{\mu}\right) \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} \\ &\qquad = \frac{\beta_\mathrm{ref}}{2} \sum_\sigma \hat{n}_{0\sigma}(\psi_0) \hat{q}_\sigma \sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}} \int_0^\infty \int_{-\infty}^\infty \pi \hat{B}_\parallel^* \left(\frac{\partial \hat{f}_\sigma}{\partial \hat{t}}\right)^\star \hat{v}_\parallel \,\mathrm{d}\hat{v}_\parallel \,\mathrm{d}\hat{\mu} \,. \end{aligned} \tag{4.19}$$

The explicit form of the normalised gyrokinetic Vlasov-Maxwell system, expressed in the locally field-aligned coordinate system, is given in Appendix A.2.

4.1.2. Normalisation of the local and canonical Maxwellian

In order to initialise the simulation or set radial boundary conditions, it is useful to have a normalised Maxwellian distribution function at our disposal. We denote the Maxwellian distribution function with $F_{0\sigma}$ and normalise it in the same way as f_σ. It reads as a function of the flux surface label ψ, $\hat{v}_\parallel$ and $\hat{\mu}$

$$\hat{F}_{0\sigma}(\psi, \hat{v}_\parallel, \hat{\mu}) = \frac{\hat{n}_{p\sigma}(\psi)}{\left(\pi \hat{T}_{p\sigma}(\psi)\right)^{3/2}} e^{-\frac{\hat{v}_\parallel^2 + \hat{\mu}\hat{B}}{\hat{T}_{p\sigma}(\psi)}} \,, \tag{4.20}$$

where the normalised temperature and density profiles $\hat{T}_{p\sigma}(\psi) = T_{0\sigma}(\psi)/T_{0\sigma}(\psi_0)$ and $\hat{n}_{p\sigma}(\psi) = n_{0\sigma}(\psi)/n_{0\sigma}(\psi_0)$ were used. Next to the local Maxwellian, we make use of a so-called canonical Maxwellian. The canonical Maxwellian is defined in terms of constants of motion and will be discussed in detail in Section 4.5. It uses the canonical toroidal momentum p_φ as a replacement for the flux surface label ψ. The new modified radial coordinate ψ_C is defined

by

$$\begin{aligned}\psi_C &= \frac{c}{q_\sigma} p_\varphi \\ &= \psi_p + \frac{c}{q_\sigma} m_\sigma v_\parallel b_\varphi R \,,\end{aligned} \tag{4.21}$$

and normalised as

$$\psi_C = \psi_p + \frac{\rho_{\text{ref}}}{L_{\text{ref}}} \sqrt{2 \hat{m}_\sigma \hat{T}_{0\sigma}(\psi_0)} \hat{q}_\sigma \hat{v}_\parallel b_\varphi \hat{R} \,. \tag{4.22}$$

4.2. Discretisation

We use the method of lines to construct the discretisation scheme. The gyrokinetic Vlasov-Maxwell system can be written in the form

$$\frac{\partial f}{\partial t} = L[f] \,, \tag{4.23}$$

where L is a nonlinear, integro-partial differential operator acting on f[†]. In the method of lines the operator L acting on phase space is discretised first. This yields a system of ordinary differential equations

$$\frac{\mathrm{d}(f_i)}{\mathrm{d}t} = L_D[(f_i)] \,, \tag{4.24}$$

where (f_i) corresponds to a vector containing the values of the distribution function at the phase space grid points i, and L_D denotes the discretised differential operator. The initial value problem in Eq. (4.24) can then be solved with a method of choice. In the following, we discuss different spatial and temporal discretisation strategies.

4.2.1. Spatial discretisation

There are multiple different strategies to construct a spatial discretisation of an integro-partial differential operator. Most schemes fall in the class of *finite difference*, *finite volume* or *finite element* schemes. There are also schemes in between these classes. A good example are schemes developed in the area of geometric discretisations and mimetic finite differences [123, 124, 125]. Therein, differential operators are discretised based on the preservation of mathematical properties of the continuous system. Further, spectral methods based on Fourier decomposition in real space [68, 44] or an expansion in orthogonal Hermite and Laguerre polynomials in velocity space [126, 127, 128] are often used to exploit symmetries of magnetically confined fusion plasmas.

Spatial discretisation of the gyrokinetic Vlasov equation

We discuss different spatial discretisation schemes in the language of differential forms to explicitly clarify the structure and properties of the quantities and operators involved. As

[†]In this form the function f is a multi-dimensional function with a component for each species and the nonlinear, integro-partial differential operator L represents the whole gyrokinetic Vlasov-Maxwell system including the quasi-neutrality equation, Ampère's law and the generalised Ohm's law.

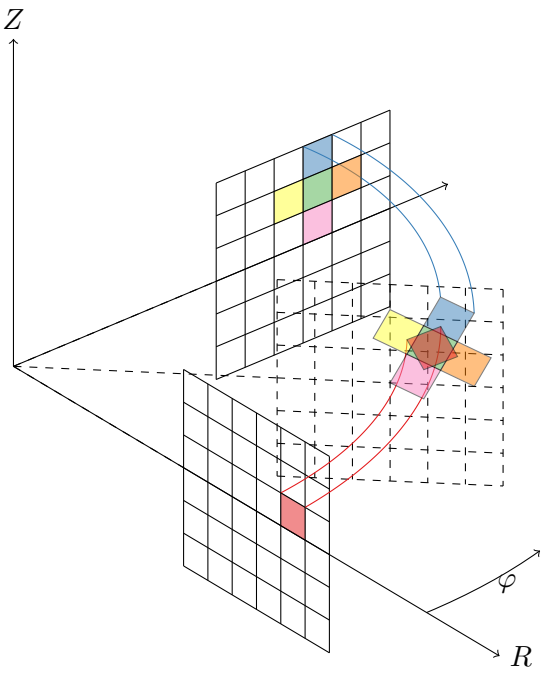

Figure 4.1.: Figure displaying the overlap of flux boxes originating from two poloidal planes at $\varphi = 0$ and $\varphi = \pi/2$. In a conservative finite volume or finite element discretisation, fluxes are exchanged between different boxes. In the locally field-aligned coordinate system, the flux boxes overlap in a non-trivial way demonstrated on a staggered poloidal plane at $\varphi = \pi/4$.

discussed in Chapter 2, the gyrokinetic Vlasov equation can be written in advective and conservative form. On the continuous level both formulations are equivalent. Which one is chosen makes a difference for the discretisation. Let $\Lambda^n(\mathcal{P})$ be the space of the differential n-forms on the phase space $\mathcal{P}$. In the conservative form of the Vlasov equation the distribution function corresponds to a six-form on the manifold, $f^{(6)} \in \Lambda^6(\mathcal{P})$, that can be integrated to obtain the particle number in a given volume. The Vlasov equation can be written as

$$\frac{\partial f^{(6)}}{\partial t} = -\,\mathrm{d}\zeta^{(5)} \tag{4.25}$$

$$\zeta^{(5)} = \iota_u f^{(6)}\,, \tag{4.26}$$

where the five-form $\zeta^{(5)}$ corresponds to the five-dimensional flux out of a six dimensional unit volume and ι_u denotes the interior product. The vector field u represents the drift velocities and is itself a nonlinear function of f. This form of the Vlasov equation is well suited for finite volume or finite element discretisation schemes because the distribution function corresponds to volume averages and $\zeta^{(5)}$ to fluxes. In the locally field-aligned coordinate system it is difficult to correctly calculate fluxes because flux boxes from different poloidal planes do not align. They overlap in a non-trivial way and sophisticated book keeping would be needed to correctly handle the fluxes at the intersection points and ensure conservation. Exemplarily, the overlap of flux boxes is demonstrated in Fig. 4.1.

In the advective form the distribution function corresponds to a scalar function or zero form

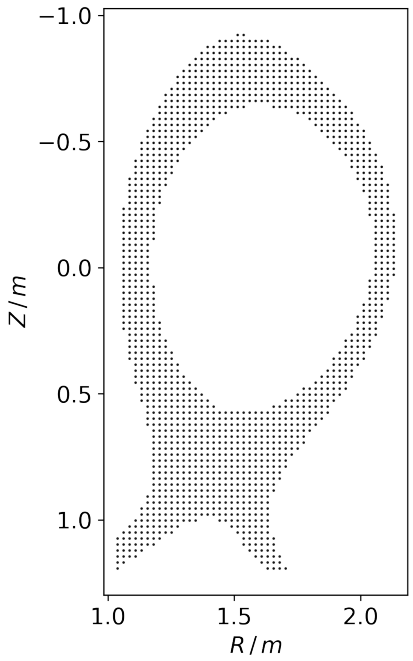

Figure 4.2.: Figure displaying the grid in the x and z coordinates at the example of AUG, used for simulations in Chapter 7. The grid is unstructured, such that the boundary of fusion devices can be modelled, and Cartesian in the neighbourhood of every point. For a clear presentation only every 20th grid point is shown.

$f^{(0)} \in \Lambda^0(\mathcal{P})$. The Vlasov equation can be written as

$$\frac{\partial f^{(0)}}{\partial t} = -\iota_u \zeta^{(1)} \tag{4.27}$$

$$\zeta^{(1)} = \mathrm{d} f^{(0)}\,, \tag{4.28}$$

where $\zeta^{(1)}$ is a one form corresponding naturally to the gradient of f. The advective form of the Vlasov equation is well suited for a finite difference discretisation because gradients correspond to differences of scalar function values.

Due to the non-conformal overlap of flux boxes, we discretise the advective form of the Vlasov equation with a finite difference discretisation scheme. As the coordinate system is defined in regions around a family of poloidal planes, we choose a Cartesian grid in the toroidal angle φ. In the coordinates x and z in the poloidal plane we use an unstructured grid that is Cartesian in the neighbourhood of every point. The grid is constructed such that $\Delta x = \Delta z$. We label each grid point by a one-dimensional index i and store the x and z coordinates in one-dimensional arrays `x[i]` and `z[i]`. This makes the implementation flexible because one can remove points from and add points to the grid and thus model the complex shape of the fusion device. Requiring that the grid is Cartesian in the neighbourhood of every point allows us to use efficient numerical schemes for Cartesian grids [87]. The grid is visualised at the example of an AUG equilibrium in Fig 4.2. In the velocity space coordinate $v_\parallel$ we use a Cartesian grid. We discretise the first derivatives in x, z and $v_\parallel$ direction, apart from the

nonlinear Poisson brackets

$$\{\hat{A}_{1\parallel}, \hat{f}_\sigma\}_{\hat{z},\hat{x}} = \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{z}} \frac{\partial \hat{f}_\sigma}{\partial \hat{x}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{x}} \frac{\partial \hat{f}_\sigma}{\partial \hat{z}} \tag{4.29}$$

$$\{\hat{\phi}_1, \hat{f}_\sigma\}_{\hat{z},\hat{x}} = \frac{\partial \hat{\phi}_1}{\partial \hat{z}} \frac{\partial \hat{f}_\sigma}{\partial \hat{x}} - \frac{\partial \hat{\phi}_1}{\partial \hat{x}} \frac{\partial \hat{f}_\sigma}{\partial \hat{z}}\,, \tag{4.30}$$

with a fourth order centred finite difference scheme

$$\left(\frac{\partial \hat{f}_\sigma}{\partial \hat{x}}\right)(\hat{x}_i) = \frac{1}{12\Delta\hat{x}} \left(\hat{f}_\sigma(\hat{x}_{i-2}) - 8\hat{f}_\sigma(\hat{x}_{i-1}) + 8\hat{f}_\sigma(\hat{x}_{i+1}) - \hat{f}_\sigma(\hat{x}_{i+2})\right) + O\left((\Delta\hat{x})^4\right). \tag{4.31}$$

For the terms in Eqs. (4.29) and (4.30), we use the second order scheme introduced by Arakawa [121]

$$\begin{aligned}
&\{\hat{\phi}_1, \hat{f}_\sigma\}_{\hat{x},\hat{z}}(\hat{x}_i, \hat{z}_j) = \\
&\frac{-1}{12(\Delta\hat{x})^2}\Bigg[\left(\hat{f}_\sigma(\hat{x}_i, \hat{z}_{j-1}) + \hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_{j-1}) - \hat{f}_\sigma(\hat{x}_i, \hat{z}_{j+1}) - \hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_{j+1})\right) \\
&\left(\hat{\phi}_1(\hat{x}_{i+1}, \hat{z}_j) - \hat{\phi}_1(\hat{x}_i, \hat{z}_j)\right) + \left(\hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_{j-1}) + \hat{f}_\sigma(\hat{x}_i, \hat{z}_{j-1}) - \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_{j+1}) - \hat{f}_\sigma(\hat{x}_i, \hat{z}_{j+1})\right) \\
&\left(\hat{\phi}_1(\hat{x}_i, \hat{z}_j) - \hat{\phi}_1(\hat{x}_{i-1}, \hat{z}_j)\right) + \left(\hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_j) + \hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_{j+1}) - \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_j) - \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_{j+1})\right) \\
&\left(\hat{\phi}_1(\hat{x}_i, \hat{z}_{j+1}) - \hat{\phi}_1(\hat{x}_i, \hat{z}_j)\right) + \left(\hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_{j-1}) + \hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_j) - \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_{j-1}) - \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_j)\right) \\
&\left(\hat{\phi}_1(\hat{x}_i, \hat{z}_j) - \hat{\phi}_1(\hat{x}_i, \hat{z}_{j-1})\right) + \left(\hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_j) - \hat{f}_\sigma(\hat{x}_i, \hat{z}_{j+1})\right)\left(\hat{\phi}_1(\hat{x}_{i+1}, \hat{z}_{j+1}) - \hat{\phi}_1(\hat{x}_i, \hat{z}_j)\right) \\
&+ \left(\hat{f}_\sigma(\hat{x}_i, \hat{z}_{j-1}) - \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_j)\right)\left(\hat{\phi}_1(\hat{x}_i, \hat{z}_j) - \hat{\phi}_1(\hat{x}_{i-1}, \hat{z}_{j-1})\right) \\
&+ \left(\hat{f}_\sigma(\hat{x}_i, \hat{z}_{j+1}) - \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_j)\right)\left(\hat{\phi}_1(\hat{x}_{i-1}, \hat{z}_{j+1}) - \hat{\phi}_1(\hat{x}_i, \hat{z}_j)\right) \\
&+ \left(\hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_j) - \hat{f}_\sigma(\hat{x}_i, \hat{z}_{j-1})\right)\left(\hat{\phi}_1(\hat{x}_i, \hat{z}_j) - \hat{\phi}_1(\hat{x}_{i+1}, \hat{z}_{j-1})\right)\Bigg] + O\left((\Delta\hat{x})^2\right).
\end{aligned} \tag{4.32}$$

The y direction follows magnetic field lines originating on a poloidal plane. We define the grid in the y direction by the intersection points of magnetic field lines, originating from the x, z grid points, with neighbouring poloidal planes. The intersection points in general do not fall on a grid point. We use bicubic interpolation to approximate the value of the distribution function on the intersection points. The intersection of the magnetic field line with neighbouring poloidal planes and the bicubic interpolation is visualised in Fig. 4.3. The intersection points are calculated during the initialisation step of `GENE-X` by solving Eqs. (3.75), (3.76), (3.77) and (3.78). As the length of the magnetic field line differs between different positions on the poloidal plane, the grid in y direction is non-uniform. Similar to Eq. (4.31), we discretise the first derivatives in the y direction with a fourth order centred finite difference scheme. We construct the scheme on the non-uniform y grid with the help of Lagrange polynomials. Let $(\hat{y}_{-2}, \hat{y}_{-1}, \hat{y}_0, \hat{y}_1, \hat{y}_2)$ describe the grid points in y direction local to a poloidal plane and $(\hat{f}_\sigma(\hat{x}_i, \hat{y}_{-2}, \hat{z}_j), \hat{f}_\sigma(\hat{x}_i, \hat{y}_{-1}, \hat{z}_j), \hat{f}_\sigma(\hat{x}_i, 0, \hat{z}_j), \hat{f}_\sigma(\hat{x}_i, \hat{y}_1, \hat{z}_j), \hat{f}_\sigma(\hat{x}_i, \hat{y}_2, \hat{z}_j))$ the corresponding interpolated values of the distribution function. The polynomial interpo-

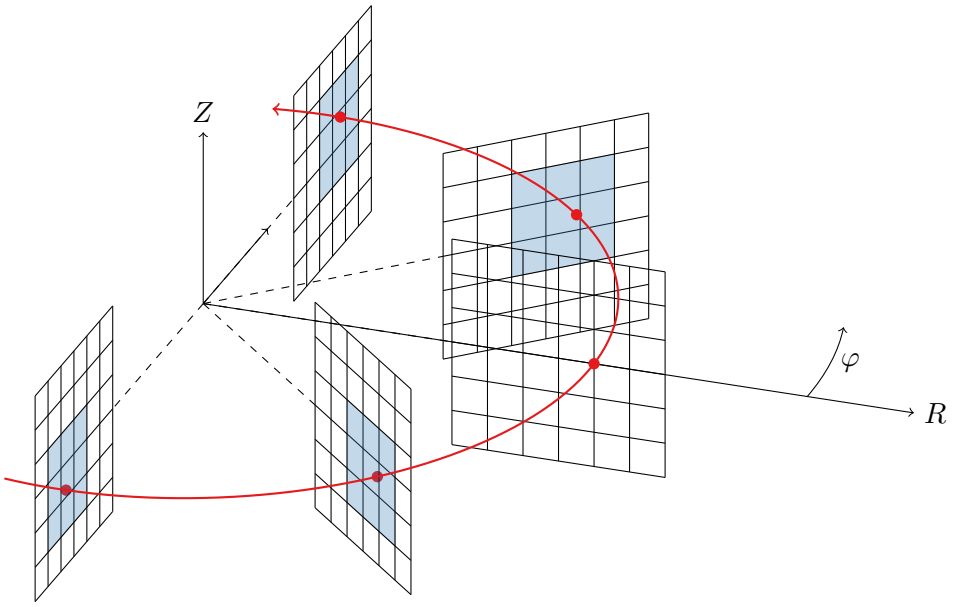

Figure 4.3.: Figure displaying a magnetic field line originating at the poloidal plane at $\varphi = 0$ and hitting different poloidal planes at $\varphi \in \{-\pi/2, -\pi/4, \pi/4, \pi/2\}$. The points where the field line intersects a poloidal plane are denoted by red dots. The value of the distribution function is obtained via bicubic interpolation around the intersection point. The area included in the interpolation on every poloidal plane is shaded in blue.

lation of the distribution function along the field line is given by

$$\hat{f}_{\sigma,\mathrm{int}}(\hat{x}_i, \hat{y}, \hat{z}_j) = \sum_{k=-2}^{2} \hat{f}_\sigma(\hat{x}_i, \hat{y}_k, \hat{z}_j)\ell_k(\hat{y})\,, \tag{4.33}$$

where ℓ_k denotes the Lagrange polynomial

$$\ell_k(\hat{y}) = \prod_{\substack{-2\leq m\leq 2 \\ m\neq k}} \frac{\hat{y} - \hat{y}_m}{\hat{y}_k - \hat{y}_m}\,. \tag{4.34}$$

The first derivative in $\hat{y}$ direction is discretised as

$$\begin{aligned}\left(\frac{\partial \hat{f}_\sigma}{\partial \hat{y}}\right)(\hat{x}_i, 0, \hat{z}_j) = \sum_{k=-2}^{2} \hat{f}_\sigma(\hat{x}_i, \hat{y}_k, \hat{z}_j) \left(\frac{\mathrm{d}\ell_k}{\mathrm{d}\hat{y}}\right)(0) \\ + O\left((\Delta\hat{y})^4\right) + O\left(\frac{(\Delta\hat{x})^3}{\Delta\hat{y}}\right) + O\left(\frac{(\Delta\hat{z})^3}{\Delta\hat{y}}\right)\,.\end{aligned} \tag{4.35}$$

The derivative of the Lagrange polynomials can be evaluated analytically and stored in an array for every grid point in $\hat{x}$, $\hat{z}$ at initialisation of the code such that the evaluation of Eq. (4.35) is fast during time stepping. The explicit form of the derivatives of the Lagrange polynomials is given in Appendix A.1.

Spatial discretisation of the field equations

The quasi-neutrality equation, Ampère's law and the generalised Ohm's law can be written in the general form

$$-\hat{\xi}\left(\frac{1}{\hat{\mathcal{J}}_{\mathrm{cyl}}}\frac{\partial}{\partial\hat{x}}\left(\hat{\mathcal{J}}_{\mathrm{cyl}}\hat{c}\frac{\partial\hat{\phi}_1}{\partial\hat{x}}\right)+\frac{\partial}{\partial\hat{z}}\left(\hat{c}\frac{\partial\hat{\phi}_1}{\partial\hat{z}}\right)\right)+\hat{\lambda}\hat{\phi}_1=\hat{b}\,, \tag{4.36}$$

with the coefficients ξ, c, λ and b. The latter of the two coefficients consist out of velocity space integrals over the distribution function. The velocity space integrals are discretised as

$$\int\hat{f}_\sigma\,\mathrm{d}\hat{v}_\parallel\,\mathrm{d}\hat{\mu}\approx\sum_{l=1}^{N_{\hat{v}_\parallel}}w_l\sum_{m=1}^{N_{\hat{\mu}}}w_m\hat{f}_\sigma(\hat{v}_{\parallel l},\hat{\mu}_m)\,, \tag{4.37}$$

with the weights w_l and w_m for the $v_\parallel$ and μ integration respectively. For the weights and the μ grid different options exist. In the $v_\parallel$ direction the grid is, as discussed above, Cartesian and integration weights corresponding to trapezoidal and Simpson integration are available. The trapezoidal weights read

$$w_l=\frac{\Delta\hat{v}_\parallel}{2}\,(1,2,\ldots,2,1)) \tag{4.38}$$

and the Simpson weights read [129, Section 4.1]

$$w_l=\frac{\Delta\hat{v}_\parallel}{48}(17,59,43,49,48,\ldots,48,49,43,59,17)\,. \tag{4.39}$$

In the μ direction also two options are available. The first option is a quadratic grid in μ

$$\hat{\mu}_m=\left((m-1)+\frac{1}{2}\right)^2\frac{L_{\hat{\mu}}}{N_{\hat{\mu}}^2}\,, \tag{4.40}$$

with trapezoidal weights w_m according to

$$w_m=\begin{cases}\frac{1}{2}\,(\hat{\mu}_2-\hat{\mu}_1) & \text{for}\quad m=1\\ \frac{1}{2}\,(\hat{\mu}_N-\hat{\mu}_{N-1}) & \text{for}\quad m=N_{\hat{\mu}}\\ \frac{1}{2}\,(\hat{\mu}_{m+1}-\hat{\mu}_{m-1}) & \text{otherwise}\end{cases}\,. \tag{4.41}$$

The second option are grid points and weights implementing Gaussian quadrature on the interval $[0,\infty)$. As there is no differentiation in μ, the grid can be optimized regarding numerical integration. The grid nodes $\hat{\mu}_m$ are formed by the roots of the Laguerre polynomials L_m and the integration weights are given by [130]

$$w_m=\frac{\hat{\mu}_m e^{\hat{\mu}_m}}{(m+1)^2L_{m+1}(\hat{\mu}_m)^2}\,. \tag{4.42}$$

The default options are Simpson quadrature in $v_\parallel$ and Gauss-Laguerre quadrature in μ. Equipped with a discrete expression for λ and b, we discretise the left-hand side of Eq. (4.36)

with second order centred finite differences

$$
\begin{aligned}
-\hat{\xi}\left(\frac{1}{\hat{\mathcal{J}}_{\mathrm{cyl}}}\frac{\partial}{\partial\hat{x}}\left(\hat{\mathcal{J}}_{\mathrm{cyl}}\hat{c}\frac{\partial\hat{\phi}_1}{\partial\hat{x}}\right)+\frac{\partial}{\partial\hat{z}}\left(\hat{c}\frac{\partial\hat{\phi}_1}{\partial\hat{z}}\right)\right)+\hat{\lambda}\hat{\phi}_1 = \\
&-\frac{\hat{\xi}(\hat{x}_i,\hat{z}_j)}{\hat{\mathcal{J}}_{\mathrm{cyl},i}\Delta\hat{x}}\left[\left(\frac{\hat{\mathcal{J}}_{\mathrm{cyl},i+1}\hat{c}(\hat{x}_{i+1},\hat{z}_j)+\hat{\mathcal{J}}_{\mathrm{cyl},i}\hat{c}(\hat{x}_i,\hat{z}_j)}{2}\right)\left(\frac{\hat{\phi}_1(\hat{x}_{i+1},\hat{z}_j)-\hat{\phi}_1(\hat{x}_i,\hat{z}_j)}{\Delta\hat{x}}\right)\right. \\
&\qquad\left.-\left(\frac{\hat{\mathcal{J}}_{\mathrm{cyl},i}\hat{c}(\hat{x}_i,\hat{z}_j)+\hat{\mathcal{J}}_{\mathrm{cyl},i-1}\hat{c}(\hat{x}_{i-1},\hat{z}_j)}{2}\right)\left(\frac{\hat{\phi}_1(\hat{x}_i,\hat{z}_j)-\hat{\phi}_1(\hat{x}_{i-1},\hat{z}_j)}{\Delta\hat{x}}\right)\right] \\
&-\frac{\hat{\xi}(\hat{x}_i,\hat{z}_j)}{\hat{\mathcal{J}}_{\mathrm{cyl},i}\Delta\hat{z}}\left[\left(\frac{\hat{\mathcal{J}}_{\mathrm{cyl},i}\hat{c}(\hat{x}_i,\hat{z}_{j+1})+\hat{\mathcal{J}}_{\mathrm{cyl},i}\hat{c}(\hat{x}_i,\hat{z}_j)}{2}\right)\left(\frac{\hat{\phi}_1(\hat{x}_i,\hat{z}_{j+1})-\hat{\phi}_1(\hat{x}_i,\hat{z}_j)}{\Delta\hat{z}}\right)\right. \\
&\qquad\left.-\left(\frac{\hat{\mathcal{J}}_{\mathrm{cyl},i}\hat{c}(\hat{x}_i,\hat{z}_j)+\hat{\mathcal{J}}_{\mathrm{cyl},i}\hat{c}(\hat{x}_i,\hat{z}_{j-1})}{2}\right)\left(\frac{\hat{\phi}_1(\hat{x}_i,\hat{z}_j)-\hat{\phi}_1(\hat{x}_i,\hat{z}_{j-1})}{\Delta\hat{z}}\right)\right] \\
&+\hat{\lambda}(\hat{x}_i,\hat{z}_i)\hat{\phi}_1(\hat{x}_i,\hat{z}_i) \\
&+O\left((\Delta\hat{x})^2\right)+O\left((\Delta\hat{z})^2\right) \qquad (4.43)
\end{aligned}
$$

Due to the elliptic operator and the Arakawa scheme, the spatial discretisation is of order two.

4.2.2. Temporal discretisation

We solve the remaining initial value problem, presented in Eq. (4.24), with an explicit Runge-Kutta method of order four [122, pp. 132–140] specified by the Butcher tableau

$$
\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array} \quad . \qquad (4.44)
$$

According to [131], the fourth order Runge-Kutta method combined with centred finite differences of fourth order, as introduced above, is a stable and efficient scheme to solve multi-dimensional linear advection equations. The timestep remains constant throughout the simulation. The size of the timestep is typically limited by either the Courant-Friedrichs-Lewy condition of the parallel electron advection or the frequency of kinetic shear Alvén waves [132, 133]. The timestep restriction due to the parallel electron advection can be estimated as [134, Section II 1.4]

$$
\Delta\hat{t} \lesssim 2.82\sqrt{\frac{\hat{m}_e}{2\hat{T}_{0\sigma}(\psi_0)}}\frac{\Delta\hat{y}_{\min}}{\hat{v}_{\parallel\max}} \, . \qquad (4.45)
$$

4.3. Numerical diffusion

The discretisation scheme presented in the last section is almost free of numerical diffusion. This is desirable because it mimics the property that the gyrokinetic Vlasov-Maxwell system is purely advective and free of diffusion on the continuous level. However, numerical diffusion is useful to stabilise the simulations and remove spurious effects caused by the discretisation scheme. As mentioned above, the chosen numerical scheme can integrate multi-dimensional linear advection problems stably and efficiently. Nevertheless, the gyrokinetic Vlasov-Maxwell system solved by the `GENE-X` code is significantly more complicated than a multi-dimensional linear advection problem and involves bicubic interpolation as well as the evaluation of integrals. Instabilities are typically caused by modes with a high wavenumber such that the scale of their oscillation is close to the spacing of the numerical grid. High wavenumber modes can be damped with the help of a diffusion operator. We explain the effect of the diffusion operator without loss of generality at the example of the parallel y direction. A general diffusion operator of order n, in the direction y and with strength ϵ is given by [135]

$$D_n = (-i)^n \epsilon \, (\Delta \hat{y})^n \, \frac{\partial^n}{\partial \hat{y}^n} \, . \tag{4.46}$$

Applying the diffusion operator D_n to a Fourier mode in y yields

$$D_n e^{i\hat{k}_{\hat{y}}\hat{y}} = -\epsilon \hat{k}_{\hat{y}}^n e^{i\hat{k}_{\hat{y}}\hat{y}} \, . \tag{4.47}$$

As can be seen from Eq. (4.47), the diffusion operator reproduces the mode with a negative prefactor resulting in effective damping. The damping strength ϵk_y^n is stronger for high k_y and weaker for low k_y modes. Therefore, the high frequency waves are efficiently damped while low frequency waves remain relatively unperturbed. The higher the derivative in the diffusion operator, the better the damping of high frequency and the preservation of low frequency modes. In numerical simulations a compromise between the order of the derivative and the cost of the evaluation must be found. With a centred finite difference scheme the evaluation of the second, fourth, sixth and eighth derivative to second order accuracy requires a stencil of size three, five, seven and nine respectively. As explained in the last section, we use centred finite difference stencils of order four with a stencil size of five to evaluate the first derivative. Similarly, we use a five-point stencil, comprising the fourth derivative evaluated up to second order accuracy, for the numerical diffusion operator.

Unstable, high wavenumber modes appear in all spatial and in the $v_\parallel$ direction. In the x, z direction the main driver of grid scale oscillations is caused by Runge's phenomenon due to the bicubic interpolation used to calculate parallel derivatives. We use the two-dimensional

biharmonic operator discretised as [136]

$$\begin{aligned}\epsilon_{\hat{x},\hat{z}}\hat{\Delta}\hat{x}^4\Delta_{\hat{x},\hat{z}}^2\hat{f}_\sigma = \epsilon_{\hat{x},\hat{z}}\Big(&\quad \hat{f}_\sigma(\hat{x}_{i-2},\hat{z}_j) + \hat{f}_\sigma(\hat{x}_i,\hat{z}_{j-2}) + \hat{f}_\sigma(\hat{x}_i,\hat{z}_{j+2}) + \hat{f}_\sigma(\hat{x}_{i+2},\hat{z}_j) \\ &+ 2\left(\hat{f}_\sigma(\hat{x}_{i-1},\hat{z}_{j-1}) + \hat{f}_\sigma(\hat{x}_{i-1},\hat{z}_{j+1}) + \hat{f}_\sigma(\hat{x}_{i+1},\hat{z}_{j-1}) + \hat{f}_\sigma(\hat{x}_{i+1},\hat{z}_{j+1})\right) \\ &- 8\left(\hat{f}_\sigma(\hat{x}_{i-1},\hat{z}_j) + \hat{f}_\sigma(\hat{x}_{i+1},\hat{z}_j) + \hat{f}_\sigma(\hat{x}_i,\hat{z}_{j-1}) + \hat{f}_\sigma(\hat{x}_i,\hat{z}_{j+1})\right) \\ &+ 20\hat{f}_\sigma(\hat{x}_i,\hat{z}_j)\Big) + O(\Delta\hat{x}^2) \end{aligned} \tag{4.48}$$

to smooth the interpolation. For the diffusion coefficient $\epsilon_{\hat{x},\hat{z}}$ we use values between $\epsilon_{\hat{x},\hat{z}} = 1$ and $\epsilon_{\hat{x},\hat{z}} = 5$ in typical simulations.

In the parallel direction the discretisation scheme destabilises high wavenumber modes that are stable in the non-discrete system [135]. The diffusion operator in the y direction is discretised, similarly to Eq. (4.35), as

$$\begin{aligned}\left(\frac{\partial^4\hat{f}_\sigma}{\partial\hat{y}^4}\right)(\hat{x}_i,0,\hat{z}_j) = &\sum_{k=-2}^{2}\hat{f}_\sigma(\hat{x}_i,\hat{y}_k,\hat{z}_j)\left(\frac{\mathrm{d}^4\ell_k}{\mathrm{d}\hat{y}^4}\right)(0) \\ &+ O\left((\Delta\hat{y})^4\right) + O\left(\frac{(\Delta\hat{x})^3}{\Delta\hat{y}}\right) + O\left(\frac{(\Delta\hat{z})^3}{\Delta\hat{y}}\right). \end{aligned} \tag{4.49}$$

The explicit form of the derivatives of the Lagrange polynomials is given in Appendix A.1. Due to the non-uniform grid in y direction, the grid spacing is not multiplied to the operator as in Eq. (4.48). Therefore, the diffusion coefficient must be adapted when the number of poloidal planes is increased. In typical simulations with 32 poloidal planes we choose values between $\epsilon_{\hat{y}} = 5\cdot10^{-3}$ and $\epsilon_{\hat{y}} = 1\cdot10^{-2}$.

Finally, we introduce numerical diffusion in the $v_\parallel$ direction. Solutions of the Vlasov equation develop filament structures superimposed over a smooth part of the distribution function over time [137]. Similar to studying Fourier modes on periodic domains, the mode structure on the domain $(-\infty,\infty)$ in $v_\parallel$ can be studied by expanding the distribution function in Hermite polynomials. The filamentation process causes excitations of higher and higher mode numbers. When the largest mode in the discrete system gets excited, the truncation acts as a reflecting boundary and the perturbation travels back towards lower mode numbers [138]. As a result, recurrence occurs and the distribution function is restored to a state close to the initial condition. Recurrence can be prevented by introducing an entropy producing, diffusive term that prevents the filamentation of the velocity space to scales that are not resolved by the grid resolution [135]. We discretise the numerical diffusion operator in $v_\parallel$ with centred finite differences as

$$\begin{aligned}\epsilon_{\hat{v}_\parallel}\Delta\hat{v}_\parallel^4\left(\frac{\partial\hat{f}_\sigma}{\partial\hat{v}_\parallel}\right)(\hat{v}_{\parallel l}) = &\epsilon_{v_\parallel}\left(\hat{f}_\sigma(\hat{v}_{\parallel l-2}) - 4\hat{f}_\sigma(\hat{v}_{\parallel l-1}) + 6\hat{f}_\sigma(\hat{v}_{\parallel l}) - 4\hat{f}_\sigma(\hat{v}_{\parallel l+1}) + \hat{f}_\sigma(\hat{v}_{\parallel l+2})\right) \\ &+ O\left((\Delta\hat{v}_\parallel)^2\right). \end{aligned} \tag{4.50}$$

According to the analysis presented in [135], we typically choose a diffusion coefficient of $\epsilon_{\hat{v}_\parallel} = 0.0125$.

4.4. Boundary conditions

For the simulations presented in this work we use Dirichlet boundary conditions. We set the distribution function to zero on the $v_\parallel$ boundary and equal to the initial distribution $f_{0\sigma}$ on the x, z boundary. In the y direction we apply the same Dirichlet boundary condition when a magnetic field line hits the x, z boundary in between two planes. In μ there is no differentiation and therefore no boundary conditions is needed. We apply the Dirichlet boundary condition for both the advective inflow and outflow through the boundary. As the Vlasov equation is advective, only a boundary condition for the inflow is needed. To avoid spurious effects due to the boundary condition for the advective outflow, we include the option to apply diffusion in a small buffer zone around the x, z boundary[†]. The diffusion operator is discretised with a second order finite difference scheme

$$-\epsilon_{\text{buffer}}(\hat{x}_i, \hat{z}_j)\,(\Delta\hat{x})^2\,\Delta_{\hat{x},\hat{z}}\hat{f}_\sigma = -\epsilon_{\text{buffer}}(\hat{x}_i, \hat{z}_j)\Big(\quad \hat{f}_\sigma(\hat{x}_{i-1}, \hat{z}_j) + \hat{f}_\sigma(\hat{x}_{i+1}, \hat{z}_j) + \hat{f}_\sigma(\hat{x}_i, \hat{z}_{j-1}) + \hat{f}_\sigma(\hat{x}_i, \hat{z}_{j+1}) - 4\hat{f}_\sigma(\hat{x}_i, \hat{z}_j)\Big) + O(\Delta\hat{x}^2)\,. \tag{4.51}$$

We set the strength of the diffusion in the buffer zone as large as the timestep of the simulation permits. The electrostatic potential ϕ_1 as well as the parallel component of the magnetic vector potential $A_{1\parallel}$ are set to zero on the x, z boundary. This corresponds to a grounded, perfectly conducting wall.

