Maurice Maurer

GENE-3D – a global gyrokinetic turbulence code for stellarators and perturbed tokamaks

Technische Universität München

Max-Planck-Institut für Plasmaphysik

GENE-3D - a global gyrokinetic turbulence code for stellarators and perturbed tokamaks

Maurice Maurer

Department of Informatics

Technische Universität München

Ph.D. thesis

GENE-3D - a global gyrokinetic turbulence code for stellarators and perturbed tokamaks

Author:	Maurice Maurer
Supervisor:	Prof. Dr. Frank Jenko
Submission Location:	Munich
Submission Date:	29.04.2020
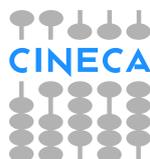
Abstract

One of the key challenges in magnetic confinement fusion research is to understand, predict, and control turbulent transport. Modern stellarators like Wendelstein 7-X are optimized to reduce collisional transport, thus bringing to the foreground the importance of turbulence-induced transport. The latter determines the energy confinement time of any modern fusion device and with it its size and cost. Hence, plasma microturbulence has been a prime subject to computational efforts since the 1980s. However, these studies have been confined mainly to the area of tokamak physics, building on various simplifications stemming from the axisymmetry of the systems. Meanwhile, investigations of non-axisymmetric devices, like stellarators or perturbed tokamaks, are far less developed at this point. In particular, turbulence simulations taking magnetic field variations in all three dimensions into account at the same time have been elusive in the literature until now. As stellarators have several advantages like long term operation compared to tokamaks, this thesis aims at closing the knowledge gap mentioned above. Here, a detailed description of the newly developed stellarator version of the well-established plasma turbulence code GENE, called GENE-3D, is given. GENE-3D makes it possible for the first time in the GENE-code family to describe plasma turbulence in magnetic geometries with variations in all three spatial dimensions. GENE-3D furthermore offers world-wide unique capabilities, such as the consideration of an arbitrary number of gyrokinetic species in such simulations. GENE-3D is massively parallelized. Hence, performance benchmarks are presented including parallelization and memory scaling tests demonstrating the suitability for present-day supercomputers. Furthermore, the production chain around GENE-3D is presented, which includes an interface to the ideal magnetohydrodynamic code GVEC providing GENE-3D with the necessary geometric input. Also, postprocessing tools based on Python and ParaView used to quantify and visualize simulation results are introduced. GENE-3D is verified through extensive benchmarks with other codes both for axisymmetric and non-axisymmetric equilibria. Additionally, the influence of numerical precision is tested in this context. Different geometries of Wendelstein 7-X are investigated for their influence on turbulent transport in the context of turbulence optimization of stellarators. Here it is found that small differences in geometry do not lead to different simulation results. This paves the way for larger tolerances for the magnetic field coils and hence lower cost as far as turbulent transport is concerned. Furthermore, the influence of the machine size is studied and compared for 3D equilibria and a flux surface geometry. Here, the importance of three-dimensional magnetic geometries is demonstrated as reduced models considering only one flux surface are not capable of capturing all geometric effects.

Contents

1   Introduction

This section aims at providing the motivation for developing the GENE-3D code and for using it to study geometric effects in the Wendelstein 7-X stellarator. To this end, the need for fusion energy is emphasized first. Then, the turbulent transport processes leading to currently low confinement times are introduced. Subsequently, strategies to numerically describe turbulence in fusion devices are explained. Finally, the various applications of GENE-3D like stellarator optimization are outlined.

1.1   The world's energy supply and demand

The world's population   is most likely to continue growing throughout the 21st-century [1]. With an 80% probability, the current population of 7.2 billion people will increase to $9.6 - 12.3$ billion by the year 2100. At the same time, despite reductions in Europe and North America in the energy consumption per capita the global energy demand per capita is increasing due to a higher standard of living in developing countries (see Fig. 1.1a). Fig. 1.1b shows that these effects combined lead to an increase of global energy production from coal and oil by more than a factor of 2 and by more than a factor of 3 for natural gas in the 50 years between 1970 and 2020.

The heavy reliance on fossil fuels   has led to the greenhouse effect [3], which results in global warming [4]. The energy production from hydropower has not experienced the same type of growth as the geographical conditions allowing to extract energy from strong currents of water are not available everywhere. The public opinion of nuclear fission power has suffered severely since the meltdowns in Tschernobyl 1986 [5] and Fukushima 2011 as both disasters had a direct impact on the environment and the population [6]. Hence, the energy production curve of nuclear fission power has flattened in Fig. 1.1b. In the last few years, other renewable energies like wind and solar power are increasing production similarly fast as the fossil fuels mentioned above. However, their total contribution to global energy production is still just around 4% [2]. Their growth is partly hindered due to environmental concerns regarding wind power [7] and the slow increase of efficiency of solar panels visualized in Fig. 1.2.

Fusion energy   is a possible alternative that has been researched over the last half a century. Utilizing nuclear fusion as an energy source would have many benefits. The power output of a power plant could be controlled similarly to conventional fission or fossil fuels and would not depend on weather conditions. At the same time, neither greenhouse gases nor long-lived radioactive isotopes would be produced. Catastrophes like in Tschernobyl or Fukushima could not happen. Furthermore, using a power plant to produce weapons is also not an issue compared to nuclear fission. From a geopolitical

(a)

(b)
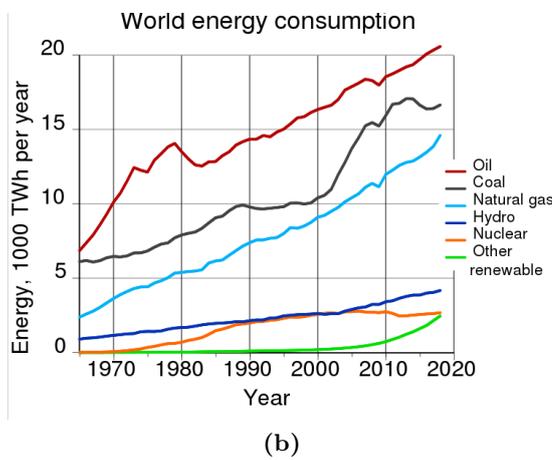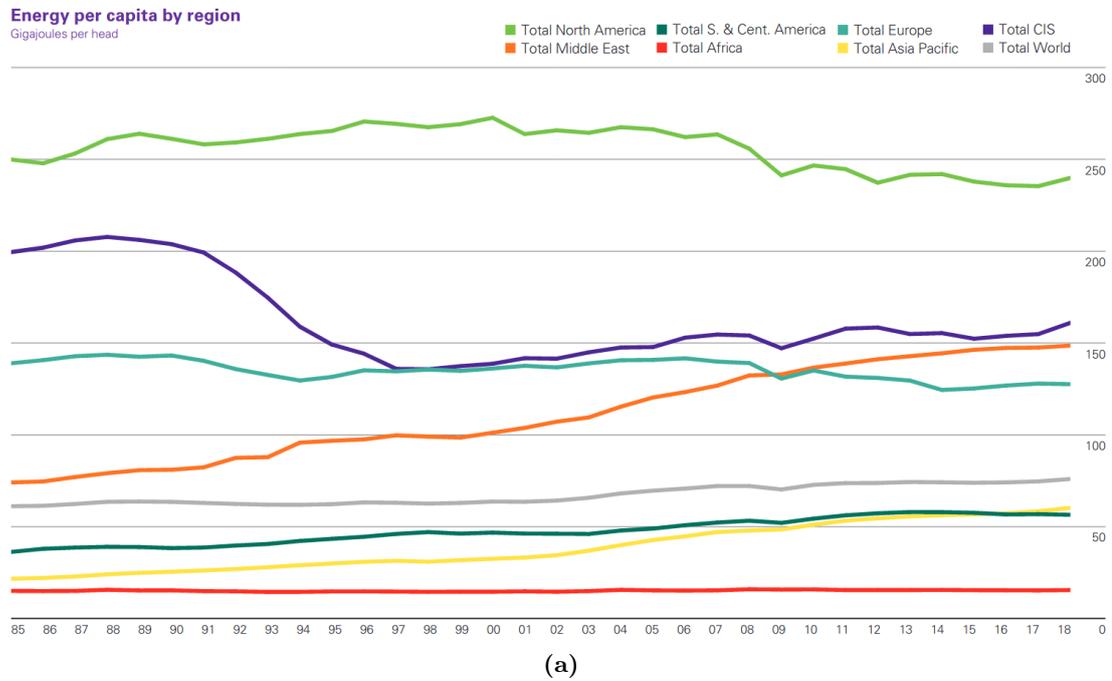
Figure 1.1: Energy demand per capita plotted per region (a). The energy demand of the world per capita has increased over the last 35 years. The total energy demand has increased even faster (b) due to an increase in population. The main sources of energy remain fossil fuels by far Cf. [2].
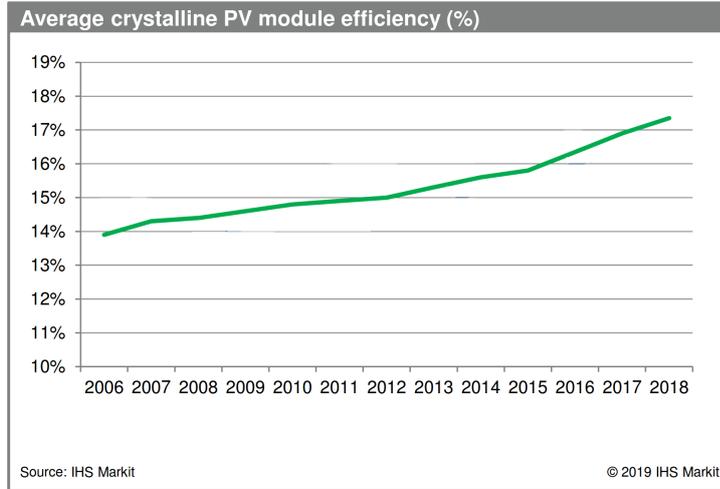
Figure 1.2: The efficiency of the average commercially used photovoltaics (PV) panels has increased slowly Cf. [8]. The use of this content was authorized in advance. Any further use or redistribution of this content is strictly prohibited without written permission by IHS Markit.

perspective, another advantage is the availability of the fuel sources required for fusion energy. The reaction which is intended to be utilized in a reactor is the fusion of the two heavy hydrogen isotopes deuterium and tritium to a helium atom, a neutron and the energy of the mass defect

$$^2\mathrm{H} + {}^3\mathrm{H} \rightarrow {}^4\mathrm{He} + \mathrm{n} + 17.6\,\mathrm{MeV}\,. \tag{1.1}$$

Deuterium constitutes 0.015% of all hydrogen atoms and can be extracted from water. Tritium does not naturally occur in significant quantities. It can, however, be bred from lithium,

$$^6\mathrm{Li} + n \rightarrow {}^4\mathrm{He} + {}^3\mathrm{H} + 4.78\,\mathrm{MeV}\,, \tag{1.2}$$

such that the neutron from the fusion reaction can be used. To fuse deuterium and tritium high temperatures T are needed to overcome the Coulomb repulsion between the two nuclei. Also, a sufficiently high density n of deuterium and tritium particles is needed for the reaction rate to be high enough. Together with the confinement time τ_e the condition for a self-sustained fusion reaction is set by the Lawson criterion [9]

$$nT\tau_e > 3 \times \frac{10^{21}\mathrm{keVs}}{\mathrm{m}^3}\,. \tag{1.3}$$

In stars like the sun, the gravitational force leads to high densities and high confinement times in the core so that the temperatures are relatively low at 15 million Kelvin compared to terrestrial fusion devices. On earth, gravity in fusion experiments can mostly be neglected.

1.2 Magnetic confinement fusion

In magnetic confinement fusion, strong magnetic fields are used to confine the plasma. Perpendicular to the magnetic field, the charged particles move in gyro-orbits (see Fig. 1.7). In contrast, the motion along the field line is unhindered except when magnetic field variations lead to mirror effects. Plasmas contained in purely toroidal magnetic fields are poorly confined due to drifts [10]. A poloidal component of the magnetic field alleviates this issue by twisting the magnetic field. The ratio of toroidal to poloidal revolutions of a field line is called the safety factor q (see Eq. (2.47)). If q is not a rational number, a field line covers a so-called flux-surface. The field lines at different radial positions then describe a set of nested flux surfaces (see Fig. 1.3).
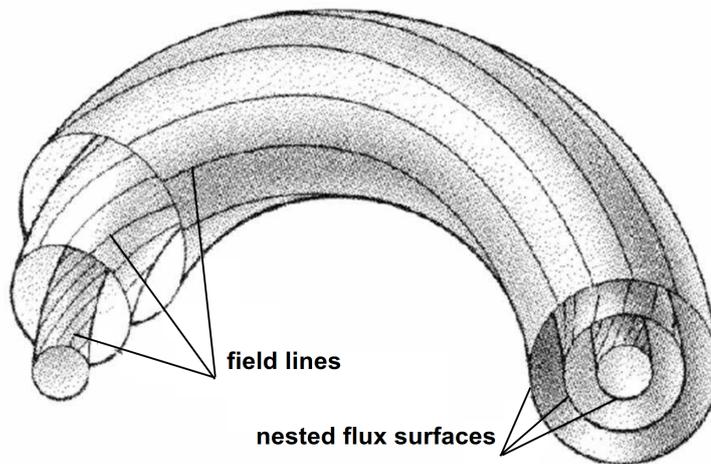
Figure 1.3: Illustration of field lines and nested flux surfaces in a tokamak geometry Cf. [11].

The two main concepts to realize such a magnetic configuration are visualized in Fig. 1.4. In a tokamak, external coils produce a toroidal magnetic field. Additionally, a current is induced through the transformer coils into the plasma to generate the poloidal component of the magnetic field. In a stellarator like Wendelstein 7-X (see right picture in Fig. 1.4) the entire magnetic field is produced by the external coils which leads to a sophisticated coil design. The two main experiments in Germany are ASDEX Upgrade (AUG) - a tokamak - and Wendelstein 7-X (W7-X) - a stellarator. The International Thermonuclear Experimental Reactor (ITER [12]), which is supposed to be the first fusion experiment with a positive energy balance, also implements a tokamak design. Wendelstein 7-X has demonstrated excellent performance in the first operational phases [13, 14]. Therefore, at the Max Planck Institute for Plasma Physics in Garching, discussions are ongoing to replace the AUG tokamak with a hybrid between a tokamak and a stellarator. One design study of such a "Stellatok" design is investigated in Ref. [15]. A snapshot of a GENE-3D simulation using the ParaView diagnostic is visualized in
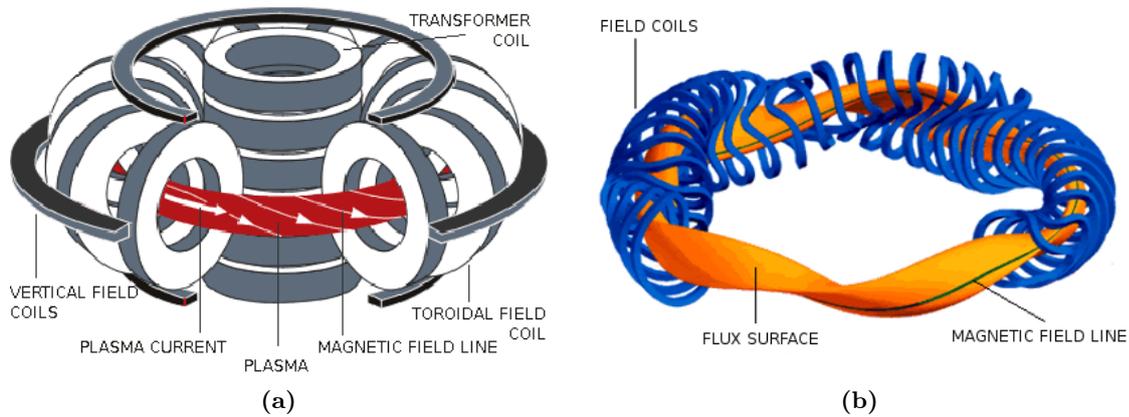
Figure 1.4: Schematic illustration of the coil system and a resulting field line and flux surface for the two main concepts for magnetic confinement fusion devices: the tokamak (a) and the stellarator (b). Source: IPP

Fig. 1.5. Hence for the successor of ITER called DEMO, different studies are currently
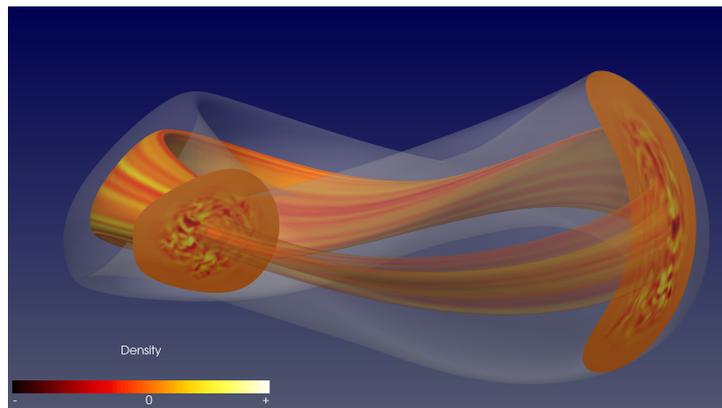
Figure 1.5: Snapshot of a GENE-3D simulation of the Stellatok design study, which is a potential successor of ASDEX Upgrade. The grey area marks the last closed flux surface. The poloidal cuts have very different shapes at different toroidal positions.

underway testing both tokamak and stellarator designs. DEMO will be used to demonstrate the feasibility and capabilities of a fusion power plant.

Plasma parameters of current and future machines

The densities in a magnetic confinement fusion experiment are lower than in the core of the sun, so the temperatures need to be higher. E.g., in ASDEX Upgrade (AUG), temperatures of up to 150 million Kelvin are reached, and confinement times up to $0.2s$ are possible [16]. The particle density in AUG is of the order of 10^{20} particles per cubic meter. In comparison, the atmospheric density on earth is of the order of 10^{25} particles per cubic meter. A current overview of the triple product for various machines,

including the first two operational phases of the Wendelstein 7-X, is plotted in Fig. 1.6a. Wendelstein 7-X has set the world record for the triple product for stellarators [14] with the first experimental campaigns. It is expected to further increase performance in the future. A bit outdated, but interesting, comparison is plotted in Fig. 1.6b. The progress of Moore's law is slower compared to the progress made by fusion experiments from the 1960s to the year 2000. The temperatures and densities necessary for a fusion

(a)   **(b)**

Figure 1.6: Triple product of current machines including the first two operational phases of Wendelstein 7-X (a) Cf. [14]. The increase of the triple product compared to Moore's law over time (b). The triple product is increasing faster. Source: EFDA

reactor with a positive energy balance can be produced with current technology. The confinement time is still the limiting factor on the way to commercially usable fusion power. There are three known forms of transport limiting confinement time.

1.3 Transport processes in fusion plasmas

Classical transport considers particle collisions in a homogeneous magnetic field. The resulting Brownian motion can be described with a random walk model with the gyroradius as the step size. This type of transport predicts confinement times in the order of hours. Hence, classical transport can be neglected, considering that the record confinement time in ASDEX Upgrade is 0.2 seconds.

Neoclassical transport considers particle collisions in an inhomogeneous magnetic field. Magnetic wells lead to trapped particle collisions. These lead to magnetic drift effects acting on larger length scales resulting in an increased transport-level compared to classical transport.

Anomalous transport is the name given to the cause of the unexpected transport levels an order of magnitude higher than predicted by the classical and neoclassical models. The original name of the Japanese machine JT-60 (see Fig. 1.6a) designed in the 1970s was "Breakeven Plasma Test Facility" (BPTF) [17]. When operations began in 1985, it performed far under the predictions. The same happened for the European machine JET and the US-American machine TFTR, which all began operation in the 1980s. Nowadays, it is widely accepted that these low levels of confinement are caused by turbulent transport. Electric and magnetic field fluctuations created by the plasma on the gyroradius scale lead to this type of transport.

Understanding the processes present in plasma microturbulence could lead to higher confinement times and the optimization of future fusion experiments for turbulent transport. Hence, this field of research has been very active in the past decades. Different types of instabilities exist, which can be distinguished clearly by their characteristic wavenumbers and frequencies for tokamak geometries. Three main drivers for turbulence are ion temperature gradient (ITG) modes, trapped electron (TEM) modes, and electron temperature gradient (ETG) modes. In this thesis, the primary linear mode is the ion temperature gradient (ITG) mode due to most simulations employing adiabatic electrons. In the following, the process leading to ITGs is discussed, starting from a single-particle picture.

The ITG mode

This description follows Ref. [18]. A charged particle moving in a homogeneous magnetic field gyrates around a magnetic field line due to the Lorentz force as illustrated in Fig. 1.7. The gyration is described by the frequency Ω and the gyroradius ρ
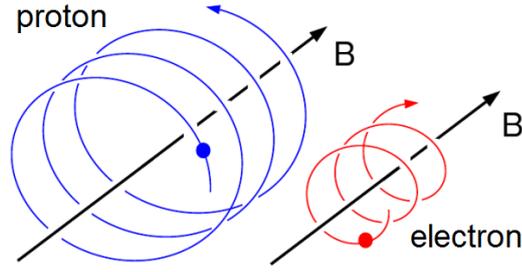
Figure 1.7: Motion of a proton and an electron around a homogeneous and stationary magnetic field line. Due to the greater mass of the ion its gyroradius is larger than the electron's see (1.4). Due to the different charges the electron and the ion rotate in different directions Cf. [11].

$$\Omega = \frac{qB}{m} \,, \tag{1.4}$$

$$\rho = \frac{mv_\perp}{qB} \,, \tag{1.5}$$

with the charge q and the mass m of the particle, the magnetic field strength B and the particles velocity component perpendicular to the magnetic field $v_\perp$.

The $\mathbf{E} \times \mathbf{B}$ drift

The next step is to add an electric field $\mathbf{E} = E_\perp \mathbf{e}_y + E_\parallel \mathbf{e}_z$ to the magnetic field $\mathbf{B} = B\mathbf{e}_z$. The component of the electric field parallel to the magnetic field accelerates the particle along the field line. Additionally, the gyrocenter is moved perpendicular to both the magnetic field and the electric field by the velocity

$$\mathbf{v}_E = \frac{\mathbf{E} \times \mathbf{B}}{B^2} \,. \tag{1.6}$$

This so called $\mathbf{E} \times \mathbf{B}$ drift moves electrons and ions in the same direction with the same velocity as it neither depends on charge nor mass of the particle. The independence on species can be explained by considering that the $E_\perp$ field accelerates the particles on one half of the gyroring and decelerates the particles on the other half. The velocity dependence of the gyroradius (1.4) results in the drift. When considering protons instead of electrons, the sign changes twice. Once for the direction of gyration and once for the charge. Hence, both particle species drift in the same direction. This is visualized in Fig. 1.8.

The ∇B drift

The next step is considering a gradient of the magnetic field ∇B. If the magnetic field points in z direction and the gradient points in y direction the resulting drift of the
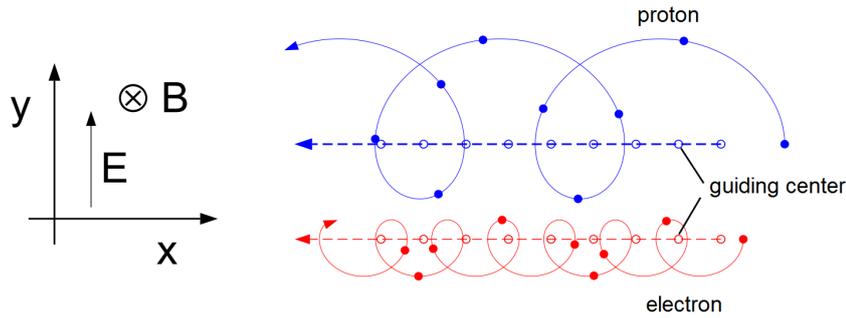
Figure 1.8: Movement of an electron and a proton in a homogeneous and stationary electromagnetic field. The $\mathbf{E} \times \mathbf{B}$ drift move the gyrocenter in the same direction with the same velocity Cf. [11].

gyrocenter is once again perpendicular to both and is calculated as

$$\mathbf{v}_{\nabla B} = \frac{mv_\perp^2}{2qB^3}\mathbf{B} \times \nabla B\,. \tag{1.7}$$

This ∇B drift depends on the mass and charge of the particle in contrast to the $\mathbf{E} \times \mathbf{B}$ drift. The gyroradius is affected by the change of the magnetic field strength. This process is independent of the charge but does depend on the direction of gyration as illustrated in Fig. 1.9. Hence, there is only one change in sign and the ∇B drift shifts the gyrocenters of protons and electrons opposite directions. Combined these two effects
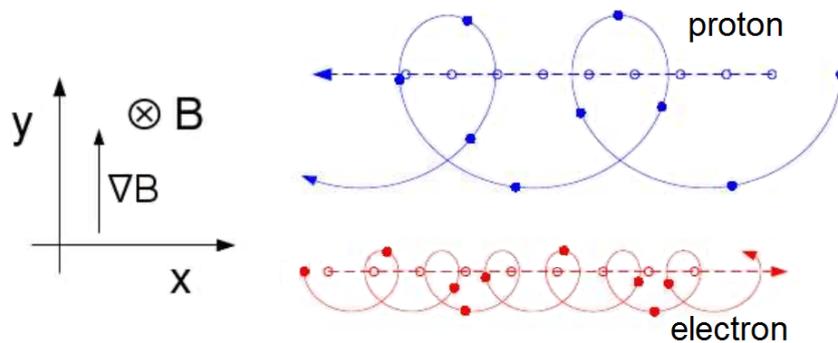
Figure 1.9: Movement of an electron and a proton in a stationary magnetic field with a gradient. The ∇B drift moves the gyrocenters of electrons and protons in opposite direction Cf. [11].

lead to the emergence of ITG modes (see Fig. 1.10).

Geometric dependence of the ITG mode

Consider the outboard side of a tokamak. Here, the temperature is higher at the magnetic axis than at the edge due to heating mechanisms, which mainly target the center of the plasma. The plasma closer to the magnetic axis hence moves with higher velocities (see the left picture in Fig. 1.10). If there is a small disturbance (right picture in Fig. 1.10), the

∇B drift leads to a separation of electrons and ions. This charge separation generates an electric field, causing an $\mathbf{E} \times \mathbf{B}$ drift, which amplifies the disturbance. As a consequence, hot plasma is moved radially outward, leading to low confinement times. These radial streamers can only grow where ∇T and ∇B point in the same directions. These areas are called bad curvature regions. In a tokamak, the bad curvature region is simply on the outboard plane of the torus. In a stellarator like Wendelstein 7-X, the bad curvature regions can be located in different areas due to the complex magnetic field geometry. This is explained in detail in Sec. 5. The basis of gyrokinetic theory is to look at the
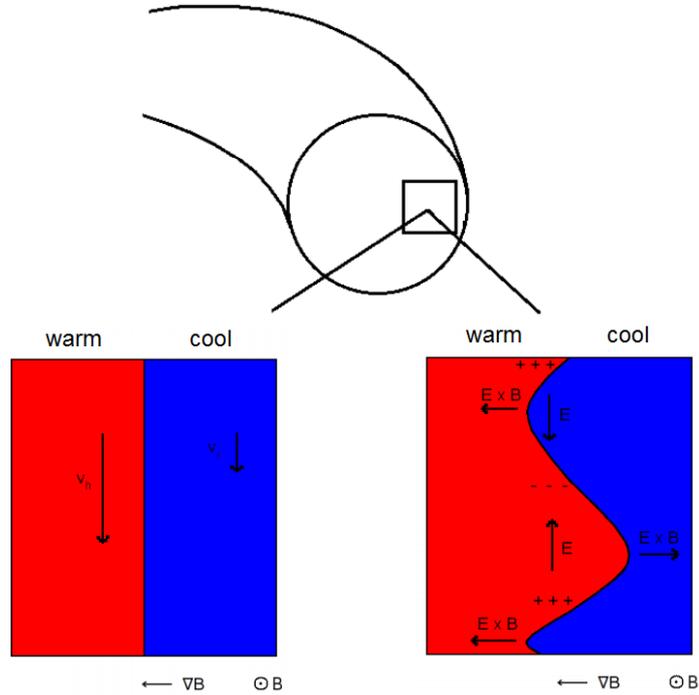
Figure 1.10: Schematic illustration of the beginning of an ITG mode from a temperature gradient and a magnetic field gradient in a region of bad curvature. The initial disturbance is amplified by the ∇B and $\mathbf{E} \times \mathbf{B}$ drift Cf. [19].

evolution of the gyrocenter and average out forces like the above mentioned drifts acting on the high frequency gyromotion. Gyrokinetic theory is explained in detail in Sec. 2.

1.4   Simulation codes employing the gyrokinetic model

Experiments often cannot measure all quantities to the degree of accuracy necessary to develop predictive models. Also, experiments are expensive to run and limited in the parameter ranges. Analytic models, on the other hand, can only be used to describe simple geometries. Hence, a numerical description of plasma microturbulence is required.

In the simulated plasma volume, the inhomogeneities in density, temperature, and magnetic field give rise to microinstabilities as the system tries to relax to thermodynamic

equilibrium. High temperature, low-density fusion plasmas are weakly collisional. Hence, kinetic theory is needed to capture the fine details of the physics taking place. In principle, the six dimensional Vlasov equation

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \dot{\mathbf{x}} \cdot \frac{\partial f}{\partial \mathbf{x}} + \dot{\mathbf{v}} \cdot \frac{\partial f}{\partial \mathbf{v}} = 0 \, , \tag{1.8}$$

needs to be solved for each particle species while also taking the coupling to the electromagnetic fields into account via Maxwell's equations. Fortunately, turbulent fluctuations typically have a much lower frequency than the high frequency gyration of the charged particles around the gyrocenters. Therefore, the 6D problem can be reduced to a 5D problem by averaging over the gyromotion. This model is known as the gyrokinetic model. It is explained in detail in Sec. 2.

However, even with this reduction of complexity, developing an efficient simulation tool remains challenging. The codes developed for this purpose all make extensive use of high-performance computing (HPC) resources. Still, there is not a single code that can simulate all relevant physics on present-day supercomputers while considering multiple species and self-consistently evaluated electromagnetic fluctuations. Therefore, the available simulation tools can be grouped into categories, each with their strengths and weaknesses attributed to the individual simplifications and approximations. An overview of different simplification strategies to reduce numerical cost and comparison to actual experiments can be found in Ref. [20].

The flux tube approach

One method to reduce computation costs by orders of magnitude is the flux tube approach: Here, the simulation domain is reduced to a small plasma volume aligned with the magnetic field lines. Radial dependencies of macroscopic quantities such as density and temperature and their gradients are neglected. The local version of GENE [21] - among others - applies this technique. In contrast, the radially global approach takes radial dependencies of equilibrium quantities into account. As a consequence, a large number of grid points is necessary, making these simulations expensive. However, the radially global model has to be used when the length scales of the turbulence are comparable to the length scale of the geometry like the minor radius.

Full-f and δf codes

Another distinction can be made between full-f and δf codes: They differ in their representation of the distribution function f. The δf model assumes a constant background distribution f_0 (usually Maxwellian) and a typically small perturbation δf, which is

variable in time. In contrast, full-f codes evolve the entire distribution function. This accounts for the back reaction of turbulent transport when evolving the equilibrium.

Grid-based and particle-in-cell codes

Another difference is the implemented numerical schemes. The majority of gyrokinetic codes can be categorized into two groups: Lagrangian particle-in-cell (PIC) codes and the grid-based Eulerian codes. PIC methods have historically been very successful as they are easy to parallelize. The PIC method for gyrokinetics was pioneered by Lee (1983) [22] and is used by many codes today like XGC [23], EUTERPE [24, 25], ORB5 [26, 27] and many others. However, one crucial drawback of PIC methods is the inherent numerical noise, limiting the accuracy of the distribution function on long time scales. Numerical noise only decreases slowly as $1/\sqrt{N}$ with the number of particles N in the simulation. The main problem for nonlinear gyrokinetic simulations is that the noise level can accumulate in time [28]. Small errors in the evaluation of moments can cause a systematic error even in relatively short time frames.

Eulerian codes discretize phase space on a fixed grid and apply finite differences, finite volumes or (pseudo-) spectral methods to calculate derivatives and integrals (see Ref. [29] for a review). Most Eulerian codes implement an explicit time integration scheme which has the drawback of the Courant-Friedrichs-Lewy (CFL) condition. It puts a small upper limit for the time step used in the simulations. With increased grid resolution, this limitation can make the simulations computationally expensive. All versions of GENE [21, 30, 31, 32] use the Eulerian approach.

The GENE code family

In the GENE code family, different codes have been developed for different applications (see sec. 3.1). The flux tube version does not consider radial or toroidal geometric variations. The radially global or tokamak version takes radial variations into account but no toroidal variations, which is an accurate model for axisymmetric tokamaks. The flux surface version is capable of taking toroidal variations into account while neglecting radial dependencies. Hence, a tool was missing which was capable of simulating 3D geometries to represent stellarators and tokamaks with resonant magnetic field perturbations accurately. This gap is now closed with the development of GENE-3D. Currently only EUTERPE [24, 25] and XGC [33] have the same capability. Despite implementing different models and numerical schemes, the codes should yield the same physical results when considering the same physical system. Hence, benchmarks between these codes are presented for two different stellarator geometries in Sec. 4.2.

GENE-3D

In summary, GENE-3D is a gyrokinetic, electrostatic[1], nonlinear, Eulerian, δf code which is capable of simulating an arbitrary number of species in any kind of geometry. Specifically, the equilibrium magnetic field B_0 is a function of all three spatial coordinates $B_0 = B_0(x, y, z)$ in GENE-3D. Also, the metric tensor g^{ij} is dependent on all three spatial coordinates. The background profiles in density and temperature are still only functions of the radial coordinate $n = n(x)$ and $T = T(x)$. The significant development steps leading to the current version of GENE-3D are outlined in Sec. 3.

In a non-axisymmetric device, the magnetic geometry varies among flux-tubes on the same flux-surface. Therefore, flux-tube simulations of core plasmas do not cover all relevant physical effects, and flux-surface simulations are an improvement [34]. However, to cover a 3D geometry entirely a fully global approach is necessary, which can simulate multiple flux surfaces. Radially global effects matter when the turbulent correlation length is on comparable length scales as the gradient lengths. As the minor radius of W7-X is about 0.5m, this condition is - dependent on the setup of the shot - fulfilled. Examples of simulations where the flux surface simulations are not enough to cover all relevant geometric effects in W-7X are performed in Sec. 5.2.

Also, tokamaks are not perfectly axisymmetric towards the edge, and axisymmetry is broken in tokamaks with resonant magnetic field perturbations. A recent project studying ITG turbulence in the presence of magnetic perturbations in ASDEX Upgrade using GENE-3D can be found in Ref. [35]. Hence, the fully global GENE-3D code developed throughout this project has a variety of applications.

1.5   Stellarator gyrokinetics

In a fusion reactor the net power gain scales nonlinearly with the confinement time $P_{fus} \propto \tau_E^{4/3}$ [36]. As modern machines like Wendelstein-7X are already optimized for neoclassical transport the contribution of turbulence becomes dominant in the radial transport of heat and particles. In fact, in the second operation phase of W7-X turbulence was the dominant driver of transport [13]. Microturbulence in stellarators is shown to have different characteristics than tokamak turbulence [37, 38, 34, 39]. Therefore, understanding turbulent transport in stellarators becomes increasingly relevant.

It has been found that the distinction of different modes for stellarators is not always as simple compared to tokamaks. In Ref. [37], the eigenvalue solver of the flux tube version of GENE is used to solve the linear system directly. Ten and more modes with similar growth rates and different characteristics have been found compared to tokamaks, where there usually are just one or two dominant modes in the system. The number of unstable modes increased with the level of the temperature gradient. Different field lines

---

[1]the development of an electromagnetic version is currently ongoing

on the same flux surface are not equivalent and lead to different flux tubes. It was found that the level of transport could vary strongly for different geometric configurations in the flux tube model. This dependence is taken as motivation for Sec. 5.1 where different geometric configurations of Wendelstein 7-X are investigated with GENE-3D for differences in linear stability and nonlinear transport of heat. Here, all flux surfaces and field lines are taken into account while all modes in k_y and k_x can grow at the same time.