4.5. Initial conditions

In order to study the behaviour and evolution of turbulence, a suitable initial condition for the simulation must be selected. The initial condition should fulfil two properties. First, it should be close to experimental input parameters, which are usually present in the form of density and temperature profiles for different species. Second, it should be an equilibrium of the gyrokinetic Vlasov-Maxwell system. If the initial condition fulfils both properties, instabilities can be triggered by adding a small perturbation to the initial condition. These instabilities grow exponentially and develop turbulence. The property that the initial condition without the added perturbation is an exact equilibrium of the gyrokinetic Vlasov-Maxwell system is crucial for the development of turbulence. If this property is not met, dynamics generated from the initial condition are present from the start and might hide, alter or even prevent the generation of turbulence in the system.

We generate the initial condition with the help of constants of motion of the particle trajectories. The kinetic energy of a particle, $E = 1/2 m_\sigma v_\parallel^2 + \mu B$, and the magnetic moment μ are conserved. The conservation of μ can be broken, if additional terms like a collision operator are added to the gyrokinetic Vlasov-Maxwell system. The presence of additional conserved quantities is dependent on the magnetic geometry used. In the following, we discuss suitable

[†]By adding a diffusion term, the Vlasov equation turns from a hyperbolic to a parabolic problem. For the parabolic problem Dirichlet boundary conditions can be set consistently for the in- and outflow boundary.

initial conditions for different geometries. The first is a slab geometry, where the magnetic field is given by

$$\hat{\boldsymbol{B}} = \hat{B}\boldsymbol{e}_{\varphi} \tag{4.52}$$

and the second is a circular geometry with concentric flux surfaces, where the magnetic field is given by

$$\hat{\boldsymbol{B}} = \hat{B}\left(\boldsymbol{e}_{\varphi} + \frac{\hat{r}}{q(\hat{r})}\boldsymbol{e}_{\theta}\right). \tag{4.53}$$

The function $q(\hat{r})$ denotes the q profile. For both the slab and circular geometry we use a large aspect ratio assumption such that φ represents a Cartesian coordinate labelling poloidal planes. In slab and circular geometry the gradient and curvature drift of a particle are confined to a flux surface. Therefore, the poloidal flux function ψ_p is a constant of motion. The poloidal flux function is converted into a flux surface label ψ by normalising it with the value of the poloidal flux function at the separatrix ψ_{sep} and the magnetic axis ψ_{ax}. In addition, we take the square root of the normalised result

$$\psi(\hat{\psi}_p) := \begin{cases} \sqrt{\frac{\hat{\psi}_p - \hat{\psi}_{\text{ax}}}{\hat{\psi}_{\text{sep}} - \hat{\psi}_{\text{ax}}}} & \text{for } \hat{\psi}_p > 0 \\ 0 & \text{for } \hat{\psi} \leq 0 \end{cases}. \tag{4.54}$$

The case distinction is needed to account for negative values of the normalised flux surface label. In slab and circular geometry we initialise the simulation with local Maxwellians with given normalised temperature and density profiles $\hat{T}_{p\sigma}$ and $\hat{n}_{p\sigma}$, defined in Section 4.1.2,

$$\hat{F}_{0\sigma}(\psi(\psi_p), \hat{v}_{\parallel}, \mu) = \frac{\hat{n}_{p\sigma}(\psi(\psi_p))}{\left(\pi \hat{T}_{p\sigma}(\psi(\psi_p))\right)^{3/2}} e^{-\frac{\hat{v}_{\parallel}^2 + \hat{\mu}\hat{B}}{\hat{T}_{p\sigma}(\psi(\psi_p))}}. \tag{4.55}$$

For toroidal geometries with a background magnetic field $\hat{B} \propto 1/\hat{R} = 1/(\hat{R}_0 + \hat{r}\cos(\theta))$ the $\boldsymbol{\nabla}\hat{B}$ drift points out of a flux surface and, consequently, the flux surface label is no constant of motion anymore. For the following discussion we assume that the equilibrium is axisymmetric, that the model contains no electromagnetic fluctuations and that the initial condition is free of an electrostatic potential. We discuss the implications of the assumptions at the end of this section. Under the assumption of axisymmetry and the absence of an electromagnetic vector potential, the canonical toroidal momentum, defined in Section 2.3.6, is conserved. Following [139, 140, 42], we define a modified poloidal flux function with the help of the canonical toroidal momentum. The corrected flux function is given by Eq. (4.21) and in normalised units by Eq. (4.22). The so-called canonical Maxwellian is obtained by replacing the poloidal flux function $\hat{\psi}_p$ with the corrected poloidal flux function $\hat{\psi}_C$, i.e.

$$\hat{F}_{C\sigma} := \hat{F}_{0\sigma}(\psi(\hat{\psi}_C), \hat{v}_{\parallel}, \mu). \tag{4.56}$$

The canonical Maxwellian is a function of constants of motion and therefore conserved. Due to the $\sqrt{2m_\sigma}$ dependence of Eq. (4.22), the correction to the local Maxwellian is by a factor

of 60 larger for deuterium than for electrons.

The corrected flux function ψ_C has an explicit dependence on $v_\parallel$ and therefore introduces a mean flow, similar to the Pfirsch-Schlüter current [8], in parallel direction to the magnetic field. This flow has been investigated in detail in previous studies [139, 140] and has been mentioned to be problematic for the onset and the properties of turbulence. To overcome the flow, a correction to ψ_C has been defined that is a constant of motion for passing particles. It reads [139, 140]

$$\psi_{C,s} = \psi_C + \frac{m_\sigma c}{q_\sigma} R_0 \overline{v}_\parallel \,, \tag{4.57}$$

with

$$\overline{v}_\parallel = \mathrm{sign}(v_\parallel)\sqrt{2\left(E - \mu B_{\max}\right)}\mathcal{H}(E - \mu B_{\max}) \,, \tag{4.58}$$

and in normalised units

$$\hat{\psi}_{C,s} = \hat{\psi}_C + \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\sqrt{2\hat{m}_\sigma \hat{T}_{0\sigma}(\psi_0)}\,\mathrm{sign}(\hat{v}_\parallel)\sqrt{\hat{E} - \hat{\mu}\hat{B}_{\max}}\mathcal{H}(\hat{E} - \hat{\mu}\hat{B}_{\max}) \,. \tag{4.59}$$

In the following, we call the canonical Maxwellian constructed with $\hat{\psi}_{C,s}$ instead of $\hat{\psi}_C$ shifted canonical Maxwellian. To visualise the difference between the local Maxwellian, canonical Maxwellian and shifted canonical Maxwellian, we display the ion distribution function for all three Maxwellians, for the cyclone base case setup described in Section 6.5, in Fig. 4.4. The distribution function is shown for the first $\hat{\mu} = 0.044$ grid point and at the outboard mid-plane at $\psi = 0.62$. The shape of the canonical Maxwellian is different from the local Maxwellian. Due to the sign function in Eq. (4.58), the shifted canonical Maxwellian is non-differentiable close to $\hat{v}_\parallel = 0$. This can cause stability and convergence problems based on the discretisation scheme chosen. The density at the outboard mid-plane, resulting from an initialisation of the distribution function with a given density profile $\hat{n}_{p\sigma}$, is shown for the different Maxwellians in Fig. 4.5.

Initialising ions and electrons with canonical Maxwellians results in good conservation of the initial condition if the assumptions above are met. Particularly, the electrostatic potential and the induced electromagnetic vector potential need to vanish. It has also been shown that initialising the ions with a canonical Maxwellian works well in the limit of an adiabatic electron response [139].

With both kinetic ions and electrons, the situation changes. It is not possible to initialise both ions and electrons with canonical Maxwellians and achieve exact quasi-neutrality at the same time. As mentioned above, electrons are lighter than ions and therefore their canonical Maxwellian is close to a local Maxwellian. As shown in Fig. 4.5, this results in different density profiles which generate a non-zero charge density. Further, the canonical Maxwellian is not constant on magnetic flux surfaces due to the explicit dependency on b_φ. The variation of the canonical Maxwellian on a flux surface is again different for ions and electrons. We have tried different approaches to solve this problem.

1. The canonical Maxwellian for electrons is close to a local Maxwellian. Therefore, it is reasonable to initialise the ions with a canonical Maxwellian, calculate the resulting ion

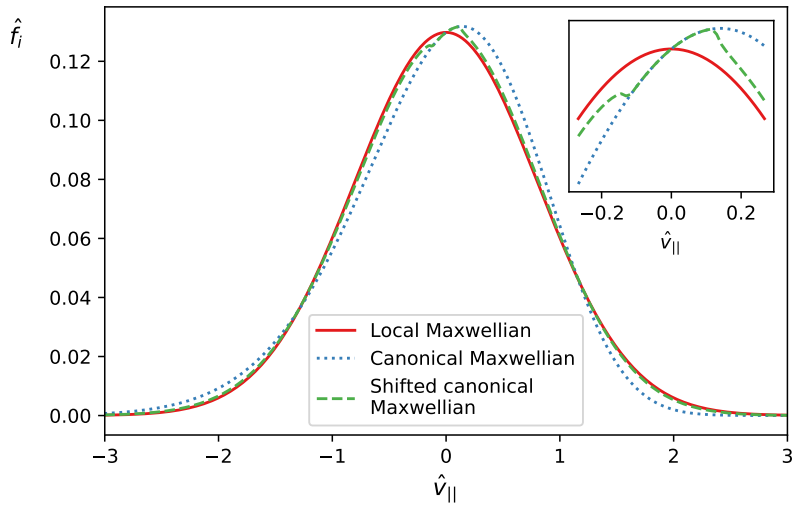

Figure 4.4.: Figure displaying a slice of the ion distribution initialised with a local Maxwellian, a canonical Maxwellian and a shifted canonical Maxwellian for the cyclone base case configuration described in Section 6.5. The distribution function is shown for the first grid point in $\hat{\mu} = 0.044$ and at the outboard mid-plane at $\psi = 0.62$. Due to the sign function in Eq. (4.58), the shifted canonical Maxwellian is non-differentiable close to $\hat{v}_{||} = 0$.

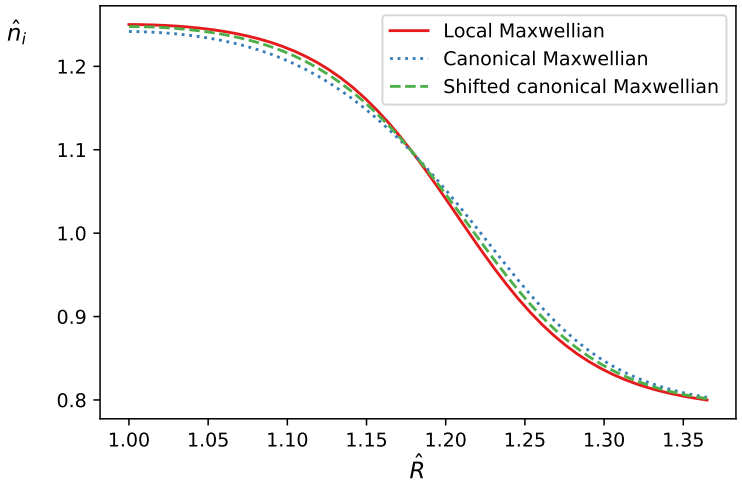

Figure 4.5.: Figure displaying the ion density profile calculated from an ion distribution function initialised with a local Maxwellian, a canonical Maxwellian and a shifted canonical Maxwellian for the cyclone base configuration described in Section 6.5. The density of both the canonical and shifted canonical Maxwellian differ from the local Maxwellian. The shifted canonical Maxwellian is closer to the local Maxwellian than the canonical Maxwellian.

density and initialise the electrons with a local Maxwellian and the given ion density. While this initial state is quasi-neutral per construction, we find that the conservation of the electron distribution is so bad that quickly after initialisation large charge densities and, consequently, a radial electric field is generated.

2. In the second approach, ions and electrons are initialised with a canonical Maxwellian. As this state is not quasi-neutral, we calculate the ion density n_i and electron density n_e after loading the distribution function and correct the electron distribution function by the factor n_i/n_e. This ensures exact quasi-neutrality of the initial condition because ion and electron densities are equal. Nevertheless, the correction breaks the conservation of the initial electron distribution. The correction factor n_i/n_e is a function of the coordinates R, Z, which are no constants of motion. If the correction factor n_i/n_e is small enough, the charge density generated by the electron dynamics is small and turbulence can develop freely. We find that this is not the case for realistic profiles. The reason is simple. As shown in Fig. 4.4 and Fig. 4.5, the distribution function of the ions is significantly different from a local Maxwellian. Consequently, the ion density is different from the initial profile and different from the electron density. This results in a large difference between ion and electron density and a large correction factor n_i/n_e, which effectively breaks the conservation.

3. The third approach proceeds similarly to the second one with the only difference that the shifted instead of the standard canonical Maxwellian is used. Consequently, the ion distribution function is closer to a Maxwellian and the resulting ion density profile is closer to the profile generated by a local Maxwellian. As a result, ion and electron density are approximately equal and the correction factor n_i/n_e is smaller than in the second approach. This is the standard initial condition implemented in the `GENE-X` code for toroidal equilibria. It works for a lot of profiles and geometries and allows for the stable simulation of turbulence. Nevertheless, the solution is still dependent on the right parameters.

The above analysis holds true in the limit of vanishing electromagnetic vector potential $A_{1\parallel}$. As soon as electromagnetic fluctuations are activated, the situation changes. First, the canonical toroidal momentum and the canonical Maxwellian are modified and obtain a dependence on $A_{1\parallel}$. Let us call the electromagnetic vector potential entering the canonical Maxwellian $A_{1\parallel\text{init}}$. As discussed above, the canonical Maxwellian introduces a mean flow, which leads to an electromagnetic vector potential $A_{1\parallel\text{generated}}$. The canonical Maxwellian is only conserved if $A_{1\parallel\text{init}} = A_{1\parallel\text{generated}}$, which is in general not fulfilled. Therefore, the initial condition implemented in the `GENE-X` code for toroidal equilibria is only stable for small β_{ref}. In addition, the conservation is broken in non-axisymmetric equilibria.

4.6. Conclusion and outlook

In this chapter, we presented the numerical scheme implemented in the `GENE-X` code. The numerical scheme was based on centred finite differences and motivated by the use of the locally field-aligned coordinate system, presented in Chapter 3. The chosen numerical scheme contains little numerical diffusion and therefore respects the property that the gyrokinetic Vlasov-Maxwell system is free of diffusion on the continuous level. The total scheme is of order two, although multiple operators are implemented to fourth order accuracy. Furthermore, we explained why it is difficult to construct a stable initial condition for the full-f gyrokinetic Vlasov-Maxwell system and detailed the initial condition for our simulations.
In the future, three upgrades to the numerical scheme would be advantageous. First, the elliptic operators and the nonlinear Poisson brackets should be discretised with numerical schemes that are fourth order accurate. Consequently, this would render the discretisation of the complete gyrokinetic Vlasov-Maxwell system accurate to fourth order. Second, the scheme in its current form does neither preserve the number of particles, energy or momentum to machine precision nor the positivity of the distribution function. It would be interesting to construct a conservative numerical scheme within the framework of mimetic finite differences that preserves the particle number and if possible even momentum and energy. For the inclusion of collisions into the simulations it is necessary to take roots of the moments of the distribution function. This is only well-defined if the moments remain positive. It will be important to explore schemes that preserve the positivity of the distribution function in the future. Third, it would be desirable to develop and implement a numerical solver that can calculate an initial condition for the simulations that is stable to machine precision.

5. Implementation

Our goal is to create a scientific code that is both fast and extensible. These two requisites contradict each other to a certain degree. For a code being extensible data encapsulation, well-structured interfaces and code abstraction is necessary. For execution speed it is best to work on raw data such that capabilities of modern hardware, like vectorisation, can be used efficiently. In the following chapter, we present and discuss our code design and prove that it provides excellent performance while being extensible. We start by presenting the combined project structure with the `GRILLIX` code. Next, we discuss the software engineering of `GENE-X` and introduce the data structures and algorithms, called operators. In addition, we discuss the structure of the time stepping and the main program. We validate the performance of the implementation with the help of the roofline model [141] and the measurement of strong [142] as well as weak scaling [143] up to 20480 CPU cores.

5.1. Project structure

As detailed in the last two chapters, `GENE-X` uses the FCI approach and unstructured, locally Cartesian grids. Consequently, it shares a lot of technology with the Braginskii fluid code `GRILLIX`. To strengthen the collaboration between both codes, we created a library based on `GRILLIX`, called `PARALLAX`, that contains the functionality to generate the grid for different magnetic equilibria, trace magnetic field lines using `dop853`, perform bicubic polynomial interpolation and solve elliptic equations within poloidal planes. The library is now used by both `GRILLIX` and `GENE-X` and allows us to join forces and reduce maintenance work in both codes.

5.2. Software engineering

`GENE-X` is written in Fortran 2008. We decided to choose Fortran 2008 over C++ because it combines modern software engineering concepts, like object orientation, with powerful array handling capabilities.

5.2.1. Data structures

We store the data of the distribution function as a five-dimensional and the data of the electromagnetic fields as two-dimensional contiguous arrays[†]. As all performance demand-

[†]As explained in Section 4.2, the dimensions x and z are stored as an unstructured grid in a one-dimensional array. Therefore, the distribution function is stored as a five-dimensional instead of a six-dimensional array consisting out of the unstructured grid, φ, $v_\parallel$, μ and σ. The electromagnetic fields are stored as two-dimensional

ing operations are performed on this data, it is crucial to store it in contiguous memory and such allow for vectorisation, cache line optimisation and optimisations by the compiler. The five-dimensional array is stored in the `data_storage_5d_t` class and the two-dimensional arrays in the `data_storage_2d_t` class respectively. The data is distributed over different Message Passing Interface (`MPI`) [144] processes. The exchange of ghost cells, necessary for the calculation of stencil operations, is encapsulated in the classes. The `MPI` ghost exchange is implemented via non-blocking communication and is performed automatically by calling `start_exchange` and `finish_exchange`. The arrays themselves are not encapsulated and a pointer is retrieved by calling `get_pointer`. These design decisions allow for the efficient implementation of operators acting on the fields. The array storing the data is either allocated on CPU or GPU. CPU allocation is implemented in the child class `data_storage_5d_cpu_t` and GPU allocation in `data_storage_5d_gpu_t`. GPU and CPU implementations are instantiated and chosen at runtime. The design of the data structures is depicted as an Unified Modelling Language (UML) diagram in Fig. 5.1.

5.2.2. Operators

The calculation of the right-hand side of the Vlasov equation, the solution of the field equations as well as diagnostics are implemented as operators acting on raw data. The interface of the operators is defined in a base operator class. Different realisations of the operator are implemented as children of the base class. A good example is the operator evaluating the part $(\partial f_\sigma/\partial t)^\star$, independent of explicit time derivatives, of the right-hand side of the Vlasov equation, called `op_rhs_vlasov_eq_static_t` in `GENE-X`. The class `op_rhs_vlasov_eq_static_t` defines the interface of the operator and the children `op_rhs_vlasov_eq_static_cpu_t` or `op_rhs_vlasov_eq_static_gpu_t` contain CPU or GPU implementations of the operator. This way of selecting different execution schemes is known as the *strategy* design pattern [145, pp. 315–323]. The operators receive contiguous pointers to raw data as input. As different operators require different inputs, each operator is implemented as its own class. Next to the operator for evaluating the right-hand side, there are for example operators to obtain scalar and multi-dimensional moments of the distribution function called `op_mom_0d_t` and `op_mom_2d_t`. The design of the operators is depicted as UML diagrams in Fig. 5.2.

5.2.3. The state vector and time stepping

After having introduced the data structures and operators, we describe how they interact in the code. The state of the electromagnetic model, often called state vector, is defined by the distribution function, the electrostatic potential and the parallel component of the electromagnetic vector potential. They are encapsulated in the `state_vector_t` class. With that also the raw data, accessible by the `get_pointer` method of the data storage types, is encapsulated. The operators need access to the raw data. This is achieved in an encapsulated way by passing the operator as a *visitor* [145, pp. 331–344] to the state vector by the `apply`

instead of three-dimensional arrays consisting out of the unstructured grid and φ.

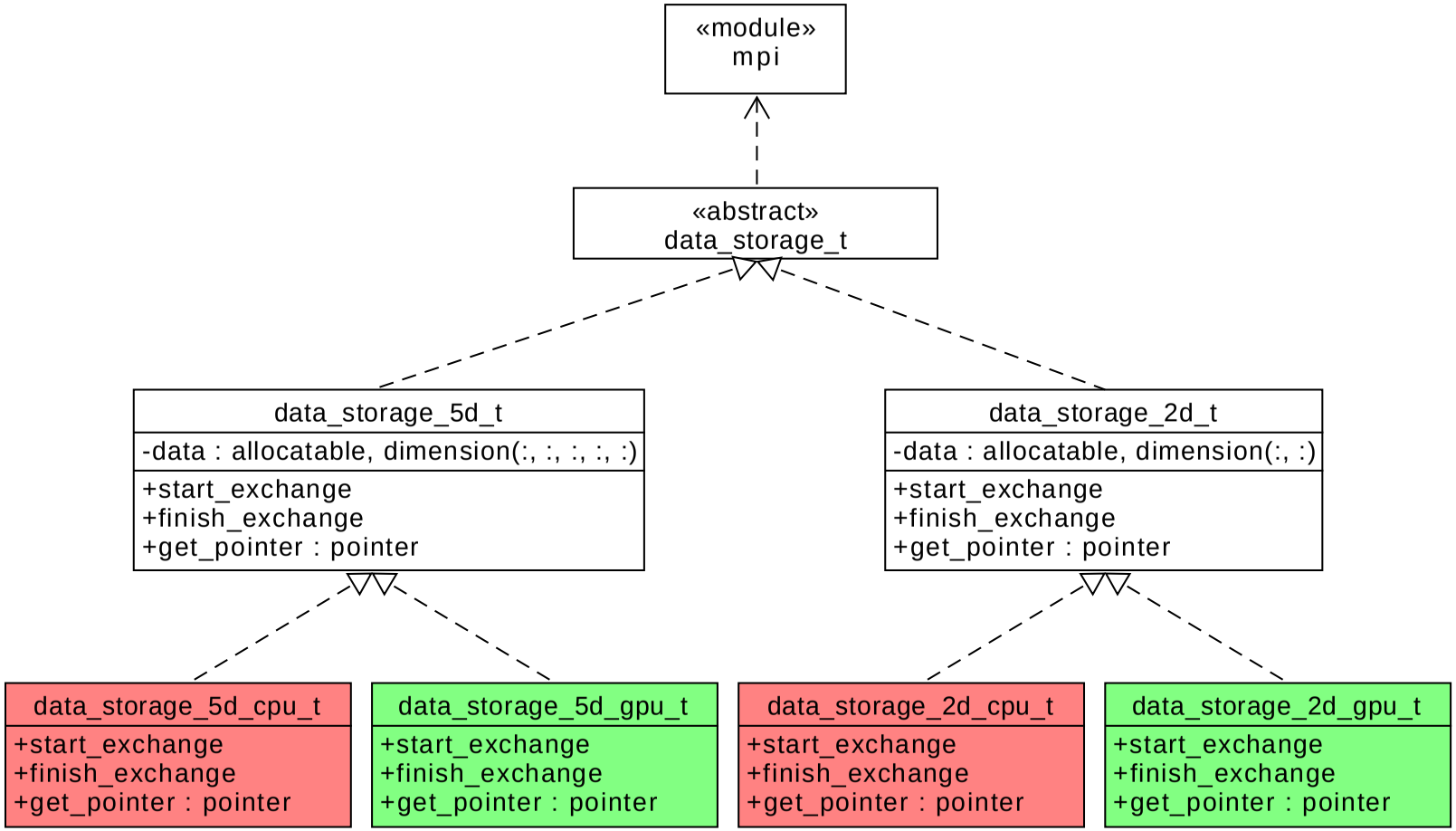

Figure 5.1.: UML diagram representing the design of the data structures in `GENE-X`. The data for the distribution function, the electrostatic potential and the parallel component of the electromagnetic vector potential are stored in 2d and 5d arrays in the classes `data_storage_5d_t` and `data_storage_2d_t`. The `MPI` ghost exchange is encapsulated and initiated with `start_exchange` and `finish_exchange`. The data is stored either on CPU or GPU.

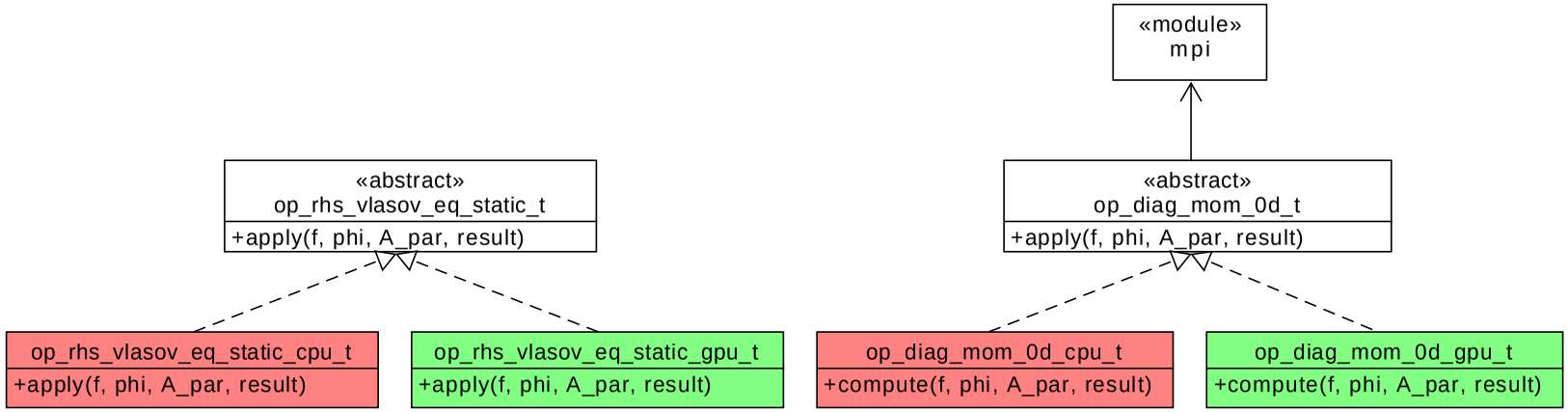

Figure 5.2.: UML diagrams representing the design of the operators in `GENE-X`. The operators are organized in classes that perform computations on contiguous pointers to raw data. The parent class is abstract and defines the interface of the operation. The children provide different implementations of the operator. Implementations differ by target, e.g. CPU or GPU, or by the numerical scheme used.

method. Within the `apply` method the operator instance has access to all private data in the state vector and can perform operations on the raw data. This design is depicted as an UML diagram in Fig. 5.3. With this software structure it is simple to extend the state vector by introducing more fields or extend operators acting on the state vector. The interface of the operators is likely to change in the future when additional phenomena, like the perpendicular component of the electromagnetic vector potential, is included in the model. As the operators are implemented as their own class, further fields and effects can be added by modifying the operator locally.

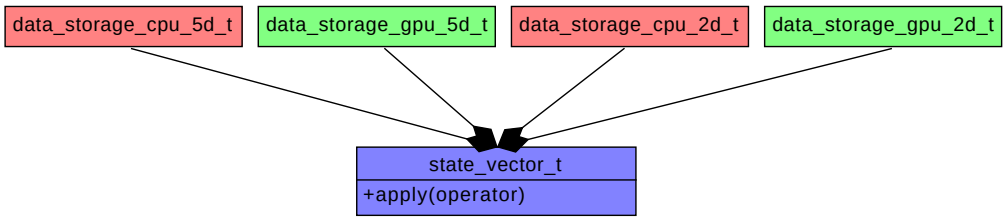

Figure 5.3.: UML diagram representing the structure of the state vector. The state vector contains multiple instances of the data storage types to store the electrostatic potential, the electromagnetic vectorpotential, the distribution function or other fields. Operators are applied to the state vector by passing them as a visitor to the `apply` method. This ensures that the raw data, stored in the data structure, remains in the state vector and is not accessible from the outside.

In order to the evolve the physical system, we need to apply operators to the state vector in a way specified by the time stepping scheme. For simulations of different models and geometries, it might be beneficial to have multiple time stepping schemes at our disposal. The interface of the time stepping is defined in the abstract `time_step_t` class. It contains one or more different state vectors, depending on the specific time stepping scheme implemented. A time step is calculated by calling `step`, diagnostics are written by calling `diagnose` and the state vector is saved to disk as a checkpoint by calling `save`. Our workhorse time stepping scheme is, as presented in Section 4.2, the fourth order Runge-Kutta method implemented as a child of `time_step_t` in `rk4_t`. The time stepping is controlled from the main program `genex`. The time stepping design is once again depicted as an UML diagram in Fig. 5.4.

5.2.4. Continuous integration

Next to the actual software design, automating and optimising the development process is an important issue for us. We employ the software engineering practice called *Continous Integration* (CI) [146] in the development process.

Due to the modular software design, presented in the last subsections, the code consists out of many independent and flexible units with well-defined interfaces which are simple to test. We test the functionality of the interfaces with the help of the Fortran unit test library `pFUnit`[147]. `pFUnit` allows to automatically execute tests for a different amount of MPI processes and OpenMP threads making the unit tests comprehensive.

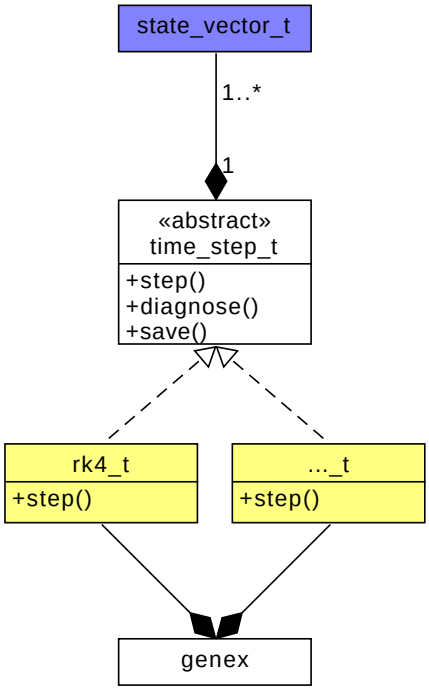

Figure 5.4.: UML diagram representing the structure of the main program. The time step is handled by the `time_step_t` class. `time_step_t` is responsible for storing one or multiple instances of the `state_vector_t` representing the state of the simulation. It evolves the simulation one time step by calling `step`, it writes diagnostics by calling `diagnose` and it saves the current state of the simulation as a checkpoint to disk by calling `save`.

The build process is automatized with `CMake` and supports the GNU as well as Intel Fortran compiler. Upon integration of new implementations in the master branch, the code is built automatically with both the GNU and Intel compiler for debug and release configurations. Furthermore, the unit tests and fast integration tests are run for all configurations. New functionality can only be incorporated into the master branch when all tests succeed.

5.3. Parallelisation and performance

The implementation is parallelised with a hybrid approach using both `OpenMP` [148] and `MPI` [144]. Currently, no GPU implementation of the operators and datastructures is available. The reason for choosing a hybrid approach is twofold.

First, modern supercomputers are heterogeneous. CPU based machines usually consist out of compute nodes with one or multiple sockets each. The nodes themselves are connected with a fast network. With a hybrid approach it is straightforward to adjust the parallelisation to the hardware needs. `OpenMP` is well suited to parallelise tasks within a node because it makes use of the shared memory architecture. `MPI` is used to exchange information over the network connecting the nodes.

Second, by separating intra- and inter-node parallelisation it is easier to optimise for the

use of accelerators in the future. For instance, `OpenMP` can be replaced with `OpenACC` [149] or a specific `CUDA` [150] implementation can be written for the datastructures and operators introduced in Section 5.2.

We use `OpenMP` to parallelise over the x, z direction within the poloidal plane. Therefore, no inter node communication is necessary during the solution of the elliptic field equations. We employ a domain decomposition with `MPI` over the $v_\parallel$, μ, φ and species dimensions.

All benchmarks are performed on the supercomputer Cobra of the Max-Planck Computing and Data Facility. All compute nodes of Cobra feature two Intel Xeon Gold 6148 processors with 20 cores and 27.5Mib L3 Cache each. The software is compiled with the Intel Fortran Compiler Version 19.1.1.

5.3.1. Node level performance

We analyse the node level performance with the roofline model [141]. The goal of the roofline model is to calculate the maximal attainable rate of floating-point operations per second (Flop/s) for a given program. Before we discuss the model, we introduce the terminology needed. The arithmetic intensity I is the quotient of floating-point operations per bytes loaded. We explain the term at the example of the `saxpy` operation

```
do i = 1, N
    y(i) = y(i) + sa * x(i)
enddo .
```

The arithmetic intensity is $I = 2/24 = 1/12 \approx 0.083$ because in each iteration there are two floating-point operations, two eight-byte loads and one eight-byte store. Further, we call the maximal memory bandwidth of the hardware B and the hardware limit of the rate of floating-point operations F. The theoretically achievable flop rate, or roofline, of a program is then given by

$$\mathrm{R} = \min\{B \cdot I, F\}\,. \tag{5.1}$$

In order to apply the roofline model to our implementation, we measure the maximum memory bandwidth with a stream benchmark provided by the `likwid` [151, 152] toolkit[†]. The measured maximum main memory bandwidth is $B = 150$Gb/s and the maximum L3 Cache memory bandwidth is $B = 660$Gb/s. For the hardware limit of the rate of floating-point operations we use the specifications by Intel. The maximum rate of scalar addition operations is $F = 190$GFlop/s and of vectorised addition operations is $F = 1400$GFlop/s. All our operators are bound by memory bandwidth such that the hardware limit of the rate of floating-point operations is not of high significance. For the `saxpy` example the roofline rate of floating-point operations evaluates to $R = 12.5$GFlop/s.

We perform the roofline analysis for the operators consuming major parts of the computation time in `GENE-X`. The operators are given in Tab. 5.1. We select a problem size with 677000 grid points in the poloidal plane, $n_\varphi = 1$, $n_{v_\parallel} = 16$, $n_\mu = 8$ that reflects a typical load for

[†]We found that the theoretical memory bandwidth specified by Intel for the Intel Xeon Gold 6148 dual socket system is not achievable in real applications.

Operator	Arithmetic intensity	Performance / (GFlop/s)
op_saxpy_cpu_t	0.083	11.9
op_lin_comb_cpu_t	0.13	14.8
op_diag_mom_0d_cpu_t	0.10	121
op_diag_mom_2d_cpu_t	0.12	163
op_rhs_vlasov_eq_static_cpu_t	0.092	28.3
op_rhs_vlasov_eq_dynamic_cpu_t	0.19	51.9

Table 5.1.: Table displaying the arithmetic intensity and the measured performance of operators consuming major parts of the computation time in GENE-X. op_saxpy_cpu_t and op_lin_comb_cpu_t are used to combine the stages of the RK4 time stepping scheme. op_diag_mom_0d_cpu_t and op_diag_mom_2d_cpu_t represent diagnostic operators with scalar and two-dimensional output. Finally, op_rhs_vlasov_eq_static_cpu_t and op_rhs_vlasov_eq_dynamic_cpu_t evaluate the right-hand side of the Vlasov equation.

one compute node after domain decomposition via MPI. The total number of grid points for one node is $173 \cdot 10^6$[†].