Already in Ref. [37] considering the flux-tube version of GENE, it is discussed that for an accurate description of W7-X five times more points in parallel direction are required compared to a tokamak. The increased resolution reduces the time step significantly due to the CFL criterion [40]. Although GENE-3D is parallelized, highly scalable (see Sec. 3.2.7), and optimized, the resolution and time step requirements for simulations with 3D geometries require an enormous computational effort. The simulations performed during this project, which are presented in this dissertation, are amongst the first few 3D gyrokinetic stellarator simulations.

1.6 Stellarator optimization

This subsection gives a short overview over stellarator optimization based on Ref. [41]. A more detailed description of the entire process can be found in Ref. [41]. As mentioned before and illustrated in Fig. 1.4b, the coil design of stellarators is rather complicated. Great effort is made to develop and design the coils such that the magnetic field configuration is optimized for different quality criteria. In the predecessor of Wendelstein 7-X, Wendelstein 7-AS, which started operation in 1988, the magnetic configuration could not be changed significantly, which lead to limitations in the experiments. Hence, its successor Wendelstein 7-X was developed with a configurational space allowing flexibility in the magnetic field geometry. This is realized by not only having the non-planar coil set illustrated in Fig. 1.4b but also a set of planar coils. Due to the symmetry of W7-X the 70 coils consist of five different non-planar coil types and two different planar coil types as illustrated in Fig. 1.11.

The classic stellarator optimization process can be summarized as follows: First, an "ideal" magnetic field is designed that fits best to certain performance criteria. In the second step, the coil optimization procedure is undergone in which magnetic field coils are found, which produce a magnetic field as close to the "ideal" magnetic field as possible. One improvement of this process is realized in the ONSET code [43]. Here, the magnetic field generated by the coils is also optimized for the quality criteria. These two types of stellarator optimization lead to designs for Wendelstein 7-X and NCSX [44], which are introduced in the following.
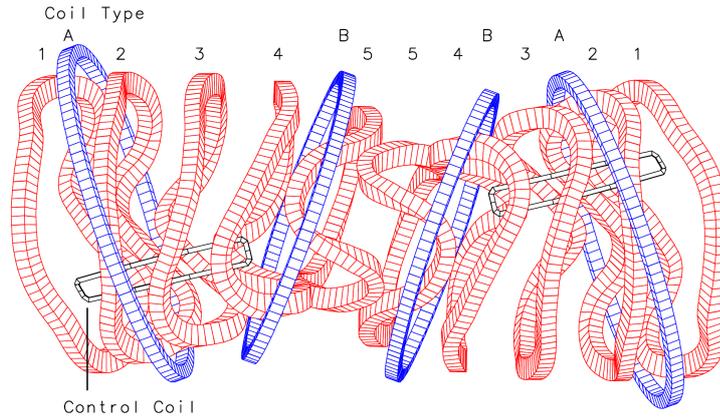
Figure 1.11: The five (1-5) non-planar coil types and two planar coil types (A-B) Cf. [42].

Construction of Wendelstein 7-X

Wendelstein 7-X has the largest major radius of all stellarators and started operation in 2015. It was shown that the magnetic field topology needed for good confinement could be accurately reproduced with deviations of the order of 1:100,000 [45]. W7-X aims at answering physics and engineering questions relevant for a future fusion reactor. This includes planned continuous plasma discharges of up to 30 minutes. To achieve the accurate reproduction of the magnetic field geometry coil tolerances of less than 0.17% were required [46]. While these strict tolerance requirements were met, they led to high cost and construction times, as stated in Ref. [47]: "The assembly process which took about 1,000,000 man-hours up to March 2014, was essentially dominated by the high demands on tolerances for the position of the superconducting coils".

Construction of NCSX

The National Compact Stellarator eXperiment (NCSX) is a compact stellarator designed at Princeton Plasma Physics Laboratory (PPPL) in the United States. Its construction tolerances needed to be even smaller ($< 0.08\%$) than the ones of Wendelstein 7-X. The construction starting in 2003 could not meet these strict tolerance requirements. After exceeding time and money budgets for years, the project was finally entirely canceled in 2008 [48].

These two examples show the importance of construction tolerances for the feasibility and cost of stellarator projects.

Construction of CNT

The Columbia Non-neutral Torus (CNT), which went into operation in 2004, followed a different approach. A Monte-Carlo type algorithm was used to investigate perturbations to the coil design [49]. Instead of choosing the magnetic field configuration with the best

performance, a configuration was chosen with the largest resilience to coil displacement and current deviations. This method led to construction tolerances of up to 1% [50]. The construction was hence fast and inexpensive as discrepancies to the design were acceptable. It was found that the coils were displaced compared to the original design [51]. Also, the robustness of the magnetic flux surfaces was confirmed in Ref. [52] proving the robustness of the chosen coil geometry against errors in the construction and installation.

Stellarator optimization with the stochastic version of ONSET and GENE-3D

The stochastic version of ONSET extended the CNT approach to the general multivariate nonlinear coil optimization problem and demonstrated a better performance of the resulting coil configurations in form of a better approximation of the magnetic field and increased resilience against coil perturbations [53]. The neoclassical optimization of Wendelstein 7-X could be demonstrated experimentally [54] and the turbulent transport was found to be the main driver of radial transport of heat in the core plasma of Wendelstein 7-X [13]. Hence, combining the stochastic coil optimization strategy with GENE-3D could lead to future devices that are optimized for turbulent transport and are cheaper and faster to build due to higher construction tolerances. Sec. 5.2 demonstrates that geometric effects can only be fully captured by a 3D global code. Hence, the GENE-3D simulations in Sec. 5.1 are the first to investigate gyrokinetic turbulence optimization of a stellarator in 3D geometries. Here, linear stability and nonlinear transport of heat are investigated for magnetic field geometries produced by a current sheet, a coil design calculated by ONSET, and a coil design calculated by the stochastic extension of ONSET.

1.7 Thesis outline

The thesis is organized as follows, the gyrokinetic equations underlying GENE-3D are outlined in Sec. 2. The implementation of the gyrokinetic equations and their discretization in GENE-3D is described in 3.1. Sec. 3.2 discusses the developmental steps necessary on the way to using GENE-3D as production code. Sec. 3.3 then introduced the production chain around GENE-3D, including the geometric preprocessing by GVEC [55] and the postprocessing with Python and ParaView. The GENE-3D code will be used for the investigations in the following sections starting with benchmarks against the tokamak version of GENE for axisymmetric systems and EUTERPE and XGC for stellarator geometries in Sec. 4. Sec. 5.1 investigates the impact of geometric variations on linear stability and turbulent transport in Wendelstein 7-X. Sec. 5.2 then tests the influence of the ratio of the machine size to the gyroradius on linear stability and turbulent trans-

port for a Wendelstein 7-X geometry. Lastly, Sec. 6 summarizes the results and gives an outlook for future projects.

2   Gyrokinetic theory and application to GENE-3D

This chapter describes the model and the underlying equations for the nonlinear gyrokinetic Vlasov code GENE-3D similar to Ref. [31] which was written at the same time as this thesis. To this end, first, the assumptions for the gyrokinetic model are explained in Sec. 2.1. Then the gyrokinetic Vlasov equation is outlined in Sec. 2.2. The gyrokinetic Vlasov equation is coupled to the gyrokinetic Poisson equation, which is introduced in Sec. 2.3. Finally, a coordinate system following the magnetic field geometry is outlined in Sec. 2.4.

2.1   Assumptions and ordering in gyrokinetics

Fusion plasmas have high temperatures and low densities and are therefore weakly collisional. Hence, fluid models that are based on high collisionality are insufficient, and a kinetic description is needed. The most fundamental way to model a plasma is by a six-dimensional distribution function $f(x, v, t)$ in the Vlasov equation coupled to Maxwell's equations with three spatial coordinates and three velocity coordinates. A six-dimensional description of plasma microturbulence would, however, not be feasible to simulate even on the most powerful supercomputers to date. For the study of low frequency ($\omega \ll \Omega_i$ with the ion gyration frequency $\Omega_i = eB/m_i c$) plasma microturbulence, the description can be simplified by averaging over the gyration of the charged particles around the magnetic field line.

Only the slow drift of the gyrocenters leads to radial transport of heat and particles, while the fast gyromotion is of no practical interest. Therefore, the reduction of the gyromotion to charged rings moving in the ring averaged fields is a valid simplification. These assumptions lead to the gyrokinetic model, which has first been developed throughout the 1980s [56, 57, 58] and is the basis for all versions of the GENE code. An easy to read introduction to the topic can be found in Ref. [59]. The derivation for the flux-tube version of GENE is given in the Ph.D. thesis in Ref. [60] while the extension to the so-called global tokamak or "x-global" GENE version taking radial dependencies of temperature, density and geometry into account is described in Refs. [61, 30].

The next evolution of the GENE code, namely the extension to 3D geometries, is described in this section. As the gyrokinetic derivation outlined in the sources above is general and includes three-dimensional geometry variations, it is not repeated here, but rather the differences to the x-global GENE version are emphasized in the current and the following sections describing the implementation of the gyrokinetic equations in GENE-3D. Still, all relevant equations and the assumptions underlying them are outlined here similarly to Ref. [31].

The main ideas and assumptions entering the gyrokinetic derivation are summarized in the following.

- **Low frequency:** The characteristic fluctuation frequency is small compared to the ion gyration frequency $\rightarrow \omega \ll \Omega$

- **Anisotropy:** The turbulent fluctuations have perpendicular correlation lengths of around $10 - 100$ gyroradii while the parallel length scales can be of the order of meters. Expressed in wave numbers parallel and perpendicular to the background magnetic field this statement becomes $k_\perp \gg k_\parallel$

- **Strong magnetization:** The larmor radius $(\rho_\sigma = v_\perp/\Omega_\sigma = v_\perp m_\sigma c/q_\sigma B)$ is small compared to the gradient length scales of the background density $L_n = n_0/(dn_0/dx)$, the gradient length scales of the background temperature $L_T = T_0/(dT_0/dx)$ and the gradient length scales of the background magnetic field $L_B = B_0/(dB_0/dx) \rightarrow \rho_\sigma \ll |L_n|$, $\rho_\sigma \ll |L_T|$ and $\rho_\sigma \ll |L_B|$

- **Small fluctuations:** The fluctuating part of the gyrocenter distribution function is assumed to be small compared to the background distribution function $\rightarrow F_1 \ll F_0$. Also the potential energy of the charged particles in the fluctuating part of the electrostatic potential is small compared to the energy of the background temperature $\rightarrow q\phi_1 \ll T_0$

Taken together these assumptions lead to the gyrokinetic ordering [56, 62]

$$\frac{\omega}{\Omega} \sim \frac{k_\parallel}{k_\perp} \sim \frac{\rho}{L_T} \sim \frac{\rho}{L_n} \sim \frac{F_1}{F_0} \sim \frac{q\phi_1}{T_0} \sim \epsilon_\delta \,, \quad \frac{\rho}{L_B} \sim \epsilon_B \,, \tag{2.1}$$

where $\epsilon_B \approx \epsilon_\delta^2$ means that the background magnetic field varies on even larger spatial scales than the other background quantities. The gyrokinetic model used for the derivation of the equations implemented in GENE-3D is valid up to $\mathcal{O}(\epsilon_\delta)$. The gyrokinetic ordering is based on many years of experimental observations and theoretical analysis of microinstabilities in fusion plasmas [57].

The area of validity of the gyrokinetic model

The gyrokinetic model is valid in the core region, with exceptions for small fusion devices. At the edge region the validity of the assumptions can be questioned [63, 64]. For the application of core stellarator physics, the gyrokinetic ordering is assumed to be valid. However, the gyrokinetic model does not represent many plasma phenomena both on small and large time scales. Among others, the small spatio-temporal effects are Debye shielding, cyclotron resonances, and compressional Alfvén waves. On large time scales, features like the time evolution of the magnetic flux surfaces are lost as the turbulence

time scales are much smaller and the Magnetohydrodynamic (MHD) equilibrium is assumed to be constant over the time scale of a gyrokinetic simulation. MHD codes that capture these features are run beforehand to provide a magnetic configuration for the gyrokinetic simulations. In the case of GENE-3D, the Galerkin Variational Equilibrium Code (GVEC [55]) (see sec. 3.3.1) is used to create a geometric input file for GENE-3D.

In the following, the coupled gyrokinetic Vlasov-Maxwell system of equations is outlined, which are the underlying equations of GENE-3D.

2.2   The gyrokinetic equations

The derivation of the gyrokinetic equations   can be done by starting with the Lagrangian of a particle with charge q in a electromagnetic field with vector potential $\mathbf{A}(\mathbf{x})$ and scalar potential $\phi(\mathbf{x})$

$$\mathcal{L}(\mathbf{x}, \mathbf{v}) = \left(m\mathbf{v} + \frac{q}{c}\mathbf{A}(\mathbf{x}) \right) \cdot \dot{\mathbf{x}} - \frac{mv^2}{2} - q\phi(\mathbf{x}) \,. \tag{2.2}$$

In order to adapt the coordinate system to the helical motion of charged particles around the field lines the one-form γ is needed which is calculated from the action integral $\mathcal{S}$ as

$$\mathcal{S} = \int \mathcal{L}(\mathbf{x}, \mathbf{v}) dt = \int d\gamma \,. \tag{2.3}$$

The transformation from particle space to guiding center space   can then be performed which describes the center of gyration $\mathbf{X}$ and the gyroradius vector or distance vector from the center of gyration to the particle position orthogonal to the magnetic field $\mathbf{r}(\mathbf{X}, \mu, \theta)$ with the magnetic moment

$$\mu = \frac{mv_\perp^2}{2B_0} \,, \tag{2.4}$$

and the gyroangle θ. With the velocity component parallel to the magnetic field $v_\parallel$, the velocity component perpendicular to the magnetic field $v_\perp$, the unit vector $\mathbf{a}_0$ pointing from the gyrocenter to the particle,

$$\mathbf{a}_0(\theta) = \hat{x}\cos\theta - \hat{y}\sin\theta \,, \tag{2.5}$$

the unit vector in magnetic field direction $\mathbf{b}_0$ and the unit vector $\mathbf{c}_0$ pointing along the gyroring

$$\mathbf{c}_0(\theta) = \frac{\partial}{\partial\theta}\mathbf{a}_0(\theta) = -\hat{x}\sin\theta - \hat{y}\cos\theta \,, \tag{2.6}$$

the relation between particle and guiding center space can be written as

$$\mathbf{x} = \mathbf{X} + \mathbf{r}(\mathbf{X}, \mu, \theta),$$
$$\mathbf{v} = v_\parallel \mathbf{b}_0(\mathbf{X}) + v_\perp(\mathbf{X}, \mu)\mathbf{c}_0(\theta).$$

$$(2.7)$$

The one-form in guiding center coordinates $\Gamma(\mathbf{X})$ can be obtained by utilizing the transformation of coordinates on the one-form $\gamma(\mathbf{x})$,

$$\gamma_\mu dz^\mu = \gamma_\mu \frac{\partial z^\mu}{\partial Z^\mu} dZ^\mu = \Gamma_\mu dZ^\mu.$$

$$(2.8)$$

Before the actual transformation is performed the one-form γ is split into an unperturbed part with a time-independent background magnetic field $\mathbf{B_0} = \nabla \times \mathbf{A_0}$ and a background scalar potential ϕ_0 and a fluctuating part containing perturbations in the potentials ϕ_1 and $\mathbf{A}_1$

$$d\gamma_0 = \left(m\mathbf{v} + \frac{q}{c}\mathbf{A}_0(\mathbf{x})\right) \cdot d\mathbf{x} - \left(\frac{1}{2}mv^2 + q\phi_0(\mathbf{x})\right) dt,$$
$$d\gamma_1 = \frac{q}{c}\mathbf{A}_1(\mathbf{x}) \cdot d\mathbf{x} - q\phi_1(\mathbf{x})dt.$$

$$(2.9)$$

Using the gyrokinetic assumption of spatially slowly varying background quantities the unperturbed quantities at particle position are replaced by their values at the gyrocenter. Furthermore using the gyroaverage operator $\mathcal{G}$ defined as

$$\mathcal{G}[f(x)] = \bar{f}(\mathbf{X}) = \frac{1}{2\pi} \oint f(\mathbf{X} + \mathbf{r}(\alpha)) \, d\alpha,$$

$$(2.10)$$

to remove the gyroangle dependence the unperturbed one-form in guiding center coordinates reads

$$\Gamma_0 = \left(mv_\parallel \mathbf{b}_0(\mathbf{X}) + \frac{q}{c}\mathbf{A}_0(\mathbf{X})\right) \cdot d\mathbf{X} + \frac{\mu mc}{q} d\theta - \left(\frac{1}{2}mv_\parallel^2 + q\phi_0(\mathbf{X}) + \mu B_0(\mathbf{X})\right) dt. \quad (2.11)$$

The perturbed one-form in guiding center coordinates Γ_1 cannot be obtained by simply applying the gyroaverage as the fluctuating quantities vary on the length scale of the gyroradius. Hence, another coordinate system is required for which the one-form Γ_1 is independent of the gyroangle. The gyrocenter coordinates fulfill this requirement. The guiding center one-form can be transformed into the gyrocenter one-

form by applying (near-identity) Lie transformations. The final result is then

$$
\begin{aligned}
\Gamma = {} & \Gamma_0 + \Gamma_1 \\
= {} & \left(\left(\left(mv_\parallel + \frac{q}{c} \bar{A}_{1\parallel} \right) \mathbf{b}_0(\mathbf{X}) + \frac{q}{c} \mathbf{A}_0(\mathbf{X}) \right) \cdot d\mathbf{X} + \frac{\mu mc}{q} d\theta \right. \\
& \left. - \left(\frac{1}{2} mv_\parallel^2 + q \left(\phi_0(\mathbf{X}) + \bar{\phi}_1(\mathbf{X}) \right) + \mu \left(B_0(\mathbf{X}) + \bar{B}_{1\parallel}(\mathbf{X}) \right) \right) dt \right) ,
\end{aligned}
\tag{2.12}
$$

with the unperturbed one-form being the same in guiding center and gyrocenter coordinates.