We measure the arithmetic intensity as well as the flop rate of the operations with Intel Advisor 2020. Intel Advisor reports the arithmetic intensity of the L1 Cache. That means all loads and stores from and to L1 cache are counted in the calculation of the arithmetic intensity. The result of the roofline analysis is shown in Fig. 5.5. The op_saxpy_cpu_t and op_lin_comb_cpu_t operators perform close to the main memory bandwidth roofline. This is expected and a nice validation of the model because there is no possibility for cache reuse. The diagnostic operators op_diag_mom_0d_cpu_t and op_diag_mom_2d_cpu_t fall above the L3 cache roofline. As they load the distribution function once to calculate six different moments a high amount of cache reuse is expected. The operators evaluating the right-hand side of the gyrokinetic Vlasov equation, op_rhs_vlasov_eq_static_cpu_t and op_rhs_vlasov_eq_dynamic_cpu_t, fall in between the DRAM and L3 cache roofline indicating a decent amount of cache reuse. It is difficult to estimate the maximal theoretical performance of these more sophisticated operators because the cache reuse is unknown.

5.3.2. Strong scaling

After discussing the intra-node performance, we proceed by testing the MPI scaling of the implementation. We start with the strong scaling that was first described by Amdahl [142]. In strong scaling a problem with a fixed size is selected and the speedup of the execution time is measured as a function of the number of compute nodes [153]. We use the AUG example, presented in Chapter 7, with a resolution of $\Delta x = \Delta z = 1.91 \cdot 10^{-4}$mm resulting in 223000 grid points in the poloidal plane, $n_\varphi = 24$, $n_{v_\parallel} = 64$ and $n_\mu = 16$. The resolution has been reduced such that we are able to fit the simulation onto 1 node. The memory usage with 1 node is approximately 700GB exhausting the available 768GB of main memory on the fat partition on Cobra. The total number of grid points is equal to $11.0 \cdot 10^9$. We start with 1 node and scale up to 512 nodes. We first fill the μ and species communicator with nodes as

[†]The number of grid points includes a factor of two from the two species.

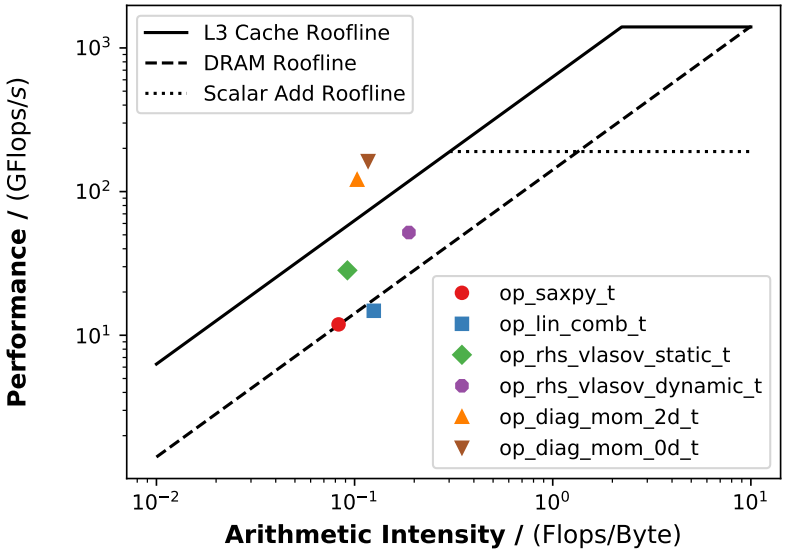

Figure 5.5.: Figure displaying the roofline analysis of the operators presented in Tab. 5.1. The operators `op_lin_comb_t` and `op_saxpy_t` are bound by the main memory bandwidth and show optimal performance. For the more involved operators, we observe a higher performance, demonstrating efficient cache reuse.

minimal `MPI` communication is needed in these directions. After they are filled, we use the φ, and $v_{\parallel}$ communicator that require a ghost exchange via `MPI`. We measure the wall clock time per time step averaged over 100 iterations. For the run with only 1 node, we average the execution time over 50 timesteps to be able to run the benchmark within the time limit of 24h imposed by our supercomputing centre. The time of the first iteration is discarded due to the presence of memory allocations and initialisations. The results are shown in Fig. 5.6. The times are given in Appendix B.1. We define the efficiency of the strong scaling by

$$\eta := \log_2 \left(\frac{t_n}{t_{2n}} \right) \tag{5.2}$$

where t_n and t_{2n} denote the time per timestep with n nodes and $2n$ nodes respectively. We measure an average reduction of execution time by a factor of 1.9 upon doubling the number of nodes. This corresponds to an efficiency of $\eta = 0.93$, demonstrating excellent strong scaling.

5.3.3. Weak scaling

We continue with the weak scaling that was first described by Gustafson [143] as a re-evaluation of Amdahl's work. In weak scaling the number of compute nodes is increased while keeping the problem size per node constant [153]. This results in larger problems the more nodes are chosen. Similar to the strong scaling, we select the problem based on the AUG simulation presented in Chapter 7. As we have more memory per node available than in the strong scaling, we choose a higher resolution of $\Delta x = \Delta z = 9.54 \cdot 10^{-4}$mm resulting in 892000 points in the poloidal plane. We start the scaling measurement with 1 node and

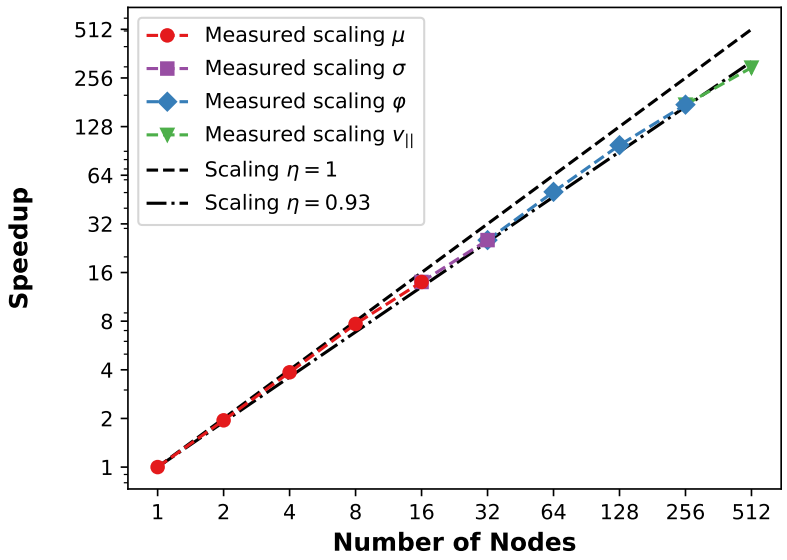

Figure 5.6.: Figure displaying the strong scaling analysis of the GENE-X code over different MPI communicators. For reference, the ideal scaling with an efficiency of $\eta = 1$ and the scaling with an efficiency of $\eta = 0.93$ is shown. On average, GENE-X shows an efficiency of $\eta = 0.93$, demonstrating excellent strong scaling.

$n_\varphi = 2$, $n_{v_\parallel} = 8$, $n_\mu = 2$ and fill the MPI communicators one after another up to 512 nodes and $n_\varphi = 16$, $n_{v_\parallel} = 64$ and $n_\mu = 16$. The number of grid points is equal to $57.1 \cdot 10^6$ per node. Again, we measure the time per timestep by averaging the execution time over 100 timesteps. The results are shown in Fig. 5.7. We observe that the time per timestep is almost constant from 1 to 512 nodes, demonstrating excellent weak scaling.

5.4. Conclusion and outlook

We started this chapter with the goal to create a scientific code that is fast and extensible. We achieved both by implementing a software design that is focused around data structures and operators, acting on raw data. Both are encapsulated in classes and implementations, specialised to different compute architectures, can be provided. We demonstrated that the implemented software design is fast by validating the intra-node performance against the roofline model and the inter-node performance by measuring the strong and weak scaling from 1 to 512 compute nodes. The strong scaling showed an efficiency of 93% and the weak scaling was nearly ideal.

Many supercomputers today feature accelerated compute nodes with GPUs [154]. To make use of these architectures, it will be important to exploit the modular code structure of GENE-X and provide a GPU implementation of the data structures and operators in the future.

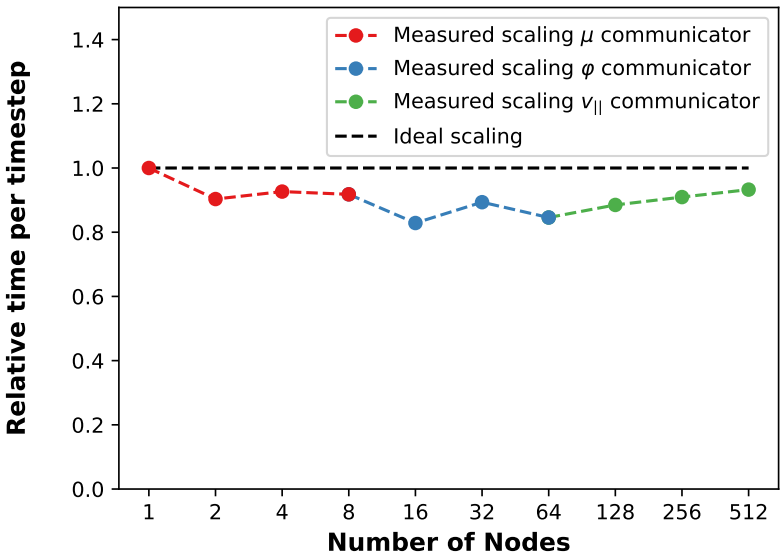

Figure 5.7.: Figure displaying the weak scaling analysis of the `GENE-X` code. The measured time per timestep is almost constant from 1 to 512 nodes, demonstrating excellent weak scaling.

6. Verification

With the help of good software engineering practices, presented in Chapter 5, we ensure that the implementation of the numerical scheme is correct. The next step is to show that the numerical scheme converges and returns approximate solutions of the gyrokinetic Vlasov-Maxwell system.

To this end, we present different tests of the `GENE-X` code in this chapter. We start with purely numerical tests and analyse the accuracy of the field line tracing, which is central for calculating derivatives in the locally field-aligned coordinate system, and perform a convergence order analysis of the whole gyrokinetic Vlasov-Maxwell system with the *Method of Manufactured Solutions* (MMS) [155]. Next, we test different physical properties of the system. Each of the physical tests probes a different aspect of the code. We start with testing the linear regime of the gyrokinetic system in the limit of no electromagnetic fluctuations, i.e. $A_{1\parallel} = 0$, by calculating and comparing ion temperature gradient growth rates in screw pinch geometry against analytical predictions. This test is well suited to compare the performance of the locally field-aligned coordinate system to standard flux-coordinates. We test the nonlinear phase by checking the conservation of energy during a turbulence simulation. We proceed with testing the electromagnetic implementation by measuring and comparing the frequency and damping rate of Alfv́en waven in slab geometry. Finally, we perform a well studied electromagnetic *Cyclone Base Case* (CBC) benchmark [156] that tests the full gyrokinetic Vlasov-Maxwell system in toroidal geometry.

6.1. Accuracy of the field line tracing

As explained in Chapter 3, the length of a magnetic field line as well as the intersection points with neighbouring poloidal planes is calculated by solving a system of ordinary differential equations. We use the algorithm `dop853` for solving the system. `dop853` is an embedded Runge-Kutta method of order eight that uses a fifth order error estimator with a third order correction [122, II. 10]. The local approximation error is controlled by providing relative and absolute error tolerances. It is kept roughly below `rtol*abs(sol)+atol` [122, Appendix], where `rtol` and `atol` represent the relative and absolute error tolerance and `sol` the value of the solution. In the following, we test the convergence and precision of the integrator for our application.

We choose a circular geometry with the magnetic field specified in Eq. (4.53) and a large aspect ratio assumption such that φ represents a Cartesian coordinate labelling poloidal planes. The q profile is given by

$$q(\hat{r}) = q_0 + \hat{s} \cdot \hat{r}. \tag{6.1}$$

For this geometry the system of equations presented in Eqs. (3.75), (3.76), (3.77) and (3.78) can be solved analytically. The position and the length of a magnetic field line starting at $(\hat{r}_0, \theta_0, \varphi_0)$, parametrised as a function of the poloidal angle φ, read

$$\hat{r}(\varphi) = \hat{r}_0 \tag{6.2}$$

$$\theta(\varphi) = \theta_0 + \frac{(\varphi - \varphi_0)}{q(\hat{r})} \tag{6.3}$$

$$\hat{s}(\varphi) = \sqrt{1 + \left(\frac{\hat{r}}{q(\hat{r})}\right)^2}\,(\varphi - \varphi_0)\,. \tag{6.4}$$

First, we analyse the accuracy of the position of the field line. We compare the coordinate $\hat{z} = \hat{Z} = \hat{R}_0 + \hat{r}\sin(\theta)$, obtained from the numerical solution with `dop853`, against the analytical solution for the angles $\varphi \in \{\pi/8, \pi, 100\pi\}$, corresponding to 1/16, 1/2 and 50 toroidal turns. For the z coordinate the absolute error

$$\epsilon = |\hat{z}_{\mathrm{dop}} - \hat{z}| \tag{6.5}$$

is measured. We set the relative tolerance of `dop853` to machine precision and scale the absolute tolerance. The results are shown in the upper image in Fig. 6.1. For $\varphi = \pi/8$ the field line tracing is precise to machine precision for every tolerance. For larger trace angles we observe a convergence with decreasing tolerance. The measured error is always larger than the provided tolerance. This is expected because the tolerance limits the local and not the global approximation error of the numerical solution.
Next, we analyse the accuracy of the calculation of the field line length. As the field line length increases with increasing trace angle φ, we measure the relative numerical error

$$\epsilon = \frac{|\hat{s}_{\mathrm{dop}} - \hat{s}|}{|\hat{s}|} \tag{6.6}$$

for the angles $\varphi \in \{\pi/8, \pi, 100\pi\}$. We set the absolute tolerance to machine precision and scale the relative tolerance. The results are shown in the lower image in Fig. 6.1. We observe a similar behaviour as for the z coordinate. For $\varphi = \pi/8$ the numerical integration is exact to machine precision. For larger trace angles we observe convergence. In the simulations presented in this work, we use 16 or more poloidal planes. The trace angle required to calculate parallel derivatives with the fourth order scheme, presented in Chapter 4, is equal to $\pi/8$ or smaller. Therefore, the accuracy of the tracing close to machine precision.

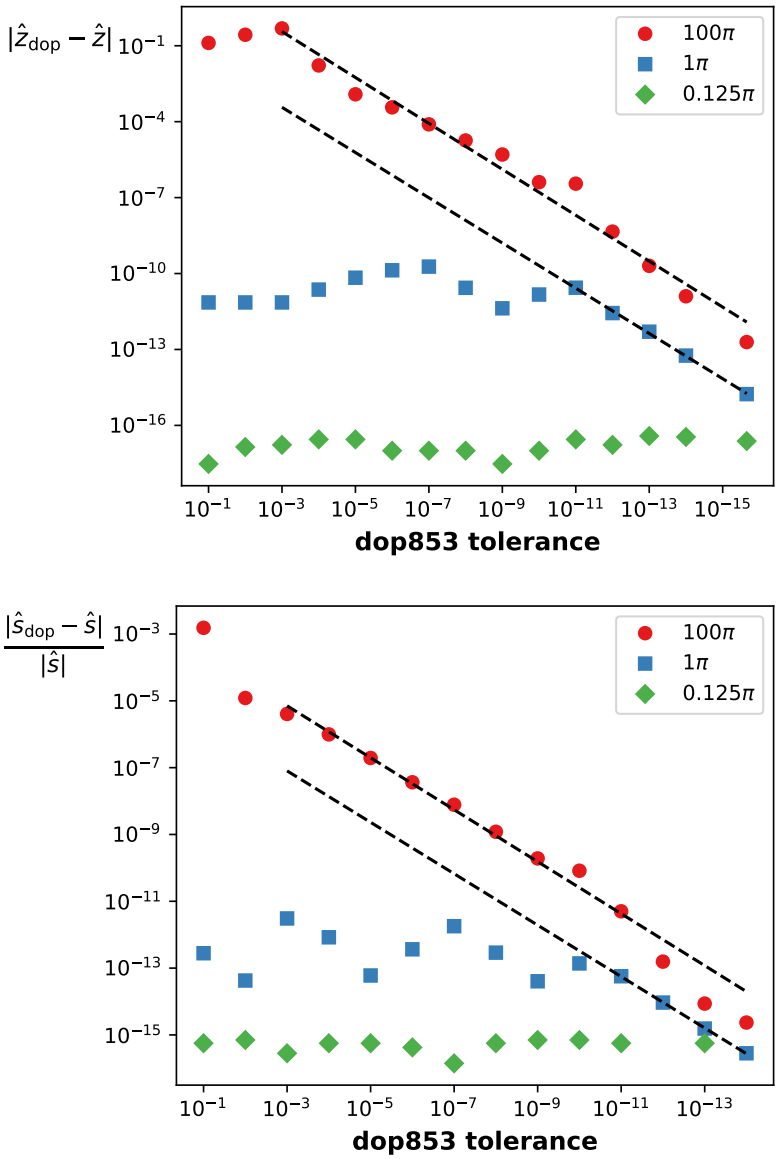

Figure 6.1.: Figure displaying the accuracy of the field line tracing with the `dop853` algorithm. We test three different trace angles $\varphi \in \{\pi/8, \pi, 100\pi\}$. Above the absolute error of the z coordinate of the field line and below the relative error of the field line length is shown. With increasing tolerance the field line tracing converges. For clarity, the convergence is highlighted with a dashed line.

6.2. Method of manufactured solutions

In this section, we present a full convergence order analysis of the numerical scheme in slab, circular and toroidal geometry with MMS. The explanation and introduction into MMS have been published in [87]. In this section, the results published in [87] are generalised from the electrostatic to the electromagnetic model, detailed in Chapter 2, with the improved numerical scheme described in Chapter 4.

The gyrokinetic Vlasov-Maxwell system is a complicated integro-partial differential system of equations and exact solutions are only known for special cases. Therefore, it is not possible to comprehensively check the convergence of the numerical solution by comparing it against exact solutions. Nevertheless, it is possible to add extra terms to the Vlasov-Maxwell system such that it obtains any solution. This is known as the method of manufactured solutions (MMS) [155]. In order to explain it in more detail, we rewrite the gyrokinetic Vlasov-Maxwell system in a different form. Consider the integro-partial differential operators $L(\phi)$, $N(f)$ and $A(f)$ such that

$$L(\phi_1, A_{1\parallel})[f] = 0 \tag{6.7}$$
$$N(f)[\phi_1] = 0 \tag{6.8}$$
$$A(f)[A_{1\parallel}] = 0 \tag{6.9}$$

represents the gyrokinetic Vlasov-Maxwell system. L represents the Vlasov operator which depends on the electrostatic potential and the parallel component of the electromagnetic vector potential, N represents the operator that forms the quasi-neutrality equation and A the operator that forms Ampère's law. In the method of manufactured solutions a modified system of equations

$$L(\phi_1, A_{1\parallel})[f] = S_f \tag{6.10}$$
$$N(f)[\phi_1] = S_{\phi_1} \tag{6.11}$$
$$A(f)[A_{1\parallel}] = S_{A_{1\parallel}} \tag{6.12}$$

is solved, where the sources S_f, S_{ϕ_1} and $S_{A_{1\parallel}}$ are included in the right-hand side of the equations such that the given functions f_{MMS}, $\phi_{1\text{MMS}}$ and $A_{1\parallel\text{MMS}}$ are a solution to the system. The functions f_{MMS}, $\phi_{1\text{MMS}}$ and $A_{1\parallel\text{MMS}}$ can be chosen arbitrarily. The sources S_{ϕ_1}, S_f and $S_{A_{1\parallel}}$ are calculated by plugging the solutions into Eqs. (6.7), (6.8) and (6.9). As the solutions are known this is a straightforward process and can be done with a computer algebra system. We use `Mathematica` [157]. The functions f_{MMS}, $\phi_{1\text{MMS}}$ and $A_{1\parallel\text{MMS}}$ are called the manufactured solutions.

Numerical solutions of the modified Vlasov-Maxwell system, presented in Eqs. (6.10), (6.11) and (6.12), can be compared to the exact solutions f_{MMS}, $\phi_{1\text{MMS}}$ and $A_{1\parallel\text{MMS}}$. This makes it possible to calculate relative errors of the numerical solution and to study the order of convergence of the numerical scheme.

$\hat{\omega}$	n_{pol}	n_{tor}	n_{rad}	q_0	α	$\hat{m}_i$	$\hat{m}_e$
2π	1	1	1	0.86	2.12	1	1

Table 6.1.: Table displaying the parameters used in the MMS analysis. This table has been published in [87].

As manufactured solutions we choose the functions

$$\begin{aligned} f_{\text{MMS}}(\hat{r},\varphi,\theta,\hat{v}_\parallel,\hat{\mu},\hat{t}) &:= \cos^2(\hat{\omega}\hat{t})\cos^2(n_{\text{tor}}\varphi)\sin^2(n_{\text{pol}}\theta) \\ &\quad\times \sin^2(\pi n_{\text{rad}}(\hat{r}-\hat{r}_{\min})/(\hat{r}_{\max}-\hat{r}_{\min}))e^{-\left(\hat{v}_\parallel^2+\hat{\mu}\hat{B}\right)}, \\ A_{1\parallel\text{MMS}}(\hat{r},\varphi,\theta,\hat{t}) &:= \phi_{1\text{MMS}}(\hat{r},\varphi,\theta,\hat{t}) := 2\cos^2(\hat{\omega}\hat{t})\cos(n_{\text{tor}}\varphi)\sin(2n_{\text{pol}}\theta) \\ &\quad\times \sin(2\pi n_{\text{rad}}(\hat{r}-\hat{r}_{\min})/(\hat{r}_{\max}-\hat{r}_{\min})), \end{aligned} \tag{6.13}$$

representing a mode. We run the verification in three different magnetic geometries testing different aspects of the implementation. For the first test we choose a slab geometry with the magnetic field specified in Eq. (4.52). In slab geometry multiple terms in the Vlasov equations like the curvature or $\boldsymbol{\nabla}B$ drift vanish. Furthermore, the magnetic field is uniform and points in the φ direction. Therefore, the magnetic field is automatically aligned to the coordinate system and the locally field-aligned coordinate system coincides with the intrinsic coordinate system of the manifold. This renders the field line tracing trivial and thus enables us to test the implementation without complications arising from the locally field-aligned coordinate system. For the second test we choose a circular geometry with the magnetic field specified in Eq. (4.53) and a constant q profile $q(\hat{r}) = q_0$. In the circular geometry the magnetic field lines are twisted and the field line tracing as well as the interpolation is tested. For both the slab and circular geometry we use a large aspect ratio assumption such that φ represents a Cartesian coordinate labelling poloidal planes. Third, we test a toroidal magnetic equilibrium that incorporates all terms in the gyrokinetic Vlasov equation. The magnetic field is given by

$$\hat{\boldsymbol{B}} = \frac{\hat{B}}{\hat{R}_0 + \hat{r}\cos(\theta)}\left(\boldsymbol{e}_\varphi + \frac{\hat{r}}{q(\hat{r})}\boldsymbol{e}_\theta\right), \tag{6.14}$$

with the q profile

$$q(\hat{r}) = q_0 + \alpha\left(\frac{\hat{r}}{\hat{a}}\right)^2, \tag{6.15}$$

minor radius a and major radius R_0. We run the MMS analysis with the parameters given in Table 6.1 with four different resolutions presented in Table 6.2. The radial box size is limited to $\hat{r}_{\min} = 0$ and $\hat{r}_{\max} = 1$ in slab geometry and $\hat{r}_{\min} = 0.1$ and $\hat{r}_{\max} = 0.5$ in both circular and toroidal geometry. In velocity space $(\hat{v_\parallel}, \hat{\mu})$, we choose a domain size of $[-3,3]\times[0,9]$ and in the toroidal angle φ of $[0, 2\pi)$. We calculate one period of the manufactured solution up to $\hat{t} = 1$. We analyse the results by calculating the relative L_2 and L_∞ error of the numerical solution given by $||f - f_{\text{MMS}}||_p/||f_{\text{MMS}}||_p$ where $p \in \{2, \infty\}$. The results are shown in Fig. 6.2.

We observe that the numerical scheme converges to second order for all three geometries. Initially, the convergence is faster than second order because the linear advection terms in the

N_φ	$\Delta\hat{x} = \Delta\hat{z}$	$N_{v_\parallel}$	N_μ	$\Delta\hat{t}$
16	0.025	10	5	0.01
32	0.0125	20	10	0.005
64	0.00625	40	20	0.0025
128	0.003125	80	40	0.00125

Table 6.2.: Table displaying the different resolutions used for the method of manufactured solutions.

gyrokinetic Vlasov equation are discretised with a fourth order scheme. For higher resolutions the numerical error introduced by the second order discretisation of the field equations and the nonlinear Poisson brackets starts to dominate, and we observe the expected second order convergence.

6.3. Screw pinch studies

After having demonstrated the convergence of the numerical scheme, we verify if the code reproduces the correct physical results. The starting point is a physical test where we measure growth rates of an ion temperature gradient mode in a screw pinch geomety. This section has been published in [87].

We use a setup similar to the work in [158, 83]. We model the screw pinch with the circular geometry, described in the last section, with a constant q profile $q(r) = q_0$. We simulate two different cases. In the first case we choose $q_0 \to \infty$, corresponding to a slab like geometry[†], and in the second case we choose a finite q. This allows us to compare the performance of ordinary field-aligned coordinates, in the slab case, to the locally field-aligned coordinate system with interpolation for the finite q case. We choose the reference length parameters $L_{\text{ref}} = 7.74 \cdot 10^{-1}$ m, $B_{\text{ref}} = 1$ T, $m_{\text{ref}} = 1$ u and $T_{\text{ref}} = 1$ keV such that the ion gyroradius $\rho_{\text{ref}} = 3.23 \cdot 10^{-3}$ m and $L_{\text{ref}}/\rho_{\text{ref}} = 239$. The simulation box ranges from $\hat{r}_{\min} = 4.17 \cdot 10^{-3}$ to $\hat{r}_{\max} = 6.05 \cdot 10^{-2}$ spanning over $14.5\rho_{\text{ref}}$. In velocity space $(\hat{v}_\parallel, \hat{\mu})$ we choose a domain size of $[-4, 4] \times [0, 16]$ and in the toroidal angle φ of $[0, 2\pi)$. We use the trapezoidal integration weights and the quadratic grid in μ, described in Section 4.2, and no hyperdiffusion as well as no buffer zone. We use a realistic mass ratio with $m_e = 1/3600\, m_i$.

We initialise the simulation with local Maxwellians, presented in Eq. (4.20), and given temperature $\hat{T}_{0\sigma}$ and density profiles $\hat{n}_{0\sigma}$. Maxwellians are an exact equilibrium of the gyrokinetic Vlasov equation in the given geometry and thus provide a stable initial condition. In order to study growth rates, we initialise a mode as a perturbation of the Maxwellian

$$\hat{f}_\sigma(\hat{t} = 0, \hat{r}, \theta, \varphi, \hat{v}_\parallel, \hat{\mu}) = \hat{F}_{0\sigma}(\hat{r}, \theta, \varphi, \hat{v}_\parallel, \hat{\mu}) \times \left(1 + \epsilon \exp\left(-\left(\frac{\hat{r} - \hat{r}_{\text{mid}}}{\delta \hat{r}_{\text{mode}}}\right)^2\right) \cos\left(m\theta + n\varphi\right)\right) . \quad (6.16)$$

We choose the profiles $\hat{T}_{0i}$, $\hat{T}_{0e}$, $\hat{n}_{0i}$ and $\hat{n}_{0e}$, appearing in the Maxwellian distribution function,

[†]Numerically we implement $q_0 \to \infty$ by setting $q_0 = 10^8$.

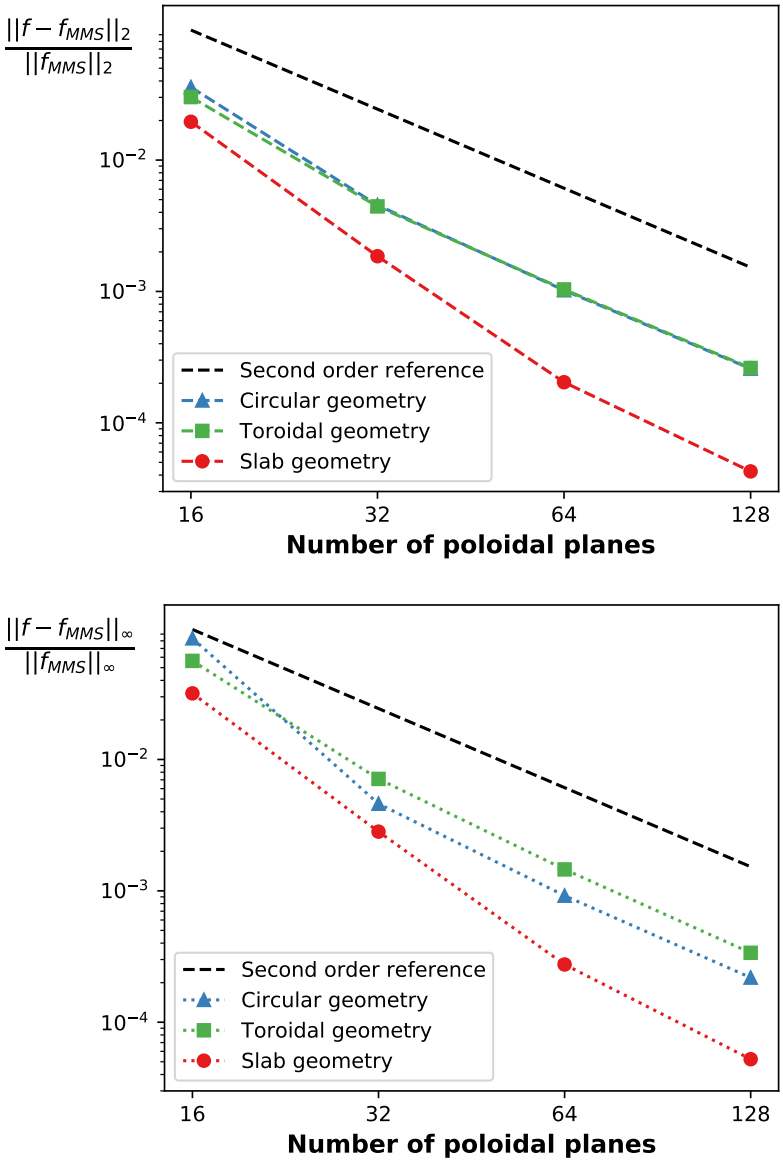

Figure 6.2.: Figure displaying the relative L_2 (top) and L_∞ (bottom) error of the numerical solution in the MMS test as a function of the resolution. The resolution is represented by the number of poloidal planes. The complete resolutions are specified in Tab. 6.2. The test is performed for a slab, circular and toroidal geometry. For reference a dashed line representing second order convergence is drawn in black. The numerical scheme converges to second order for all geometries.

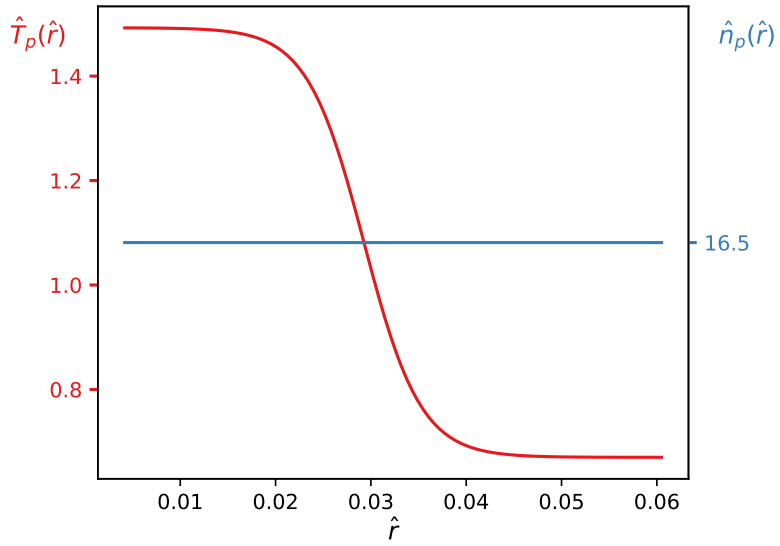

Figure 6.3.: Figure showing the normalised initial temperature and density profile. The profile is the same for ions and electrons. This figure has been published in [87].

according to

$$P(\hat{r}) = C_P \exp\left(-\kappa_P \, \delta \, \hat{r}_P \tanh\left(\frac{\hat{r} - \hat{r}_{\text{mid}}}{\delta \hat{r}_P}\right)\right) . \tag{6.17}$$

The parameters for the profile and the initial condition are given in Table 6.3. The initial density profile is chosen to be constant by setting $\kappa_n = 0$. The profiles are depicted in Figure 6.3.

In order to compare the growth rates to analytical predictions, we use an approximate dispersion relation derived in [83, Appendix]. The dispersion relation has been derived with the assumption of adiabatic electron response and large aspect ratio, i.e. $r/(qR_0) \to 0$. Both approximations are valid for the parameters chosen and hence we expect the numerical results to be close to the prediction of the dispersion relation. In order to present the dispersion relation, we need to define a single Fourier mode of $\hat{\phi}_1$:

$$\hat{\phi}_1 = \hat{\phi}_{1,m,n,\hat{\omega}} \exp\left(i(m\theta + n\varphi - \hat{\omega}\hat{t})\right) . \tag{6.18}$$

ϵ	$1.00 \cdot 10^{-4}$
$\hat{r}_{\text{mid}}$	$3.05 \cdot 10^{-2}$
$\delta\hat{r}_{\text{mode}}$	$5.90 \cdot 10^{-3}$
$C_{n_i} = C_{n_e}$	$1.65 \cdot 10^{1}$
$C_{T_i} = C_{T_e}$	1.00
$\delta\hat{r}_{T_i} = \delta\hat{r}_{T_e}$	$6.05 \cdot 10^{-3}$
$\delta\hat{r}_{n_i} = \delta\hat{r}_{n_e}$	$1.21 \cdot 10^{-2}$
$\kappa_{T_i} = \kappa_{T_e}$	$6.62 \cdot 10^{1}$
$\kappa_{n_i} = \kappa_{n_e}$	0

Table 6.3.: Table defining the parameters for the screw pinch profile for ions and electrons specified in Equation (6.17). This table has been published in [87].

The dispersion relation reads [83, Appendix]

$$
\begin{aligned}
&-\left(\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\right)^2\left[\frac{\partial^2}{\partial\hat{r}^2}+\left(\frac{1}{\hat{r}}+\frac{1}{\hat{n}_0}\frac{\partial\hat{n}_0}{\partial\hat{r}}\right)\frac{\partial}{\partial\hat{r}}-\frac{m^2}{\hat{r}^2}\right]\hat{\phi}_{1,m,n,\hat{\omega}}(\hat{r})=\\
&\left\{-\frac{1}{\hat{T}_{0i}}\left(1+zZ(z)\right)-\frac{1}{\hat{T}_{0e}}+\frac{m}{\hat{k}^*\hat{r}\hat{B}_0 L}\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\right.\\
&\left.\left[Z(z)\left(\frac{1}{\hat{n}_0}\frac{\partial\hat{n}_0}{\partial\hat{r}}-\frac{1}{2\hat{T}_{0i}}\frac{\partial\hat{T}_{0i}}{\partial\hat{r}}\right)+z(1+zZ(z))\frac{1}{\hat{T}_{0i}}\frac{\partial\hat{T}_{0i}}{\partial\hat{r}}\right]\right\}\\
&\times\hat{\phi}_{1,m,n,\hat{\omega}}(\hat{r}),
\end{aligned}
\tag{6.19}
$$

with $z=\hat{\omega}/\hat{k}^*$, $\hat{k}^*=\hat{k}_\parallel\sqrt{2\hat{T}_{0i}}$ and the parallel wavenumber $\hat{k}_\parallel=(m/q+n)b_\varphi$. Furthermore, we used the plasma dispersion function

$$
\begin{aligned}
Z(u)&=\frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty}\frac{\exp\left(-x^2\right)}{x-u}\,\mathrm{d}x=\mathrm{i}\sqrt{\pi}\exp\left(-u^2\right)\left(1+\operatorname{erf}(\mathrm{i}u)\right)\\
\operatorname{erf}(x)&=\frac{2}{\sqrt{\pi}}\int_0^x\exp\left(-t^2\right)\mathrm{d}t\,.
\end{aligned}
\tag{6.20}
$$

The dispersion relation is a nonlinear, differential eigenvalue problem that has to be solved for the radial electrostatic profile $\hat{\phi}_{1,m,n,\omega}$ and the complex eigenvalue $\hat{\omega}$ simultaneously. As we are not aware of methods to solve the equation analytically, we employ a numerical method. First, we discretise the differential operators to obtain an ordinary nonlinear eigenvalue problem. Similar to the numerical scheme presented in Chapter 4, we choose fourth order centred finite differences to discretise the r derivative. At the boundary, we implement Dirichlet boundary conditions and set $\hat{\phi}_{1,m,n,\omega}(\hat{r}_{\min})=\hat{\phi}_{1,m,n,\omega}(\hat{r}_{\max})=0$. We choose 128 points for the discretisation.

The remaining nonlinear eigenvalue problem can be solved by calculating the roots of the characteristic polynomial. We use a determinant free method based on the Newton iteration described in [159, pp. 28–29]. As an initial guess for the Newton iteration we choose $\omega_{\mathrm{init}}=-2+4i$. The calculation is done with `Mathematica` [157]. The results are presented in Appendix B.2.

The dispersion relation and thus the eigenvalues $\hat{\omega}$ are a function of the poloidal mode number m and the parallel wavenumber $\hat{k}_\parallel$. Hence, the eigenvalues $\hat{\omega}$ do not depend on q explicitly. For a given poloidal mode number m, we can select two toroidal mode numbers n_{slab} and n_{screw}, for the $q=\infty$ and $q\neq\infty$ case respectively, such that the parallel wavenumbers and, consequently, the growth rates are equal. For the slab case the poloidal magnetic field vanishes, and the parallel wavenumber reads $\hat{k}_{\parallel\mathrm{slab}}=n_{\mathrm{slab}}$. For the $q\neq\infty$ case we choose the toroidal mode number n_{screw} such that $k_{\parallel\mathrm{screw}}=k_{\parallel\mathrm{slab}}$, i.e.

$$
\begin{aligned}
n_{\mathrm{screw}}&=k_{\parallel\mathrm{slab}}-\frac{b_\theta m}{r}\\
&=n_{\mathrm{slab}}-\frac{m}{q}\,.
\end{aligned}
\tag{6.21}
$$

q	m	n	$\hat{k}_\parallel$
∞	5	1	1
∞	10	1	1
∞	15	1	1
∞	20	1	1

q	m	n	$\hat{k}_\parallel$
5/3	5	-2	1
10/3	10	-2	1
15/3	15	-2	1
20/3	20	-2	1

Table 6.4.: Table displaying the different initial mode configurations for the simulation. On the left, the mode numbers for the $q = \infty$ case and on the right, the mode numbers for the $q \neq \infty$ case are shown. The poloidal and toroidal mode numbers are constructed such that the parallel mode number $k_\parallel$ is equal to one in all cases. This table has been published in [87].

Having two setups with the same growth rates allows us to compare the performance of field-aligned coordinates in the $q = \infty$ case to the locally field-aligned coordinates in the $q \neq \infty$ case.
The q factors and mode numbers chosen for the simulation are shown in Table 6.4. They are selected such that a broad range of modes and q factors are tested.

6.3.1. Growth rates

We measure the growth rate from the L_2 norm of the electrostatic potential. The L_2 norm is calculated numerically with the trapezoidal quadrature rule similar to the charge density in the quasi-neutrality equation. We determine the growth rate by performing a linear fit of $\log\|\hat{\phi}_1\|$ between the times $\hat{t} = 1$ and $\hat{t} = 3.5$. This is shown exemplarily in Fig. 6.4.
We use a resolution of $\Delta\hat{x} = \Delta\hat{z} = 6.46 \cdot 10^{-4}$, corresponding to $0.2\,\rho_{\text{ref}}$, $n_\varphi = 16$, $n_{v_\parallel} = 256$, $n_\mu = 16$ and a timestep of $\Delta t = 1.4 \cdot 10^{-3}\,t_{\text{ref}} = 3.5\text{ns}$. The timestep is constant throughout the simulation and limited by the parallel electron streaming. We study the convergence of the simulation at the example of the $q = 15/3$ case. Upon doubling the resolution in each dimension, $\Delta\hat{x} \to 0.5\Delta\hat{x}$, $\Delta\hat{z} \to 0.5\Delta\hat{z}$, $n_\varphi \to 2n_\varphi$, $n_{v_\parallel} \to 2n_{v_\parallel}$ and $n_\mu \to 2n_\mu$, the growth rate of the $q = 15/3$ mode changes less than 1%. Therefore, we assume a 1% error on the growth rates obtained from our simulations.
The growth rates obtained from the simulation as well as the solution of the approximate dispersion relation are shown in Fig. 6.5. The simulations for $q = \infty$ and $q \neq \infty$ agree well with each other within the 1% error margin, as predicted by the dispersion relation. Furthermore, the numerical results are close to the solution of the dispersion relation. They agree within the 1% numerical error margin for the poloidal mode numbers $m = 15$ and $m = 20$. For the poloidal mode numbers $m = 5$ and $m = 10$ the measured growth rates are slightly higher than growth rates obtained from the dispersion relation. As discussed above, the dispersion relation approximates the growth rates and thus exact agreement is not expected.

6.3.2. Energy and particle conservation

In order to study the behaviour of the nonlinear phase of the simulation, we investigate the conservation of the particle number and energy during the onset of turbulence. We choose

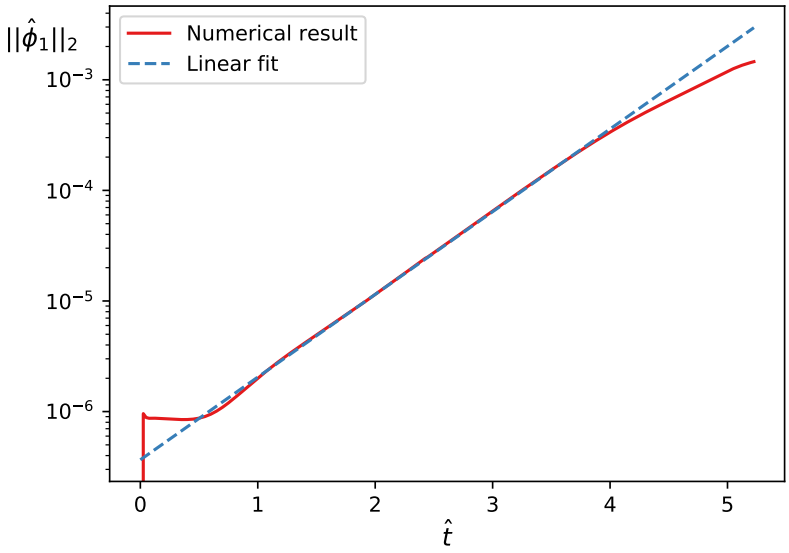

Figure 6.4.: Figure displaying the L_2 norm of the electrostatic potential as a function of time. The growth rate is obtained by performing a linear fit of $\log\|\hat{\phi}_1\|_2$ between the times $\hat{t} = 1$ and $\hat{t} = 3.5$. The fit is drawn in a blue dashed line. Shown are the results from the simulation with $q = 15/3$ and $m = 15$. This figure has been published in [87].

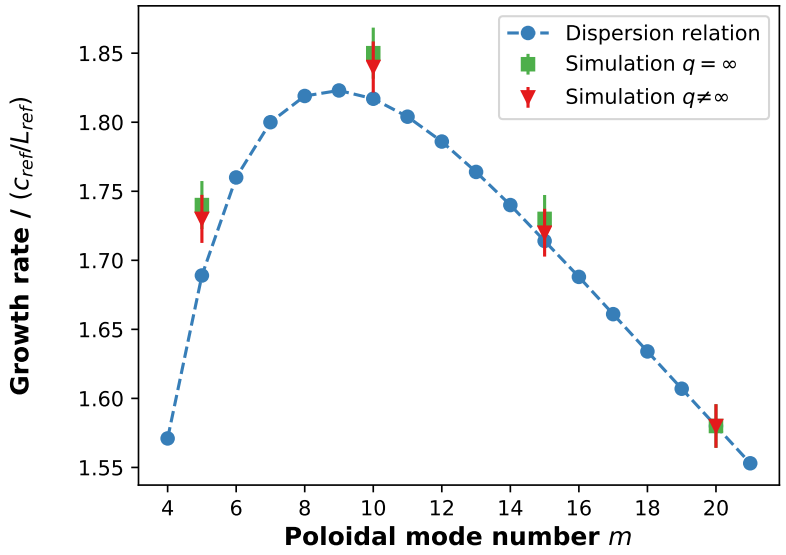

Figure 6.5.: Figure displaying the growth rates obtained from the approximate dispersion relation and the simulations with different q factors and mode numbers specified in Table 6.4. Vertical lines through the markers represent the error of the simulation. The measured growth rates are close to the prediction of the dispersion relation. This figure has been published in [87].

the same screw pinch setup as presented in the last section and run the $q = 15/3$ case. We use a resolution of $\Delta\hat{x} = \Delta\hat{z} = 3.23 \cdot 10^{-4}$, corresponding to $0.1\,\rho_{\text{ref}}$, $n_\varphi = 32$, $n_{v_\parallel} = 64$ and $n_\mu = 16$. We run the simulation until $\hat{t} = 10$, when the turbulence starts to hit the wall. The chosen Dirichlet boundary conditions allow an energy flux through the domain boundaries, which becomes effective when eddies hit the wall. As an example, the ion density is depicted in the linear and nonlinear phase in Fig. 6.6. The energy of the ions and electrons is shown in Fig. 6.7. The number of particles is conserved up to a precision of 10^{-7}. The energy is conserved up to a precision of $10^{-8\dagger}$. In the linear phase the conservation is improved. As soon as the nonlinear phase starts energy gets transferred between electrons and ions.

6.4. Alfv́en waves in slab geometry

We continue with the physical verification of the electromagnetic part of the model. In this section, we compare the frequency and damping rate of Alfv́en waves in slab geometry between numerical simulation with `GENE-X` and analytical estimates, following the work [132, 97].

6.4.1. Derivation of the dispersion relation

We start by deriving a dispersion relation that provides the analytical estimates for the frequencies and damping rates of the Alfv́en waves. The dispersion relation is similar to the one used in [132, 97]. We perform the derivation explicitly in order to control the assumptions made and obtain the correct equation in our normalisation. Electrons are, due to their light weight, more mobile than ions and the main generator of electromagnetic fields. Therefore, we assume that the ions are stationary, i.e. the frequency of the Alfv́en waves fulfils $\omega_{\text{Alfv́en}} \gg k_\parallel v_{T_i}$. Further, we employ a δf splitting such that $\hat{f}_e = \delta\hat{f}_e + \hat{f}_M$ with

$$\hat{f}_M(\hat{v}_\parallel) = \frac{1}{\sqrt{\pi}} e^{-\hat{v}_\parallel^2}\,. \tag{6.22}$$

We neglect all nonlinear terms. We choose two spatial dimensions x and y. In compliance with the locally field aligned coordinate system presented in Chapter 3, y labels the direction parallel to the magnetic field. The normalisation is chosen, as introduced in Section 4.1, such that $\hat{m}_i = 1$ and $\hat{q}_e = -1$. The normalised gyrokinetic Vlasov equation for the electrons, only taking the parallel dynamics into account, reads

$$\frac{\partial \delta\hat{f}_e}{\partial \hat{t}} = -\sqrt{\frac{2}{\hat{m}_e}}\hat{v}_\parallel \left(\frac{\partial \delta\hat{f}_e}{\partial \hat{y}} + \hat{q}_e \left(\frac{\partial \hat{\phi}_1}{\partial \hat{y}} + \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} \right) \hat{F}_M \right)\,, \tag{6.23}$$

[†]The energy conservation of 10^{-8}, presented here, is significantly better than the result published in [87]. The reason is that the factor $1/2$ in front of the $q_\sigma\phi_1$ term in Eq. (2.99) was missing in the energy diagnostic of the `GENE-X` code. This resulted in an overestimation of the electrostatic energy contribution and, consequently, an incorrect energy measurement.

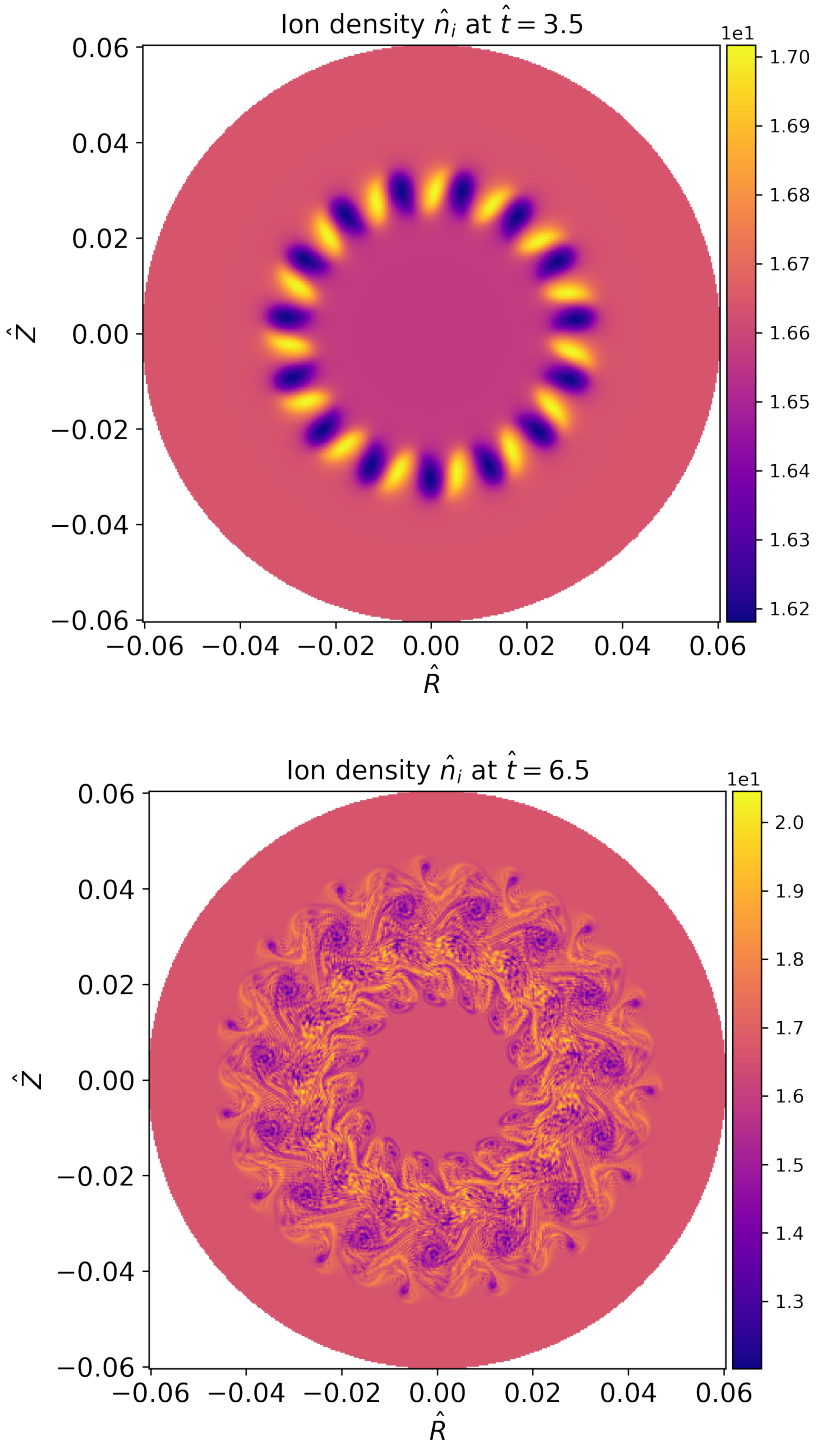

Figure 6.6.: Figure displaying the ion density in the poloidal plane located at $\varphi = \pi$ in the linear phase at $\hat{t} = 3.5$ and the nonlinear phase at $\hat{t} = 6.5$ of a screw pinch simulation with $q = 15/3$. The simulation is initialised with the temperature and density profile shown in Fig. 6.3 and perturbed with an unstable $m = 15$, $n = -2$ mode. This figure has been published in [87]. For a unified presentation, the color bars and font sizes have been adapted.

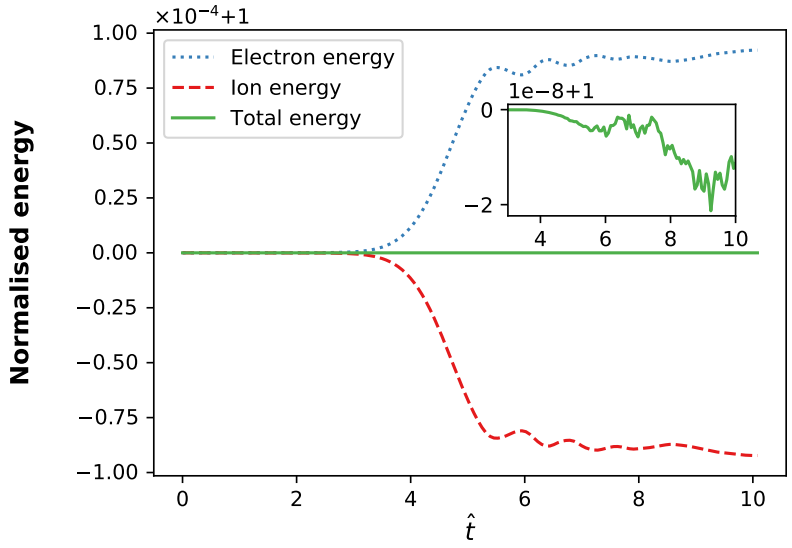

Figure 6.7.: Figure displaying the normalised energy $E(t)/E(0)$ for electrons and ions of a simulation in screw pinch geometry with $q = 15/3$. The simulation is initialised with the temperature profile shown in Figure 6.3 and perturbed with an unstable $m = 15$, $n = -2$ mode. The simulation develops turbulence at approximately $\hat{t} = 4$. The total energy is conserved to a precision of 10^{-8}. This figure has been published in a similar form in [87]. The origin of the difference is explained in the footnote on page 86.

with the corresponding field equations

$$-\left(\frac{\rho_{\text{ref}}}{L_{\text{ref}}}\right)^2 \frac{\partial^2 \hat{\phi}_1}{\partial \hat{x}^2} = \hat{q}_e \int_{-\infty}^{\infty} \delta \hat{f}_e \, \mathrm{d}\hat{v}_{\parallel} \tag{6.24}$$

$$-\left(\frac{\rho_{\text{ref}}}{L_{\text{ref}}}\right)^2 \frac{\partial^2 \hat{A}_{1\parallel}}{\partial \hat{x}^2} = \frac{\beta_{\text{ref}}}{2} \hat{q}_e \sqrt{\frac{2}{\hat{m}_e}} \int_{-\infty}^{\infty} \hat{v}_{\parallel} \delta \hat{f}_e \, \mathrm{d}\hat{v}_{\parallel}. \tag{6.25}$$

Similar to Eq. (6.18), we switch to a Fourier representation of a single mode

$$\delta \hat{f}_e(\hat{x}, \hat{y}, \hat{v}_{\parallel}, \hat{t}) = \delta \hat{f}_{\hat{k}_y, \hat{\omega}}(\hat{x}, \hat{v}_{\parallel}) e^{i(\hat{k}_y \hat{y} - \hat{\omega} \hat{t})} \tag{6.26}$$

$$\hat{\phi}_1(\hat{x}, \hat{y}, \hat{t}) = \hat{\phi}_{\hat{k}_x, \hat{k}_y}(\hat{t}) e^{i(\hat{k}_x \hat{x} + \hat{k}_y \hat{y})} \tag{6.27}$$

$$\hat{A}_{1\parallel}(\hat{x}, \hat{y}, \hat{t}) = \hat{A}_{\hat{k}_x}(\hat{y}, \hat{t}) e^{i\hat{k}_x \hat{x}} \,. \tag{6.28}$$

Inserting Eqs. (6.26), (6.27) and (6.28) into Eqs. (6.23), (6.24) and (6.25) gives the linear system

$$-i\hat{\omega} \delta \hat{f}_e = -\sqrt{\frac{2}{\hat{m}_e}} \hat{v}_{\parallel} \left(i\hat{k}_y \delta \hat{f}_e - \hat{q}_e \left(i\hat{k}_y \hat{\phi}_1 + \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} \right) \hat{F}_M \right) \tag{6.29}$$

$$-\left(\frac{\rho_{\text{ref}}}{L_{\text{ref}}}\right)^2 \hat{k}_x^2 \hat{\phi}_1 = \hat{q}_e \int_{-\infty}^{\infty} \delta \hat{f}_e \, \mathrm{d}\hat{v}_{\parallel} \tag{6.30}$$

$$\left(\frac{\rho_{\text{ref}}}{L_{\text{ref}}}\right)^2 \hat{k}_x^2 \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} = i\hat{\omega} \frac{\beta_{\text{ref}}}{2} \hat{q}_e \sqrt{\frac{2}{\hat{m}_e}} \int_{-\infty}^{\infty} \hat{v}_{\parallel} \delta \hat{f}_e \, \mathrm{d}\hat{v}_{\parallel} \,. \tag{6.31}$$

We define $\bar{\omega} = \sqrt{\hat{m}_e/2}\hat{\omega}/\hat{k}_y$ and rewrite Eq. (6.29)

$$\delta\hat{f}_e = -\frac{\hat{v}_\parallel \hat{q}_e \left(\frac{i}{\hat{k}_y}\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} - \hat{\phi}_1\right) F_M}{\hat{v}_\parallel - \bar{\omega}}\,. \tag{6.32}$$

Inserting Eq. (6.32) into Eq. (6.30) yields

$$\begin{aligned}\hat{\phi}_1 &= \left(\frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}\right)^2 \frac{1}{\hat{k}_x^2}\left(\int_{-\infty}^{\infty} \frac{\hat{v}_\parallel F_M(\hat{v}_\parallel)}{\hat{v}_\parallel - \bar{\omega}}\,\mathrm{d}\hat{v}_\parallel\right)\left(\frac{i}{\hat{k}_y}\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} - \hat{\phi}_1\right) \\ &= \left(\frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}\right)^2 \frac{1}{\hat{k}_x^2}\left(1 + \bar{\omega}Z(\bar{\omega})\right)\left(\frac{i}{\hat{k}_y}\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} - \hat{\phi}_1\right)\,,\end{aligned} \tag{6.33}$$

where we have used the plasma dispersion function and Eq. (A.19). Similarly, we insert Eq. (6.32) into Eq. (6.31)

$$\begin{aligned}\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} &= -i\hat{\omega}\left(\frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}\right)^2 \frac{1}{\hat{k}_x^2}\frac{\beta_{\mathrm{ref}}}{2}\sqrt{\frac{2}{\hat{m}_e}}\left(\int_{-\infty}^{\infty} \frac{\hat{v}_\parallel^2 F_M(v_\parallel)}{\hat{v}_\parallel - \bar{\omega}}\,\mathrm{d}\hat{v}_\parallel\right)\left(\frac{i}{\hat{k}_y}\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} - \hat{\phi}_1\right) \\ &= -i\hat{\omega}\left(\frac{L_{\mathrm{ref}}}{\rho_{\mathrm{ref}}}\right)^2 \frac{1}{\hat{k}_x^2}\frac{\beta_{\mathrm{ref}}}{2}\sqrt{\frac{2}{\hat{m}_e}}\left(\bar{\omega}(1 + \bar{\omega}Z(\bar{\omega})\right)\left(\frac{i}{\hat{k}_y}\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}} - \hat{\phi}_1\right)\,,\end{aligned} \tag{6.34}$$

where we used Eq. (A.20). Subtracting Eq. (6.33) from $i/\hat{k}_y$ times Eq. (6.34) gives the final dispersion relation

$$\hat{k}_x^2\left(\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\right)^2 + \left(1 + \bar{\omega}Z(\bar{\omega})\right)\left(1 - \frac{\beta_{\mathrm{ref}}}{\hat{m}_e}\bar{\omega}^2\right) = 0\,. \tag{6.35}$$

We solve the dispersion relation numerically with the `FindRoot` function of `Mathematica` [157]. We choose $\bar{\omega} = 1 - 0.1i$ as the starting value of the iterative algorithm. The results for different mode numbers and plasma beta are given in Appendix B.4.

6.4.2. Numerical simulation

We continue with describing the setup of the numerical simulation with `GENE-X`. We choose the reference parameters $L_{\mathrm{ref}} = 1.0\mathrm{m}$, $T_{\mathrm{ref}} = 1.0\mathrm{KeV}$ and $B_{\mathrm{ref}} = 3.231 \cdot 10^{-3}\mathrm{T}$ such that $\rho_{\mathrm{ref}} = L_{\mathrm{ref}} = 1\mathrm{m}$. We choose a realistic mass ratio with $\hat{m}_i = 1$ and $\hat{m}_e = 1/3600$. We simulate in slab geometry and use the magnetic field described in Eq. (4.52). In slab geometry, the toroidal angle φ is interpreted as a Cartesian coordinate and coincides with the locally field-aligned y coordinate. We choose a box size of $(\hat{x}, \hat{y}, \hat{z}, \hat{v}_\parallel, \hat{\mu}) \in [0, 2\pi] \times [0, 2\pi) \times [0, 2\pi) \times [-3, 3] \times [0, 9]$. The initial condition is given by

$$\hat{f}_i(\hat{t} = 0, \hat{x}, \hat{y}, \hat{z}, \hat{v}_\parallel, \hat{\mu}) = \pi^{-3/2} e^{-(\hat{v}_\parallel^2 - \hat{\mu})} \tag{6.36}$$

$$\hat{f}_e(\hat{t} = 0, \hat{x}, \hat{y}, \hat{z}, \hat{v}_\parallel, \hat{\mu}) = \pi^{-3/2} e^{-(\hat{v}_\parallel^2 - \hat{\mu})} \cdot \left(1 + \epsilon \cos\left(\hat{k}_y \hat{y}\right) \sin\left(\hat{k}_x \hat{x}\right)\right)\,. \tag{6.37}$$

We choose the strength of the mode perturbation as $\epsilon = 10^{-2}$. We simulate with a constant parallel mode number $\hat{k}_y = 1$. The orthogonal mode number is varied from $\hat{k}_x \in$

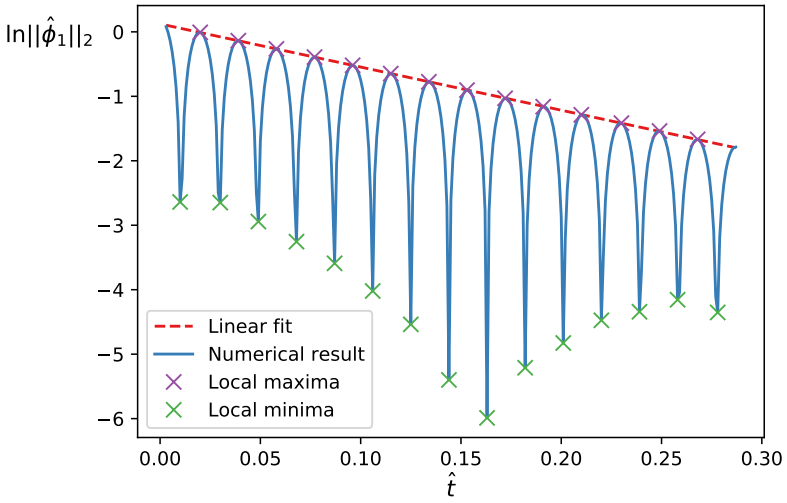

Figure 6.8.: Figure displaying the time trace of the logarithm of the L_2 norm of the electrostatic potential for $\beta_{\text{ref}}/\hat{m}_e = 0.2$ and $\hat{k}_x = 1/4$. The red crosses indicate local maxima and the green crosses local minima. The damping rate is determined by a linear fit through the local maxima, drawn as a red dashed line.

$\{1/128, 1/64, 1/32, 1/16, 1/8, 1/4, 1/2, 1\}$. The distribution function is constant along z and constant on the boundary. For the electromagnetic fields we choose periodic boundary conditions in z and Dirichlet boundary conditions in x. We set the electromagnetic fields to zero for $x = 0$ and $x = 2\pi$. Due to the periodic boundary condition, there is no dynamic in the z direction and the system is effectively two dimensional, as in Eq. (6.35). With the mode perturbation in x we set the perpendicular mode in the electrostatic field $\hat{\phi}_1$. We use a resolution of $\Delta\hat{x} = \Delta\hat{z} = 0.05$, $n_\varphi = 32$, $n_{\hat{v}_\parallel} = 64$, $n_{\hat{\mu}} = 16$ and a constant timestep of $\Delta\hat{t} = 1 \cdot 10^{-3}$. We use no hyperdiffusion and no buffer zone.

We measure the frequency and damping rate of the Alfv́en wave at the L_2 norm of the electrostatic potential. An example of the time trace of the L_2 norm is shown in Fig. 6.8. To determine the damping rate and frequency, we search for local minima and maxima of the oscillatory time trace. We define a local maximum as a point $\hat{t}_i$ such that is $(\|\hat{\phi}_1\|_2)(\hat{t}_i) > (\|\hat{\phi}_1\|_2)(\hat{t}_j)\,\forall j \in \{i-2, i-1, i+1, i+2\}$, i.e. $(\|\hat{\phi}_1\|_2)(\hat{t}_i)$ has to be larger than all of its four nearest neighbours. In a similar fashion, we define a local minimum by the condition $(\|\hat{\phi}_1\|_2)(\hat{t}_i) < (\|\hat{\phi}_1\|_2)(\hat{t}_j)\,\forall j \in \{i-2, i-1, i+1, i+2\}$ such that the L_2 norm is smaller than its four nearest neighbours. The matched local maxima and minima of the example are shown in Fig. 6.8 as crosses. We determine the growth rate of the oscillation as the slope of the linear fit through the local maxima. For the kinetic shear Alfv́en waves the growth rate is always negative and the waves are damped via Landau damping. Let $\{\hat{t}_{\min,i} \,|\, i \in \{1, \dots N_{\min}\}\}$ label the position of the minima. We determine the frequency of the oscillation as

$$\operatorname{Re}\hat{\omega} \approx \frac{1}{N_{\min} - 1} \sum_{i=1}^{N_{\min}-1} \frac{\pi}{\hat{t}_{\min,i+1} - \hat{t}_{\min,i}}\,. \tag{6.38}$$

6.4.3. Results

The results from the numerical simulation with GENE-X as well as from the solution of the dispersion relation are shown in Fig. 6.9. The numerical results agree well with the prediction of the dispersion relation for all plasma beta and mode numbers tested. For low perpendicular mode numbers and large plasma beta the damping of the Alfv́en waves is very weak. In that case the induced electric field $\partial \hat{A}_{1\parallel}/\partial \hat{t}$ is determined by the two integrals in Eq. (2.89). These integrals need to be discretised consistently such that numerical errors arising from the discretisation cancel and the physics is correctly resolved [97, Appendix. A]. This is also known as the Ampère cancellation problem [98, 99, 100]. As shown by the results, the consistent discretisation scheme used in GENE-X does not suffer from the cancellation problem.

6.5. Cyclone base case

We close this chapter by performing a well-studied CBC benchmark [156] that probes the full electromagnetic model in toroidal geometry. Following [156], we measure the growth rate and frequency of different toroidal modes at a fixed plasma β. We choose the reference parameters $L_{\text{ref}} = 1.67\text{m}$, $B_{\text{ref}} = 2.0\text{T}$, $m_{\text{ref}} = 1\text{u}$, $T_{\text{ref}} = 2.14\text{keV}$ and $n_{\text{ref}} = 4.66 \cdot 10^{19}\ \text{m}^{-3}$ such that the plasma β is given by $\beta_{\text{ref}} = 1\%$. The reference ion gyroradius reads $\rho_{\text{ref}} = 3.34 \cdot 10^{-3}\text{m}$. The minor radius is given by $a = 0.60\text{m} = 0.36 L_{\text{ref}}$. We choose a radial extent of $\hat{r}_{\text{min}} = 3.60 \cdot 10^{-2} = 0.1\hat{a}$ to $\hat{r}_{\text{max}} = 3.24 \cdot 10^{-1} = 0.9\hat{a}$. The magnetic field is given by the toroidal geometry described in Eq. (6.14) with the q profile

$$q(\hat{r}) = 0.86 - 0.16 \left(\frac{\hat{r}}{\hat{a}}\right) + 2.52 \left(\frac{\hat{r}}{\hat{a}}\right)^2 . \tag{6.39}$$

In velocity space $(\hat{v}_\parallel, \hat{\mu})$ we choose a domain size of $[-4, 4] \times [0, 16]$ and in the toroidal angle φ of $[0, 2\pi)$. We use a realistic mass ratio of $m_e = 1/3600 m_i$. Similar to the screw pinch benchmark, we initialise the simulation with local Maxwellians, presented in Eq. (4.20), with the profiles $\hat{T}_{0i}$, $\hat{T}_{0e}$, $\hat{n}_{0i}$ and $\hat{n}_{0e}$, chosen according to Eq. (6.17). The parameters for the profiles are given in Tab. 6.5 and the profiles are shown in Fig. 6.10. This test poses challenges for GENE-X. As discussed in Section 4.5, the initial condition constructed with local Maxwellians is unstable and prevents us from studying the growth rate and frequency

$\hat{r}_{\text{mid}}$	$5.00 \cdot 10^{-1}$
$C_{n_i} = C_{n_e}$	1.00
$C_{T_i} = C_{T_e}$	1.00
$\delta\hat{r}_{T_i} = \delta\hat{r}_{T_e}$	$3.00 \cdot 10^{-1}$
$\delta\hat{r}_{n_i} = \delta\hat{r}_{n_e}$	$3.00 \cdot 10^{-1}$
$\kappa_{T_i} = \kappa_{T_e}$	2.49
$\kappa_{n_i} = \kappa_{n_e}$	$8.00 \cdot 10^{-1}$

Table 6.5.: Table defining the parameters for the CBC profile for ions and electrons specified in Equation (6.17).

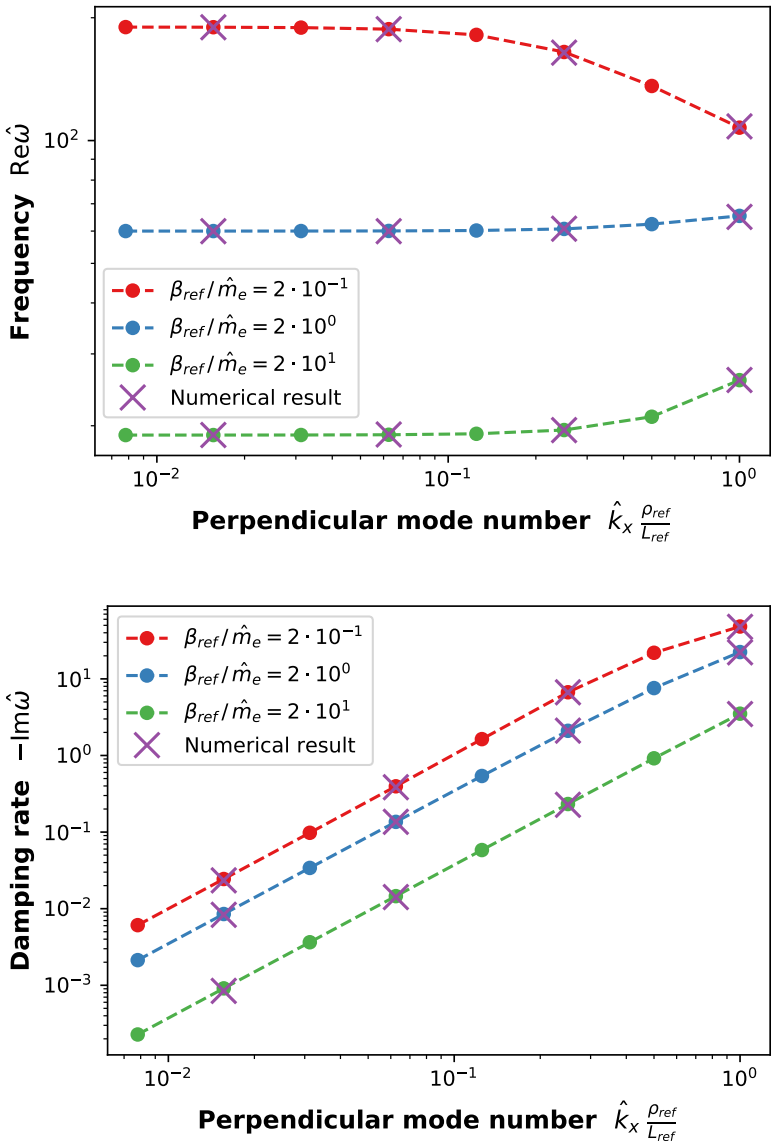

Figure 6.9.: Figure displaying the frequency $\mathrm{Re}\,\hat{\omega}$ and the damping rate $-\mathrm{Im}\,\hat{\omega}$ for different perpendicular mode numbers and different $\beta_{\mathrm{ref}}/\hat{m}_e$. The results obtained with the numerical calculation from `GENE-X` agree well with the results from the dispersion relation for all parameters tested.