Vlasov equation

The equation describing the collisionless time evolution of the gyrocenter distribution function $F_\sigma(\mathbf{X}, v_\parallel, \mu, t)$ for each particle species σ is the so-called Vlasov equation

$$
\frac{dF_\sigma}{dt} = \frac{\partial F_\sigma}{\partial t} + \frac{d\mathbf{X}}{dt} \cdot \nabla F_\sigma + \frac{dv_\parallel}{dt} \frac{\partial F_\sigma}{\partial v_\parallel} = 0 ,
\tag{2.13}
$$

where $d\mu/dt = 0$ is utilized as one of the aims of the gyrokinetic derivation is to have μ as a constant of motion [57]. By using the Euler-Lagrange equations

$$
\frac{d}{dt} \frac{\partial L}{\partial \dot{Z}_i} - \frac{\partial L}{\partial Z_i} = 0 ,
\tag{2.14}
$$

expressions for $d\mathbf{X}/dt$ and $dv_\parallel/dt$ can be derived from the gyrocenter one-form (2.12)

$$
\frac{d\mathbf{X}}{dt} = v_\parallel \mathbf{b}_0 + \mathbf{v}_{E_0} + \mathbf{v}_{E_1} + \mathbf{v}_{\nabla B_0} + \mathbf{v}_c ,
\tag{2.15}
$$

$$
\frac{dv_\parallel}{dt} = -\frac{d\mathbf{X}/dt}{m_\sigma v_\parallel} \cdot \left(q_\sigma \nabla \left(\phi_0 + \bar{\phi}_1 \right) + \mu \nabla \left(B_0 + \bar{B}_{1\parallel} \right) \right) .
\tag{2.16}
$$

The physical meaning of the terms in the equation of motion for $\mathbf{X}$ are (in order) the undisturbed motion along the magnetic field line, the $\mathbf{E} \times \mathbf{B}$ terms of the equilibrium and the perturbed part of the electric field

$$
\mathbf{v}_{E_0} = \frac{c}{B_0^2} \mathbf{B}_0 \times \nabla \phi_0 ,
\tag{2.17}
$$

$$
\mathbf{v}_{E_1} = \frac{c}{B_0^2} \mathbf{B}_0 \times \nabla \bar{\phi}_1 ,
\tag{2.18}
$$

where the unperturbed part of the electrostatic potential ϕ_0 can be utilized to investigate long-wavelength radial electric field effects. The velocity stemming from the gradient of

the magnetic field is given by

$$\mathbf{v}_{\nabla B_0} = \frac{\mu c}{q_\sigma B_0^2} \mathbf{B}_0 \times \nabla B_0 \,, \tag{2.19}$$

while the velocity imposed by the magnetic field curvature is given by

$$\mathbf{v}_c = \frac{v_\parallel^2}{\Omega_\sigma} \left(\nabla \times \mathbf{b}_0 \right)_\perp \,. \tag{2.20}$$

Note that these equations are electrostatic as is the version of GENE-3D at the time of writing this thesis.

The δF splitting

A method commonly used in the derivation of equations for gyrokinetic codes is the splitting of the distribution function F into an equilibrium part F_0 and a fluctuating part F_1. This so-called δF splitting exploits the relation $F_1 \ll F_0$ of the gyrokinetic ordering in Eq. (2.1). This method is known to save computational resources compared to evolving the entire distribution function [20]. One of the reasons for this is that a full-F code would require a nonlinear field solver, which is computationally much more expensive than the linear ones used in δF codes. Also, as soon as numerical discretization is applied, the distribution function can occasionally become negative, which violates the positivity of F. Numerical schemes enforcing positivity of the distribution function are generally computationally expensive. Here, the gyrocenter distribution function of species σ is split into an equilibrium part $F_{0\sigma}$ and a fluctuating part $F_{1\sigma}$

$$F_\sigma = F_{0\sigma} + F_{1\sigma} \,, \tag{2.21}$$

where the equilibrium distribution function $F_{0\sigma}$ is assumed to be constant in time as it evolves on the heating time-scale while the fluctuating part $F_{1\sigma}$ evolves on the much smaller turbulent time scale. In line with the gyrokinetic ordering, the equilibrium distribution function is assumed to vary on a macroscopic length scale both parallel and perpendicular to the magnetic field

$$(\hat{\mathbf{z}} \cdot \nabla F_{0\sigma}) \sim |\hat{\mathbf{z}} \times \nabla F_{0\sigma}| \sim \mathcal{O}(F_{0\sigma}/L_F) \,,$$

while the fluctuating part is assumed to be varying on a macroscopic length scale in parallel direction and on microscopic length scales of the order of the gyroradius in

perpendicular direction

$$(\hat{\mathbf{z}} \cdot \nabla F_{1\sigma}) \sim \mathcal{O}(F_{1\sigma}/L_F)\,,$$
$$|\hat{\mathbf{z}} \times \nabla F_{1\sigma}| \sim \mathcal{O}(F_{1\sigma}/\rho_\sigma)\,.$$

The background distribution function F_0 needs to fulfill the following condition derived from Eq. (2.13) by keeping only terms of zeroth order in ϵ,

$$v_\| \hat{\mathbf{b}}_{\mathbf{0}} \cdot \left(\nabla F_{0a} - \frac{\mu}{m_\sigma v_\|} \frac{\partial F_{0\sigma}}{\partial v_\|} \nabla B_0 \right) = 0\,. \tag{2.22}$$

Different functions fulfill the condition given by Eq. (2.22). A Maxwellian is commonly used in gyrokinetic codes as it solves the gyrokinetic equations for thermalized plasmas locally and only requires information about density, temperature, and magnetic field strength of the background as input parameters for the simulation. Hence, in GENE-3D the equilibrium distribution function F_0 is assumed to be a local Maxwellian constant in time

$$F_{0\sigma}(\mathbf{X}, v_\|, \mu) = \frac{n_{0\sigma}(\mathbf{X})}{\pi^{3/2} v_{T\sigma}^3(\mathbf{X})} \exp\left[-\frac{m_\sigma v_\|^2/2 + \mu B_0(\mathbf{X})}{T_{0\sigma}(\mathbf{X})} \right]\,. \tag{2.23}$$

"Local" here means that the thermal equilibrium is different for each flux surface. The coordinate system is formally introduced in Sec. 2.4.

The derivatives of the background distribution function in the Vlasov equation (2.13) can be calculated from the analytic expression for the equilibrium. The gradient of the equilibrium distribution function $\nabla F_{0\sigma}$ can be split into its components as the equilibrium density profiles and temperature profiles are only dependent on the radial coordinate,

$$\partial_x F_{0\sigma} = \left[\frac{\partial_x n_{0\sigma}}{n_{0\sigma}} + \frac{\partial_x T_{0\sigma}}{T_{0\sigma}} \left(\frac{m_\sigma v_\|^2/2 + \mu B_0}{T_{0\sigma}} - \frac{3}{2} \right) - \frac{\mu}{T_{0\sigma}} \partial_x B_0 \right] F_{0\sigma}\,, \tag{2.24}$$

$$\partial_{y,z} F_{0\sigma} = \left(-\frac{\mu}{T_{0\sigma}} \partial_{y,z} B_0 \right) F_{0\sigma}\,, \tag{2.25}$$

$$\partial_{v_\|} F_{0\sigma} = \left(-\frac{m_\sigma v_\|}{T_{0\sigma}} \right) F_{0\sigma}\,. \tag{2.26}$$

The δF-splitted gyrokinetic Vlasov equation   in vector form is then derived by keeping only terms of first order in ϵ for the fluctuating part of the distribution function,

$$
\frac{\partial F_{1\sigma}}{\partial t} + v_\parallel \hat{\mathbf{b}}_0 \cdot \Gamma_\sigma - \hat{\mathbf{b}}_0 \cdot \frac{\mu}{m_\sigma} \nabla B_0 \frac{\partial F_{1\sigma}}{\partial v_\parallel} + \left(\mathbf{v}_{\nabla B_0} + \mathbf{v}_c \right) \cdot \Gamma_\sigma + \left(\mathbf{v}_{E_0} + \mathbf{v}_{E_1} \right) \cdot \nabla F_{1\sigma}
$$
$$
+ \mathbf{v}_{E_1} \cdot \left[\nabla F_{0\sigma} + \mu \nabla B_0 \frac{F_{0\sigma}}{T_{0\sigma}} \right] + \overbrace{\left(\mathbf{v}_{\nabla B} + \mathbf{v}_c \right) \cdot \nabla F_{0\sigma}}^{0} = 0 \,,
\tag{2.27}
$$

with the abbreviation

$$
\Gamma_\sigma = \nabla F_{1\sigma} + \frac{q_\sigma}{T_{0\sigma}} F_{0\sigma} \nabla \bar{\phi}_1 \,.
\tag{2.28}
$$

The last term in Eq. (2.27) couples turbulent transport with neoclassical transport. It can affect the long term evolution of the system when collisions are considered [65]. As both collisions and long time scale simulations are currently not feasible in terms of computational cost, this term is neglected in the following.

Terms of order $\mathcal{O}(\epsilon^2)$ are neglected in GENE-3D as they are small in comparison. The largest term of order $\mathcal{O}(\epsilon^2)$ is the parallel nonlinearity which is given in its full electromagnetic form as

$$
\left(-\frac{B_0}{B_{0\parallel}^*} \left(\mathbf{x}_{\bar{\chi}} + \mathbf{v}_{\nabla B} + \mathbf{v}_c \right) \cdot \left(\frac{1}{m_\sigma v_\parallel} (\mu \nabla B_0 + q_\sigma \nabla \bar{\chi}) \right) + \frac{q_\sigma}{m_\sigma} \left(\frac{B_0}{B_{0\parallel}^*} \hat{\mathbf{b}}_0 \cdot \nabla \bar{\chi} - \frac{1}{c} \frac{\partial \bar{A}_\parallel}{\partial t} \right) \right) \frac{\partial F_{1\sigma}}{\partial v_\parallel} \,,
\tag{2.29}
$$

where the gyrokinetic potential χ has been introduced defined as

$$
\bar{\chi} = \bar{\Phi} - \frac{v_\parallel}{c} \bar{A}_\parallel + \frac{\mu}{q_\sigma} \bar{B}_\parallel \,,
\tag{2.30}
$$

together with the modified parallel background magnetic field $B_{0\parallel}^*$ defined as

$$
B_{0\parallel}^* = B_0 + \frac{m_\sigma c}{q_\sigma} v_\parallel \hat{\mathbf{b}}_0 \cdot (\nabla \times \hat{\mathbf{b}}_0) \,,
\tag{2.31}
$$

and the electromagnetic vector potential $\mathbf{A}$. The parallel nonlinearity is implemented in the x-global GENE version as it can be relevant for simulations close to the edge where the gyrokinetic ordering is not entirely valid anymore [63, 64]. Also, for fast particle investigations with non-Maxwellian background distributions as recently investigated in a separate Ph.D. project by Alessandro di Siena [66], the parallel nonlinearity plays a role due to strong gradients in velocity space. This type of investigation could also be interesting in a stellarator context using GENE-3D in the future, especially since the groundwork has already been done for x-global GENE. Core simulations, as investigated

in this thesis, are not affected by the parallel nonlinearity, and it is therefore neglected in the following.

The resulting equation (2.27) appears more complex than the six-dimensional Vlasov model. However, it is computationally much cheaper to simulate. As the fast gyro-motion is averaged out in the gyrokinetic model, the time step limit of simulations is larger, while the resolution requirements are lower compared to kinetic simulations. Both lead to fewer resources being spent while the result stays accurate as long as the gyrokinetic ordering is fulfilled.

2.3   The gyrokinetic Poisson equation

The gyrokinetic Vlasov equation (2.27) is coupled to Maxwell's equations as it contains the electrostatic field potential ϕ_1 which needs to be determined from the Poisson equation.

Assuming a quasi-neutral background and utilizing the anisotropy of the gyrokinetic ordering (Eq. (2.1)) to replace the Laplacian by its perpendicular component the Poisson equation becomes

$$\nabla^2 \phi_1 \approx \nabla_\perp^2 \phi_1 = -4\pi \sum_\sigma q_\sigma n_{1\sigma}(\mathbf{x}) \,. \tag{2.32}$$

Even small displacements of the electron density relative to the ions lead to strong restoring forces causing high frequency oscillations with the plasma frequency $\omega_p = 4\pi e^2 n_0 / m_e$. In low-frequency plasma physics, the assumption of a quasi-neutral plasma is often made. Quasi-neutrality implies the Debye length to be small compared to other perpendicular length scales in the system. The gyrokinetic Poisson equation then loses its Laplace part $\nabla_\perp^2 \phi_1 \approx 0$ and reduces to the quasi-neutrality condition

$$\sum_\sigma q_\sigma n_{1\sigma}(\mathbf{x}) = 0 \,. \tag{2.33}$$

The main difficulty is now to determine the fluctuating density $n_{1\sigma}$ of charged particles as a function of the gyrocenter distribution function. Generally, the perturbed density is defined as

$$n_{1\sigma}(x) = \int d^3 v F_{1\sigma}(\mathbf{x}, \mathbf{v}) \,. \tag{2.34}$$

As the electrostatic charges are located at the particle positions $\mathbf{x}$ rather than at the gyrocenter positions $\mathbf{X}$ the perturbed density in (2.33) has to be evaluated at the particle positions. The perturbed density $n_{1\sigma}$ can be expressed as the zeroth velocity space moment of the distribution function $n_{1\sigma}(\mathbf{x}) = M_{00,\sigma}(\mathbf{x})$.

An arbitrary velocity space moment reads in particle, guiding-center and gyro-center space as

$$
M_{mn,\sigma} = \int d^3v\, v_\parallel^m v_\perp^n F_{1\sigma}^{(particle)}(\mathbf{x}, \mathbf{v})
$$

$$
= \int d^3X\, dv_\parallel\, d\mu\, d\theta\, \delta(\mathbf{X} - (\mathbf{x} - \rho)) F_{1\sigma}^{(guiding-center)}(\mathbf{X}, \mathbf{V}) v_\parallel^m v_\perp^n \frac{B_{0\parallel}^*(\mathbf{x}, \mathbf{v})}{m_\sigma}
$$

$$
= \int d^3X\, dv_\parallel\, d\mu\, d\theta\, \delta(\mathbf{X} - (\mathbf{x} - \rho)) T^*\{F_{1\sigma}(\mathbf{X}, \mathbf{V})\} v_\parallel^m v_\perp^n \frac{B_{0\parallel}^*(\mathbf{x}, \mathbf{v})}{m_\sigma}
$$

$$
= \int d^3X\, dv_\parallel\, d\mu\, d\theta\, \delta(\mathbf{X} - (\mathbf{x} - \rho)) \left(F_{1\sigma}(\mathbf{X}, \mathbf{V}) - (q_\sigma \phi(\mathbf{x}) - q_\sigma \bar{\phi}(\mathbf{X}) - \mu \overline{B}_{1\parallel}(\mathbf{X})) \frac{F_{0\sigma}}{T_{0\sigma}} \right) \cdot
$$
$$
\cdot\, v_\parallel^m v_\perp^n \frac{B_{0\parallel}^*(\mathbf{x}, \mathbf{v})}{m_\sigma}
$$

$$(2.35)$$

where $F_{1\sigma}$ is the distribution function in gyrocenter coordinates as before and T^* is the first order pull-back operator acting between gyro-center and guiding center coordinates described in Ref. [57] defined as

$$
T^*\{F_{1\sigma}\} = F_{1\sigma} - \left(q_\sigma \phi(\mathbf{x}) - q_\sigma \bar{\phi} - \mu \overline{B}_\parallel \right) \frac{F_{0\sigma}}{T_{0\sigma}} \,. \tag{2.36}
$$

The fluctuating density $n_{1\sigma}$ can then be calculated by neglecting magnetic field perturbations $\overline{B}_{1\parallel}(\mathbf{X}) \approx 0$ and using the gyrokinetic ordering to simplify the expression for the modified magnetic field which has implicitly already been used in Eq. (2.15),

$$
B_{0\parallel}^* = B_0 + \frac{m_\sigma c}{q_\sigma} v_\parallel \hat{\mathbf{b}}_0 \cdot (\nabla \times \hat{\mathbf{b}}_0) \approx B_0 \,. \tag{2.37}
$$

This results in,

$$
n_{1\sigma}(\mathbf{x}) = M_{00,\sigma}(\mathbf{x})
$$

$$
= \frac{B_0(\mathbf{x}, \mathbf{v})}{m_\sigma} \int d^3X\, dv_\parallel\, d\mu\, d\theta\, \delta(\mathbf{X} - (\mathbf{x} - \rho)) \left(F_{1\sigma}(\mathbf{X}, \mathbf{V}) - (q_\sigma \phi(\mathbf{x}) - q_\sigma \bar{\phi}(\mathbf{X})) \frac{F_{0\sigma}}{T_{0\sigma}} \right)
$$

$$
= \frac{2\pi B_0}{m_\sigma} \int \left[\langle F_{1\sigma} \rangle(\mathbf{x}) - \frac{q_\sigma F_{0\sigma}}{T_{0\sigma}} \left(\phi_1(\mathbf{x}) - \langle \bar{\phi}_1 \rangle(\mathbf{x}) \right) \right] dv_\parallel\, d\mu \,.
$$

$$(2.38)$$

Integrating the δ function over space results in the average over the distribution function at all gyrocenter positions $\mathbf{X}$ which contain $\mathbf{x}$ in their gyrotrajectory

$$\mathcal{K}(F_{1\sigma}) = \langle F_{1\sigma} \rangle(\mathbf{x}) = \frac{1}{2\pi} \oint F_{1\sigma}(\mathbf{x} - \mathbf{r}(\alpha)) \, d\alpha \,, \tag{2.39}$$

and the double gyroaverage operator which performs a gyroaverage of the field value at all gyrocenter positions $\mathbf{X}$ which have $\mathbf{x}$ in their gyrotrajectory

$$\langle \bar{\phi}_1 \rangle(\mathbf{x}) = \frac{1}{(2\pi)^2} \oint d\alpha \int d\mathbf{X} \, \delta(\mathbf{X} + \mathbf{r}(\alpha) - \mathbf{x}) \oint d\alpha' \, \phi_1(\mathbf{X} + \mathbf{r}(\alpha'))$$
$$= \frac{1}{(2\pi)^2} \oint d\alpha \oint d\alpha' \, \phi_1(\mathbf{x} - \mathbf{r}(\alpha) + \mathbf{r}(\alpha')) \,. \tag{2.40}$$

The field equation   which is the basis for the GENE-3D implementation can then be written as

$$\sum_\sigma \frac{q_\sigma^2}{m_\sigma} \int \frac{F_{0\sigma}}{T_{0\sigma}} \left(\phi_1(\mathbf{x}) - \langle \bar{\phi}_1 \rangle(\mathbf{x}) \right) dv_\parallel d\mu = \sum_\sigma \frac{q_\sigma}{m_\sigma} \int \langle F_{1\sigma} \rangle(\mathbf{x}) dv_\parallel d\mu \,, \tag{2.41}$$

where the right-hand side is simply the gyrocenter charge density. The left hand side is the polarization density which arises when switching from guiding center coordinates to gyrocenter coordinates.