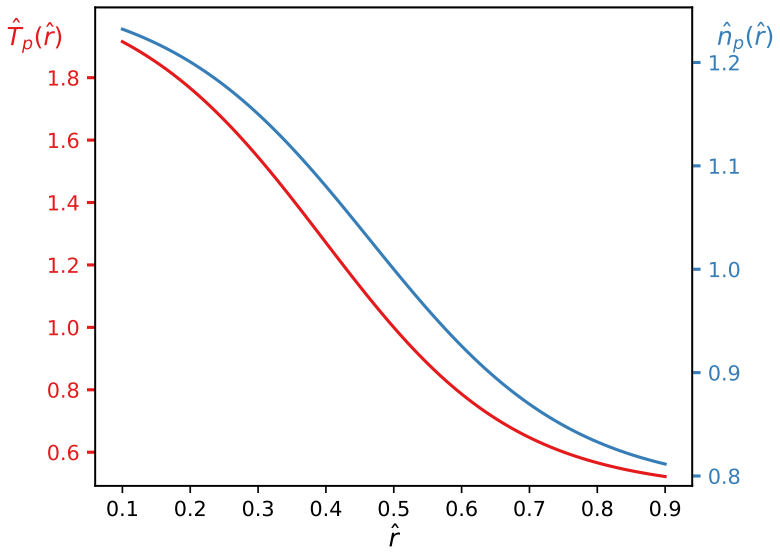

Figure 6.10.: Figure showing the normalised initial temperature and density profiles. The profiles are the same for ions and electrons.

of specific toroidal modes. Therefore, we remove the dynamics of the initial condition from the system by evaluating the right-hand side of the gyrokinetic Vlasov-Maxwell system on the initial condition and subtracting the result every timestep. Using this filtering process is similar to using the assumption that the $\boldsymbol{\nabla} B$ drift of the background vanishes, which is usually made in δf codes [139]. To trigger the growth of unstable modes, we perturb the initial distribution function with a sample x from the uniform distribution on the interval $[0, 10^{-6}]$ according to

$$\hat{f}_\sigma \to \hat{f}_\sigma + x\,. \tag{6.40}$$

We select modes with a fixed toroidal mode number by using a toroidal Fourier filter on the electrostatic potential and the parallel component of the electromagnetic vector potential. The dominant unstable modes in the prescribed system are either ion temperature gradient (ITG) or trapped electron modes (TEM) [156]. They are aligned to the magnetic field and obtain part of their toroidal structure through the poloidal rotation of the magnetic field. Using the FCI approach, the parallel dynamics is resolved via field line tracing and a high resolution within the poloidal plane. The toroidal resolution is usually sparse and of the order of $n_\varphi = 16$. This makes toroidal Fourier filtering difficult because the highest resolvable mode number is $n_\varphi/2$[†]. To filter higher toroidal mode numbers, the number of poloidal planes needs to be increased. The computational demand of the simulations increases quadratically with

[†]At this point we would like to emphasise that only the Fourier filter and not the code itself is limited to modes with a mode number $n_\varphi/2$. In the simulation ITGs and TEMs of significantly higher mode numbers are resolved. The reason for this is that the modes are field-aligned and most of the toroidal structure of the mode is generated by the poloidal rotation of the magnetic field. This rotation and the parallel structure of the modes along the field line is well resolved because the coordinate system is aligned to the magnetic field. For instance, in a simulation with $n_\varphi = 20$ and a toroidal Fourier filter of $n = 5$ we obtain the mode $n = 15$. The mode $n = 15$ is not covered by the filter because the number of poloidal planes is too low. It is still present in the system due to the high parallel resolution of FCI and grows faster than the mode $n = 5$.

n_φ because, in addition to the increase in number of grid points, the timestep decreases proportional to the increase in n_φ. Therefore, we restrict the analysis to the toroidal mode numbers $n = 5, 10$ and 15. As discussed in Chapter 2, we solve the gyrokinetic Vlasov-Maxwell system in a long wavelength limit. The modes $n = 5, 10$ and 15 correspond to perpendicular mode numbers $k_\perp = nq(a/2)/(a/2)$ of $k_\perp \rho_{\text{ref}} = 0.078, 0.157$ and 0.235. These are significantly smaller than $k_\perp \rho_{\text{ref}} = 1$ such that a good agreement between the literature and the simulations performed with `GENE-X` can be expected.

We measure the growth rate and the frequency from the maximum of the parallel component of the electromagnetic vector potential on the poloidal plane located at $\varphi = \pi$. Similar to the screw pinch benchmark, we determine the growth rate by performing a linear fit of $\log \max_{\hat{x},\hat{z}}(\hat{A}_{\parallel_1}(\hat{x}, \hat{z}, \varphi = \pi))$. We calculate the frequency by subtracting the fit from the signal, measuring the position of two minima of the oscillations $\{\hat{t}_{\text{min},1}, \hat{t}_{\text{min},2}\}$ and using $\text{Re}\,\hat{\omega} = 2\pi/(\hat{t}_{\text{min},2} - \hat{t}_{\text{min},1})$. As `GENE-X` is a nonlinear code and the frequency is small in relation to the growth rate, we are in general not able to measure the position of more than two minima before the nonlinear phase starts.

Growth rates and frequencies

We use a resolution of $\Delta\hat{x} = \Delta\hat{z} = 3.34 \cdot 10^{-3}$ corresponding to $1.67\rho_{\text{ref}}$, $n_{v_\parallel} = 48$ and $n_\mu = 12$. For simulations with the $n = 5$ toroidal filter we use $n_\varphi = 32$, for the $n = 10$ toroidal filter we use $n_\varphi = 48$ and for the $n = 15$ toroidal filter we use $n_\varphi = 64$. The timestep is constant throughout the simulation and reads $\Delta t = 1 \cdot 10^{-3} t_{\text{ref}}$, $\Delta t = 0.5 \cdot 10^{-3} t_{\text{ref}}$ and $\Delta t = 0.25 \cdot 10^{-3} t_{\text{ref}}$ for the three different toroidal filters.

We study the convergence of the simulation at the example of the simulations with the $n = 10$ and $n = 15$ toroidal filters. A higher toroidal mode number leads to a higher poloidal mode number and therefore requires a higher resolution within the poloidal plane. We increase the resolution of the two simulations to $\Delta\hat{x} \to 0.7\Delta\hat{x}$, $\Delta\hat{z} \to 0.7\Delta\hat{z}$ and $n_\varphi \to 2n_\varphi$ and measure the growth rate and frequency. We do not observe significant changes in the frequency upon increasing the resolution. The error of the frequency is limited by the measurement error of the minima of the oscillations. We observe significant changes in the growth rate. The growth rate for the $n = 10$ simulation increases by 10% from 0.186 to 0.205 and the growth rate of the $n = 15$ simulation by 30% from 0.271 to 0.356. Therefore, we assume an error of 10% on the growth rate of the $n = 5$ and $n = 10$ simulations and an error of 30% on the growth rate of the $n = 15$ simulation. To obtain more precise results for the $n = 15$ simulation, a higher resolution is needed, which is expensive due to the specific properties of the locally field-aligned coordinate system discussed above. The measured growth rates and frequencies are shown in Fig. 6.11. Both the growth rates and frequencies agree well with the reference results obtained with the `GENE` code [156].

Exemplarily, a cross section of the electrostatic potential and the parallel component of the electromagnetic vector potential for the simulations with a Fourier filter of the mode $n = 10$ are shown in Fig. 6.12. The structure of the $n = 10$ mode is clearly visible in both fields.

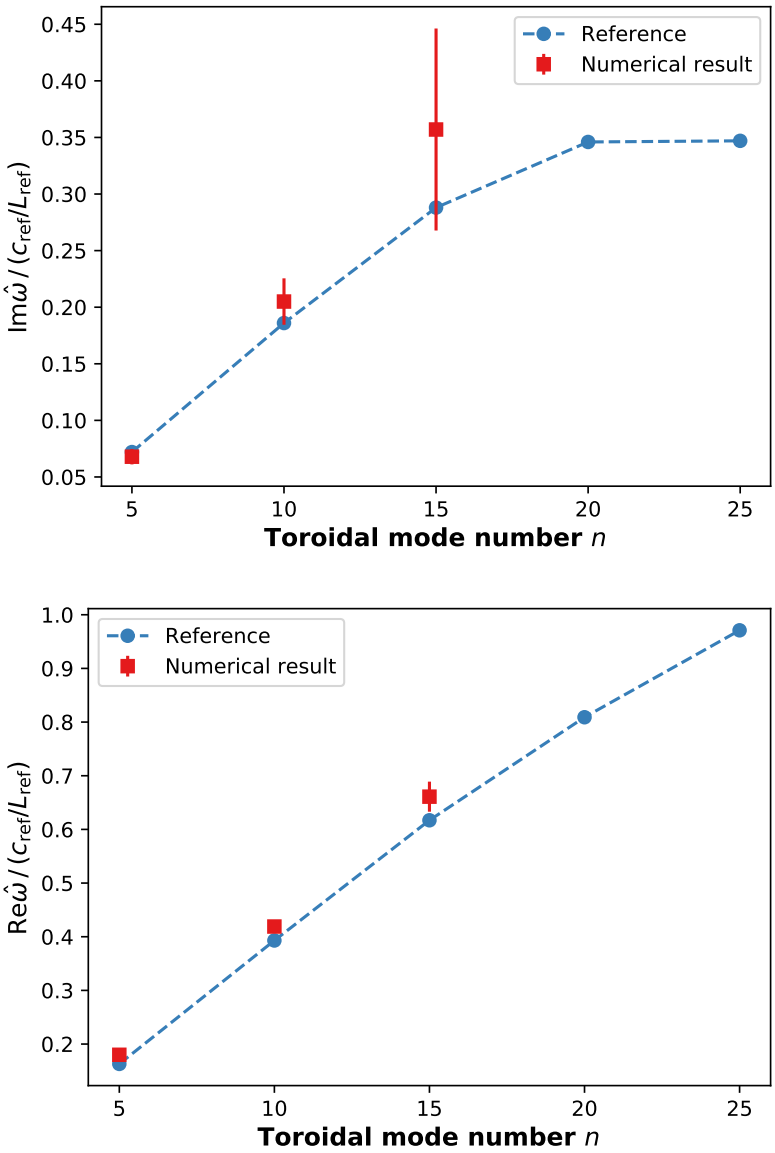

Figure 6.11.: Figure displaying the measured growth rate (top) and frequency (bottom) of modes with a different toroidal mode number. For reference, the results obtained with the `GENE` code [156] are drawn as a dashed line. The measured growth rates and frequencies agree well.

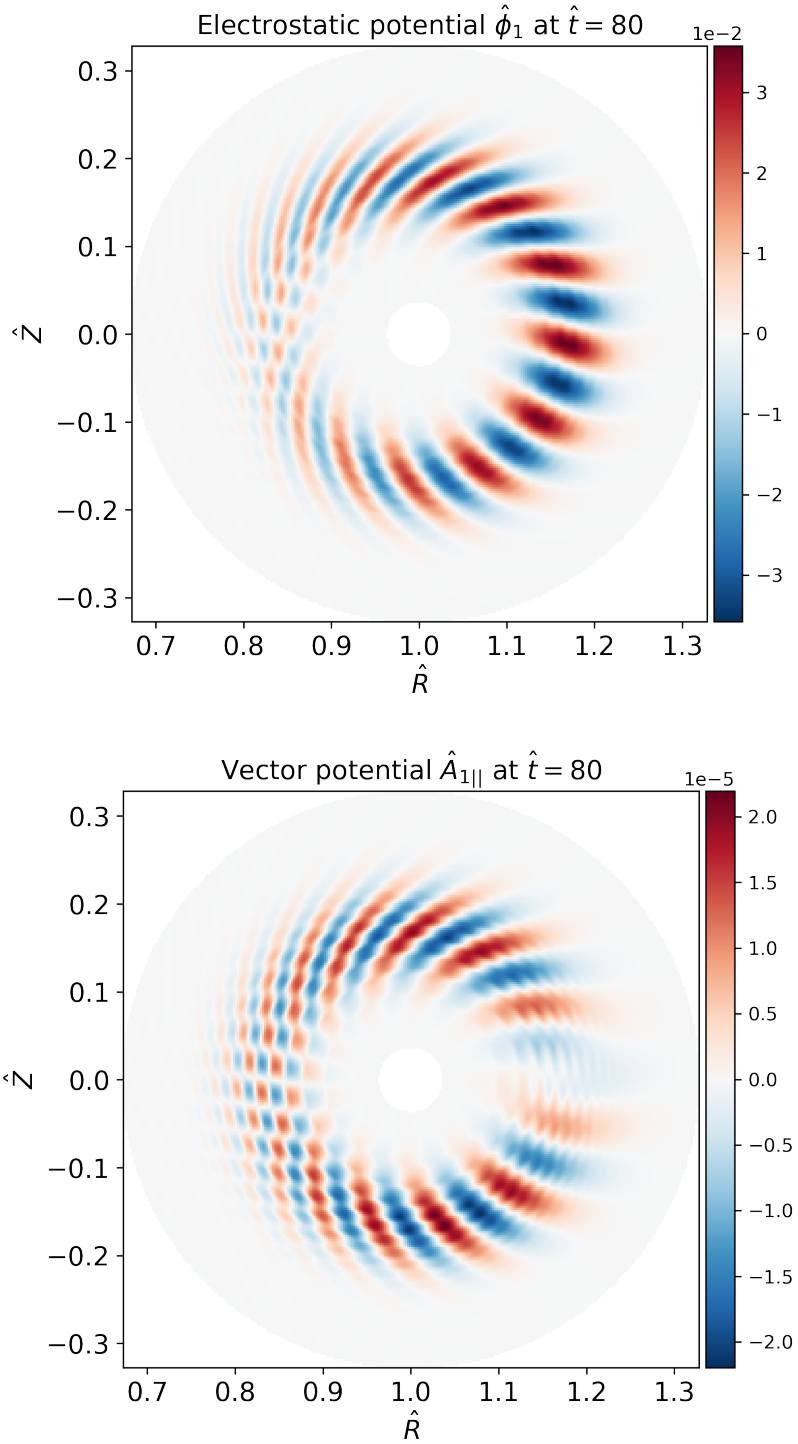

Figure 6.12.: Figure displaying the electrostatic potential (top) and the parallel component of the electromagnetic vector potential (bottom) on the poloidal located at $\varphi = \pi$ at the time $t = 80t_{\text{ref}}$ for the simulation with a toroidal Fourier filter of $n = 10$.

6.6. Conclusion and outlook

In this chapter, we presented thorough tests of the `GENE-X` code. The test were designed to comprehensively probe different parts of the numerical scheme and their implementation in the code. We started by performing precise numerical tests to demonstrate that the chosen numerical scheme, in combination with the locally field-aligned coordinate system, is capable of solving the gyrokinetic Vlasov-Maxwell system. This included the measurement of the accuracy of the field line tracing and a comprehensive convergence order analysis with MMS. We demonstrated that the numerical scheme converges to second order accuracy. We proceeded with physically motivated benchmarks and probed the electrostatic part of the model by comparing growth rates in a screw pinch geometry against analytical predictions and tested the nonlinear regime by studying the conservation of energy. We tested the electromagnetic implementation by calculating the damping rates and frequencies of Alfv́en waves in slab geometry. Finally, we tested the whole gyrokinetic Vlasov-Maxwell system in toroidal geometry by performing a well studied CBC benchmark. `GENE-X` passed all tests.
In the future, it would be beneficial to perform further tests that probe the nonlinear regime. A suitable candidate would be a nonlinear version of the CBC benchmark [160]. Furthermore, it would be interesting to study the damping of zonal flows with the Rosenbluth-Hinton test [161, 160].

7. Simulations of the edge and scrape-off layer of ASDEX Upgrade

In this chapter, we demonstrate the full capability of the `GENE-X` code by performing simulations of the edge and SOL of ASDEX Upgrade. The simulations are performed in the diverted X-point geometry of the L-mode discharge #36190.

We present the results from three simulations at different physical parameters. The first simulation uses a reduced electron-to-ion mass ratio of $\mu = m_e/m_i = 1/400$, the second simulation a realistic electron-to-deuterium mass ratio of $\mu = 1/3600$ and the third simulation a realistic mass ratio and a collision operator. This allows us to study the effect of the mass ratio and collisions on edge and SOL turbulence. The simulations presented are performed over a timescale of 1ms until profile saturation. This chapter is structured as follows. First, we describe the setup of the simulation and discuss the parameters used. We proceed by presenting the final plasma density and temperature profiles and the radial electric field. The results are compared to Braginskii fluid [52] simulations, performed with the `GRILLIX` code [79, 80], and experimental measurements. Finally, we look at the power balance of the simulations and study the $E \times B$ heat flux over the separatrix and the parallel heat flux in the SOL. In particular, we measure the parallel heat flux onto the divertor and the SOL fall-off length λ_q by performing an Eich fit [26] of the parallel heat flux profile.

7.1. Simulation setup

We use the magnetic equilibrium of the AUG discharge #36190 at the time $t = 3.3s$ for the simulations. The simulations are performed in the edge and SOL ranging from the flux surface $\psi = 0.9$ to $\psi = 1.05$. We use the canonical Maxwellian described in Section 4.5 to initialise the simulations. The initial condition is constructed to allow for comparison with previous Braginskii fluid simulations with `GRILLIX` [79, 80]. The profiles for electrons and ions, entering the canonical Maxwellian, are equal. The form of the profiles is given by the function

$$P(\psi) = \begin{cases} P_{\max} & \text{if} \quad \psi \leq \psi_{\min} \\ c_1 \sin(c_2\psi + c_3) + c_4 & \text{if} \quad \psi_{\min} < \psi < \psi_{\max} \\ P_{\min} & \text{else} \end{cases}, \tag{7.1}$$

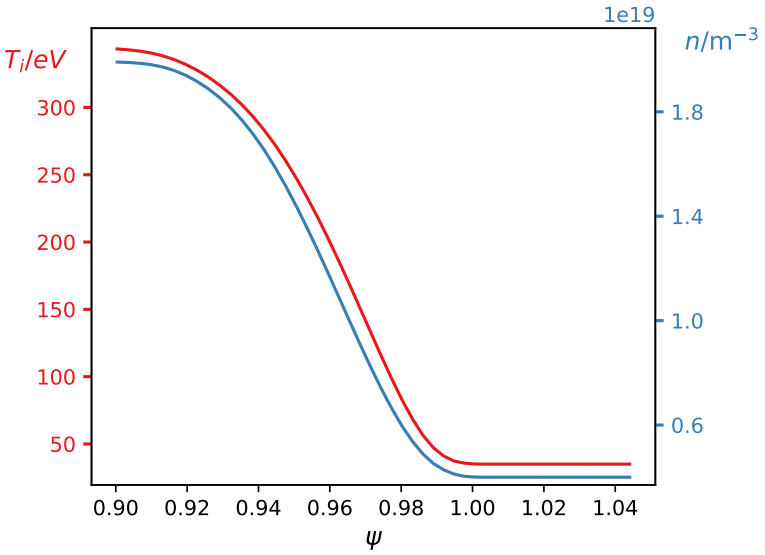

Figure 7.1.: Figure displaying the initial temperature and density profiles for the simulations. Ion and electron temperatures are equal. The functional for of the profile is a sine function described in Eq. (7.1). The measured initial profile, presented here, differs slightly from the analytical form, described in Eq. (7.1), because a canonical Maxwellian is used.

with the coefficients

$$c_1 = (P_{\max} - P_{\min})/2 \tag{7.2}$$

$$c_2 = \pi/(\psi_{\max} - \psi_{\min}) \tag{7.3}$$

$$c_3 = \pi/2 - c_2\psi_{\min} \tag{7.4}$$

$$c_4 = (P_{\max} + P_{\min})/2\,. \tag{7.5}$$

For both the temperature and density profile, we choose $\psi_{\min} = 0.92$ and $\psi_{\max} = 0.995$. In order to obtain a stable initial condition, it is important that the distribution function is constant in regions with $\psi > 1$ at the start of the simulation. On the core boundary $\psi_{\min} = 0.90$ we choose a temperature of $T_{\max} = 350\text{eV}$ and a density of $n_{\max} = 2 \cdot 10^{19}\text{m}^{-3}$, coinciding with the experimental measurements of the ion temperature and density at AUG and the `GRILLIX` simulations. On the SOL boundary $\psi_{\max} = 1.05$ we choose a density of $n_{\min} = 0.2 \cdot 10^{19}\text{m}^{-3}$ and a temperature of $T_{\min} = 35\text{eV}$. The density is selected according to experimental measurements. The temperature boundary value of 35eV is motivated by the available velocity space resolution. In gyrokinetic simulations the velocity information about the distribution function is retained. In the plasma core, the temperature is by orders of magnitude larger than in the SOL. Consequently, the dynamics of fast particles in the plasma core and slow particles in the SOL need to be resolved on the velocity space grid. In order to reduce the computational demand of the simulations, we limit the factor between the temperature at the core boundary and the SOL boundary to ten. To trigger the growth of unstable modes, we perturb the initial condition, according to Eq. (6.40), with a sample x from

the uniform distribution on the interval $[0, 10^{-3}]$. The plasma profiles after initialisation with the canonical Maxwellian are depicted in Fig. 7.1. The initial ion density of the simulation in the poloidal plane is shown in Fig. 7.2.

As explained in Section 4.4, we use Dirichlet boundary conditions in the simulations. The boundary value of the distribution function in real space is equal to the initial condition and remains constant throughout the simulation. This implies that the temperature and density on the core boundary are fixed to 350eV and $2.0 \cdot 10^{19}\mathrm{m}^{-3}$, and the core boundary functions as heat and particle source. On the SOL boundary the temperature and density is fixed to 35eV and $0.2 \cdot 10^{19}\mathrm{m}^{-3}$, and the boundary functions as a heat and particle sink. By fixing the temperature at the SOL boundary to 35eV, we ensure that the temperature in the SOL does not reach values significantly below 35eV, and the dynamics is well resolved within the available velocity space resolution.

We choose the reference length $L_{\mathrm{ref}} = 1.65\mathrm{m}$, corresponding to the major radius, the reference magnetic field $B_{\mathrm{ref}} = 2.5T$, corresponding to the on-axis magnetic field, $m_{\mathrm{ref}} = 1u$, corresponding to the mass of a hydrogen ion and $T_{\mathrm{ref}} = 0.050\mathrm{keV}$, corresponding to the ion temperature in the near SOL. The reference gyroradius reads $\rho_{\mathrm{ref}} = 2.89 \cdot 10^{-4}\mathrm{m}$ and $L_{\mathrm{ref}}/\rho_{\mathrm{ref}} = 5.71 \cdot 10^{3}$. We use a resolution of $\Delta x = \Delta z = 1.43\mathrm{mm}$ in the poloidal plane which corresponds to roughly 1.9 ion gyroradii at a (separatrix) temperature of 100eV and a (outboard mid-plane) magnetic field of 2T. As the temperature and magnetic field vary over the device, so does the resolution in terms of local gyroradii. In velocity space $(\hat{v}_{\parallel}, \hat{\mu})$ we use different resolutions for the simulations with and without collisions. For the conservation of energy, particles and momentum in the simulations with collision operator, it is vital to increase the domain size in velocity space such that larger parts of the tail of distribution function are included. In the simulations without collisions, we use a resolution of $n_{v_{\parallel}} = 72$ and $n_{\mu} = 16$ and a domain size of $[-7, 7] \times [0, 49]$. In the simultions with collisions, we use a resolution of $n_{v_{\parallel}} = 80$ and $n_{\mu} = 16$ and a domain size of $[-8, 8] \times [0, 81]$. In the toroidal angle φ, we use a resolution of $n_{\varphi} = 24$ and a domain size of $[0, 2\pi)$, covering the whole torus. To study the convergence of the numerical results, we performed simulations at an in plane resolution of $\Delta\hat{x} \to 0.7\Delta\hat{x}$, $\Delta\hat{z} \to 0.7\Delta\hat{z}$. We did not observe significant changes of the results.

The timestep is $\Delta t = 6 \cdot 10^{-4} t_{\mathrm{ref}} = 14.3\mathrm{ns}$ and constant throughout the simulation. The simulation is run for 42 time units corresponding to 1ms of plasma evolution.

Close to the boundary and before the divertor plates we apply diffusion in a buffer zone. We set the parameter controlling the strength of the diffusion, introduced in Section 4.2, to $\epsilon_{\mathrm{buffer}} = 200$, which is close to the maximum value allowed by the CFL condition and the chosen timestep. Furthermore, we use hyperdiffusion in the dimensions x, y, z and $v_{\parallel}$. The parameters controlling the strength of the hyperdiffusion, introduced in Section 4.2, are chosen as $\epsilon_{\hat{x},\hat{z}} = 5$, $\epsilon_{\hat{y}} = 0.06$ and $\epsilon_{\hat{v}_{\parallel}} = 0.025$. The strength of the hyperdiffusion in x, y, z is estimated via trial and error. We use the estimation presented in [135] for the value of the hyperdiffusion in $v_{\parallel}$. It is worth discussing the computational cost of the simulations. The simulations presented in this work were performed on the supercomputer Cobra of the Max-

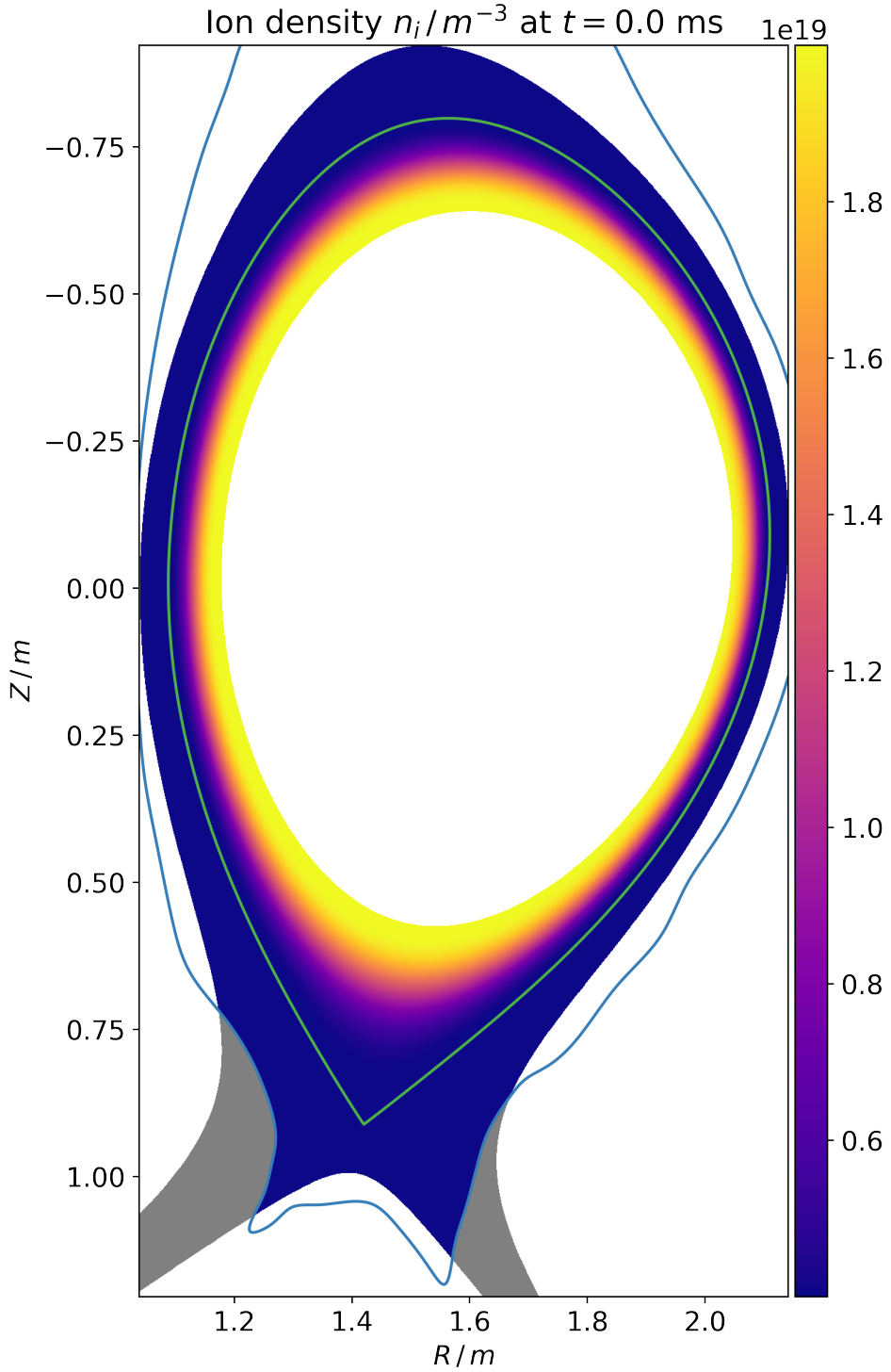

Figure 7.2.: Figure displaying the ion density of the initial state of the simulation on the poloidal plane at $\varphi = \pi$. The separatrix is drawn in green, the device wall in blue and the divertor legs in grey. The distribution function is initialised according to Eq. (7.1) and constant in the SOL.

Planck Computing and Data Facility (MPCDF) and the A3 partition of the supercomputer Marconi of Cineca. Both systems feature compute nodes with Intel SkyLake Processors with 40 and 48 cores respectively. The cost for one simulation at the resolution presented is $2.5 \cdot 10^6$ CPU hours. We used 256 nodes for the simulations which required a runtime of approximately 9 days. Doubling the resolution within the poloidal plane quadruples the computational cost because the resolution is increased uniformly in the x and z direction. This means that a resolution of $\Delta x = \Delta z = 7.53 \cdot 10^{-4}$mm, corresponding to resolving one thermal ion Larmor radius on the separatrix, increases the computational demand by a factor of roughly 3.6. Of further interest is a higher resolution in velocity space to reduce the temperature in the SOL. Upon decreasing the temperature by a factor of 2, the resolution requirement in $v_\parallel$ and μ increases by a factor of $\sqrt{2}$, and the total computational cost increases by a factor of $\sqrt{2}^2 = 2$. Doubling the resolution in the φ direction leads to a quadrupling of the computational cost because the number of points increase and the timestep needs to be reduced at the same time. This means that an increase of the number of poloidal planes from $n_\varphi = 24$ to $n_\varphi = 32$ leads to increase in the computational demand by roughly 1.8.

Collision model

For the simulations with collisions, we use a simple collision model based on the Bhatnagar-Gross-Krook (BGK) collision operator [113]. The collision operator supports multi-species collisions and reads

$$\hat{C}_{\alpha\beta}\hat{f}_\alpha = \hat{\nu}_{\alpha\beta}\left(\frac{\hat{B}}{\hat{B}^*_\parallel}\hat{\mathcal{M}}_{\alpha\beta} - \hat{f}_\alpha\right) \tag{7.6}$$

$$\hat{\mathcal{M}}_{\alpha\beta} = \frac{\hat{n}_\alpha}{\left(\pi\hat{T}_{\alpha\beta}\right)^{3/2}} \exp\left(-\frac{\left(\hat{v_\parallel} - \hat{u}_{\alpha\beta}\right)^2 + \hat{\mu}\hat{B}}{\hat{T}_{\alpha\beta}}\right), \tag{7.7}$$

where α, β denote the different species and $\hat{n}_\alpha$ the current plasma density. The coefficients $\hat{u}_{\alpha\beta}, \hat{T}_{\alpha\beta}$ are chosen such that the collision operator conserves the particle number, parallel momentum, energy and Boltzmann's $\mathcal{H}$-Theorem on the continuous level [162]. In addition, the collision frequency $\hat{\nu}_{\alpha\beta}$ is chosen such that the temperature relaxation rates match the rates of the Boltzmann collision operator [162]. They read

$$\hat{u}_{\alpha\beta} = \frac{\hat{m}_\alpha\hat{u}_\alpha + \sqrt{\hat{m}_\beta\hat{m}_\alpha}\hat{u}_\beta}{\hat{m}_\alpha + \hat{m}_\beta} \tag{7.8}$$

$$\hat{T}_{\alpha\beta} = \frac{\hat{T}_\alpha + \hat{T}_\beta}{2} + \frac{\hat{m}_\alpha\hat{m}_\beta}{3\left(\hat{m}_\alpha + \hat{m}_\beta\right)}\left(\frac{\hat{u}_\alpha}{\sqrt{\hat{m}_\alpha}} - \frac{\hat{u}_\beta}{\sqrt{\hat{m}_\beta}}\right)^2 \tag{7.9}$$

$$\hat{\nu}_{\alpha\beta} = \frac{n_{\mathrm{ref}}L_{\mathrm{ref}}}{T^2_{\mathrm{ref}}}\frac{16\hat{n}_\beta\sqrt{2\pi\hat{m}_\alpha\hat{m}_\beta}\left(Z_\alpha Z_\beta e^2\right)^2}{3\left(\hat{m}_\alpha\hat{T}_\beta + \hat{m}_\beta\hat{T}_\alpha\right)^{3/2}}\ln\Lambda\,. \tag{7.10}$$

The Coulomb logarithm $\ln\Lambda$ is calculated at the beginning of the simulation according to model number four described in [163] and varies over the computational domain. For more

details on the collision operator in GENE-X we refer to [116]. The numerical scheme, presented in Section 4.2, does not preserve the positivity of the distribution function. The temperature $\hat{T}$ can become negative and cause the collision operator to become imaginary. As the temperature and density get closer to zero, the collision frequency $\hat{\nu}$ diverges. This requires the timestep of the simulation to go to zero. To prevent these problems, we use a density floor in the collision operator of $n = 0.1 \cdot 10^{19}\mathrm{m}^{-3}$ for the ion and electron density and $T = 21\mathrm{eV}$ for the ion and electron temperature.

7.2. Plasma profiles and fluctuation amplitudes

We start the analysis of the AUG simulations by looking at the plasma density and the electrostatic potential of the final state at $t = 1\mathrm{ms}$. We present the analysis at the example of the simulation with realistic mass ratio. The plasma density is shown in Fig. 7.3 and the electrostatic potential in Fig. 7.4. Compared to the initial condition, shown in Fig. 7.2, plasma density has been transported out radially and populates the whole confined region. Turbulent structures are dominant on the low field side and most prominent in the confined region. Turbulent streamers start close to the core boundary and end at the separatrix. The density in the SOL is significantly lower than in the confined region. On the outboard mid-plane (OMP) small density blobs are visible that have passed the separatrix and entered the SOL. The electrostatic potential in the confined region is dominated by zonal flows [164]. There is a rapid drop in the electrostatic potential close to the separatrix giving rise to a radial electric field that generates a poloidal rotation of the plasma along the separatrix. In all simulations there is little dynamics in the private flux region. This finding is consistent with the simulations of the discharge performed with GRILLIX. There are two possible explanations for the findings. First, the private flux region is cold and the Larmor radius of thermal ions small. Therefore, the in-plane resolution could not be high enough to resolve the small scale dynamics in the private flux region. A second explanation could be the absence of physical effects related to neutral gases in the simulation. Recent simulations performed with GRILLIX demonstrated turbulent structures in the private flux region when a neutral gas model was added [80].