Adiabatic electrons

To save computation time it helps to consider the electrons in the simulation to be massless compared to the ions $m_e/m_i \to 0$. This implies that the thermal velocity of the electrons is much larger than the thermal velocity of the ions $v_{Te}/v_{Ti} \propto \sqrt{m_i/m_e} \to \infty$. As a consequence, all parallel electric field fluctuations are almost instantaneously balanced by the electrons. The "infinite" parallel electron heat conductivity prevents thermal fluctuations within a flux surface by short-circuiting it. This assumption simplifies the expression for the fluctuating electron density to

$$n_{1e} = n_{0e} \frac{e}{T_{0e}} (\phi_1 - \langle \phi_1 \rangle_{\mathrm{FS}}) \,, \tag{2.42}$$

with the flux surface average $\langle \cdot \rangle_{\mathrm{FS}}$ [67]

$$\langle \, \cdot \, \rangle_{\mathrm{FS}} = \frac{\partial}{\partial V} \int_V \cdot \, dV' \,, \tag{2.43}$$

and the volume enclosed by that flux surface V.

2.4 Non-axisymmetric toroidal magnetic equilibria

The previous subsections introduced the gyrokinetic Vlasov-Maxwell system of equations, which is adapted to complex three-dimensional MHD equilibria in this subsection. GENE-3D uses a coordinate system utilizing the anisotropic nature of the problem. It takes advantage of the splitting into the parallel and perpendicular components of the magnetic field in Eq. (2.27).

These so-called field-aligned coordinate systems have been shown to cut down computational cost for gyrokinetic core turbulence simulations for more than two decades [68]. The number of grid points necessary for a converged resolution can be reduced by multiple orders of magnitude, depending on the problem at hand compared to non-aligned grids. With the choice of a field aligned-coordinate system, transformation metrics need to be included, e.g., in the gradients of Eq. (2.27).

Straight-field-line coordinates

The mapping of the computational domain of GENE-3D is based on the geometry of a three-dimensional ideal MHD equilibrium with closed nested flux surfaces. Two coordinate systems are used to describe the magnetic field geometry. A right-handed cylindrical coordinate system with coordinates (R, Z, ϕ) and a right-handed straight-field-line coordinate system with coordinates (ρ, θ, ζ). Both coordinate systems together with an example geometry based on a stellarator shape generated by a rotated ellipse are shown in Fig. 2.1. Here, R and Z are the radial distance from the axis and the axial coordinate respectively and determine the position within a poloidal plane. ϕ is the geometrical toroidal angle, ρ is a convenient flux surface label while θ and ζ denote the straight-field-line poloidal and toroidal angles. They can be either PEST [69] or Boozer [70] angles.

Fig. 2.1 shows that $\nabla\zeta$ points in the same direction as $\nabla\phi$. For the PEST coordinate system ζ and ϕ coincide $\zeta = \phi$. The contravariant representation of the equilibrium magnetic field is then

$$\mathbf{B}_0 = \frac{d\Psi_{\text{tor}}/d\rho}{2\pi}(\nabla\rho \times \nabla\theta) - \frac{d\Psi_{\text{pol}}/d\rho}{2\pi}(\nabla\rho \times \nabla\zeta)\,, \qquad (2.44)$$

with the ρ-derivatives of the toroidal and poloidal flux functions [67]

$$\Psi_{\text{tor}} = \iint\limits_{S_{\text{tor}}} \mathbf{B}_0 \cdot d\mathbf{S} = \frac{1}{2\pi}\iiint\limits_{V} \mathbf{B} \cdot \nabla\zeta d^3R\,, \qquad (2.45)$$

$$\Psi_{\text{pol}} = \iint\limits_{S_{\text{pol}}} \mathbf{B}_0 \cdot d\mathbf{S} = \frac{1}{2\pi}\iiint\limits_{V} \mathbf{B} \cdot \nabla\theta d^3R \qquad (2.46)$$
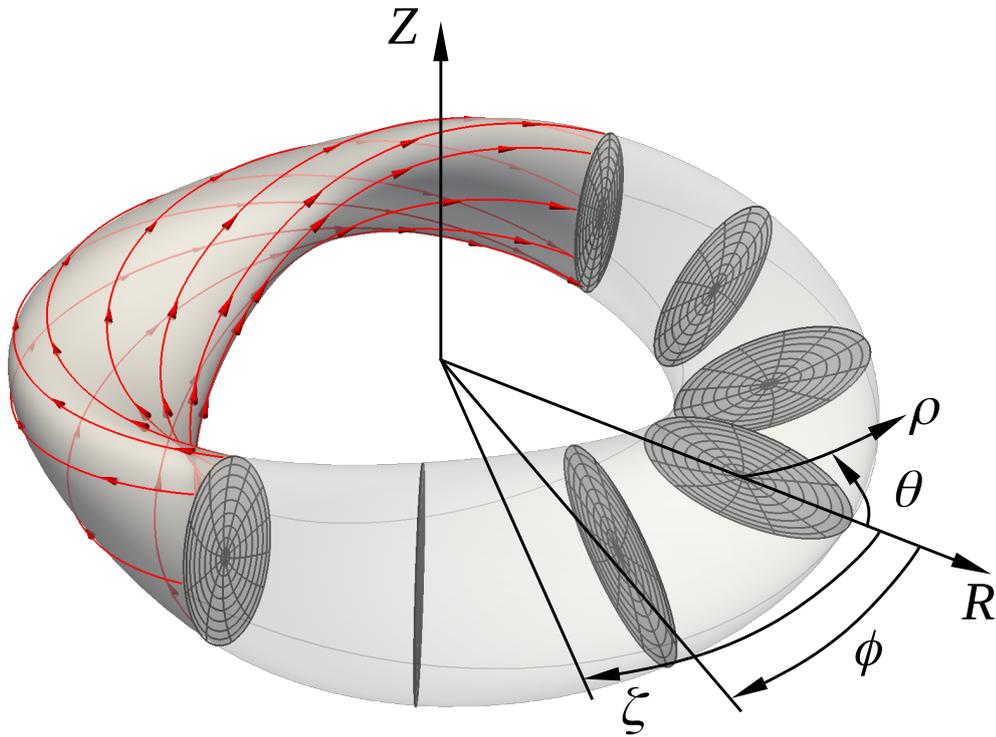
Figure 2.1: Right-handed cylinder coordinates (R, Z, ϕ) and right-handed straight field line coordinates (ρ, θ, ζ) for the definition of the geometry of the magnetic field of a MHD equilibrium. Shown for a simple stellarator shape and PEST coordinates, where poloidal planes are $\zeta = \phi =$ const., with lines of ρ=const. and θ=const.. Magnetic field lines are plotted in red on the last flux surface $(\rho = 1, q = -1.11, \Phi_{\text{edge}} = 1.5)$ Cf. Ref. [31].

where S_{tor} is a poloidal cut of the flux surface with $\zeta = const$ and S_{pol} is a $\theta = const$ toroidal cut of the flux surface. Both Ψ_{tor} and Ψ_{pol} are only dependent on the flux surface label ρ as both are flux functions which by definition are uniform over a magnetic surface and therefore independent of θ and ζ. For positive Ψ'_{tor} the toroidal magnetic field points in $\nabla\zeta$ and for positive Ψ'_{pol} the poloidal magnetic field points in $\nabla\theta$. The safety factor q is defined by the derivative flux functions as

$$q(\rho) = \frac{\Psi'_{tor}(\rho)}{\Psi'_{\text{pol}}(\rho)}\,, \tag{2.47}$$

while the sign of the poloidal magnetic flux at the edge σ_{B_p} is defined as

$$\sigma_{B_p} = \text{sgn}(\Psi'_{\text{pol}}(\text{edge})) = \text{sgn}(\Psi'_{\text{tor}}(\text{edge}))\,\text{sgn}(q(\text{edge}))\,. \tag{2.48}$$

In Fig. 2.1 the magnetic field has $\text{sgn}(\Psi'_{\text{tor}}) = 1$ and $\text{sgn}(\Psi'_{\text{pol}}) = -1$ and therefore $\sigma_{B_p} = -1$.

Field-aligned coordinates

GENE-3D uses a field-aligned coordinate system (x, y, z) derived from this straight-field-line coordinate system. The field-aligned coordinates used by GENE-3D are defined as

$$x = \rho_{\text{tor}}\,, \tag{2.49}$$

$$y = \sigma_{B_p} C_y (q\theta - \zeta)\,, \tag{2.50}$$

$$z = \sigma_{B_p}\theta\,. \tag{2.51}$$

In order to have a radial (flux-surface label) coordinate (x) ascending from the magnetic axis towards the edge the flux surface label ρ is chosen to be

$$\rho_{\text{tor}} = \frac{\sqrt{\Psi_{\text{tor}}}}{\Psi_{\text{tor}}(\text{edge})}\,, \tag{2.52}$$

where $\Psi_{\text{tor}}(\text{edge})$ denotes the toroidal flux at the last closed flux surface. Hence, ∇x always points outward from the flux surface. The (y)-coordinate selects a field line on a given flux surface. The $C_y = x_0/|q_0|$ normalization factor converts the angular coordinate $\alpha = \sigma_{B_p}(q\theta - \zeta)$ to a length. The subscript 0 denotes that these radial dependent quantities are taken at a reference position x_0, which is usually in the middle of the radial simulation box.

For determining wave numbers k_y, it is useful to have the y coordinate as a length. In order to compare different equilibria, it is more useful to use the field line angle α as the minimum, and the maximum values of different equilibria of the same machine are equivalent. The (z) coordinate denotes the position along the field line selected by

the (y) coordinate. The covariant basis vector e_z is always parallel to the equilibrium magnetic field B_0. Hence, (z) is often called the parallel coordinate. The y coordinate is periodic while the z coordinate is quasi periodic as a field line usually does not close onto itself after one poloidal turn as depicted in Fig. 2.2. The boundary conditions
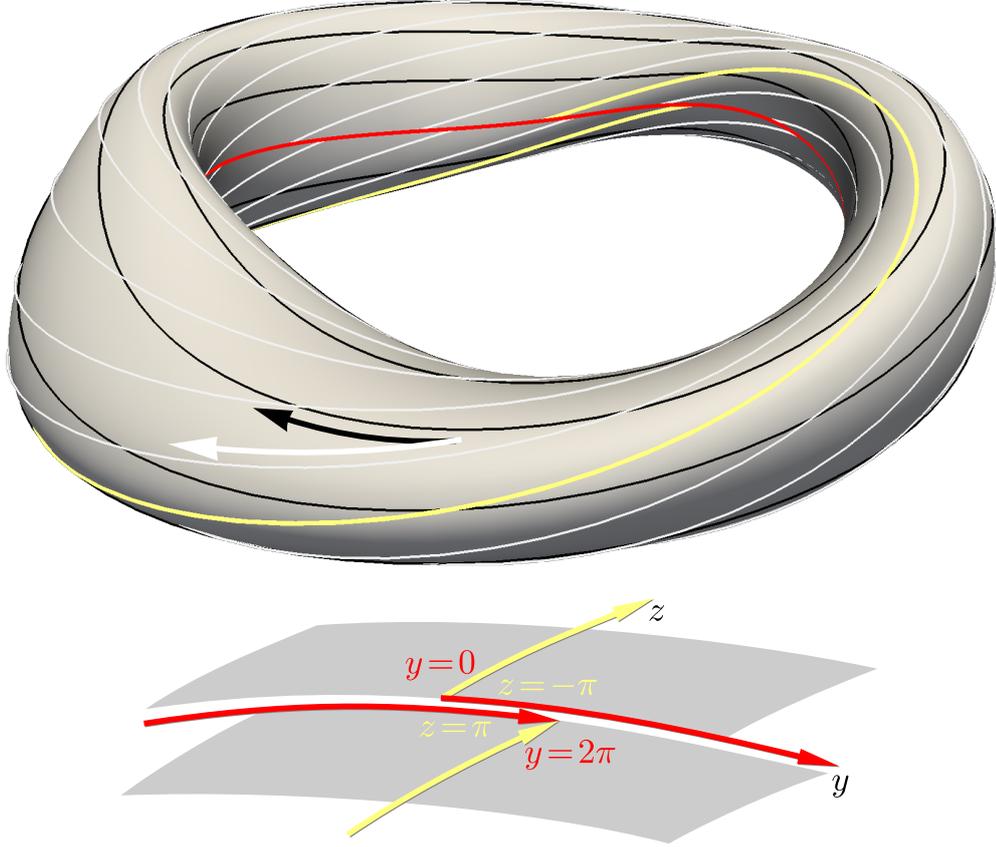
Figure 2.2: "Representation of the (y,z) coordinates on a flux surface of a simple stellarator geometry. Top: The white and black arrows point, respectively, in the z and y direction. White and black lines indicate, respectively, locations of constant y and z. The solid yellow line ($y = const = 0, 2\pi C_y$) and red line ($z = const = -\pi, \pi$) depict the boundary of the domain. Bottom: Depiction of the shifted boundary" Cf. [31].

implemented in GENE-3D are described in Sec. 3.2.3.

The equilibrium magnetic field $\mathbf{B_0}$ (Eq. (2.44)) can be expressed utilizing the above definitions as

$$\mathbf{B}_0 = \frac{\Psi'_{\text{pol}}}{2\pi} \nabla \rho_{\text{tor}} \times \nabla \left[q(\rho_{\text{tor}})\theta - \phi \right] = \mathcal{C}(x)\nabla x \times \nabla y \,, \tag{2.53}$$

with the normalized contravariant basis vectors

$$\nabla x = a\nabla\rho_{\text{tor}} \,, \tag{2.54}$$

$$\nabla y = a\sigma_{B_p}C_y(q'(x)\theta\nabla\rho_{\text{tor}} + q(x)\nabla\theta - \nabla\phi) \,, \tag{2.55}$$

$$\nabla z = a\sigma_{B_p}\nabla\theta, \tag{2.56}$$

where a is the average minor radius of the device and defining

$$\mathcal{C}(x) = \frac{x}{|q(x)|}\frac{B_{\text{axis}}}{C_y} \,. \tag{2.57}$$

From Eq. (2.53) follows that $\mathbf{B} \cdot \nabla x = \mathbf{B} \cdot \nabla y = 0$, so that $x = \text{const}$ and $y = \text{const}$ define a magnetic field line and the GENE-3D coordinates (x, y, z) are a valid field-aligned coordinate system.

The metric coefficients g^{ij} and the Jacobian J,   required for the expression of the differential operators appearing in the gyrokinetic equations, are defined as

$$g^{ij} = \nabla u^i \cdot \nabla u^j \,, \tag{2.58}$$

$$J^{-1} = (\nabla x \times \nabla y) \cdot \nabla z = \frac{\mathbf{B_0} \cdot \nabla z}{\mathcal{C}(x)} \,, \tag{2.59}$$

with i and j taking on the values 1, 2, and 3, and $(u^1, u^2, u^3) = (x, y, z)$. To simplify notation, the following combinations of metric coefficients are defined

$$\gamma_1 = g^{xx}g^{yy} - g^{xy}g^{yx} \,, \tag{2.60}$$

$$\gamma_2 = g^{xx}g^{yz} - g^{yx}g^{xz} \,, \tag{2.61}$$

$$\gamma_3 = g^{xy}g^{yz} - g^{yy}g^{xz} \,. \tag{2.62}$$

MHD equilibria as input for GENE-3D

The geometrical terms appearing in Eq. (2.27) strongly depend on the given MHD equilibrium. In order to benchmark against other versions of GENE, the analytical axisymmetric equilibrium with circular concentric flux surfaces, as described in Ref. [30], can be used. The benchmark results are described in Sec. 4.

In order to simulate realistic magnetic field geometries of stellarators like Wendelstein 7-X and non-axisymmetric tokamaks with resonant magnetic perturbations, the ideal MHD equilibrium has to be computed beforehand, which is done with the Galerkin Variational Equilibrium Code (GVEC [55]) described in section 3.3.1. An interface to GVEC is implemented in GENE-3D, and GVEC is linked to GENE-3D as a library to compute the geometric coefficients and equilibrium quantities. Examples of global

quantities computed by GVEC are a and $\Psi_{\text{tor}}(\text{edge})$. Also, the safety factor and pressure profiles $q(x), p(x)$ are given at all radial positions. Additionally, the gradients (2.54)-(2.56) needed for the metric coefficients and the gradient of the absolute magnetic field $\frac{\partial B_0}{\partial y}, \frac{\partial B_0}{\partial x}, \frac{\partial B_0}{\partial z}$, needed in 4 are provided.

Summary

In this section, the ideas of gyrokinetic theory and the derivation of the gyrokinetic equations have been outlined. Furthermore, the adaptation to complex 3D geometries has been described. This has laid the foundation for understanding the next sections.

3 The development of the global stellarator version of GENE

This section describes the development stages of the GENE-3D code. To this end, first, the state of GENE-3D before this Ph.D. project is outlined in Sec. 3.1. Here, the setup of the gyromatrices and field matrices are described as well as discretization schemes similar to Ref. [31] which was written at the same time as this thesis.

Then, the significant developmental steps during this Ph.D. project, which were necessary to turn GENE-3D into production code, are described in Sec. 3.2. Here, the transition from a Fourier space representation to a real space description in the y-coordinate (see Eq. (2.49)) is described in Sec. 3.2.1 which is necessary to take 3D geometries into account. Then the parallelization schemes, boundary conditions, and boundary exchanges are outlined in Sec. 3.2.3. In this context, also the interpolation schemes necessary for the 3D visualization of simulation results are introduced in Sec. 3.2.4. Then, the PETSc linear equation solver used to solve (2.41) is explained, and typical matrices are visualized in Sec. 3.2.5. Lastly, the code structure and performance aspects, including parallelization performance and memory usage, are discussed in subsections 3.2.6 and 3.2.7.