In the following, we analyse different plasma profiles at the OMP. The profile data at the OMP is measured by interpolating the results along a line that starts at the magnetic axis and is orthogonal to the magnetic flux surfaces via cubic spline interpolation. The data for the density and temperature profiles is taken at $t = 1\mathrm{ms}$ and averaged over all 24 poloidal planes. The fluctuation amplitude shown is calculated as the standard deviation of the profiles over the 24 poloidal planes. The average and standard deviation of the radial electric field is calculated including the last $100\mu\mathrm{s}$ of the simulation. This averages over the fluctuations in the radial electric fields due to oscillations of the geodesic acoustic mode (GAM) [165].

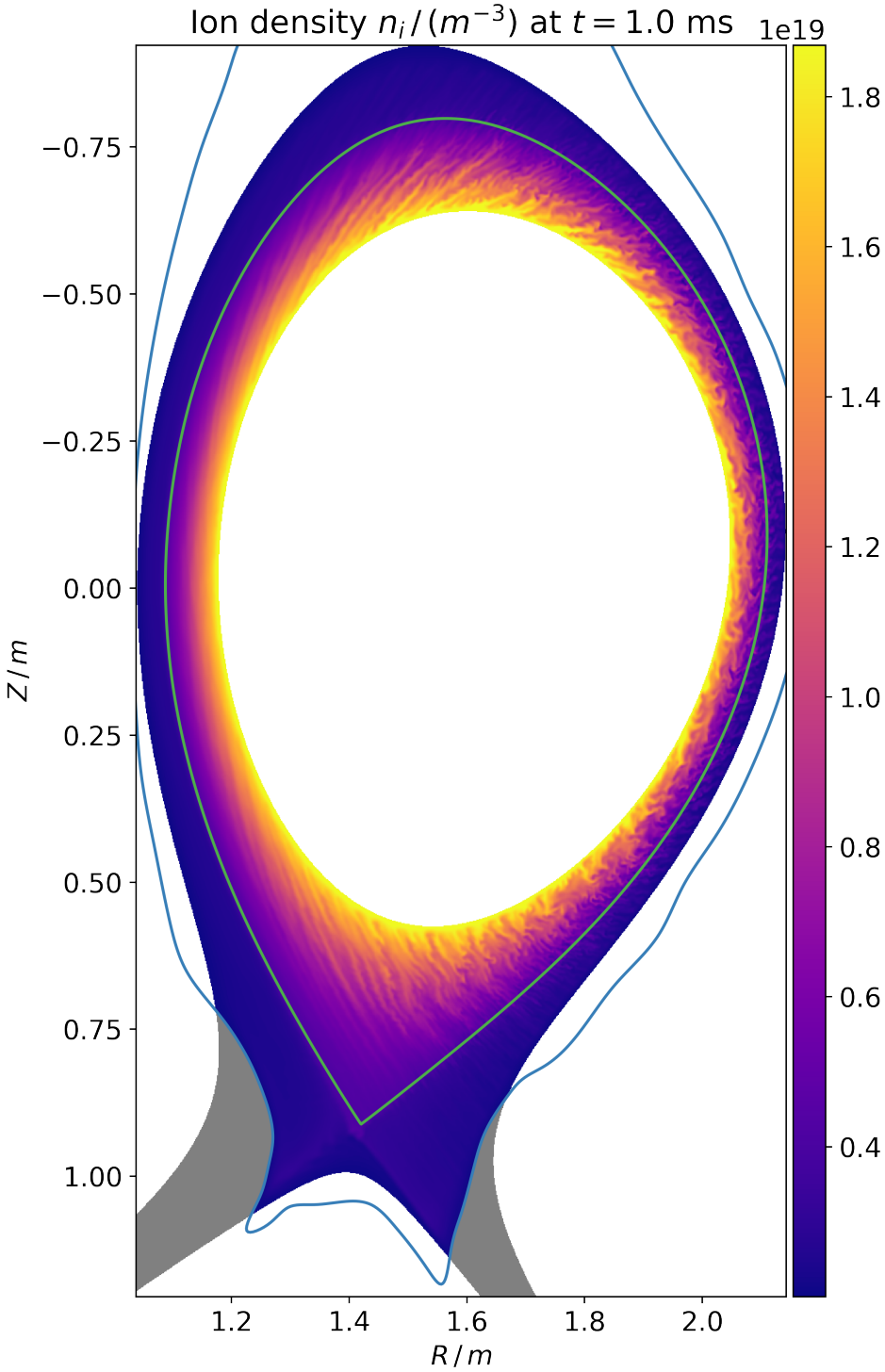

Figure 7.3.: Figure displaying the ion density of the final state of the simulation on the poloidal plane at $\varphi = \pi$. Similar to Fig. 7.2, the separatrix, wall and divertor is shown. Compared to the initial condition presented in Fig. 7.2, plasma density populates the whole confined region, and the dynamics is dominated by turbulence. In the SOL on the low field side density blobs are visible that have been transported over the separatrix.

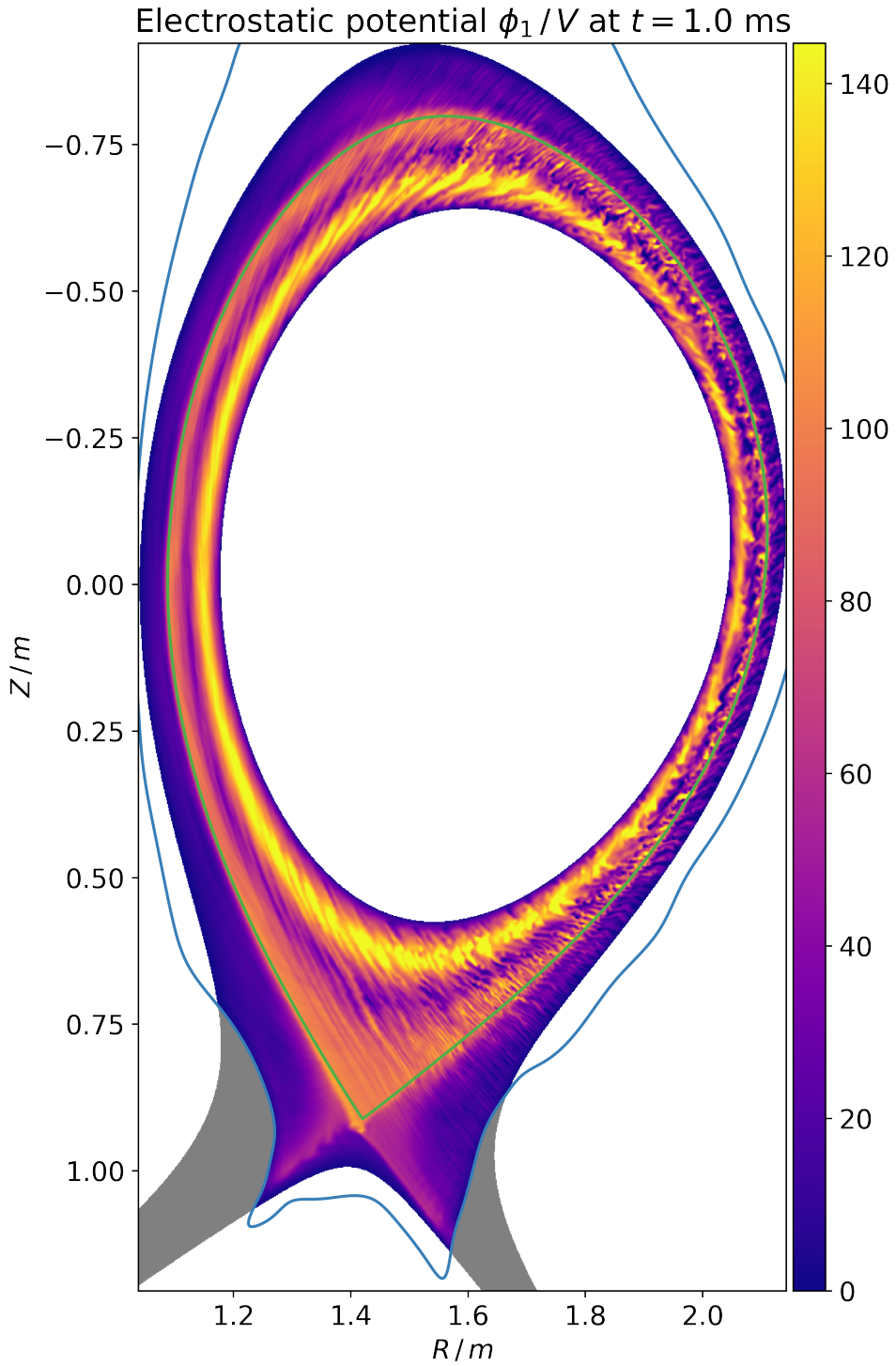

Figure 7.4.: Figure displaying the electrostatic potential of the final state of the simulation on the poloidal plane at $\varphi = \pi$. Similar to Fig. 7.2, the separatrix, wall and divertor is shown. In the confined region the dynamics is dominated by zonal flows that change their structure over time. Upon crossing the separatrix there is a significant drop in the electrostatic potential giving rise to a radial electric field.

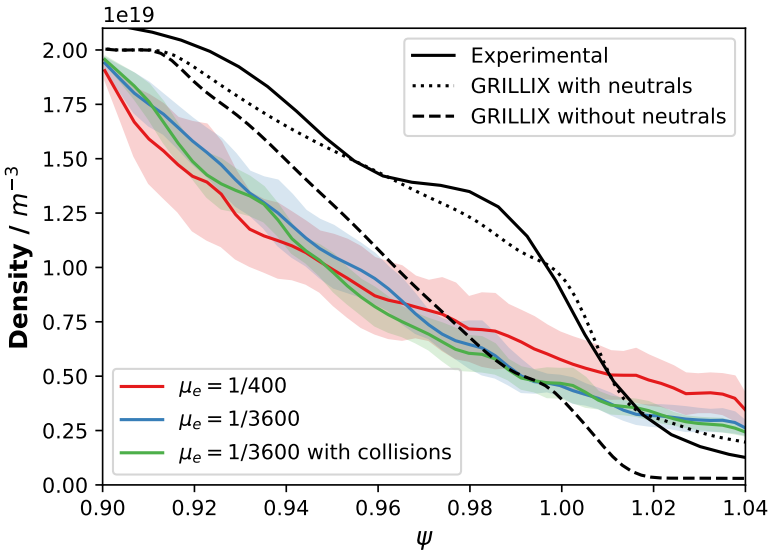

Figure 7.5.: Figure displaying the mean density profiles measured at the outboard mid-plane of the three different GENE-X simulations performed. In addition, the fluctuation amplitude is drawn as a shaded area around the mean. For comparison, the density profile of GRILLIX simulations with and without neutrals and the experimental density measurements are shown.

7.2.1. Density

The density profiles are shown in Fig. 7.5. The measured average value of the density profile is similar for all three simulations performed. We observe a significantly larger fluctuation amplitude in the simulation with reduced mass ratio indicating a stronger turbulent drive. The density in the far SOL of the GENE-X simulations is limited by the boundary condition. The shape of density profiles is similar to the shape of density profile obtained with GRILLIX without neutral physics. The slope of the density profiles is shallower. The density in the SOL in the GRILLIX simulations is lower. This is because the density in the GRILLIX simulations is not fixed by the boundary condition. The shape of the measured density profile of the experiment differs significantly. GRILLIX simulations with neutrals reproduce the experimental measurement quite well. This could be a hint that neutrals are important for the formation of the density profile and should be included in gyrokinetic edge and SOL simulations.

7.2.2. Temperature

Following [52], we define the temperature of the species σ as the kinetic energy of the plasma in a coordinate system in which the parallel flow $u_\sigma = 0$, i.e.

$$T_\sigma := \frac{2}{3n_\sigma} \int_0^\infty \int_{-\infty}^\infty \left(\frac{1}{2} m_\sigma \left(v_\parallel - \frac{u_\sigma}{n_\sigma} \right)^2 + \mu B \right) f_\sigma \, \mathrm{d}W \,, \tag{7.11}$$

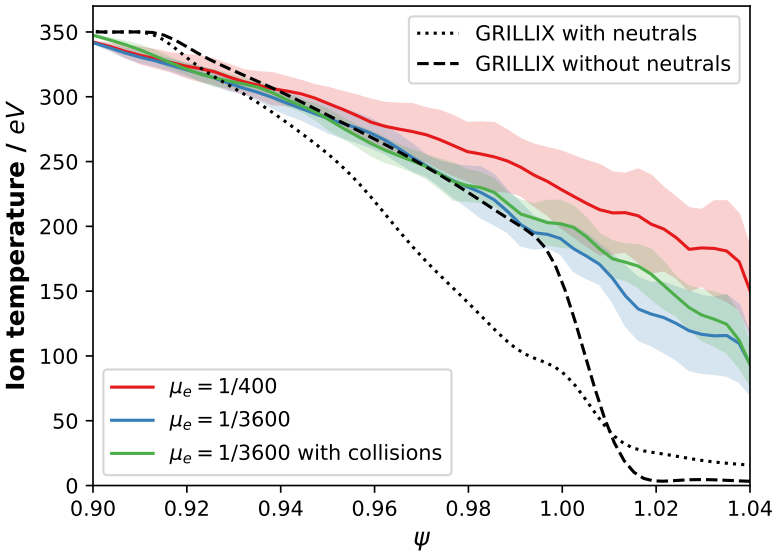

Figure 7.6.: Figure displaying the mean ion temperature profiles measured at the outboard mid-plane of the three different `GENE-X` simulations performed. In addition, the fluctuation amplitude is drawn as a shaded area around the mean. For comparison, the ion temperature profile of `GRILLIX` simulations with and without neutrals is shown.

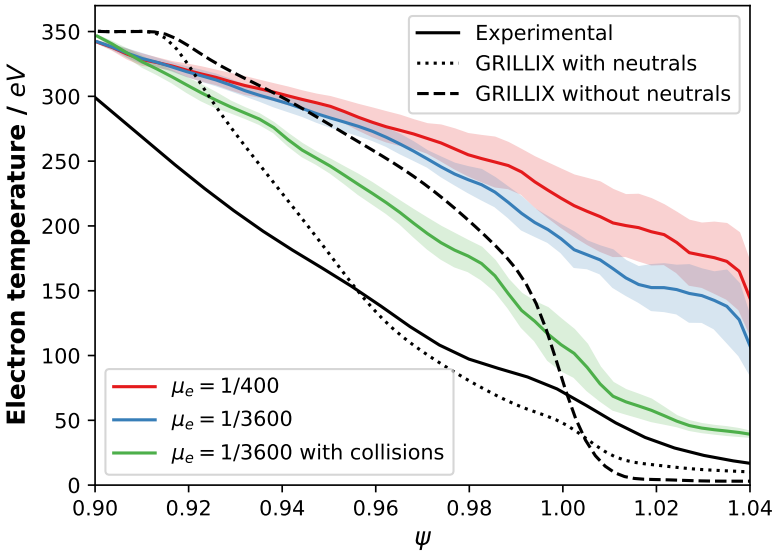

Figure 7.7.: Figure displaying the mean electron temperature profiles measured at the outboard mid-plane of the three different `GENE-X` simulations performed. In addition, the fluctuation amplitude is drawn as a shaded area around the mean. For comparison, the electron temperature profile of `GRILLIX` simulations with and without neutrals is shown.

where the expression for the density

$$n_\sigma := \int_0^\infty \int_{-\infty}^\infty f_\sigma \, \mathrm{d}W \tag{7.12}$$

and parallel flow

$$u_\sigma := \int_0^\infty \int_{-\infty}^\infty v_\parallel f_\sigma \, \mathrm{d}W \tag{7.13}$$

were used. The ion temperature profiles are shown in Fig. 7.6 and the electron temperature profiles in Fig. 7.7. We start with the analysis of the ion temperature profiles. In the confined region the ion temperature profiles of the simulations with realistic mass ratio and the `GRILLIX` simulation without neutrals agree well. The density of the simulation with reduced mass ratio is slightly increased. Similiar to the density profile, we observe that the fluctuation amplitude of the simulation with reduced mass ratio is increased. In the SOL we observe differences in the profiles. In the simulations with realisitic mass ratio the temperature is slightly reduced. Collisions do not seem to have an impact on the ion temperature. Nevertheless, the ion temperature of all three simulations is in the range of 100eV–250eV and significantly larger than the temperature floor of 35eV. We proceed with analysing the electron temperature profile. In contrast to the ion temperature, none of the measured electron temperature profiles agree with the `GRILLIX` simulations. The profiles of the simulations without collision are close to each other. Again, the temperature of the simulation with reduced mass ratio is slightly increased and the fluctuation amplitude is larger. The electron temperature profile with collisions differs significantly from the simulations without collisions. It is significantly lower and reaches values close to the temperature boundary value of 35eV in the SOL. Its slope agrees well with the experimental measurement. Comparing the velocity space of the simulations with and without collisions, we observe that the electrons are efficiently de-trapped. The perpendicular kinetic energy is transferred to the parallel kinetic energy and efficiently absorbed by the divertor plates. As observed above, this process is not as efficient for the ions because the collision frequency is lower. A possible explanation for this finding is the temperature boundary condition of 35eV that limits the temperature in the SOL. The ion-ion collision frequency scales as $T_i^{-3/2}$ and the ion-electron collision frequency approximately as $T_e^{-3/2}$. This implies that decreasing the ion and electron temperature boundary condition to more realistic values of 2eV would allow both the ion-ion and ion-electron collisionality to increase by a factor 73. Hence, the energy transfer between ions and electrons would become more efficient resulting in lower SOL temperatures for the ions. These findings demonstrate that collisions are essential for efficient heat transport in the SOL.

7.2.3. Radial electric field

The profiles of the radial electric field are shown in Fig. 7.8. The profiles are approximately flat in the confined region. The main generator of the radial electric field in the confined region is given by GAMs. They change their structure over time and average out over longer

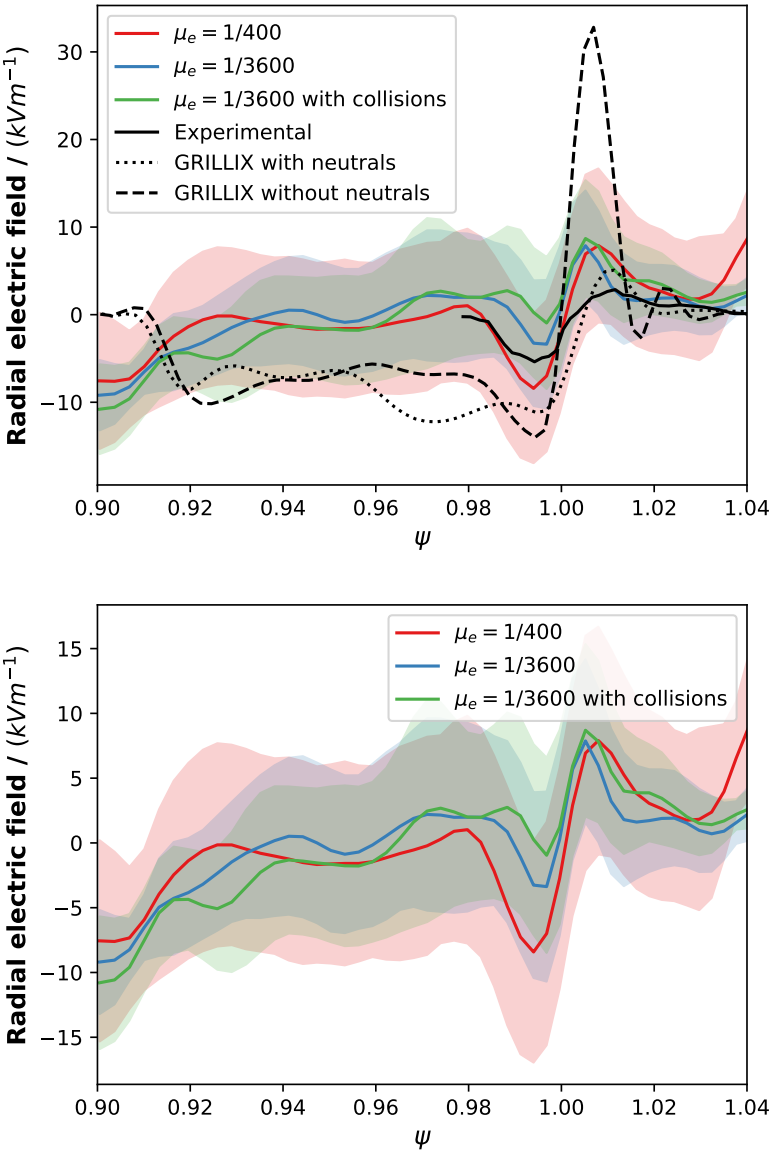

Figure 7.8.: Figure displaying the mean radial electric field measured at the outboard midplane of the three different `GENE-X` simulations performed. In addition, the fluctuation amplitude is drawn as a shaded area around the mean. For comparison, the radial electric field of `GRILLIX` simulations with and without neutrals and the experimental measurements is shown. The picture in the bottom excludes the `GRILLIX` simulations such that a close comparison between the `GENE-X` results and the experiment is possible.

time periods [165]. Consequently, the profiles in the confined region are flat after averaging over 100μs. Close to the separatrix we observe a well in the radial electric field in the confined region, and a hill in the SOL. This leads to an opposite poloidal rotation of the plasma inside and outside of the separatrix. The fluctuation amplitude of the radial electric field is large and on the same order of magnitude as the value of the field itself. The simulations performed with `GENE-X` reproduce the shape and amplitude of the experimental measurement better than the simulations performed with `GRILLIX`.

7.3. Power exhaust

Next, we analyse the power balance of the simulations. As we solve for the full-f Vlasov equation, we do not prescribe any input power to the simulations. Plasma density and energy is transferred into the plasma by the core and SOL boundaries. The plasma evolves freely and establishes a quasi-steady state. As the plasma β of the simulation is low, the main driver of cross field heat transport in the confined region is the radial component of the $E \times B$ heat flux. We define it as

$$q_{E\times B} := \sum_\sigma \int_0^\infty \int_{-\infty}^\infty \left(\frac{1}{2} m_\sigma v_\parallel^2 + \mu B\right) \left(\frac{c\boldsymbol{b}}{B_\parallel^*} \times \boldsymbol{\nabla}\phi_1\right) \cdot \boldsymbol{e}_\psi f_\sigma \, \mathrm{d}W \tag{7.14}$$

$$\approx \left(\frac{c\boldsymbol{b}}{B} \times \boldsymbol{\nabla}\phi_1\right) \cdot \boldsymbol{e}_\psi \sum_\sigma E_\sigma \,, \tag{7.15}$$

where E_σ denotes the kinetic energy of the plasma. In slight abuse of terminology, we call the radial component of the $E \times B$ heat flux the $E \times B$ heat flux in the following. The $E \times B$ heat flux of the simulations with realistic mass ratio, measured at $t = 1$ms, is shown in Fig. 7.9. The $E \times B$ heat flux is dominant on the low field side. We observe large radial streamers in the top and bottom of the device ranging from the core boundary to the separatrix. After crossing the separatrix the $E \times B$ heat flux falls off rapidly. We do not observe any significant $E \times B$ heat flux in the divertor region.

In the SOL the dominant driver of heat transport is the parallel heat flux. We define the parallel heat flux as

$$q_\parallel := \sum_\sigma \int_0^\infty \int_{-\infty}^\infty v_\parallel \left(\frac{1}{2} m_\sigma v_\parallel^2 + \mu B\right) f_\sigma \, \mathrm{d}W \,. \tag{7.16}$$

Compared to the temperature, presented in Eq. (7.11), that is calculated in a frame where $u_\sigma = 0$, Eqs. (7.15) and (7.16) account for the total energy flux via $E \times B$ and parallel transport and can therefore be used to calculate the transport of power through the device. We measure the power transported over the separatrix via the $E \times B$ drift by performing a surface integral of the $E \times B$ heat flux over the flux surface at $\psi = 0.999$. Like the electrostatic potential, the $E \times B$ heat flux fluctuates over time. Therefore, we average the result over the 24 poloidal planes and the last 100μs of the simulation and calculate the fluctuation amplitude as the standard deviation. The results are shown in Tab. 7.1. We observe that

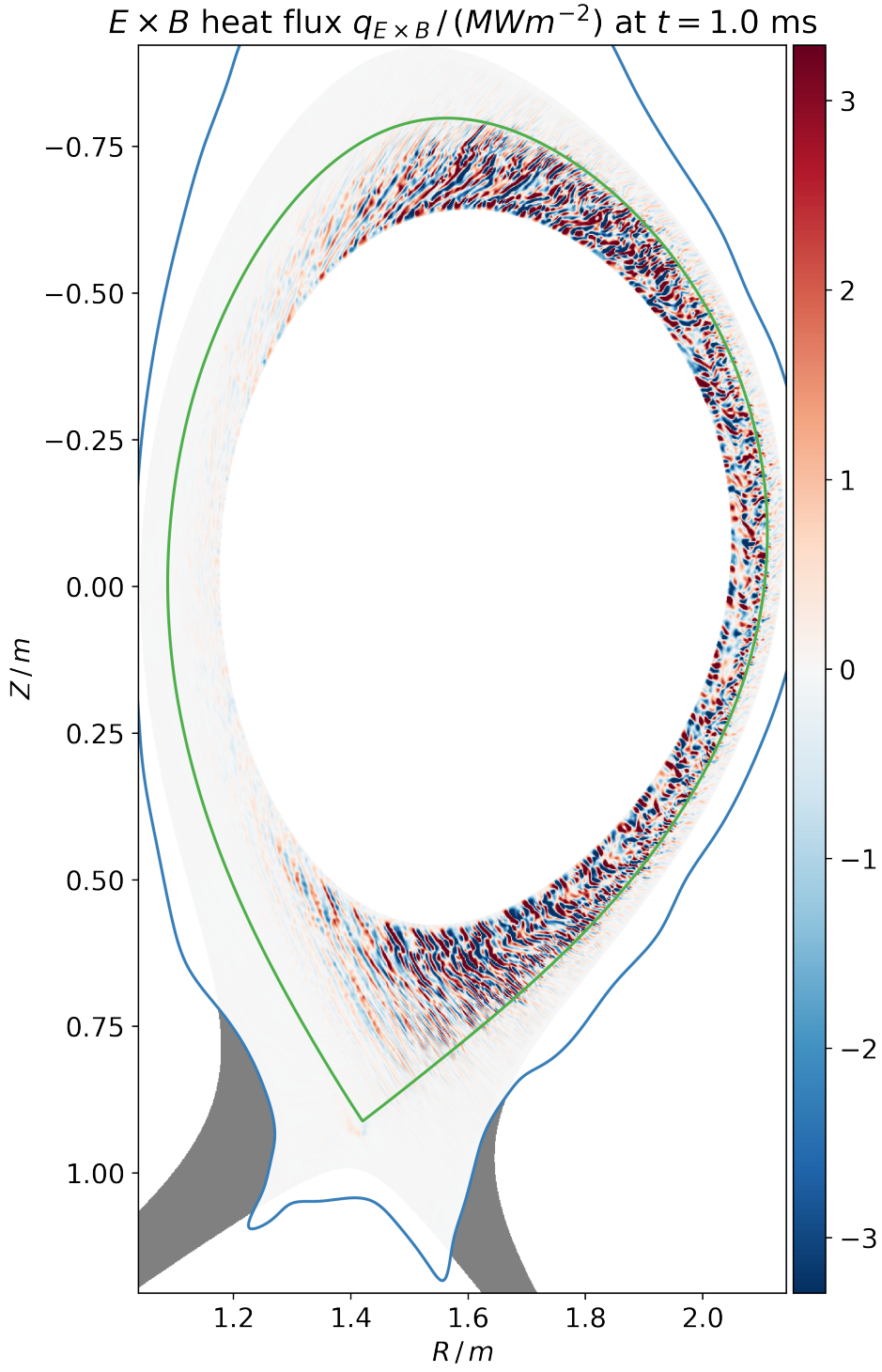

Figure 7.9.: Figure displaying the radial component of the $E \times B$ heat flux of the final state of the simulation on the poloidal plane at $\varphi = \pi$. Similar to Fig. 7.2, the separatrix, wall and divertor is shown. The $E \times B$ heat flux is dominant on the low field side and falls off rapidly after crossing the separatrix. We do not observe any significant $E \times B$ heat flux in the divertor region.

	Separatrix power	Divertor power
$\mu = 1/400$	$5.8\text{MW} \pm 0.4\text{MW}$	2.3MW
$\mu = 1/3600$	$1.8\text{MW} \pm 0.3\text{MW}$	0.7MW
$\mu = 1/3600$ with collisions	$1.4\text{MW} \pm 0.3\text{MW}$	1.1MW

Table 7.1.: Table displaying the total power flowing over the separatrix and onto the divertor for the three simulations performed.

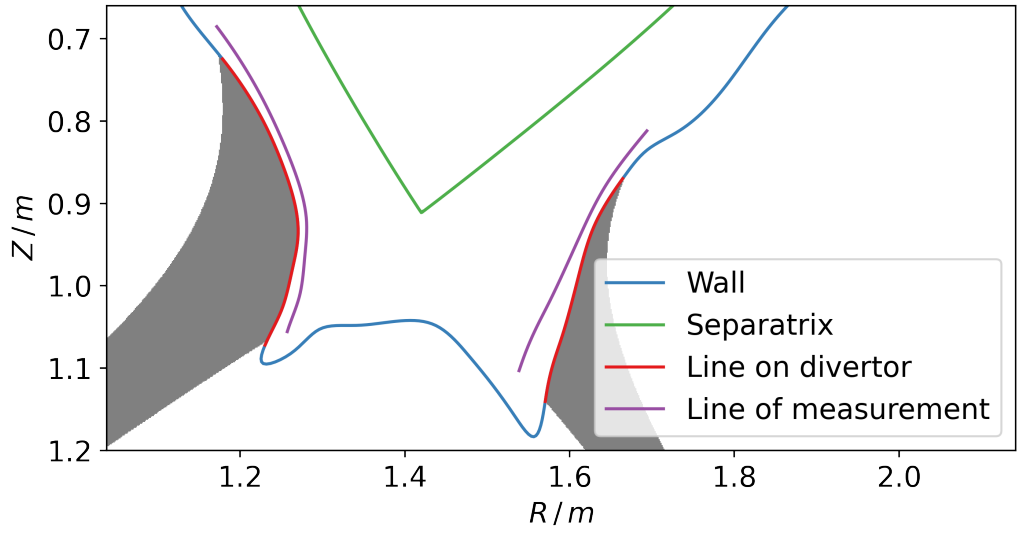

Figure 7.10.: Figure displaying the line of measurement of the parallel heat flux onto the divertor. The separatrix is drawn in green, the device wall in blue, the divertor legs in grey and the boundary line of the divertor in red. The line of measurement, drawn in purple, is calculated by following the points on the boundary line of the divertor in opposite direction of the magnetic field by a toroidal angle of $\pi/8$. This moves the line of measurement away from the divertor such that the heat flux is not given by the chosen boundary condition.

the power transported over the separatrix is the largest for the simulation with reduced mass ratio. This finding is consistent with the increased level of turbulence that we discussed in the analysis of the plasma profiles. For the simulations with realistic mass ratio the power fluctuates around 1.4MW–1.8MW. The heating power in the experiment, determined by the power balance, is equal to 475 kW. After crossing the separatrix and leaving the confined region, the energy is transported along the field lines onto the divertor. We measure the heat flux and total power hitting the divertor. This process is not straightforward as the plasma at the divertor is set by the boundary conditions, the divertor has a complicated shape and the magnetic field lines hit the divertor at a shallow incidence angle. We solve the first problem by measuring the heat flux on a line in front of the divertor. The line is generated by following the points on the divertor along the magnetic field lines by a toroidal angle of $\pi/8$. The line of measurement is shown in Fig. 7.10. On the line of measurement we interpolate the parallel

heat flux with cubic spline interpolation. To solve the last problem and account for the reduction of heat flux onto the divertor due to the shallow incident angle, we calculate the normalised normal vector $\boldsymbol{n}$ to the line of measurement and reduce the interpolated parallel heat flux by the factor $\sin\theta = \boldsymbol{n} \cdot \boldsymbol{b}$, where θ denotes the incidence angle of the magnetic field line. The measured $\sin\theta$ varies between 0.03 on the lower part of the divertor and 0.01 on the upper part of the divertor. To measure the SOL fall-off length λ_q, we fit the function

$$q(\overline{s}) = \frac{q_0}{2} \exp\left(\left(\frac{s}{2\lambda_q}\right)^2 - \frac{\overline{s}}{\lambda_q}\right) \operatorname{erfc}\left(\frac{s}{2\lambda_q} - \frac{\overline{s}}{s}\right) + q_{\mathrm{BG}}\,, \tag{7.17}$$

described in [26], to the electron heat flux.

$$\operatorname{erfc}(x) = 1 - \operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2}\,\mathrm{d}t \tag{7.18}$$

denotes the complementary error function [166]. q_0 denotes the peak heat flux, s the power spreading factor and q_{BG} the background heat flux. To remove the influence of the flux expansion from the OMP to the line of measurement, we perform the fit with the OMP mapped distance s instead of the length of the line of measurement. The OMP mapped distance is calculated by following the points on the line of measurements along the magnetic field lines until they hit the OMP. The OMP mapped distance is measured as the distance of the intersection points to the separatrix. The measured heat flux profiles and the corresponding fit for the divertor at the low field side are shown for the simulation with reduced mass ratio in Fig. 7.11, for the simulation with realistic mass ratio in Fig. 7.12 and for the simulation with realistic mass ratio and collisions in Fig. 7.13. We observe that the electron heat flux is much larger than the ion heat flux and more peaked close to the separatrix. The measured shape of the electron and total heat flux profile is well described by the Eich model presented in Eq. (7.17). The measured λ_q reads 12mm for the run with reduced mass ratio, 1.9mm for the run with realistic mass ratio and 6.7mm for the run with realistic mass ratio and collisions. The increase in λ_q for the simulations with reduced mass ratio agrees well with the observation of increased turbulence in the profile analysis above. The stronger the turbulence, the further the particles are transported into the SOL. In addition, the reduced mass ratio decreases the electron heat conductivity which slows the parallel heat transport in comparison to the radial $E \times B$ transport and leads to a further broadening of λ_q. The reason for the increase in λ_q for the simulation with collisions is similar. Due to collisions, fast electrons are decelerated which reduces the parallel heat conductivity and leads to a broadening of λ_q by the same mechanism [80]. The measured λ_q from the experiment is around 3mm–4mm and lies in between the results of our simulations with and without collisions. It seems that collisions have an important influence on the broadening of the λ_q, and further investigations with a more sophisticated collision operator are needed. The measured peak heat flux from the experiment is around $0.6\mathrm{MW/m^2} - 0.65\mathrm{MW/m^2}$ and approximately by a factor of 2–3 smaller than the results of our simulations. This finding is consistent with the input power of the experiment, mentioned above, that is approximately by a factor of 2.9–3.8 smaller than

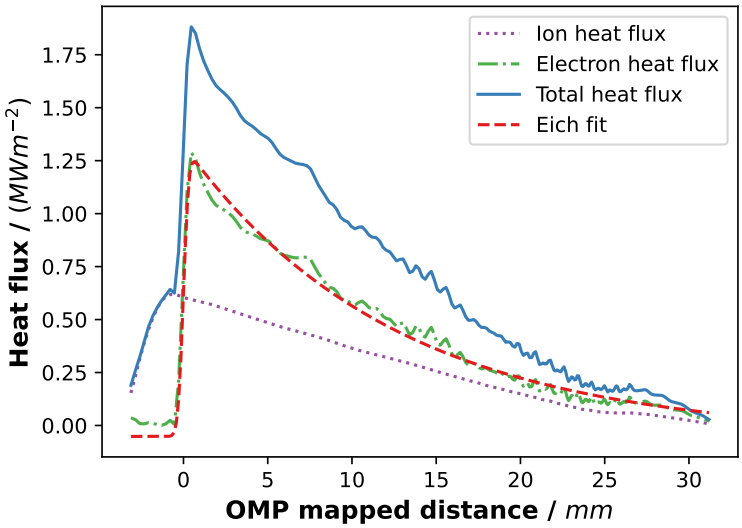

Figure 7.11.: Figure displaying the ion, electron and total heat flux onto the divertor for the simulation with reduced mass ratio. Furthermore, the Eich fit of the total heat flux is drawn. The heat flux is measured as a function of the OMP mapped distance to account for the flux expansion between the OMP and the divertor. The measured SOL fall-off length is $\lambda_q = 12\text{mm}$.

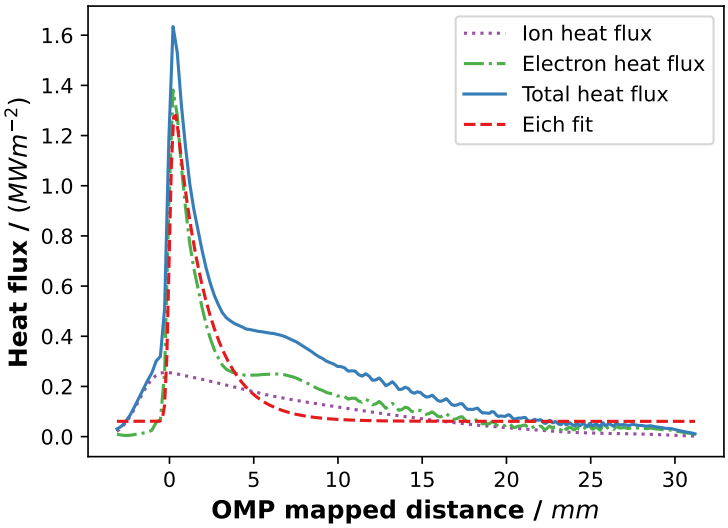

Figure 7.12.: Figure displaying the ion, electron and total heat flux onto the divertor, analogous to Fig. 7.11, for the simulation with realistic mass ratio. The measured SOL fall-off length is $\lambda_q = 1.9\text{mm}$.

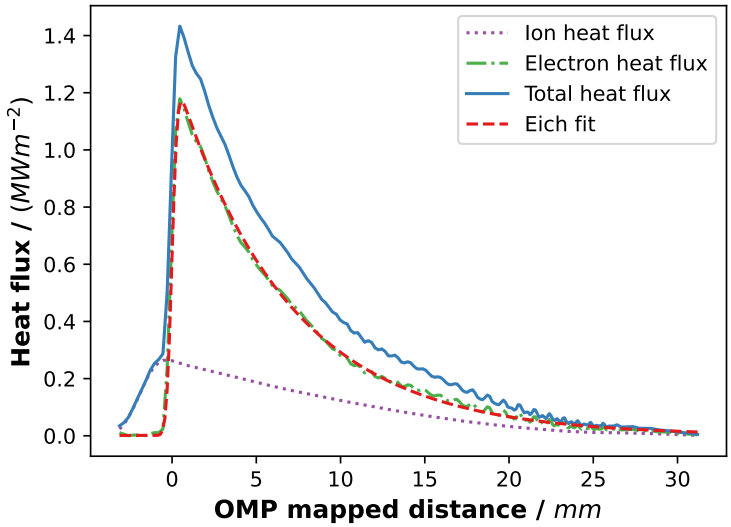

Figure 7.13.: Figure displaying the ion, electron and total heat flux onto the divertor, analogous to Fig. 7.11, for the simulation with realistic mass ratio and collisions. The measured SOL fall-off length is $\lambda_q = 6.7$mm.

in the simulation.

The total power absorbed by the divertor on the low and high field side is displayed in Tab. 7.1. The measurements are consistent with the $E \times B$ heat flux. Between 40%–80% of the power crossing the separatrix is absorbed by the divertor. The rest part is transported to the divertor via different mechanisms or is absorbed by the main wall.

7.4. Conclusion and outlook

In this chapter we presented the first `GENE-X` simulations in diverted geometry including a magnetic X-point. We performed the simulations with three different configurations. The first configuration used a reduced electron-to-ion mass ratio of $\mu = m_e/m_i = 1/400$, the second a realistic electron-to-deuterium mass ratio of $m_e/m_i = 1/3600$ and the third a realistic mass ratio with a BGK collision operator. All simulations were performed over a timescale of 1ms in the magnetic equilibrium and at the parameters of the ASDEX Upgrade discharge #36190. We compared the profiles of density, temperature and the radial electric field to experimental measurements and simulations with the Braginskii fluid code `GRILLIX`. We found that collisions have an impact on the temperature but not on the density profile. The simulations with collisions were able to reproduce aspects of the experimental temperature profiles. The radial electric field, obtained with `GENE-X`, agreed well with the experimental measurement. We analysed the power balance of the simulations by measuring the $E \times B$ heat flux flowing over the separatrix and the parallel heat flux onto the divertor. In addition, we calculated λ_q by performing an Eich fit to the parallel heat flux profile in front of the divertor. The shape of the measured heat flux profile agreed well with the theoretical prediction by the Eich model. We observed that collisions broaden λ_q. In the simulations with reduced mass ratio

the fluctuation amplitude was increased and λ_q broadened. This indicated that a realistic mass ratio is important for realistic edge and SOL simulations.
As collisions have shown to have a significant effect on the plasma temperature and λ_q, it will be important to use a more realistic collision model in the future. As mentioned in the outlook of Chapter 2, implementing a Fokker-Planck type Lennard-Bernstein/Dogherty collision operator [114, 115] is current work in progress [116]. The SOL profiles for the ions seemed to be significantly affected by the temperature boundary condition of 35eV. For future studies it is vital to reduce the SOL temperature to more realistic values around 2eV and find solutions to efficiently represent temperature differences of two orders of magnitude on the velocity space grid. One possible candidate to achieve this could be the implementation of block structured grids in `GENE-X` [167, 168]. Finally, it would be interesting to study the effect of neutral physics, similar to [80], on the plasma profiles and the heat exhaust.

8. Summary and outlook

Understanding and predicting the effects of turbulence in the edge and SOL belongs to one of the most challenging problems on the path towards a magnetic confinement fusion power plant. Turbulence is characterised by a, cascade like, transport of energy between different scales and is believed to play a crucial role in the L-H transition and the particle and heat exhaust of fusion devices.
In this work, we presented a new gyrokinetic turbulence code, `GENE-X`, that is tailored towards edge and SOL simulations. Developing such a code is a difficult task and multiple challenges had to be overcome. We presented detailed conclusions and outlooks at the end of each chapter. In the following, we summarise the highlights of this work and provide a unified outlook.

8.1. Summary

We started this work, in Chapter 2, with a precise discussion of gyrokinetic theory and the derivation of the self-consistent model implemented in the `GENE-X` code. The model included the full distribution function of ions and electrons and is, therefore, capable of handling large fluctuation amplitudes in SOL plasmas. The model was consistently derived from the second order gyrokinetic Lagrangian and conserves energy as well as the number of particles on the continuous level.
We continued, in Chapter 3, by discussing the coordinate system used for the simulations. The magnetic geometry of the edge and SOL is complex. Magnetic field lines are not confined to flux surfaces and can intersect with the device wall. In addition, there is a transition from closed to open magnetic flux surfaces which renders flux-coordinate systems, that have been used in fusion theory for decades, ineffective. We solved this problem by implementing a locally field-aligned coordinate system based on the FCI approach. We discussed the coordinate transformation, derived expressions for the differential operators and expressed the gyrokinetic Vlasov-Maxwell system in the new coordinate system. Using this technology we are able to perform simulations without restrictions by the magnetic geometry. This includes magnetic geometries with single and even multiple magnetic X-points.
In Chapter 4 we described the discretisation scheme used. The choice of the discretisation scheme was motivated by the locally field-aligned coordinate system and constructed with the method of lines. The spatial discretisation featured centred finite differences and the solution of the remaining, temporal initial value problem was done with a Runge-Kutta method of order four.

As the implementation of scientific codes is complex and their execution resource intensive, we put, in Chapter 5, a special emphasis on the software design and the computational performance of the implementation. The design principles behind the code were explained in detail and visualised with the help of UML diagrams. The analysis of the performance comprised two steps. First, we analysed the intra-node performance of the implementation and verified it against the roofline model. Second, we demonstrated that the inter-node performance is excellent by measuring the strong and weak scaling at the example of a real application. Finally, we described the continuous integration workflow with automatic testing that is used extensively in the `GENE-X` code and ensures that the implementation of the gyrokinetic Vlasov-Maxwell system is correct.

In Chapter 6, we ensured that the numerical scheme, using the FCI approach, solves the gyrokinetic Vlasov-Maxwell system correctly. We performed a comprehensive list of benchmarks testing different aspects of the code. First, we used the method of manufactured solution to demonstrate that the numerical scheme converges to second order accuracy. The tests were performed in slab, circular and screw pinch geometry. Second, we measured the growth rates of ITG modes in a screw pinch geometry. This test allowed for a precise comparison between standard flux coordinates and the locally field-aligned coordinate system. Third, we verified the electromagnetic part of the model by measuring the frequency and Landau damping of Alfv́en in slab geometry. Finally, we probed the complete gyrokinetic Vlasov-Maxwell system in toroidal geometry by measuring the frequency and growth rates of ITG modes in a cyclone base case simulation.

To close this work, we presented simulations of the edge and SOL of AUG in Chapter 7 that included the separatrix and the X-point. The simulations were performed over 1ms of plasma evolution until profile saturation. We compared the profiles of density, temperature and the radial electric field to experimental measurements and previous `GRILLIX` simulations. We found that simulation performed with a BGK collision operator were able to reproduce aspects of the experimental profiles quite well. Further, we calculated the power exhaust of the simulation and measured λ_q. The overall shape of the heat flux profile, measured close to the divertor, agreed well with the theoretical prediction. We observed that collisions broaden λ_q. The simulations performed with a reduced mass ratio of $m_e/m_i = 1/400$ showed an increased fluctuation level and a significantly broader λ_q, which makes the application of reduced mass ratio simulations for edge and SOL plasmas questionable.

8.2. Outlook

With the development of the `GENE-X` code, we have taken a first step towards a predictive tool to study turbulence in the edge and SOL of fusion devices. There are numerous interesting ways to improve and apply the code in the future. We split the future work into three categories: (i) Improvements of the plasma model, (ii) improvements of the numerical scheme and (iii) future applications of the `GENE-X` code.

8.2.1. Improvements of the gyrokinetic model

To improve the predictability of the code, it is important to model effects specific to the SOL more precisely. This includes:

- The implementation of a more sophisticated Fokker-Planck type collision operator. Currently, the implementation of an energy conservative, full-f Lennard-Bernstein/Dogherty collision operator [114, 115] is work in progress [116].
- The implementation of a neutral physics model. A solid starting point could be the simple neutral model implemented in the `GRILLIX` code recently [80].
- The implementation of sheath boundary conditions [169].
- Removing the linearisation of the quasi-neutrality equation, made in Eq. (2.76), to further improve the full-f capabilities of the code.

8.2.2. Improvements of the discretisation scheme

While the discretisation scheme chosen has been shown to work well, major improvements to the discretisation scheme can still be done on two fronts:

- The implementation of a new grid in velocity space that allows to represent the velocity structure of plasmas with temperatures that vary over multiple orders of magnitude. A possible candidate are block-structured grids, introduced in [167, 168].
- The development and implementation of a numerical scheme based on mimetic finite differences that conserves the particle number and if possible even the momentum and energy of the discrete system.

8.2.3. Future applications of the `GENE-X` code

Finally, there are many interesting ways to use the `GENE-X` code in the future. This includes:

- Performing simulations of H-mode plasmas to investigate the L-H transition.
- Performing a validation of the `GENE-X` code in X-point geometry. This could for instance be achieved by following the recent validation of Braginskii fluid codes presented in [170].
- Generalising the `GENE-X` code to 3D geometries to investigate turbulence in the edge and SOL of stellarator fusion devices – including magnetic islands and stochastic regions.
- Performing whole device turbulence simulations of a medium size tokamak, like the Tokamak à Configuration Variable (TCV) [171], that range all the way from the magnetic axis, over the separatrix into the scrape-off layer.

A. Supplemental derivations

A.1. Explicit form of the discretised parallel derivatives

In this section, we provide the explicit form of the derivatives of the Lagrange polynomials used in Eqs. (4.35) and (4.49). Consider five grid points in y direction $(y_{-2}, y_{-1}, y_0, y_1, y_2)$. The first derivatives of the Lagrange polynomials read

$$\left(\frac{\mathrm{d}\ell_{-2}}{\mathrm{d}y}\right)(0) = \frac{-(y_{-1}-y_0)(y_0-y_1)(y_0-y_2)}{(y_{-2}-y_{-1})(y_{-2}-y_0)(y_{-2}-y_1)(y_{-2}-y_2)} \tag{A.1}$$

$$\left(\frac{\mathrm{d}\ell_{-1}}{\mathrm{d}y}\right)(0) = \frac{(y_{-2}-y_0)(y_0-y_1)(y_0-y_2)}{(y_{-2}-y_{-1})(y_{-1}-y_0)(y_{-1}-y_1)(y_{-1}-y_2)} \tag{A.2}$$

$$\begin{aligned}\left(\frac{\mathrm{d}\ell_0}{\mathrm{d}y}\right)(0) = &\frac{y_{-1}(3y_0^2+y_1y_2-2y_0(y_1+y_2))+y_0(-4y_0^2-2y_1y_2+3y_0(y_1+y_2))}{(y_{-2}-y_0)(-y_{-1}+y_0)(y_0-y_1)(y_0-y_2))} \\ &+\frac{y_{-2}(3y_0^2+y_1y_2-2y_0(y_1+y_2)+y_{-1}(-2y_0+y_1+y_2))}{(y_{-2}-y_0)(-y_{-1}+y_0)(y_0-y_1)(y_0-y_2)}\end{aligned} \tag{A.3}$$

$$\left(\frac{\mathrm{d}\ell_1}{\mathrm{d}y}\right)(0) = \frac{(y_{-2}-y_0)(-y_{-1}+y_0)(y_0-y_2)}{(y_{-2}-y_1)(-y_{-1}+y_1)(-y_0+y_1)(y_1-y_2)} \tag{A.4}$$

$$\left(\frac{\mathrm{d}\ell_2}{\mathrm{d}y}\right)(0) = \frac{(y_{-2}-y_0)(-y_{-1}+y_0)(y_0-y_1)}{(y_{-2}-y_2)(-y_{-1}+y_2)(-y_0+y_2)(-y_1+y_2)} \tag{A.5}$$

and the fourth derivatives of the Lagrange polynomials read

$$\left(\frac{\mathrm{d}^4\ell_{-2}}{\mathrm{d}y^4}\right)(0) = \frac{24}{(y_{-2}-y_{-1})(y_{-2}-y_0)(y_{-2}-y_1)(y_{-2}-y_2)} \tag{A.6}$$

$$\left(\frac{\mathrm{d}^4\ell_{-1}}{\mathrm{d}y^4}\right)(0) = \frac{-24}{(y_{-2}-y_{-1})(y_{-1}-y_0)(y_{-1}-y_1)(y_{-1}-y_2)} \tag{A.7}$$

$$\left(\frac{\mathrm{d}^4\ell_0}{\mathrm{d}y^4}\right)(0) = \frac{-24}{(y_{-2}-y_0)(-y_{-1}+y_0)(y_0-y_1)(y_0-y_2)} \tag{A.8}$$

$$\left(\frac{\mathrm{d}^4\ell_1}{\mathrm{d}y^4}\right)(0) = \frac{-24}{(y_{-2}-y_1)(-y_{-1}+y_1)(-y_0+y_1)(y_1-y_2)} \tag{A.9}$$

$$\left(\frac{\mathrm{d}^4\ell_2}{\mathrm{d}y^4}\right)(0) = \frac{24}{(y_{-2}-y_2)(y_{-1}-y_2)(y_0-y_2)(y_1-y_2)}\,. \tag{A.10}$$

These relations can be used together with Eqs. (4.35) and (4.49) to calculate the first and fourth derivative on the unstructured y grid via a stencil operation.

A.2. Explicit form of the gyrokinetic Vlasov-Maxwell system

In this section, we present the explicit form of the normalised gyrokinetic Vlasov-Maxwell system, expressed in the locally field-aligned coordinate system, that is implemented in the `GENE-X` code. The gyrokinetic Vlasov equation reads

$$\begin{aligned}
&\frac{\partial \hat{f}_\sigma}{\partial \hat{t}} + \underbrace{\sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}}\hat{v}_\parallel \frac{\hat{\boldsymbol{B}}^*}{\hat{B}^*_\parallel}\cdot\hat{\boldsymbol{\nabla}}\hat{f}_\sigma}_{(a)} + \underbrace{\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{\boldsymbol{b}}{\hat{q}_\sigma \hat{B}^*_\parallel}\times\left(\hat{T}_{0\sigma}(\psi_0)\hat{\mu}\hat{\boldsymbol{\nabla}}\hat{B} + \hat{q}_\sigma\hat{\boldsymbol{\nabla}}\hat{\phi}_1\right)\hat{\boldsymbol{\nabla}}\hat{f}_\sigma}_{(b)} \\
&\quad \underbrace{- \frac{\hat{\boldsymbol{B}}^*}{\sqrt{2\hat{m}_\sigma\hat{T}_{0\sigma}(\psi_0)}\hat{B}^*_\parallel}\cdot\left(\hat{T}_{0\sigma}(\psi_0)\hat{\mu}\hat{\boldsymbol{\nabla}}\hat{B} + \hat{q}_\sigma\hat{\boldsymbol{\nabla}}\hat{\phi}_1\right)\frac{\partial \hat{f}_\sigma}{\partial \hat{v}_\parallel}}_{(c)} - \frac{\hat{q}_\sigma}{\sqrt{2\hat{m}_\sigma\hat{T}_{0\sigma}(\psi_0)}}\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{t}}\frac{\partial \hat{f}_\sigma}{\partial v_\parallel} \\
&\quad = 0\,,
\end{aligned} \tag{A.11}$$

with the components

$$\begin{aligned}
(a) = &\sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}}\hat{v}_\parallel\frac{\hat{B}}{\hat{B}^*_\parallel}\frac{\partial \hat{f}_\sigma}{\partial \hat{y}} \\
&+ \frac{2\hat{v}_\parallel^2\hat{T}_{0\sigma}(\psi_0)}{\hat{B}^*_\parallel\hat{q}_\sigma}\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{1}{b_{y_c}}\left(\frac{\partial b_{z_c}}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{x}} - \frac{\partial b_{x_c}}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{z}} + \left(\frac{\partial b_{x_c}}{\partial \hat{z}} - \frac{\partial b_{z_c}}{\partial \hat{x}}\right)\frac{\partial \hat{f}_\sigma}{\partial \hat{y}}\right) \\
&+ \sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}}\frac{\hat{v}_\parallel}{\hat{B}^*_\parallel}\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\left(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{z}}\frac{\partial \hat{f}_\sigma}{\partial \hat{y}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{z}}\right) + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{x}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{x}}\frac{\partial \hat{f}_\sigma}{\partial \hat{y}}\right)\right. \\
&\qquad\left. + \frac{1}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{x}}\frac{\partial \hat{f}_\sigma}{\partial \hat{z}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{z}}\frac{\partial \hat{f}_\sigma}{\partial \hat{x}}\right)\right),
\end{aligned} \tag{A.12}$$

$$\begin{aligned}
(b) = \frac{1}{B^*_\parallel\hat{q}_\sigma}\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\Bigg[\quad &\hat{T}_{0\sigma}(\psi_0)\hat{\mu}\left(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial \hat{B}}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{z}} - \frac{\partial \hat{B}}{\partial \hat{z}}\frac{\partial \hat{f}_\sigma}{\partial \hat{y}}\right) + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial \hat{B}}{\partial \hat{x}}\frac{\partial \hat{f}_\sigma}{\partial \hat{y}} - \frac{\partial \hat{B}}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{x}}\right)\right. \\
&\qquad\left. + \frac{1}{b_{y_c}}\left(\frac{\partial \hat{B}}{\partial \hat{z}}\frac{\partial \hat{f}_\sigma}{\partial \hat{x}} - \frac{\partial \hat{B}}{\partial \hat{x}}\frac{\partial \hat{f}_\sigma}{\partial \hat{z}}\right)\right) \\
&+ q_\sigma\left(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial \hat{\phi}_1}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{z}} - \frac{\partial \hat{\phi}_1}{\partial \hat{z}}\frac{\partial f_\sigma}{\partial \hat{y}}\right) + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial \hat{\phi}_1}{\partial \hat{x}}\frac{\partial \hat{f}_\sigma}{\partial \hat{y}} - \frac{\partial \hat{\phi}_1}{\partial \hat{y}}\frac{\partial \hat{f}_\sigma}{\partial \hat{x}}\right)\right. \\
&\qquad\left.\left. + \frac{1}{b_{y_c}}\left(\frac{\partial \hat{\phi}_1}{\partial \hat{z}}\frac{\partial \hat{f}_\sigma}{\partial \hat{x}} - \frac{\partial \hat{\phi}_1}{\partial \hat{x}}\frac{\partial \hat{f}_\sigma}{\partial \hat{z}}\right)\right)\right] \quad \text{and}
\end{aligned} \tag{A.13}$$

$$
\begin{aligned}
(c) = \frac{1}{\sqrt{2\hat{m}_\sigma \hat{T}_{0\sigma}(\psi_0)}\hat{B}^*_\parallel}\Bigg[\quad & \hat{T}_{0\sigma}(\psi_0)\hat{\mu}\Bigg(\hat{B}\frac{\partial \hat{B}}{\partial \hat{y}} + \sqrt{2m_\sigma \hat{T}_{0\sigma}(\psi_0)}\frac{\hat{v}_\parallel}{\hat{q}_\sigma}\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{1}{b_{y_c}} \\
& \left(\frac{\partial b_{z_c}}{\partial \hat{y}}\frac{\partial \hat{B}}{\partial \hat{x}} - \frac{\partial b_{x_c}}{\partial \hat{y}}\frac{\partial \hat{B}}{\partial \hat{z}} + \left(\frac{\partial b_{x_c}}{\partial \hat{z}} - \frac{\partial b_{z_c}}{\partial \hat{x}}\right)\frac{\partial \hat{B}}{\partial \hat{y}}\right) \\
& + \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\Bigg(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{z}}\frac{\partial \hat{B}}{\partial \hat{y}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{y}}\frac{\partial \hat{B}}{\partial \hat{z}}\right) \\
& + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{y}}\frac{\partial \hat{B}}{\partial \hat{x}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{x}}\frac{\partial \hat{B}}{\partial \hat{y}}\right) \\
& + \frac{1}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{x}}\frac{\partial \hat{B}}{\partial \hat{z}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{z}}\frac{\partial \hat{B}}{\partial \hat{x}}\right)\Bigg) \\
& + q_\sigma\Bigg(\hat{B}\frac{\partial \hat{\phi}_1}{\partial \hat{y}} + \sqrt{2m_\sigma \hat{T}_{0\sigma}(\psi_0)}\frac{\hat{v}_\parallel}{\hat{q}_\sigma}\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{1}{b_{y_c}}\Bigg(\frac{\partial b_{z_c}}{\partial \hat{y}}\frac{\partial \hat{\phi}_1}{\partial \hat{x}} - \frac{\partial b_{x_c}}{\partial \hat{y}}\frac{\partial \hat{\phi}_1}{\partial \hat{z}} \\
& \left(\frac{\partial b_{x_c}}{\partial \hat{z}} - \frac{\partial b_{z_c}}{\partial \hat{x}}\right)\frac{\partial \hat{\phi}_1}{\partial \hat{y}}\Bigg) \\
& + \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\Bigg(\frac{b_{x_c}}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{z}}\frac{\partial \hat{\phi}_1}{\partial \hat{y}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{y}}\frac{\partial \hat{\phi}_1}{\partial \hat{z}}\right) \\
& + \frac{b_{z_c}}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{y}}\frac{\partial \hat{\phi}_1}{\partial \hat{x}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{x}}\frac{\partial \hat{\phi}_1}{\partial \hat{y}}\right) \\
& + \frac{1}{b_{y_c}}\left(\frac{\partial \hat{A}_{1\parallel}}{\partial \hat{x}}\frac{\partial \hat{\phi}_1}{\partial \hat{z}} - \frac{\partial \hat{A}_{1\parallel}}{\partial \hat{z}}\frac{\partial \hat{\phi}_1}{\partial \hat{x}}\right)\Bigg)\Bigg)\Bigg]\frac{\partial \hat{f}_\sigma}{\partial v_\parallel}\,. \qquad \text{(A.14)}
\end{aligned}
$$

The quasi-neutrality equation reads

$$
\begin{aligned}
& -\left(\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\right)^2\left(\frac{1}{\hat{\mathcal{J}}_{\mathrm{cyl}}}\frac{\partial}{\partial \hat{x}}\left(\hat{\mathcal{J}}_{\mathrm{cyl}}\sum_\sigma \frac{\hat{m}_\sigma \hat{n}_{0\sigma}}{\hat{B}^2}\frac{\partial}{\partial \hat{x}}\right) + \frac{\partial}{\partial \hat{z}}\left(\sum_\sigma \frac{\hat{m}_\sigma \hat{n}_{0\sigma}}{\hat{B}^2}\frac{\partial}{\partial \hat{z}}\right)\right)\hat{\phi}_1 \\
& \qquad = \sum_\sigma \hat{n}_{0\sigma}(\psi_0)\hat{q}_\sigma \int_0^\infty \int_{-\infty}^\infty \pi \hat{B}^*_\parallel \hat{f}_\sigma \,\mathrm{d}\hat{v}_\parallel \,\mathrm{d}\hat{\mu}\,, \qquad \text{(A.15)}
\end{aligned}
$$

Ampère's law reads

$$
\begin{aligned}
& -\left(\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\right)^2\left(\frac{1}{\hat{\mathcal{J}}_{\mathrm{cyl}}}\frac{\partial}{\partial \hat{x}}\left(\hat{\mathcal{J}}_{\mathrm{cyl}}\frac{\partial}{\partial \hat{x}}\right) + \frac{\partial^2}{\partial \hat{z}^2}\right)\hat{A}_{1\parallel} \\
& \qquad = \frac{\beta_{\mathrm{ref}}}{2}\sum_\sigma \hat{n}_{0\sigma}(\psi_0)\hat{q}_\sigma\sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}}\int_0^\infty \int_{-\infty}^\infty \pi \hat{B}^*_\parallel \hat{f}_\sigma \hat{v}_\parallel \,\mathrm{d}\hat{v}_\parallel \,\mathrm{d}\hat{\mu} \qquad \text{(A.16)}
\end{aligned}
$$

and the generalised Ohm's law reads

$$
\begin{aligned}
-\Bigg[\left(\frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\right)^2 \Bigg(\frac{1}{\hat{\mathcal{J}}_{\mathrm{cyl}}}\frac{\partial}{\partial\hat{x}}\left(\hat{\mathcal{J}}_{\mathrm{cyl}}\frac{\partial}{\partial\hat{x}}\right)+\frac{\partial^2}{\partial\hat{z}^2}\Bigg) \\
+\frac{\beta_{\mathrm{ref}}}{2}\sum_\sigma \hat{n}_{0\sigma}(\psi_0)\frac{\hat{q}_\sigma^2}{\hat{m}_\sigma}\int_0^\infty\int_{-\infty}^\infty \pi\hat{B}_\parallel^*\frac{\partial\hat{f}_\sigma}{\partial\hat{v}_\parallel}\hat{v}_\parallel\,\mathrm{d}\hat{v}_\parallel\,\mathrm{d}\hat{\mu}\Bigg]\frac{\partial\hat{A}_{1\parallel}}{\partial\hat{t}} \\
=\frac{\beta_{\mathrm{ref}}}{2}\sum_\sigma \hat{n}_{0\sigma}(\psi_0)\hat{q}_\sigma\sqrt{\frac{2\hat{T}_{0\sigma}(\psi_0)}{\hat{m}_\sigma}}\int_0^\infty\int_{-\infty}^\infty \pi\hat{B}_\parallel^*\left(\frac{\partial f_\sigma}{\partial t}\right)^\star\hat{v}_\parallel\,\mathrm{d}\hat{v}_\parallel\,\mathrm{d}\hat{\mu}\,. \quad \text{(A.17)}
\end{aligned}
$$

In the implementation we set $\hat{T}_{0\sigma}(\psi_0)=1$ and $\hat{n}_{0\sigma}(\psi_0)=1$.

A.3. Identities of the plasma dispersion function

The plasma dispersion function is defined as [172]

$$Z : \omega \mapsto Z(\omega) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \frac{e^{-t^2}}{t-\omega} \, \mathrm{d}t \,. \tag{A.18}$$

It fulfils the identities

$$\begin{aligned} \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \frac{t e^{-t^2}}{t-\omega} \, \mathrm{d}t &= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} 1 e^{-t^2} + \frac{\omega}{t-\omega} e^{-t^2} \, \mathrm{d}t \\ &= 1 + \omega Z(\omega) \end{aligned} \tag{A.19}$$

and

$$\begin{aligned} \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \frac{t^2 e^{-t^2}}{t-\omega} \, \mathrm{d}t = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} (t+\omega) e^{-t^2} + \frac{\omega^2}{t-\omega} e^{-t^2} \, \mathrm{d}t = \omega + \omega^2 Z(\omega) \\ = \omega(1 + \omega Z(\omega)) \,. \end{aligned} \tag{A.20}$$

A.4. Proof of an integral identity of the Poisson bracket

In this section, we prove

$$\int H_\sigma \{f_\sigma, H_\sigma\}^\star \,\mathrm{d}V \,\mathrm{d}W = 0 \tag{A.21}$$

under the assumption that boundary integrals vanish. The gyrokinetic Poisson bracket $\{\,,\,\}^\star$, defined in Subsection 2.3.6, evaluated on the two functions f and g, can be written as

$$\{f, g\}^\star = \sum_{a=1}^{6} \sum_{b=1}^{6} J_{ab} \frac{\partial f}{\partial z_a} \frac{\partial g}{\partial z_b}\,, \tag{A.22}$$

where J_{ab} is a skew-symmetric matrix. Inserting Eq. (A.22) into Eq. (A.21) yields

$$\begin{aligned}
\int g\{f, g\}^\star \,\mathrm{d}V \,\mathrm{d}W &= \sum_{a=1}^{6} \sum_{b=1}^{6} \int J_{ab} g \frac{\partial f}{\partial z_a} \frac{\partial g}{\partial z_b} \,\mathrm{d}V \,\mathrm{d}W \\
&= \sum_{a=1}^{6} \sum_{b=1}^{6} \int \frac{1}{2} J_{ab} B_\parallel^* \left(g \frac{\partial f}{\partial z_a} \frac{\partial g}{\partial z_b} + g \frac{\partial f}{\partial z_a} \frac{\partial g}{\partial z_b} \right) \frac{1}{m_\sigma} \,\mathrm{d}V \,\mathrm{d}\theta \,\mathrm{d}\mu \,\mathrm{d}v_\parallel \\
&= \sum_{a=1}^{6} \sum_{b=1}^{6} \int \frac{1}{2} \Bigg(\frac{\partial}{\partial z_b} \left(J_{ab} B_\parallel^* g^2 \frac{\partial f}{\partial z_a} \right) - J_{ab} B_\parallel^* g^2 \frac{\partial^2 f}{\partial z_a \partial z_b} \\
&\qquad - g^2 \frac{\partial f}{\partial z_a} \frac{\partial}{\partial z_b} \left(J_{ab} B_\parallel^* \right) \Bigg) \frac{1}{m_\sigma} \,\mathrm{d}V \,\mathrm{d}\theta \,\mathrm{d}\mu \,\mathrm{d}v_\parallel \\
&= 0\,,
\end{aligned} \tag{A.23}$$

where we performed a partial integration over z_b, assumed that boundary integrals vanish, used the skew-symmetry of J_{ab} and Liouville's theorem $\sum_{b=1}^{6} \frac{\partial}{\partial z_b} (J_{ab} B_\parallel^*) = 0$ [94]. Setting $g = H_\sigma$ and $f = f_\sigma$ proves Eq. (A.21).

B. Supplemental results

B.1. Results of the scaling analysis

In this section, we give the measured times for the evaluation of one timestep in the analysis of the strong and weak scaling, presented in Chapter 5. The measured times for the strong scaling are contained in the first and for the weak scaling in the second table.

Number of nodes	Time per timestep
1	884s
2	454s
4	229s
8	115s
16	63.2s
32	34.8s
64	17.5s
128	9.00s
256	5.05s
512	2.98s

Number of nodes	Time per timestep
1	8.17s
2	7.38s
4	7.57s
8	7.50s
16	6.77s
32	7.30s
64	6.91s
128	7.23s
256	7.43s
512	7.62s

B.2. Solution of the electrostatic dispersion relation in screw pinch geometry

In this section, we give the results for the complex frequency $\hat{\omega}$ obtained from the dispersion relation in Eq. (6.19) up to four significant digits. They are contained in the following table.

m	$\hat{\omega}$
5	$-1.725 + 1.689i$
6	$-1.740 + 1.760i$
7	$-1.744 + 1.800i$
8	$-1.742 + 1.819i$
9	$-1.735 + 1.823i$
10	$-1.725 + 1.817i$
11	$-1.713 + 1.804i$
12	$-1.700 + 1.786i$
13	$-1.686 + 1.764i$
14	$-1.671 + 1.740i$
15	$-1.657 + 1.714i$
16	$-1.643 + 1.688i$
17	$-1.629 + 1.661i$
18	$-1.615 + 1.634i$
19	$-1.601 + 1.607i$
20	$-1.588 + 1.580i$

B.3. Numerical results for the growth rate $\mathrm{Im}\,\hat{\omega}$ in screw pinch geometry

In this section, we give the numerical results for the growth rate $\mathrm{Im}\,\hat{\omega}$ obtained from simulations with `GENE-X` described in Section 6.3. They are contained in the following table.

m	$\mathrm{Im}\,\hat{\omega}$ for $q = \infty$	$\mathrm{Im}\,\hat{\omega}$ for $q \neq \infty$
5	1.74	1.73
10	1.85	1.84
15	1.73	1.72
20	1.58	1.58

B.4. Solution of the electromagnetic dispersion relation in slab geometry

In this section, we give the results for the complex frequency $\hat{\omega}$ obtained from the dispersion relation Eq. (6.35) for the parallel mode number $\hat{k}_y = 1$, different perpendicular mode numbers $\hat{k}_x$ and different plasma beta $\bar{\beta} = \beta_{\mathrm{ref}}/\hat{m}_e$ up to four significant digits. They are contained in the following table.

$\hat{k}_x\rho_{\mathrm{ref}}/L_{\mathrm{ref}}$	$\hat{\omega}$ for $\bar{\beta} = 0.2$	$\hat{\omega}$ for $\bar{\beta} = 2$	$\hat{\omega}$ for $\bar{\beta} = 20$
1/128	$189.7 - 6.095 \cdot 10^{-3}i$	$60.00 - 2.130 \cdot 10^{-3}i$	$18.97 - 2.279 \cdot 10^{-4}i$
1/64	$189.6 - 2.440 \cdot 10^{-2}i$	$60.00 - 8.517 \cdot 10^{-3}i$	$18.98 - 9.114 \cdot 10^{-4}i$
1/32	$189.7 - 9.789 \cdot 10^{-2}i$	$60.01 - 3.405 \cdot 10^{-2}i$	$18.98 - 3.646 \cdot 10^{-3}i$
1/16	$187.5 - 3.960 \cdot 10^{-1}i$	$60.05 - 1.360 \cdot 10^{-1}i$	$19.01 - 1.458 \cdot 10^{-2}i$
1/8	$181.6 - 1.635 \cdot 10^{0}i$	$60.19 - 5.399 \cdot 10^{-1}i$	$19.11 - 5.827 \cdot 10^{-2}i$
1/4	$164.8 - 6.687 \cdot 10^{0}i$	$60.73 - 2.010 \cdot 10^{0}i$	$19.52 - 2.324 \cdot 10^{-1}i$
1/2	$136.1 - 2.192 \cdot 10^{1}i$	$62.40 - 7.584 \cdot 10^{0}i$	$21.04 - 9.187 \cdot 10^{-1}i$
1	$107.6 - 4.826 \cdot 10^{1}i$	$65.37 - 2.239 \cdot 10^{1}i$	$25.87 - 3.512 \cdot 10^{0}i$

B.5. Numerical results for the complex frequency $\hat{\omega}$ in slab geometry

In this section, we give the numerical results for the complex frequency $\hat{\omega}$ obtained from simulations with `GENE-X` described in Section 6.4. They are contained in the following table.