Sec. 3.3 then explains in detail how the production chain around GENE-3D operates. The geometric preprocessing by GVEC providing GENE with the geometric information described in sec. 2.4 is described in sec. 3.3.1. The postprocessing tools based on Python, which serve to evaluate among other observables growth rates, frequencies, and mode numbers are described in sec. 3.3.2. Finally, some examples of the Paraview 3D visualization are shown in sec. 3.3.3.

3.1 State of the GENE code prior to this project

The GENE code started in the year 2000 as a flux tube code [21] where the x and y direction are represented in Fourier space while the z coordinate is treated in real space (see sec. 2.4). In the flux tube version, no geometrical or profile variations could be taken into account in the radial or poloidal direction. Only a small volume around a magnetic field line could be simulated where profiles and geometries are assumed to be constant. GENE was later extended to the radially global (x-global or global tokamak) version [30], where also the x coordinate is treated in real space allowing for dependencies of the geometry and profiles in the radial direction. The x-global GENE version was then adjusted so that it could also be used to simulate single flux surfaces by internally interchanging the indices of x and y. By interchanging the indices of x and y, this version could later also be used to simulate a single flux surface with all geometric variations but without a radial dependence. This code is internally called the y-global or single flux

surface version of GENE. However, to study multiple flux-surfaces, a new code needed to be developed, which is GENE-3D. GENE-3D has a real space representation in all three spatial dimensions and can hence take geometric dependencies in all three dimensions into account. This allows capturing all geometric effects in simulations of stellarators like Wendelstein 7-X or tokamaks with magnetic field ripples. The capabilities of the different versions of GENE are summarized in Table 1 and visualized in Fig. 3.1.

GENE-3D can, among other things, help to better design future devices and optimize them for turbulent transport. A first step in this direction is done in Sec. 5. A non-functioning base version of GENE-3D was developed before this Ph.D. project. Algorithms and implementations of the base version retained in the current version of GENE-3D are outlined in this subsection.

Modes of operation of the gyrokinetic GENE code			
	radially global	**poloidally global**	**year**
Flux-tube GENE	O	O	2000 [21]
x-global GENE	X	O	2010 [30]
y-global GENE	O	X	2014 [34]
GENE-3D	X	X	2019 [31]

Table 1: The geometric capabilities of the different versions of GENE.
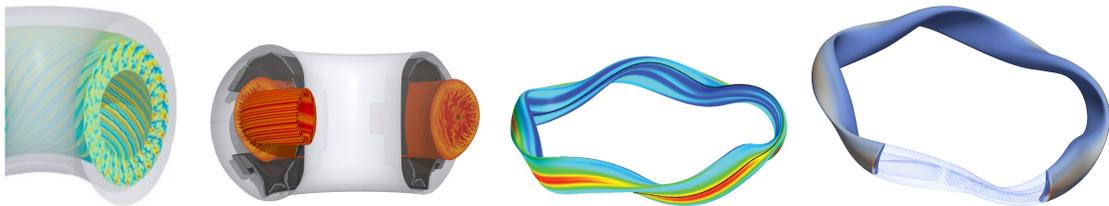
Figure 3.1: Visualization of the different versions of GENE. From flux-tube (left) to x-global Cf. [32] to y-global Cf. [34] to GENE-3D (right).

3.1.1   Normalization

For numerical simulations the physical quantities are transformed to normalized (dimensionless) quantities. Here, they are denoted with a hat. To be able to convert the dimensionless quantities, e.g., to SI units after a simulation reference quantities are kept. Here, they are denoted by the index 'ref' such that e.g. the temperature is expressed as $T = \hat{T} \cdot T_{\text{ref}}$. The basic reference values are described in Tab. 2. The normalizations

of gyrocenter coordinates, time, mass, equilibrium and fluctuating quantities as well as velocity space and geometric quantities can be found in Table 3 using the reference quantities of Table 2. The parallel coordinate z is an angle-like coordinate and, therefore, dimensionless while the x and y coordinates are normalized to a reference gyroradius. Also the velocity space quantities are normalized to radially and species dependent quantities evaluated at a reference position x_0 usually at the center of the simulation box. With the definitions

Table 2: Basic and combined reference values of GENE-3D.

Reference variable	Meaning
e	positive elementary charge
m_{ref}	reference mass
n_{ref}	reference density
T_{ref}	reference temperature
L_{ref}	macroscopic reference length
$B_{\mathrm{ref}} = \left\| \Psi_{\mathrm{tor}}(\mathrm{edge})/(\pi \mathrm{L}_{\mathrm{ref}}^2) \right\|$	reference magnetic field
$c_{\mathrm{ref}} = \sqrt{T_{\mathrm{ref}}/m_{\mathrm{ref}}}$	reference velocity
$\Omega_{\mathrm{ref}} = eB_{\mathrm{ref}}/(m_{\mathrm{ref}}c)$	reference gyrofrequency
$\rho_{\mathrm{ref}} = c_{\mathrm{ref}}/\Omega_{\mathrm{ref}}$	reference gyroradius

Table 3: Normalization of GENE-3D variables.

Basic quantities	Equilibrium quantities	Velocity & Fluctuating quantities	Geometry
$x = \rho_{\mathrm{ref}}\,\hat{x}$	$B_0 = B_{\mathrm{ref}}\,\hat{B}_0$	$v_\parallel = c_{\mathrm{ref}}\hat{v}_\parallel \hat{v}_{\mathrm{th},\sigma}(x_0)$	$J = L_{\mathrm{ref}}\hat{J}$
$y = \rho_{\mathrm{ref}}\,\hat{y}$	$n_{0\sigma} = n_{\mathrm{ref}}\,\hat{n}_{0\sigma}$	$\mu = \frac{T_{\mathrm{ref}}}{B_{\mathrm{ref}}}\hat{\mu}\hat{T}_\sigma(x_0)$	$\mathcal{C} = B_{\mathrm{ref}}\hat{\mathcal{C}}$
$z = \hat{z}$	$T_{0\sigma} = T_{\mathrm{ref}}\,\hat{T}_{0\sigma}$	$\hat{v}_{\mathrm{th},\sigma}(x_0) = \sqrt{2\hat{T}_\sigma(x_0)/\hat{m}_\sigma}$	$\gamma_1 = \hat{\gamma}_1$
$t = \frac{L_{\mathrm{ref}}}{c_{\mathrm{ref}}}\,\hat{t}$	$\phi_0 = \frac{T_{\mathrm{ref}}}{e}\,\hat{\phi}_0$	$\phi_1 = \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{T_{\mathrm{ref}}}{e}\,\hat{\phi}_1$	$\gamma_2 = \frac{1}{L_{\mathrm{ref}}}\hat{\gamma}_2$
$m_\sigma = m_{\mathrm{ref}}\,\hat{m}_\sigma$	$F_{0\sigma} = \frac{n_{\mathrm{ref}}}{c_{\mathrm{ref}}^3}\frac{\hat{n}_{0\sigma}(x_0)}{\hat{v}_{\mathrm{th}\sigma}^3(x_0)}\,\hat{F}_{0\sigma}$	$F_{1\sigma} = \frac{\rho_{\mathrm{ref}}}{L_{\mathrm{ref}}}\frac{n_{\mathrm{ref}}}{c_{\mathrm{ref}}^3}\frac{\hat{n}_{0\sigma}(x_0)}{\hat{v}_{\mathrm{th}\sigma}^3(x_0)}\,\hat{F}_{1\sigma}$	$\gamma_3 = \frac{1}{L_{\mathrm{ref}}}\hat{\gamma}_3$

$$\hat{n}_p(x) = n_{0\sigma}(x)/n_{0\sigma}(x_0)\,, \tag{3.1}$$

$$\hat{T}_p(x) = T_{0\sigma}(x)/T_{0\sigma}(x_0)\,, \tag{3.2}$$

used to simplify notation for the density and temperature profiles the dimensionless background Maxwellian (Eq. (2.23)) of species ρ can be written as

$$\hat{F}_{0\sigma}(x, y, z, v_\parallel, \mu) = \frac{\hat{n}_{p\sigma}(x)}{[\pi \hat{T}_{p\sigma}(x)]^{3/2}} \, \exp\left(\frac{-\hat{v}_\parallel^2 - \hat{\mu}\hat{B}_0(x, y, z)}{\hat{T}_{p\sigma}(x)}\right). \tag{3.3}$$

The dimensionless form of the Vlasov equation   (2.27) is given by

$$
\begin{aligned}
\frac{\partial \hat{F}_{1\sigma}}{\partial \hat{t}} =& \mathcal{V}_{\Gamma,z}\hat{\Gamma}_{\sigma,z} + \mathcal{V}_{F_1,v_\parallel}\frac{\partial \hat{F}_{1\sigma}}{\partial \hat{v}_\parallel} + \mathcal{V}_{\Gamma,x}\hat{\Gamma}_{\sigma,x} + \mathcal{V}_{\Gamma,y}\hat{\Gamma}_{\sigma,y} - \frac{1}{\hat{\mathcal{C}}}\left(\partial_{\hat{x}}\bar{\hat{\phi}}_1\hat{F}_{1\sigma,y} - \partial_{\hat{y}}\bar{\hat{\phi}}_1\hat{F}_{1\sigma,x}\right) \\
& - \frac{1}{\hat{\mathcal{C}}}\partial_{\hat{x}}\bar{\hat{\phi}}_0\hat{F}_{1\sigma,y} + \mathcal{V}_{\phi,y}\partial_{\hat{y}}\bar{\hat{\phi}}_1\,,
\end{aligned}
\tag{3.4}
$$

with the abbreviation

$$\hat{\Gamma}_{\sigma,i} = \partial_{\hat{i}}\hat{F}_{1\sigma} + \frac{\hat{q}_\sigma}{\hat{T}_{0\sigma}}\hat{F}_{0\sigma}\partial_{\hat{i}}\bar{\hat{\phi}}_1\,. \tag{3.5}$$

The $\mathcal{V}$ prefactors are used to simplify notation and are summarized and explained in Tab. 4.

Lastly, the dimensionless gyrokinetic quasi neutrality condition implemented in GENE-3D reads

$$\sum_\sigma \hat{q}_\sigma^2 \left(\frac{\hat{n}_{0\sigma}(x)}{\hat{T}_{0\sigma}(x)} - \int d\hat{\mu}'\mathcal{K}\frac{\hat{n}_{0\sigma}(x)}{\hat{T}_{0\sigma}(x)}e^{-\hat{\mu}'}\mathcal{G}\right)\hat{\phi}_1 = \sum_\sigma \hat{n}_{0\sigma}(x_0)\hat{q}_\sigma\pi \int d\hat{v}_\parallel d\hat{\mu}\mathcal{K}\hat{B}_0(x, y, z)\hat{F}_{1\sigma}\,, \tag{3.6}$$

with the definition

$$\hat{\mu}' = \hat{\mu}\hat{B}_0(x, y, z)/\hat{T}_{p\sigma}(x)\,. \tag{3.7}$$

The gyromatrix $\mathcal{G}$ is the the discretized version of the gyroaverage operator defined in Eq. (2.10). Its explicit form is derived in the following subsection. The matrix $\mathcal{K}$ is the discretized version of the operator defined in Eq. (2.39), which is approximated by the adjoint of the gyromatrix $\mathcal{G}$ as derived in Ref. [71].

Table 4: Explanation of prefactors occurring in Eq. (3.4).

Prefactor	meaning
$\mathcal{V}_{\Gamma,z} = -\hat{v}_{T\sigma}(x_0)\frac{\hat{C}}{\hat{J}\hat{B}_0}\hat{v}_{\parallel}$	Parallel derivative prefactor
$\mathcal{V}_{F_1,v_{\parallel}} = \frac{\hat{v}_{T\sigma}(x_0)}{2}\frac{\hat{C}}{\hat{J}\hat{B}_0}\hat{\mu}\partial_{\hat{z}}\hat{B}_0$	Trapping term prefactor
$\mathcal{V}_{\Gamma,x} = -\frac{\hat{T}_{0\sigma}(x_0)}{\hat{q}_{\sigma}}\frac{\hat{\mu}\hat{B}_0+2\hat{v}_{\parallel}^2}{\hat{B}_0}\hat{K}_x$	Combined curvature and gradient-B prefactors
$\mathcal{V}_{\Gamma,y} = -\frac{\hat{T}_{0\sigma}(x_0)}{\hat{q}_{\sigma}\hat{B}_0}(\hat{\mu}\hat{B}_0 + 2\hat{v}_{\parallel}^2)\hat{K}_y$	"
$\hat{K}_x = -\frac{1}{\hat{C}}\left(\frac{\partial\hat{B}_0}{\partial\hat{y}} + \frac{\hat{\gamma}_2}{\hat{\gamma}_1}\frac{\partial\hat{B}_0}{\partial\hat{z}}\right)$	Gradient of equilibrium magnetic field
$\hat{K}_y = \frac{1}{\hat{C}}\left(\frac{\partial\hat{B}_0}{\partial\hat{x}} - \frac{\hat{\gamma}_3}{\hat{\gamma}_1}\frac{\partial\hat{B}_0}{\partial\hat{z}}\right)$	"
$\mathcal{V}_{\phi,y} = -\frac{1}{\hat{C}}\left[\omega_{n\sigma} + \omega_{T\sigma}\left(\frac{\hat{v}_{\parallel}^2+\hat{\mu}\hat{B}_0}{\hat{T}_{p\sigma}} - \frac{3}{2}\right)\right]\hat{F}_{0\sigma}$	Linear drive term
$\omega_{T\sigma} = -L_{\text{ref}}\partial_x \ln T_{0\sigma}(x)$	Logarithmic temperature gradient
$\omega_{n\sigma} = -L_{\text{ref}}\partial_x \ln n_{0\sigma}(x)$	Logarithmic density gradient

3.1.2   Setup of the gyromatrix

Several terms in the gyrokinetic equations contain gyroaveraged quantities requiring an explicit representation of the gyroaverage operator $\mathcal{G}[f(x)]$, which is derived in the following based on Ref. [72]. Consider a function $f(x, y, z)$ defined in real space. The points to sample the gyro-ring at $[x(\mathbf{X}+\mathbf{r}), y(\mathbf{X}+\mathbf{r}), z(\mathbf{X}+\mathbf{r})] \approx [x(\mathbf{X}+\mathbf{r}), y(\mathbf{X}+\mathbf{r}), z(\mathbf{X})]$ will generally lie between grid points in x and y direction.

A finite element approach   is therefore used for interpolation. Here, the gyrokinetic ordering is used to neglect z-variations of fluctuating quantities. Hence, f is developed into basis functions in both x and y direction

$$f(x, y, z) = \sum_i^{n_x} \sum_j^{n_y} f_{ij}(z)\, \Lambda_{ij}(x, y)\,, \tag{3.8}$$

with the basis functions $\Lambda_{ij}(x, y)$, the finite element coefficients $f_{ij}(z) = f(x_i, y_j, z)$ and the number of grid points in x and y direction n_x and n_y. Inserting this into the definition of the gyroaverage (Eq. (2.10)) yields

$$\begin{aligned}
\bar{f}(\mathbf{X}) &= \frac{1}{2\pi} \int_0^{2\pi} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} f_{ij}\Lambda_{ij}(x(\mathbf{X}+\mathbf{r}(\alpha)), y(\mathbf{X}+\mathbf{r}(\alpha)))\, d\alpha\,. \\
&= \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} f_{ij} \frac{1}{2\pi} \int_0^{2\pi} \Lambda_{ij}(x(\mathbf{X}+\mathbf{r}(\alpha)), y(\mathbf{X}+\mathbf{r}(\alpha)))\, d\alpha\,.
\end{aligned} \tag{3.9}$$

To map the values at positions around the gyroring $(x(\mathbf{X}+\mathbf{r}(\alpha)), y(\mathbf{X}+\mathbf{r}(\alpha))$ to the field aligned coordinate system the metric coefficients defined in Sec. 2.4 are used. Linearization of the metric around the gyrocenter position $\mathbf{X}$ results in

$$x = x(\mathbf{X}+\mathbf{r}(\alpha)) \approx x(\mathbf{X}) + \mathbf{r} \cdot \nabla x = x(\mathbf{X}) + \rho\,\sqrt{g^{xx}} \cos\alpha\,, \tag{3.10}$$

$$y = y(\mathbf{X}+\mathbf{r}(\alpha)) \approx y(\mathbf{X}) + \mathbf{r} \cdot \nabla y = y(\mathbf{X}) + \rho\,(g^{xy}\cos\alpha + \sqrt{\gamma_1}\sin\alpha)/\sqrt{g^{xx}}\,. \tag{3.11}$$

To convert Eq. (3.9) into a matrix-vector multiplication an index mapping from the two-dimensional i, j space to a one-dimensional k space is applied for the indices of the basis functions and analogously for the index for the gyrocenter position q. The mapping reads

$$k = j \cdot M + i \quad with \quad j \in \{1, N\}, i \in \{1, M\}\,, \tag{3.12}$$

$$q = n \cdot M + m \quad with \quad n \in \{1, N\}, m \in \{1, M\}\,. \tag{3.13}$$

Equation (3.9) then transforms to

$$
\bar{f}_q = \bar{f}(\mathbf{X}_q) = \sum_{k=1}^{n_x \cdot n_y} f_k \frac{1}{2\pi} \int_0^{2\pi} \Lambda_k(x,y)) \, d\alpha = \sum_{k=1}^{n_x \cdot n_y} \mathcal{G}_{qk} f_k \,, \tag{3.14}
$$

with the matrix elements

$$
\mathcal{G}_{qk} = \frac{1}{2\pi} \oint \Lambda_k(x,y) \, d\alpha \,. \tag{3.15}
$$

This means that the gyroaverage at a given gyrocenter index $q(m,n)$ is calculated by summing the integrals of the basis functions Λ at gyrocenter positions with index $k(i,j)$ over the gyroring around $\mathbf{X}_q$. At the moment, the integral is solved by sampling a large number of points uniformly around the gyroring. Therefore, interpolation between grid points is required. The two-dimensional basis functions $\Lambda_{ij}(x,y)$ in Eq. (3.8) must be chosen such that the following conditions are fulfilled on the patch $[x_n, x_{n+1}] \times [y_m, y_{m+1}]$

$$
f(x_n, y_m) = f_{nm} \qquad \frac{\partial f(x_n, y_m)}{\partial x} = \partial_x f_{nm}
$$

$$
\frac{\partial f(x_n, y_m)}{\partial y} = \partial_y f_{nm} \qquad \frac{\partial^2 f(x_n, y_m)}{\partial x \partial y} = \partial_{xy} f_{nm} \,, \tag{3.16}
$$

to transition accurately between coefficient space and real space. Together with the condition

$$
\Lambda_{ij}(x_n, y_m) = \delta_{in} \delta_{jm} \,, \tag{3.17}
$$

this means that the basis function of a grid point converges to zero before reaching the neighboring grid points. Also, the function values $f(x,y,z)$ and their derivatives at the grid points should be the same in the basis function representation.