$\hat{k}_x\rho_{\mathrm{ref}}/L_{\mathrm{ref}}$	$\hat{\omega}$ for $\bar{\beta} = 0.2$	$\hat{\omega}$ for $\bar{\beta} = 2$	$\hat{\omega}$ for $\bar{\beta} = 20$
1/64	$189.6 - 2.343 \cdot 10^{-2}i$	$60.00 - 8.318 \cdot 10^{-3}i$	$18.98 - 8.450 \cdot 10^{-4}i$
1/16	$187.5 - 3.852 \cdot 10^{-1}i$	$60.00 - 1.362 \cdot 10^{-1}i$	$19.04 - 1.418 \cdot 10^{-2}i$
1/4	$164.2 - 6.680 \cdot 10^{0}i$	$60.81 - 2.110 \cdot 10^{0}i$	$19.51 - 2.271 \cdot 10^{-1}i$
1	$108.3 - 4.773 \cdot 10^{1}i$	$65.00 - 2.223 \cdot 10^{1}i$	$25.89 - 3.475 \cdot 10^{0}i$

B.6. Numerical results for the complex frequency $\hat{\omega}$ in toroidal geometry

In this section, we give the numerical results for the complex frequency $\hat{\omega}$ obtained from simulations with `GENE-X` described in Section 6.5. They are contained in the following table.

n	$\hat{\omega}$
5	$0.180 + 6.79 \cdot 10^{-2} i$
10	$0.419 + 2.05 \cdot 10^{-1} i$
15	$0.661 + 3.56 \cdot 10^{-1} i$

Bibliography

[1] IPCC. Summary for policymakers. In M. L. Parry, O. F. Canziani, J. P. Palutikof *et al.*, editors, *Climate Change 2007: Impacts, Adaptation and Vulnerability. Contribution of Working Group II to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*, pages 7–22. Cambridge University Press, Cambridge, UK (2007).

[2] J. Cook, N. Oreskes, P. T. Doran *et al.* Consensus on consensus: a synthesis of consensus estimates on human-caused global warming. *Environmental Research Letters*, **11** (4), 048002 (2016).

[3] E. Böhm-Vitense. *Introduction to Stellar Astrophysics*, volume 3. Cambridge University Press (1989).

[4] J. D. Huba. *NRL PLASMA FORMULARY Supported by The Office of Naval Research.* Naval Research Laboratory, Washington, DC (2013).

[5] J. Wesson and D. J. Campbell. *Tokamaks.* International series of monographs on physics. Clarendon Press, 3 edition (2004).

[6] J. Ongena, R. Koch, R. Wolf *et al.* Magnetic-confinement fusion. *Nature Physics*, **12** (5), 398 (2016).

[7] J. P. Freidberg. *Plasma physics and fusion energy.* Cambridge University Press (2008).

[8] J. P. Freidberg. *Ideal MHD.* Cambridge University Press (2014).

[9] A. J. Wootton, B. A. Carreras, H. Matsumoto *et al.* Fluctuations and anomalous transport in tokamaks. *Physics of Fluids B: Plasma Physics*, **2** (12), 2879 (1990).

[10] H. Meyer, C. Angioni, C. G. Albert *et al.* Overview of physics studies on ASDEX Upgrade. *Nuclear Fusion*, **59** (11), 112014 (2019).

[11] F. Wagner, G. Becker, K. Behringer *et al.* Regime of Improved Confinement and High Beta in Neutral-Beam-Heated Divertor Discharges of the ASDEX Tokamak. *Physical Review Letters*, **49**, 1408 (1982).

[12] B. Bigot. Progress toward ITER's First Plasma. *Nuclear Fusion*, **59** (11), 112001 (2019).

[13] M. Shimada, D. Campbell, V. Mukhovatov *et al.* Chapter 1: Overview and summary. *Nuclear Fusion*, **47** (6), S1 (2007).

[14] F. Wagner. A quarter-century of H-mode studies. *Plasma Physics and Controlled Fusion*, **49** (12B), B1 (2007).

[15] J. W. Connor and H. R. Wilson. A review of theories of the L-H transition. *Plasma Physics and Controlled Fusion*, **42** (1), R1 (1999).

[16] P. H. Diamond, Y.-M. Liang, B. A. Carreras *et al.* Self-Regulating Shear Flow Turbulence: A Paradigm for the L to H Transition. *Phys. Rev. Lett.*, **72**, 2565 (1994).

[17] E. Kim and P. H. Diamond. Zonal Flows and Transient Dynamics of the $L-H$ Transition. *Phys. Rev. Lett.*, **90**, 185006 (2003).

[18] K. Miki, P. H. Diamond, O. D. Gürcan *et al.* Spatio-temporal evolution of the L $\rightarrow$ I $\rightarrow$ H transition. *Physics of Plasmas*, **19** (9), 092306 (2012).

[19] L. Chôné, P. Beyer, Y. Sarazin *et al.* L-H transition dynamics in fluid turbulence simulations with neoclassical force balance. *Physics of Plasmas*, **21** (7), 070702 (2014).

[20] G. Y. Park, S. S. Kim, H. Jhang *et al.* Flux-driven simulations of turbulence collapse. *Physics of Plasmas*, **22** (3), 032505 (2015).

[21] J. J. Rasmussen, A. H. Nielsen, J. Madsen *et al.* Numerical modeling of the transition from low to high confinement in magnetically confined plasma. *Plasma Physics and Controlled Fusion*, **58** (1), 014031 (2015).

[22] C. S. Chang, S. Ku, G. R. Tynan *et al.* Fast Low-to-High Confinement Mode Bifurcation Dynamics in a Tokamak Edge Plasma Gyrokinetic Simulation. *Phys. Rev. Lett.*, **118**, 175001 (2017).

[23] A. Loarte, B. Lipschultz, A. S. Kukushkin *et al.* Chapter 4: Power and particle control. *Nuclear Fusion*, **47** (6), S203 (2007).

[24] H. Reimerdes, B. Duval, J. Harrison *et al.* TCV experiments towards the development of a plasma exhaust solution. *Nuclear Fusion*, **57** (12), 126007 (2017).

[25] R. Pitts, X. Bonnin, F. Escourbiac *et al.* Physics basis for the first ITER tungsten divertor. *Nuclear Materials and Energy*, **20**, 100696 (2019).

[26] T. Eich, A. W. Leonard, R. Pitts *et al.* Scaling of the tokamak near the scrape-off layer H-mode power width and implications for ITER. *Nuclear Fusion*, **53** (9), 093031 (2013).

[27] A. S. Kukushkin, H. Pacher, G. Pacher *et al.* Consequences of a reduction of the upstream power SOL width in ITER. *Journal of Nuclear Materials*, **438**, S203 (2013).

[28] C. S. Chang, S. Ku, A. Loarte *et al.* Gyrokinetic projection of the divertor heat-flux width from present tokamaks to ITER. *Nuclear Fusion*, **57** (11), 116023 (2017).

[29] S. I. Krasheninnikov and A. S. Kukushkin. Physics of ultimate detachment of a tokamak divertor plasma. *Journal of Plasma Physics*, **83** (5), 155830501 (2017).

[30] A. J. Brizard and T. S. Hahm. Foundations of nonlinear gyrokinetic theory. *Reviews of Modern Physics*, **79**, 421 (2007).

[31] W. W. Lee. Gyrokinetic approach in particle simulation. *The Physics of Fluids*, **26** (2), 556 (1983).

[32] A. Dimits and W. Lee. Partially Linearized Algorithms in Gyrokinetic Particle Simulation. *Journal of Computational Physics*, **107** (2), 309 (1993).

[33] S. E. Parker and W. W. Lee. A fully nonlinear characteristic method for gyrokinetic simulation. *Physics of Fluids B: Plasma Physics*, **5** (1), 77 (1993).

[34] R. E. Denton and M. Kotschenreuther. δf Algorithm. *Journal of Computational Physics*, **119** (2), 283 (1995).

[35] A. M. Dimits, T. J. Williams, J. A. Byers *et al.* Scalings of Ion-Temperature-Gradient-Driven Anomalous Transport in Tokamaks. *Physical Review Letters*, **77**, 71 (1996).

[36] Z. Lin, T. S. Hahm, W. W. Lee *et al.* Turbulent Transport Reduction by Zonal Flows: Massively Parallel Simulations. *Science*, **281** (5384), 1835 (1998).

[37] S. E. Parker, C. Kim and Y. Chen. Large-scale gyrokinetic turbulence simulations: Effects of profile variation. *Physics of Plasmas*, **6** (5), 1709 (1999).

[38] F. Jenko, W. Dorland, M. Kotschenreuther *et al.* Electron temperature gradient driven turbulence. *Physics of Plasmas*, **7** (5), 1904 (2000).

[39] W. Dorland, F. Jenko, M. Kotschenreuther *et al.* Electron Temperature Gradient Turbulence. *Physical Review Letters*, **85**, 5579 (2000).

[40] J. Candy and R. E. Waltz. Anomalous Transport Scaling in the DIII-D Tokamak Matched by Supercomputer Simulation. *Physical Review Letters*, **91**, 045001 (2003).

[41] V. Grandgirard, M. Brunetti, P. Bertrand *et al.* A drift-kinetic Semi-Lagrangian 4D code for ion turbulence simulation. *Journal of Computational Physics*, **217** (2), 395 (2006).

[42] Y. Idomura, M. Ida, T. Kano *et al.* Conservative global gyrokinetic toroidal full-f five-dimensional Vlasov simulation. *Computer Physics Communications*, **179** (6), 391 (2008).

[43] S. Jolliet, A. Bottino, P. Angelino *et al.* A global collisionless PIC code in magnetic coordinates. *Computer Physics Communications*, **177** (5), 409 (2007).

[44] A. Peeters, Y. Camenen, F. Casson *et al.* The nonlinear gyro-kinetic flux tube code GKW. *Computer Physics Communications*, **180** (12), 2650 (2009).

[45] T. Görler, X. Lapillonne, S. Brunner *et al.* The global version of the gyrokinetic turbulence code GENE. *Journal of Computational Physics*, **230** (18), 7053 (2011).

[46] T. T. Ribeiro and B. Scott. Gyrofluid turbulence studies of the effect of the poloidal position of an axisymmetric Debye sheath. *Plasma Physics and Controlled Fusion*, **50** (5), 055007 (2008).

[47] P. Ricci, F. D. Halpern, S. Jolliet *et al.* Simulation of plasma turbulence in scrape-off layer conditions: the GBS code, simulation results and code validation. *Plasma Physics and Controlled Fusion*, **54** (12), 124047 (2012).

[48] A. Stegmeir, D. Coster, O. Maj *et al.* The field line map approach for simulations of magnetically confined plasmas. *Computer Physics Communications*, **198**, 139 (2016).

[49] P. Tamain, H. Bufferand, G. Ciraolo *et al.* The TOKAM3X code for edge turbulence fluid simulations of tokamak plasmas in versatile magnetic geometries. *Journal of Computational Physics*, **321**, 606 (2016).

[50] B. D. Dudson and J. Leddy. Hermes: global plasma edge fluid turbulence simulations. *Plasma Physics and Controlled Fusion*, **59** (5), 054010 (2017).

[51] B. Zhu, M. Francisquez and B. N. Rogers. GDB: A global 3D two-fluid model of plasma turbulence and transport in the tokamak edge. *Computer Physics Communications*, **232**, 46 (2018).

[52] S. I. Braginskii. Transport Processes in a Plasma. *Reviews of Plasma Physics*, **1**, 205 (1965).

[53] A. Zeiler, J. F. Drake and B. Rogers. Nonlinear reduced Braginskii equations with ion thermal dynamics in toroidal plasma. *Physics of Plasmas*, **4** (6), 2134 (1997).

[54] G. W. Hammett and F. W. Perkins. Fluid moment models for Landau damping with application to the ion-temperature-gradient instability. *Physical Review Letters*, **64**, 3019 (1990).

[55] O. V. Batishchev, S. I. Krasheninnikov, P. J. Catto *et al.* Kinetic effects in tokamak scrape-off layer plasmas. *Physics of Plasmas*, **4** (5), 1672 (1997).

[56] F. Jenko and W. Dorland. Nonlinear electromagnetic gyrokinetic simulations of tokamak plasmas. *Plasma Physics and Controlled Fusion*, **43** (12A), A141 (2001).

[57] R. H. Cohen and X. Q. Xu. Progress in Kinetic Simulation of Edge Plasmas. *Contributions to Plasma Physics*, **48** (1-3), 212 (2008).

[58] C. S. Chang, S. Ku, P. Diamond *et al.* Whole-volume integrated gyrokinetic simulation of plasma turbulence in realistic diverted-tokamak geometry. *Journal of Physics: Conference Series*, **180**, 012057 (2009).

[59] M. A. Dorf, M. R. Dorr, J. A. Hittinger *et al.* Continuum kinetic modeling of the tokamak plasma edge. *Physics of Plasmas*, **23** (5), 056102 (2016).

[60] E. L. Shi, G. W. Hammett, T. Stoltzfus-Dueck *et al.* Gyrokinetic continuum simulation of turbulence in a straight open-field-line plasma. *Journal of Plasma Physics*, **83** (3), 905830304 (2017).

[61] Q. Pan, D. Told, E. L. Shi *et al.* Full-f version of GENE for turbulence in open-field-line systems. *Physics of Plasmas*, **25** (6), 062303 (2018).

[62] V. Grandgirard, J. Abiteboul, J. Bigot *et al.* A 5D gyrokinetic full-f global semi-Lagrangian code for flux-driven ion turbulence simulations. *Computer Physics Communications*, **207**, 35 (2016).

[63] E. Caschera, G. Dif-Pradalier, P. Ghendrih *et al.* Immersed boundary conditions in global, flux-driven, gyrokinetic simulations. *Journal of Physics: Conference Series*, **1125**, 012006 (2018).

[64] M. Boesl, A. Bergmann, A. Bottino *et al.* Gyrokinetic full-f particle-in-cell simulations on open field lines with PICLS. *Physics of Plasmas*, **26** (12), 122302 (2019).

[65] W. M. Nevins, G. W. Hammett, A. M. Dimits *et al.* Discrete particle noise in particle-in-cell simulations of plasma microturbulence. *Physics of Plasmas*, **12** (12), 122305 (2005).

[66] J. A. Krommes. Nonequilibrium gyrokinetic fluctuation theory and sampling noise in gyrokinetic particle-in-cell simulations. *Physics of Plasmas*, **14** (9), 090501 (2007).

[67] G. J. Wilkie and W. Dorland. Fundamental form of the electrostatic δf-PIC algorithm and discovery of a converged numerical instability. *Physics of Plasmas*, **23** (5), 052111 (2016).

[68] M. Kotschenreuther, G. Rewoldt and W. Tang. Comparison of initial value and eigenvalue codes for kinetic toroidal plasma instabilities. *Computer Physics Communications*, **88** (2), 128 (1995).

[69] E. Sonnendrücker, J. Roche, P. Bertrand *et al.* The Semi-Lagrangian Method for the Numerical Resolution of the Vlasov Equation. *Journal of Computational Physics*, **149** (2), 201 (1999).

[70] N. Crouseilles, T. Respaud and E. Sonnendrücker. A forward semi-Lagrangian method for the numerical solution of the Vlasov equation. *Computer Physics Communications*, **180** (10), 1730 (2009).

[71] N. Crouseilles, M. Mehrenberger and E. Sonnendrücker. Conservative semi-Lagrangian schemes for Vlasov equations. *Journal of Computational Physics*, **229** (6), 1927 (2010).

[72] A. Staniforth and J. Côté. Semi-Lagrangian Integration Schemes for Atmospheric Models—A Review. *Monthly Weather Review*, **119** (9), 2206 (1991).

[73] W. D'haeseleer, W. N. Hitchon, J. D. Callen *et al.* *Flux coordinates and magnetic field structure: a guide to a fundamental tool of plasma structure.* Springer series in computational physics. Springer-Verlag (1991).

[74] A. K. Stegmeir. *GRILLIX: A 3D turbulence code for magnetic fusion devices based on a field line map.* Ph.D. thesis, Technische Universität München (2015).

[75] M. Ottaviani. An alternative approach to field-aligned coordinates for plasma turbulence simulations. *Physics Letters A*, **375** (15), 1677 (2011).

[76] F. Hariri and M. Ottaviani. A flux-coordinate independent field-aligned approach to plasma turbulence simulations. *Computer Physics Communications*, **184** (11), 2419 (2013).

[77] M. Held, M. Wiesenberger and A. Stegmeir. Three discontinuous Galerkin schemes for the anisotropic heat conduction equation on non-aligned grids. *Computer Physics Communications*, **199**, 29 (2016).

[78] B. Shanahan, B. Dudson and P. Hill. Fluid simulations of plasma filaments in stellarator geometries with BSTING. *Plasma Physics and Controlled Fusion*, **61** (2), 025007 (2018).

[79] W. Zholobenko, T. Body, P. Manz *et al.* Electric field and turbulence in global Braginskii simulations across the ASDEX Upgrade edge and scrape-off layer. *Plasma Physics and Controlled Fusion*, **63** (3), 034001 (2021).

[80] W. Zholobenko, A. Stegmeir, M. Griener *et al.* The role of neutral gas in validated global edge turbulence simulations. *Nuclear Fusion* (accepted).

[81] M. Dorf and M. Dorr. Progress with the 5D full-f continuum gyrokinetic code COGENT. *Contributions to Plasma Physics*, **60** (5-6), e201900113 (2020).

[82] J.-M. Kwon, D. Yi, X. Piao *et al.* Development of semi-Lagrangian gyrokinetic code for full-f turbulence simulation in general tokamak geometry. *Journal of Computational Physics*, **283**, 518 (2015).

[83] G. Latu, M. Mehrenberger, Y. Güçlü *et al.* Field-Aligned Interpolation for Semi-Lagrangian Gyrokinetic Simulations. *Journal of Scientific Computing*, **74**, 1601 (2018).

[84] H. Reimerdes, R. Ambrosino, P. Innocente *et al.* Assessment of alternative divertor configurations as an exhaust solution for DEMO. *Nuclear Fusion*, **60** (6), 066030 (2020).

[85] F. Militello, L. Aho-Mantila, R. Ambrosino *et al.* Preliminary analysis of alternative divertors for DEMO. *Nuclear Materials and Energy*, **26**, 100908 (2021).

[86] A. Stegmeir, O. Maj, D. Coster *et al.* Advances in the flux-coordinate independent approach. *Computer Physics Communications*, **213**, 111 (2017).

[87] D. Michels, A. Stegmeir, P. Ulbl *et al.* GENE-X: A full-f gyrokinetic turbulence code based on the flux-coordinate independent approach. *Computer Physics Communications*, **264**, 107986 (2021).

[88] N. Tronko and C. Chandre. Second-order nonlinear gyrokinetic theory: from the particle to the gyrocentre. *Journal of Plasma Physics*, **84** (3), 925840301 (2018).

[89] P. A. Muñoz, D. Told, P. Kilian *et al.* Gyrokinetic and kinetic particle-in-cell simulations of guide-field reconnection. I. Macroscopic effects of the electron flows. *Physics of Plasmas*, **22** (8), 082110 (2015).

[90] D. Told, J. Cookmeyer, F. Muller *et al.* Comparative study of gyrokinetic, hybrid-kinetic and fully kinetic wave physics for space plasmas. *New Journal of Physics*, **18** (6), 065011 (2016).

[91] C. D. Stephens, R. W. Brzozowski and F. Jenko. On the limitations of gyrokinetics: Magnetic moment conservation. *Physics of Plasmas*, **24** (10), 102517 (2017).

[92] D. Grošelj, S. S. Cerri, A. B. Navarro *et al.* Fully Kinetic versus Reduced-kinetic Modeling of Collisionless Plasma Turbulence. *The Astrophysical Journal*, **847** (1), 28 (2017).

[93] K. Vogtmann, A. Weinstein and V. I. Arnol'd. *Mathematical Methods of Classical Mechanics*. Graduate Texts in Mathematics. Springer New York (1997).

[94] J. R. Cary and A. J. Brizard. Hamiltonian theory of guiding-center motion. *Rev. Mod. Phys.*, **81**, 693 (2009).

[95] A. Brizard. Nonlinear gyrokinetic Maxwell-Vlasov equations using magnetic coordinates. *Journal of Plasma Physics*, **41** (3), 541–559 (1989).

[96] P. C. Crandall. *Collisional and Electromagnetic Physics in Gyrokinetic Models*. Ph.D. thesis, University of California, Los Angeles (2019).

[97] N. R. Mandell, A. Hakim, G. W. Hammett *et al.* Electromagnetic full-f gyrokinetics in the tokamak edge with discontinuous Galerkin methods. *Journal of Plasma Physics*, **86** (1), 905860109 (2020).

[98] J. V. W. Reynders. *Gyrokinetic Simulation of Finite-Beta Plasmas on Parallel Architectures*. Ph.D. thesis, Princeton University (1993).

[99] J. C. Cummings. *Gyrokinetic simulation of finite-beta and self-generated sheared-flow effects on pressure-gradient-driven instabilities*. Ph.D. thesis, Princeton University (1995).

[100] A. Mishchenko, A. Bottino, R. Hatzky *et al.* Mitigation of the cancellation problem in the gyrokinetic particle-in-cell simulations of global electromagnetic modes. *Physics of Plasmas*, **24** (8), 081206 (2017).

[101] R. G. Littlejohn. A guiding center Hamiltonian: A new approach. *Journal of Mathematical Physics*, **20** (12), 2445 (1979).

[102] R. G. Littlejohn. Hamiltonian formulation of guiding center motion. *The Physics of Fluids*, **24** (9), 1730 (1981).

[103] R. G. Littlejohn. Variational principles of guiding centre motion. *Journal of Plasma Physics*, **29** (1), 111–125 (1983).

[104] J. R. Cary. Lie transform perturbation theory for Hamiltonian systems. *Physics Reports*, **79** (2), 129 (1981).

[105] S. Possanner. Gyrokinetics from variational averaging: Existence and error bounds. *Journal of Mathematical Physics*, **59** (8), 082702 (2018).

[106] H. Sugama. Gyrokinetic field theory. *Physics of Plasmas*, **7** (2), 466 (2000).

[107] A. J. Brizard. Variational principle for nonlinear gyrokinetic Vlasov–Maxwell equations. *Physics of Plasmas*, **7** (12), 4816 (2000).

[108] K. Aleynikova, A. Zocco, P. Xanthopoulos *et al.* Kinetic ballooning modes in tokamaks and stellarators. *Journal of Plasma Physics*, **84** (6), 745840602 (2018).

[109] N. Miyato, B. D. Scott and M. Yagi. On the gyrokinetic model in long wavelength regime. *Plasma Physics and Controlled Fusion*, **55** (7), 074011 (2013).

[110] B. Scott and J. Smirnov. Energetic consistency and momentum conservation in the gyrokinetic description of tokamak plasmas. *Physics of Plasmas*, **17** (11), 112302 (2010).

[111] A. Bottino and E. Sonnendrücker. Monte Carlo particle-in-cell methods for the simulation of the Vlasov–Maxwell gyrokinetic equations. *Journal of Plasma Physics*, **81** (5), 435810501 (2015).

[112] A. J. Brizard and N. Tronko. Exact momentum conservation laws for the gyrokinetic Vlasov-Poisson equations. *Physics of Plasmas*, **18** (8), 082307 (2011).

[113] P. L. Bhatnagar, E. P. Gross and M. Krook. A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems. *Physical Review*, **94**, 511 (1954).

[114] A. Lenard and I. B. Bernstein. Plasma Oscillations with Diffusion in Velocity Space. *Physical Review*, **112**, 1456 (1958).

[115] J. P. Dougherty. Model Fokker-Planck Equation for a Plasma and Its Solution. *The Physics of Fluids*, **7**, 1788 (1964).

[116] P. Ulbl, D. Michels and F. Jenko. Implementation and Verification of a Conservative, Multi-Species, Gyro-Averaged, Full-f, Lenard-Bernstein/Dougherty Collision Operator in the Gyrokinetic Code GENE-X. *Contributions to Plasma Physics* (to be submitted).

[117] R. C. Grimm, J. M. Greene and J. L. Johnson. Computation of the Magnetohydrodynamic Spectrum in Axisymmetric Toroidal Confinement Systems. In J. Killeen, editor, *Controlled Fusion*, volume 16 of *Methods in Computational Physics: Advances in Research and Applications*, pages 253–280. Elsevier (1976).

[118] S. Hamada. Hydromagnetic equilibria and their proper coordinates. *Nuclear Fusion*, **2** (1-2), 23 (1962).

[119] A. H. Boozer. Establishment of magnetic coordinates for a given magnetic field. *The Physics of Fluids*, **25** (3), 520 (1982).

[120] B. Scott. Shifted metric procedure for flux tube treatments of toroidal geometry: Avoiding grid deformation. *Physics of Plasmas*, **8** (2), 447 (2001).

[121] A. Arakawa. Computational Design for Long-Term Numerical Integration of the Equations of Fluid Motion: Two-Dimensional Incompressible Flow. Part I. *Journal of Computational Physics*, **1**, 119 (1966).

[122] E. Hairer, S. Nørsett and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer Series in Computational Mathematics. Springer Berlin Heidelberg (2008).

[123] K. Lipnikov, G. Manzini and M. Shashkov. Mimetic finite difference method. *Journal of Computational Physics*, **257**, 1163 (2014). Physics-compatible numerical methods.

[124] A. N. Hirani. *Discrete Exterior Calculus*. Ph.D. thesis, California Institute of Technology (2003).

[125] M. Desbrun, A. N. Hirani, M. Leok *et al.* Discrete exterior calculus. *arXiv preprint math/0508341* (2005).

[126] R. Jorge, P. Ricci and N. F. Loureiro. A drift-kinetic analytical model for scrape-off layer plasma dynamics at arbitrary collisionality. *Journal of Plasma Physics*, **83** (6), 905830606 (2017).

[127] N. Mandell, W. Dorland and M. Landreman. Laguerre–Hermite pseudo-spectral velocity formulation of gyrokinetics. *Journal of Plasma Physics*, **84** (1), 905840108 (2018).

[128] B. J. Frei, R. Jorge and P. Ricci. A gyrokinetic model for the plasma periphery of tokamak devices. *Journal of Plasma Physics*, **86** (2), 905860205 (2020).

[129] W. H. Press, B. P. Flannery, S. A. Teukolsky *et al. Numerical Recipes in Pascal: The Art of Scientific Computing*. Cambridge University Press (1989).

[130] P. Rabinowitz and G. Weiss. Tables of abcissas and weights for numerical evaluation of integrals of the form $\int_0^\infty e^{-x} x^n f(x) \mathrm{d}x$. *Mathematics of Computation*, **13**, 285 (1959).

[131] M. Baldauf. Stability analysis for linear discretisations of the advection equation with Runge–Kutta time integration. *Journal of Computational Physics*, **227** (13), 6638 (2008).

[132] T. Dannert and F. Jenko. Vlasov simulation of kinetic shear Alfvén waves. *Computer Physics Communications*, **163** (2), 67 (2004).

[133] H. Doerk and F. Jenko. Towards optimal explicit time-stepping schemes for the gyrokinetic equations. *Computer Physics Communications*, **185** (7), 1938 (2014).

[134] W. Hundsdorfer and J. Verwer. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*. Springer Berlin Heidelberg, Berlin, Heidelberg (2003).

[135] M. J. Pueschel, T. Dannert and F. Jenko. On the role of numerical dissipation in gyrokinetic Vlasov simulations of plasma microturbulence. *Computer Physics Communications*, **181** (8), 1428 (2010).

[136] G. J. Tee. A Novel Finite-Difference Approximation to the Biharmonic Operator. *The Computer Journal*, **6** (2), 177 (1963).

[137] G. Knorr. Plasma simulation with few particles. *Journal of Computational Physics*, **13** (2), 165 (1973).

[138] O. Pezzi, E. Camporeale and F. Valentini. Collisional effects on the numerical recurrence in Vlasov-Poisson simulations. *Physics of Plasmas*, **23** (2), 022103 (2016).

[139] P. Angelino, A. Bottino, R. Hatzky *et al.* On the definition of a kinetic equilibrium in global gyrokinetic simulations. *Physics of Plasmas*, **13** (5), 052304 (2006).

[140] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin *et al.* Defining an equilibrium state in global full-f gyrokinetic models. *Communications in Nonlinear Science and Numerical Simulation*, **13** (1), 65 (2008).

[141] S. Williams, A. Waterman and D. Patterson. Roofline: An Insightful Visual Performance Model for Multicore Architectures. *Communications of the ACM*, **52** (4), 65–76 (2009).

[142] G. M. Amdahl. Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities. In *Proceedings of the April 18-20, 1967, Spring Joint Computer Conference*, AFIPS '67 (Spring), page 483–485. Association for Computing Machinery, New York, NY, USA (1967).

[143] J. L. Gustafson. Reevaluating Amdahl's Law. *Communications of the ACM*, **31** (5), 532–533 (1988).

[144] Message Passing Interface Forum. MPI: A Message-Passing Interface Standard Version 3.1. https://www.mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf (2015). Accessed: 2021-09-10.

[145] E. Gamma, R. Helm, R. Johnson *et al. Design Patterns: Elements of Reusable Object-oriented Software*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1995).

[146] G. Booch. *Object Oriented Design with Applications*. Benjamin-Cummings Publishing Co., Inc., USA (1990).

[147] pFUnit 4.0. https://github.com/Goddard-Fortran-Ecosystem/pFUnit. Accessed: 2020-05-12.

[148] OpenMP Architecture Review Board. OpenMP Application Program Interface Version 4.5. https://www.openmp.org/wp-content/uploads/openmp-4.5.pdf (2015). Accessed: 2020-05-26.

[149] OpenACC-Standard.org. The OpenACC Application Programming Interface Version 2.7. https://www.openacc.org/sites/default/files/inline-files/OpenACC.2.7.pdf (2018). Accessed: 2020-05-26.

[150] NVIDIA Corporation. CUDA Zone. https://developer.nvidia.com/cuda-zone. Accessed: 2021-09-04.

[151] J. Treibig, G. Hager and G. Wellein. LIKWID: A Lightweight Performance-Oriented Tool Suite for x86 Multicore Environments. In *2010 39th International Conference on Parallel Processing Workshops*, pages 207–216 (2010).

[152] J. Treibig, G. Hager and G. Wellein. likwid-bench: An Extensible Microbenchmarking Platform for x86 Multicore Compute Nodes. In H. Brunst, M. S. Müller, W. E. Nagel *et al.*, editors, *Tools for High Performance Computing 2011*, pages 27–36. Springer Berlin Heidelberg, Berlin, Heidelberg (2012).

[153] H. Shoukourian, T. Wilde, A. Auweter *et al.* Predicting the Energy and Power Consumption of Strong and Weak Scaling HPC Applications. *Supercomputing Frontiers and Innovations*, **1** (2) (2014).

[154] TOP500. https://www.top500.org/lists/top500/2021/06/. Accessed: 2021-08-11.

[155] P. J. Roache. Code Verification by the Method of Manufactured Solutions. *Journal of Fluids Engineering*, **124** (1), 4 (2001).

[156] T. Görler, N. Tronko, W. A. Hornsby *et al.* Intercode comparison of gyrokinetic global electromagnetic modes. *Physics of Plasmas*, **23** (7), 072503 (2016).

[157] Wolfram Research, Inc. Mathematica, Version 12.1. Champaign, IL, 2020.

[158] D. Coulette and N. Besse. Numerical comparisons of gyrokinetic multi-water-bag models. *Journal of Computational Physics*, **248**, 1 (2013).

[159] S. Güttel and F. Tisseur. The nonlinear eigenvalue problem. *Acta Numerica*, **26**, 1–94 (2017).

[160] X. Lapillonne, B. F. McMillan, T. Görler *et al.* Nonlinear quasisteady state benchmark of global gyrokinetic codes. *Physics of Plasmas*, **17** (11), 112321 (2010).

[161] M. N. Rosenbluth and F. L. Hinton. Poloidal Flow Driven by Ion-Temperature-Gradient Turbulence in Tokamaks. *Physical Review Letters*, **80**, 724 (1998).

[162] J. R. Haack, C. D. Hauck and M. S. Murillo. A Conservative, Entropic Multispecies BGK Model. *Journal of Statistical Physics*, **168**, 826 (2017).

[163] D. O. Gericke, M. S. Murillo and M. Schlanges. Dense plasma temperature equilibration in the binary collision approximation. *Physical Review E*, **65**, 036418 (2002).

[164] P. H. Diamond, S.-I. Itoh, K. Itoh *et al.* Zonal flows in plasma–a review. *Plasma Physics and Controlled Fusion*, **47** (5), R35 (2005).

[165] G. D. Conway, A. I. Smolyakov and T. Ido. Geodesic Acoustic Modes in magnetic confinement devices. *Nuclear Fusion* (2021).

[166] L. C. Andrews. *Special Functions of Mathematics for Engineers*. Online access with subscription: SPIE Digital Library. SPIE Optical Engineering Press (1997).

[167] D. Jarema, H. Bungartz, T. Görler *et al.* Block-structured grids for Eulerian gyrokinetic simulations. *Computer Physics Communications*, **198**, 105 (2016).

[168] D. Jarema, H. Bungartz, T. Görler *et al.* Block-structured grids in full velocity space for Eulerian gyrokinetic simulations. *Computer Physics Communications*, **215**, 49 (2017).

[169] S. Parker, R. Procassini, C. Birdsall *et al.* A Suitable Boundary Condition for Bounded Plasma Simulation without Sheath Resolution. *Journal of Computational Physics*, **104** (1), 41 (1993).

[170] D. S. Oliveira, T. Body, D. Galassi *et al.* Validation of edge turbulence codes against the TCV-X21 diverted L-mode reference case. *Nuclear Fusion* (submitted).

[171] S. Coda, J. Ahn, R. Albanese *et al.* Overview of the TCV tokamak program: scientific progress and facility upgrades. *Nuclear Fusion*, **57** (10), 102011 (2017).

[172] B. D. Fried and S. D. Conte. *The Plasma Dispersion Function*. Academic Press (1961).

Acknowledgements

I would like to express my gratitude to Frank Jenko, my supervisor, for making this thesis possible and advising me throughout the time.
I would like to acknowlegde the `GRILLIX` team including Thomas Body, Andreas Stegmeir and Wladimir Zholobenko. Thomas developed the numerical equilibria of `PARALLAX`, making the simulations of AUG possible. Furthermore, he developed the diagnostic tool `TorX` that I used to generate the plots in Chapter 7. Andreas, the main developer of `GRILLIX`, was always there when I had any question about the FCI approach and Wladimir was my first contact when I had any questions about turbulence or the physics of fusion devices in general.
I would like to thank Philipp Ulbl, who joined the `GENE-X` development team at a later time and started the development of collision operators. In addition, Philipp was a great help in locating and solving problems in the code.
I would like to thank Omar Maj, Stefan Possanner and Eric Sonnendrücker from the Numerical Methods in Plasmaphysics (NMPP) department of our institute for many stimulating and encouraging discussions about discretisation schemes and gyrokinetic theory.
I would like to thank Tilman Dannert, Tobias Görler, Denis Jarema, Alejandro Bañón Navarro, Daniel Told, Felix Wilms from the `GENE` team and Alberto Bottino from the `ORB5` team for numerous discussions about gyrokinetic turbulence codes.
I would like to thank Thomas Eich and Davide Silvagni for providing the experimental measurements of the parallel heat flux and the scrape-off layer fall-off length λ_q, presented in Chapter 7.
Finally, I am deeply grateful to my family for supporting me throughout my life and during the work on this thesis.