Bicubic basis functions   are employed for an accurate interpolation of both the function values as well as the derivatives around the grid points

$$
f(x,y,z) = \sum_i^{n_x} \sum_j^{n_y} \left(G_{ij}(x,y) + H_{ij}(x,y)\partial_x + Q_{ij}(x,y)\partial_y + P_{ij}(x,y)\partial_{xy} \right) f_{ij}(z) \,, \tag{3.18}
$$

with the bicubic ansatz for $G(x,y)$

$$
G(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} \alpha_{ij} x^i y^j \,, \tag{3.19}
$$

and analogous for $H(x,y)$, $Q(x,y)$, $P(x,y)$. With conditions (3.16) and (3.17), Eq. (3.18) can be solved analytically. The solutions are

$$
\begin{aligned}
G_{ij}(x,y) &= (1 - \sigma_x \delta x)^2 \left(1 + 2\sigma_x \delta x\right) \left(1 - \sigma_y \delta y\right)^2 \left(1 + 2\sigma_y \delta y\right) , \\
H_{ij}(x,y) &= \Delta x \left(1 - \sigma_x \delta x\right)^2 \left(1 - \sigma_y \delta y\right)^2 \left(1 + 2\sigma_y \delta y\right) , \\
Q_{ij}(x,y) &= \Delta y \left(1 - \sigma_x \delta x\right)^2 \left(1 - \sigma_y \delta y\right)^2 \left(1 + 2\sigma_x \delta x\right) , \\
P_{ij}(x,y) &= \Delta x \Delta y \left(1 - \sigma_x \delta x\right)^2 \left(1 - \sigma_y \delta y\right)^2 ,
\end{aligned}
\tag{3.20}
$$

with the absolute distance $\Delta x = x - x_i$, the position of the grid point x_i of which the basis function is calculated, the relative distances $\delta x = \Delta x / dx$, the gridspacing $dx = x_{i+1} - x_i$ and the sign σ_x which is positive or negative depending on the position of x relative to x_i. These quantities are calculated analogously for y.

The basis functions are plotted in Fig. 3.2 for gridspacing $dx = dy = 1$. The solution
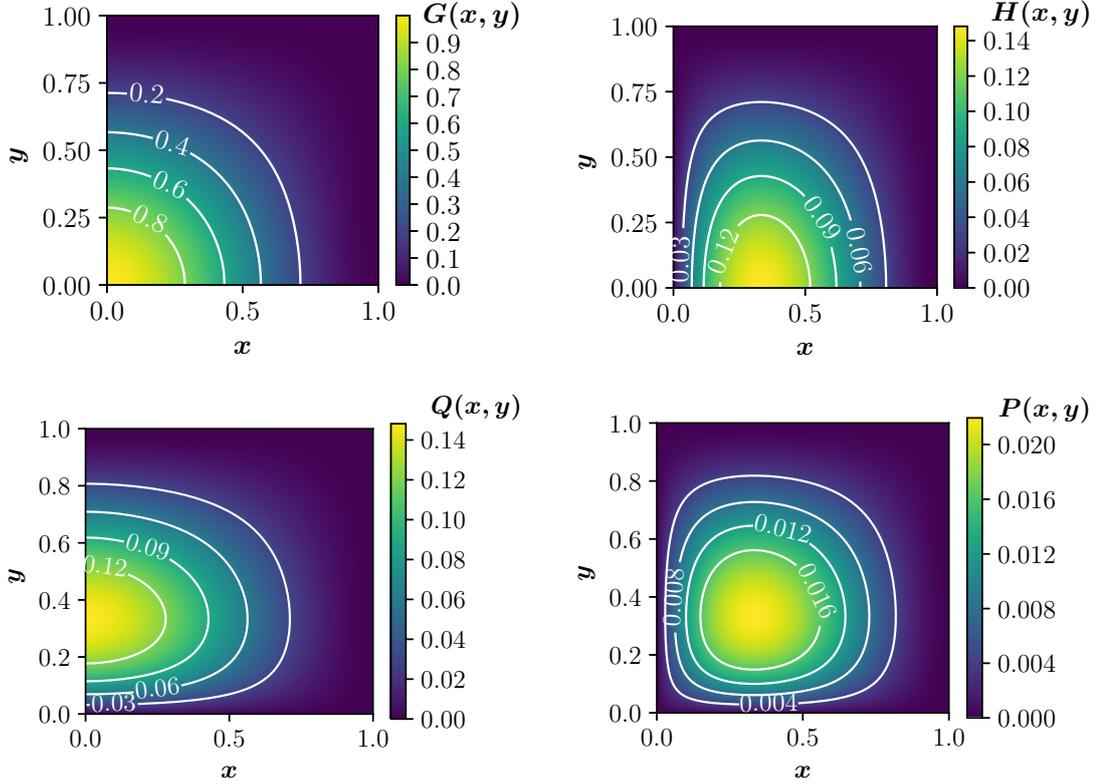

Figure 3.2: Basis functions of point x_i, y_i at $(0,0)$ used for the setup of the gyromatrix.

is then used to interpolate the points of the gyro-ring lying between grid points in x and y direction to build the gyromatrices.

The gyroaverage expressed by the basis functions   becomes

$$
\bar{f}(\mathbf{X}+\mathbf{r}) = \frac{1}{2\pi}\int_0^{2\pi}\sum_{i=1}^M\sum_{j=1}^N\Bigg(G_{ij}(x,y)+H_{i,j}(x,y)\partial_x
$$
$$
+\,Q_{i,j}(x,y)\partial_y+P_{i,j}(x,y)\partial_{xy}\Bigg)f_{ij}d\theta\,, \tag{3.21}
$$

which can be written as a matrix equation

$$
\bar{\boldsymbol{f}} = \left(\mathcal{G}_{\mathrm{basis}}+\mathcal{H}\partial_x+\mathcal{Q}\partial_y+\mathcal{P}\partial_{xy}\right)\boldsymbol{f}\,. \tag{3.22}
$$

As the derivatives are generally not given they have to be calculated numerically from the input vector $\boldsymbol{f}$. This can be done by multiplying $\boldsymbol{f}$ with a derivativative matrix $\mathcal{D}$ from the left

$$
\bar{\boldsymbol{f}} = \left(\mathcal{G}_{\mathrm{basis}}+\mathcal{H}\mathcal{D}^{(x)}+\mathcal{Q}\mathcal{D}^{(y)}+\mathcal{P}\mathcal{D}^{(x)}\mathcal{D}^{(y)}\right)\boldsymbol{f}\,. \tag{3.23}
$$

Fourth order centered differences are implemented in GENE-3D for the derivative matrices

$$
\mathcal{D}^{(x)} = \frac{1}{12dx}\begin{bmatrix} 1 & -8 & 0 & 8 & -1 \end{bmatrix}
$$

and $\mathcal{D}^{(y)}=\mathcal{D}^{(x)^T}$. Hence, the vector of the gyroaverage of f at all gyrocenter positions in the x,y plane can be written as

$$
\bar{f} = (\bar{f}(\mathbf{X}_1),\bar{f}(\mathbf{X}_2),\ldots,\bar{f}(\mathbf{X}_{n_xn_y})) = \mathcal{G}f\,. \tag{3.24}
$$

Gyromatrix properties and storage

As the gyromatrices are square matrices the number of rows N and columns M are the same

$$
N = M = n_x\cdot n_y\,. \tag{3.25}
$$

As the gyroradius ρ in Eq. (3.10) is dependent on the magnetic moment μ and species σ GENE-3D saves one gyromatrix for every point in species-, perpendicular velocity- and parallel space. Large amounts of memory are necessary to store the matrices. Since the gyroradii are small compared to the box size, the matrices are sparse (see Fig. 3.6). Storing them in a dense matrix format would quickly exhaust the memory of any modern supercomputer. Therefore the Portable, Extensible Toolkit for Scientific Computation (PETSc) [73, 74, 75] is used to store the matrices in the "MPIAIJ" sparse matrix format. Here, for every row, a list of column indices with their non-zero values are stored. As the name suggests, it is possible to distribute the matrix over different MPI cores by storing

only row ranges per core. This parallelization allows simulating at high resolutions with GENE-3D.

3.1.3   Setup of the field matrix

Additionally to the gyroaverage operator Eq. (3.6) contains the operator $\mathcal{K}$ which is approximated in GENE-3D by the adjoint of the gyromatrix $\mathcal{G}$ accoring to Ref. [71]

$$\mathcal{K}f = \frac{1}{2\pi} \oint f(\mathbf{x} - \mathbf{r}(\alpha))\, d\alpha = \mathcal{G}^\dagger f\,. \tag{3.26}$$

Hence, the integral on the left hand side of Eq. (3.6) is Hermitian,

$$\mathcal{P}''\phi_1 = \sum_\sigma \hat{n}_{0\sigma}(x_0)\hat{q}_\sigma \pi \int d\hat{v}_\parallel d\hat{\mu}\, \mathcal{G}^\dagger \hat{B}_0(x,y,z)\hat{F}_{1\sigma}\,, \tag{3.27}$$

with

$$\mathcal{P}'' = \sum_\sigma \hat{q}_\sigma^2 \frac{\hat{n}_{0\sigma}(x)}{\hat{T}_{0\sigma}(x)}\left(\mathbb{1} - \int \mathcal{G}^\dagger \frac{\hat{B}_0(x,y,z)}{\hat{T}_{0\sigma}(x)} e^{-\frac{\hat{\mu}\hat{B}_0(x,y,z)}{\hat{T}_{0\sigma}(x)}} \mathcal{G}\, d\hat{\mu}\right), \tag{3.28}$$

representing the field matrix with $\hat{\mu}' = \hat{\mu}B_0(x,y,x)/T_0(x)$. The equation can be rewritten in a simplified operator form

$$\mathcal{P}''\Phi = \rho\,, \tag{3.29}$$

with ρ being the charge density. The $\mathcal{P}''$ field matrices are also saved in the PETSc sparse matrix format, and the linear system of equations initially was solved by a direct interface to MUMPS to obtain the electrostatic field potential Φ needed in Eq. (2.27). Sec. 3.2.5 explains how the field solver evolved throughout this Ph.D. project and outlines its current form.

3.1.4   Numerical schemes

Time discretization   To time integrate Eq. (3.4) a standard four stage Runge-Kutta (RK4) scheme [76, 77] is used

$$\hat{F}_1^{t+1} = \hat{F}_1^t + dt \cdot \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right). \tag{3.30}$$

The species label index σ is omitted for better readability. The Runge-Kutta stages k are defined with the right-hand side (`rhs`) of Eq. (3.4) as

$$k_1 = \texttt{rhs}(\hat{F}_1^t, t), \tag{3.31}$$

$$k_2 = \texttt{rhs}(\hat{F}_1^t + \frac{dt}{2} k_1, t + \frac{dt}{2}), \tag{3.32}$$

$$k_3 = \texttt{rhs}(\hat{F}_1^t + \frac{dt}{2} k_2, t + \frac{dt}{2}), \tag{3.33}$$

$$k_4 = \texttt{rhs}(\hat{F}_1^t + dt\, k_3, t + dt). \tag{3.34}$$

In all four stages the field equation (3.6) is solved which is computationally expensive. Every solution is dependent on the solution of the previous time step. Hence, an initial condition for the distribution function $\hat{F}_1^0$ needs to be set. Different initial conditions like Gaussians in x, y, z directions are implemented in GENE-3D. Ideally, the initial condition should not influence the result of simulations except for relaxation tests. This is tested to be true for the cases studied in this thesis. The RK4 method is accurate to fourth-order, meaning that the (local) error caused by one iteration is of order $\mathcal{O}(dt^5)$ while the (global) cumulative error caused by many iterations is of order $\mathcal{O}(dt^4)$. However, as the Runge-Kutta method is not a symplectic integrator, it is not energy conserving, which does not matter in the GENE-3D context as already energy conservation is violated when neglecting terms of higher-order in ϵ in Eq. (2.27).

Spatial coordinate discretization and derivatives   Currently, all three spatial dimensions are discretized on a fixed equidistant grid. Finite difference schemes are used for numerical calculation of derivatives. In particular, fourth-order centered differences are considered a good compromise between time step demand, accuracy, and computational expense in GENE-3D. The finite-difference coefficients for the fourth-order first derivative at point ξ_i are

ξ_{i-2}	ξ_{i-1}	ξ_i	ξ_{i+1}	ξ_{i+2}
$1/12$	$-2/3$	0	$2/3$	$-1/12\,,$

where ξ can be any of the spatial dimensions (x, y, z). Hence, the derivative at point ξ_i is calculated as

$$f'(\xi_i) = \frac{f(\xi_{-2})/12 - 2f(\xi_{-1})/3 + 2f(\xi_{+1})/3 - f(\xi_{+2})/12}{d\xi} + \mathcal{O}(d\xi^4), \tag{3.35}$$

where $d\xi$ is the grid spacing in ξ direction. The derivative calculation at a point, therefore, requires information about the function values at the four neighboring grid points

of the given direction. Hence, at the boundaries of each core's simulation domain, ghost cells need to be exchanged when parallelizing. This is explained in detail in Sec. 3.2.3.

Velocity space discretization and integration   The fluctuating part of the distribution function converges to zero at high velocities. Hence, zero Dirichlet boundary conditions are assumed outside the parallel velocity domain. As a consequence, the maximum value of the $v_\parallel$ grid needs to be chosen carefully to capture all relevant velocity space dynamics. For collisionless simulations as performed in this thesis, no boundary conditions are necessary for the magnetic moment μ as no $\partial/\partial\mu$ terms need to be calculated. For the $v_\parallel$ integration an extended Simpson's rule [78] is implemented in GENE-3D. For the μ integration, a Gaussian quadrature scheme with Gauss-Legendre weights and knots is implemented in order to reduce the number of grid points necessary for a converged resolution [79].

Conclusion

With the previous work done, there was still a left to do to turn GENE-3D into a production code (see Sec. 3.2). Parts of the GENE-3D code still used the gyroaverage and field solver of the x-global GENE version, which are not suited for 3D geometries. Furthermore, the y-dependence of arrays was not implemented, which only made it possible to compare GENE-3D with x-global GENE but would not allow simulating actual 3D geometries. Also, with $140,000$ lines, the code was difficult to work with. Therefore, a massive refactoring is done to reduce the number of lines and to get consistent and concise software architecture. Refactoring helped on the way to a robust and well-performing 3D-code, which is easy to read, easy to debug, and easy to modify for future projects. GENE-3D is now embedded into a toolchain where GVEC [55] provides the geometrical input for the simulations (see Sec. 3.3.1). The simulation output is then read by python postprocessing tools, which then interface to ParaView visualizations (see sections 3.3.2 and 3.3.3).

3.2   Turning GENE-3D into a production code

This section outlines the changes done throughout this Ph.D. project in the GENE-3D code in order to turn it into production code. The transition from a Fourier space representation to real space is described in Sec. 3.2.1. The boundary conditions with an emphasis on the intricate parallel boundary condition (see Fig. 2.2) are explained in 3.2.3. The interpolation from the GENE coordinate system to the PEST coordinate system for visualization and comparisons to other codes is outlined in Sec. 3.2.4. The efficient and flexible PETSc linear equation solver is described in detail in Sec. 3.2.5. Furthermore, the scaling behavior is measured in Sec. 3.2.7 as well as the overall time consumption for different scenarios. Based on these measurements, possible optimizations for the future are discussed.

3.2.1   Transitioning from a Fourier representation to real space

Before this project, many calculations in GENE-3D were still done by the Fourier space implementations of the x-global GENE version. This was done by Fourier transforming the real space quantities into Fourier space, perform an operation like the gyroaverage, and then Fourier transforming the result back into real space. Hence, GENE-3D could only be used to simulate axisymmetric geometries while being slower than the x-global GENE version. For the GENE-3D function `CalFullRhs` which calculates the fields and then evaluates the right-hand-side of Eq. (2.27) this is visualized in Fig. 3.3.

Additionally, inside of the `calc_rhs_only` function a total of 15 Fourier transformations were performed. In the GENE-3D version developed throughout this project, the variables `g`, `f` and `emfields` are always real space variables. The Runge-Kutta-4 (RK4) scheme, which adds the output argument `rhs` of `CalFullRhs`, is also acting in real space so that no Fourier transformations are necessary anymore.

These changes were necessary to simulate 3D geometries as, for example, the gyroaverage operator of the x-global GENE version acting in Fourier space cannot take y-dependencies into account. Also, it makes the code more concise, easier to read, easier to debug, and faster as the FFTs have a runtime of $\mathcal{O}(n \log n)$.

3.2.2   Adding y-dependence and parallelization

In x-global GENE, as well as the original version of GENE-3D geometric quantities like the Jacobian and the magnetic field, do not have dependencies in the y-direction. Geometric quantities hence needed to be extended to represent 3D geometries like stellarators and perturbed tokamaks. Representing them requires many grid points in y-direction compared to axisymmetric systems. For example, a linear simulation of an axisymmetric tokamak geometry uses one point in y-direction to simulate a single k_y-mode in Fourier space. In contrast, GENE-3D uses 384 points in y-direction in a linear simulation
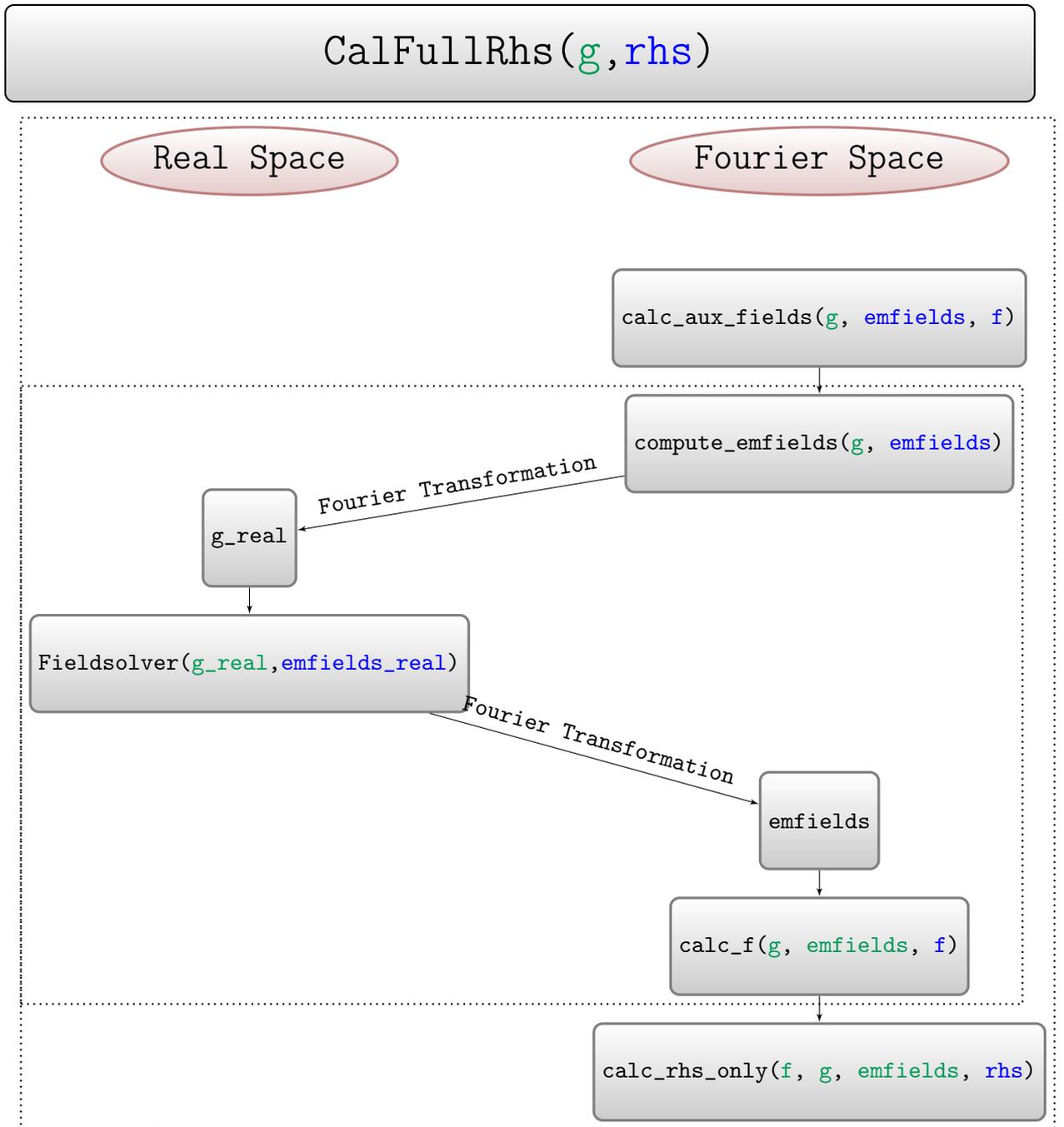
Figure 3.3: Old version of CalFullRhs: Fourier transformations were necessary. As an FFT scales with $N \cdot \log(N)$ making the simulations more expensive. Input parameters to functions are colored in green while output parameters are blue. The Runge Kutta 4 scheme (3.31) adds the complex results of CalFullRhs `rhs`.

of Wendelstein 7-X (see sec. 4.2.1). Hence y-parallelization is needed to simulate 3D systems in a reasonable amount of time and to distribute the memory requirements. Communicators need to be set up to parallelize such a system, e.g., for the real space boundary exchanges and flux surface averages. The main intricacy of parallelizing the y-direction represented in real space arose in the z-boundary exchange, which is described in the following.

3.2.3   Boundary conditions and boundary exchanges

Simulation domains are always finite. Hence, boundary conditions need to be employed, which ideally mimic the actual physical boundaries in the experiment. As realistic boundary conditions are often not feasible or connected with massive effort on the algorithmic side, simpler models have to be employed. Here, the boundary conditions and the corresponding boundary exchanges in the radial direction x, the y-direction, and the direction along the field line z (see Fig. 2.2) are outlined.

Radial x-direction   The x-coordinate in GENE-3D describes the flux surface label (see Eq. (2.49)) or in other words the radial position between the magnetic axis and the edge of the plasma (see Fig. 2.1). Radially global simulations have dependencies of geometrical coefficients and temperature and density profiles in the radial direction. However, the only radial boundary condition implemented at the start of this project were periodic boundaries

$$F(x, y, z) = F(x + L_x, y, z) \,. \tag{3.36}$$

This is a valid boundary condition in the local approximation where radial dependencies of equilibrium quantities are neglected. The periodic boundary condition in the radial direction is, therefore, mainly used for benchmarks with local codes. As soon as radial dependencies are considered, periodic boundaries can lead to numerical instabilities.

Dirichlet boundary conditions   have been implemented that were already successfully used in the x-global GENE version:

$$f(x_B, y, z) = 0 \quad \forall \ x_B < \frac{-L_x}{2}, \text{ and } \forall \ x_B > \frac{L_x}{2} \,, \tag{3.37}$$

with L_x being the total box size in x direction and f being an arbitrary function.

Here, particle and heat fluxes vanish at the boundaries which is adequate as long as the profiles are chosen such that the modes grow reasonably far away from the boundary. In the experiment, however, fluxes will propagate outwards until they reach the walls. This is still a limitation of GENE-3D, and discussions are ongoing on how to improve

the radial boundary condition both for x-global GENE and GENE-3D. One has to be careful, as the effect of the boundary condition can affect the simulation domain some distance away from the boundary. The length scale of this effect is dependent on ρ^* and the strength of the radial coupling that can be caused by the gyroaveraging procedure or radial derivatives. For this thesis, Dirichlet boundary conditions were chosen for their simplicity, positive experience of the x-global GENE version, and numerical stability. In GENE-3D the parameter `rad_bc_type` determines the radial boundary condition. For benchmarks against local codes the periodic option `rad_bc_type` $= 0$ is still available while for the Dirichlet boundary condition `rad_bc_type` $= 1$.

y-direction In the y-direction, the boundary condition is motivated by the physical condition for periodicity in the toroidal angle ζ, which can be expressed as

$$f(\rho, \zeta, \theta) = f(\rho, \zeta + 2\pi, \theta). \tag{3.38}$$

In the field-aligned coordinate system implemented in GENE-3D, this corresponds to the condition:

$$f(x, y, z) = f(x, y - \sigma_{B_p} 2\pi C_y, z). \tag{3.39}$$

To reduce computational cost, it is assumed that the correlation lengths are much smaller than a full toroidal turn and the simulation domain is assumed to be n_0-periodic. Hence, only an integer fraction $1/n_0$ of the full flux surface is simulated.

The field period of the geometry constrains n_0 for non-axisymmetric devices. For example, the five-fold symmetry of Wendelstein 7-X leaves two options for n_0 in GENE-3D. If n_0 is set to 1, the entire flux surfaces are simulated while for $n_0 = 5$, only one-fifth of a flux surface is covered. The same periodic boundary condition can be used in both cases as turbulence properties are assumed to be statistically identical at two locations with the same local magnetic field

$$f(x, y, z) = f(x, y - \sigma_{B_p} \frac{2\pi}{n_0} C_y, z). \tag{3.40}$$

The minimum mode number $k_y^{\min}$ is dependent on n_0, C_y and ρ^*

$$k_y^{\min} = \frac{n_0 \cdot \rho^*}{C_y}. \tag{3.41}$$

The reduction of the simulation domain in y-direction hence results in a thinning out of toroidal mode number space as the k_y modes are calculated as

$$k_y = k_y^{\min} \cdot j \,; \quad j \in \left[0 \dots \frac{ny0}{2}\right] \,. \tag{3.42}$$

While n_0 and C_y are determined by the geometry for a stellarator ρ^* can be chosen as a free input parameter. This means that when ρ^* is increased $k_y^{\min}$ and $k_y^{\max}$ are also increased. If k_y^{max} shall be kept constant for a ρ^* scan, for example, the number of points in y-direction $\texttt{ny0}$ has to be adjusted accordingly. For large values of ρ^*, it might be necessary to set $n_0 = 1$ to resolve large scale structures. Lastly, the size of the simulation box L_y in y-direction is calculated as

$$L_y = \frac{2\pi}{k_y^{\min}} \,. \tag{3.43}$$

Parallel z-direction   The boundary condition along the field line is the most intricate one. The safety factor q (see Eq. (2.47)) is generally an irational number. Hence, the field lines do not close onto themselves after one revolution around the torus. In 3D geometries, q is not a single value but a radially dependent profile $q(x)$, which further complicates the problem. The boundary condition implemented in GENE stems from the fact that the toroidal angle ζ should be the same after following the field line once around the torus (see Fig. 2.2)

$$f(\rho, \zeta, \theta) \stackrel{!}{=} f(\rho, \zeta, \theta + 2\pi)$$

$$\Rightarrow \zeta_1 = \frac{y_1 \cdot \rho^*}{C_{y0}} + q(x)z_1 \stackrel{!}{=} \zeta_2 = \frac{y_2 \cdot \rho^*}{C_{y0}} + q(x) \cdot (z_1 + 2\pi)$$

$$\frac{y_1 \cdot \rho^*}{C_{y0}} = \frac{y_2 \cdot \rho^*}{C_{y0}} + q(x) \cdot 2\pi$$

$$y_2 = y_1 - 2\pi \cdot q(x) \cdot \frac{C_{y0}}{\rho^*} \,.$$

The twist-and-shift boundary condition   in the ρ^* normalized coordinate system reads

$$F(x, y, z + 2\pi) = F(x, y - 2\pi \cdot q(x) \cdot \frac{C_{y0}}{\rho^*}, z). \tag{3.44}$$

The implementation of equation (3.44) in a yz-parallelized system is a bit complicated. Hence, the algorithm is described in detail in the following. First, everything that only needs to be done once per simulation is done in the initialization of the boundary exchange. The steps are outlined in algorithm 1. In the time loop only the steps shown

---

Algorithm 1 Initialization of Twist and Shift algorithm parallelized in yz

---

1: **procedure** INITIALIZE_BOUNDARY_EXCHANGE_Z

2:     `phase_shift(x)` $= 2\pi \cdot q(x) \cdot \frac{C_{y0}}{\rho^*}$ $\leftarrow$ x dependent phase shift in **y**-direction

3:     Setup `host_pey` array associating a host core to every point in **y** direction

4:     Setup shifted y positions `y_shift_positions` for upper and lower boundary

5:     Setup `y_difference` array with distance of `y_shift_position` to nearest grid point to the left

6:     Setup `shift_pey` array associating a host core to all `y_shift_positions`

7:     Setup `four_neighbor_indices` array with four neighboring grid points of `y_shift_positions`

8:     Setup `number_values_to` array for `y_shift_positions` whose `shift_pey` = `my_pey` and `host_pey` $\neq$ `my_pey`

9:     Setup `number_values_from` array for `y_shift_positions` whose `host_pey` = `my_pey` and `shift_pey` $\neq$ `my_pey`

---

in algorithm 2 need to be performed during the boundary exchange.

Algorithm 2 Time loop execution of Twist and Shift algorithm parallelized in yz

---

1: **procedure** EXCHANGE_Z_3D_REAL(ARRAY: **U**)

2:     Perform periodic boundary exchange of **u** in **z** direction

3:     Perform periodic boundary exchange of **u** in **y** direction

4:     Use function values of **u** at `four_neighbor_indices` to interpolate function values at `y_shift_positions`

5:     Send interpolated values whose `shift_pey` = `my_pey` and `host_pey` $\neq$ `my_pey` to their `host_pey`

6:     Receive interpolated values whose `host_pey` = `my_pey` and `shift_pey` $\neq$ `my_pey` from their `shift_pey`

7:     Sort received values into **u**

Additionally, the algorithm is also visualized in Fig. 3.4. The shift in y direction leads to the complication that for y-parallelized simulations, the value needed by one processor in the y-communicator needs to be interpolated by another processor. The interpolated value is then sent back to the processor needing it.

Cubic Catmull-Rom splines [80] are utilized for the interpolation. Catmull-Rom splines are a particular form of cubic Hermite splines where piece-wise interpolation
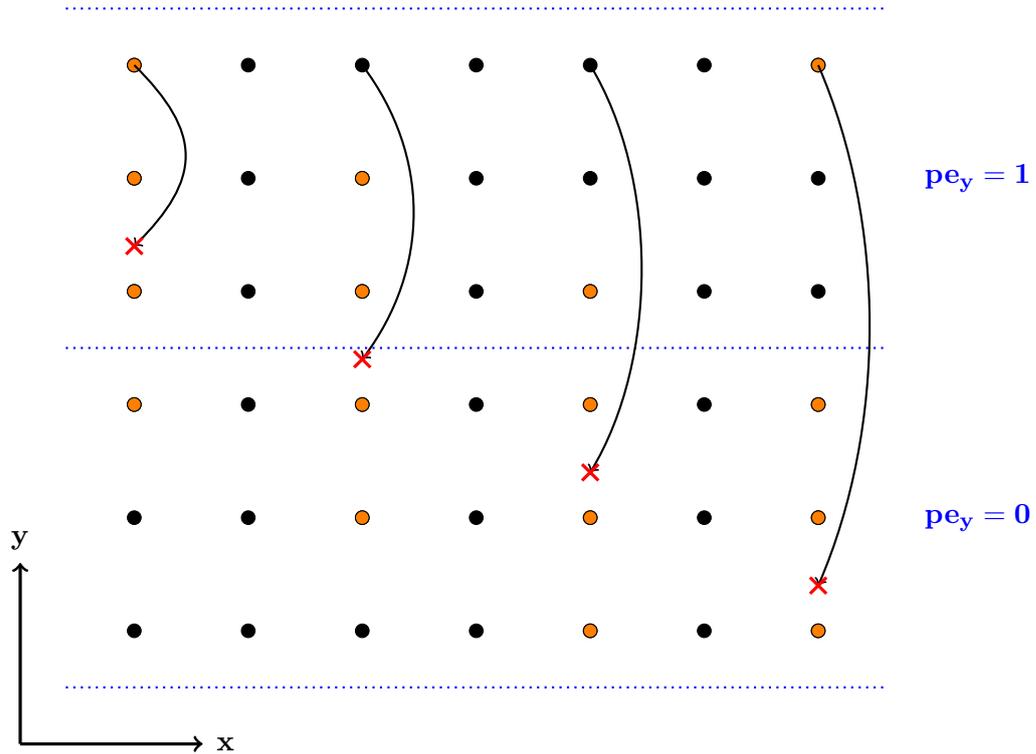
Figure 3.4: "Visualization of the phase shift of the twist and shift boundary condition. The shifted positions (red crosses) are calculated from the original grid point positions (black dots). The phase shift in y direction is dependent on x as the $q(x)$ profile has radial dependence (see Eq. (3.44)). The shifted position can be on a different core pe_y than the original grid point. Marked in gray are the grid points that are used to interpolate the function value at the shifted position. As the grid points can also be on different cores a boundary exchange in y is necessary" Cf. [31].

with third-degree polynomials is used to approximate values between known endpoints of a domain by using the values and first derivatives at the endpoints. In the particular case of Catmull-Rom splines, the first derivative at a given point is approximated by using central finite differences with a two-point stencil. Given four points with function values v_0, v_1, v_2 and v_3 and interpolating between points 1 and 2 the interpolated value $f(\delta)$ at position δ on the unit interval $[0, 1]$ is calculated as

$$a_0 = -\frac{1}{2}v_0 + \frac{3}{2}v_1 - \frac{3}{2}v_2 + \frac{1}{2}v_3$$

$$a_1 = v_0 - \frac{5}{2}v_1 + 2v_2 - \frac{1}{2}v_3$$

$$a_2 = -\frac{1}{2}v_0 + \frac{1}{2}v_2$$

$$a_3 = v_1$$

$$f(\delta) = a_0 \cdot \delta^3 + a_1 \cdot \delta^2 + a2 \cdot \delta + a_3 \,. \tag{3.45}$$

As communication along the y-communicator is necessary perfect scaling is inhibited when parallelizing in y-direction. Also, as the twist and shift boundary condition is only done by the first and last core in the z-communicator, there is a load imbalance as all other cores do nothing. In reality, this rarely matters as the twist-and-shift boundary usually takes around 0.3% of the total simulation time (see Sec. 3.2.7). The real space implementation improves the performance by a factor of two compared to the Fourier space implementation and displays the scaling properties described in Sec. 3.2.7. This is the default parallel boundary condition used in GENE-3D.

A periodic boundary condition   in the parallel direction

$$f(x, y, z) = f(x, y, z + 2\pi) \,, \tag{3.46}$$

has been shown to give similar results to the twist-and-shift boundary condition for non-axisymmetric geometries [81]. As the periodic boundary condition saves computational time compared to the twist-and-shift boundary condition, the periodic boundary condition in parallel direction is also implemented in GENE-3D.

3.2.4   Visualization interpolation to Cartesian grid

To visualize GENE-3D simulation data in Paraview (see Sec. 3.3.3) the values of the array to be visualized need to be interpolated from the GENE grid (Fig. 2.2) to PEST

coordinates (Fig. 2.1). Also, for a smooth visualization, the visualization grids need to have more points than the GENE simulation grid.

GENE z to θ coordinates   To this end, the $z = \sigma_{B_p}\theta$ interpolation is done first as outlined in algorithm 3. As the GENE-3D z coordinate is the same as the θ coordinate (except for a sign prefactor) the interpolation only has to be done to have a finer grid (see Fig. 3.5). Therefore, the finer visualization grid is initialized, and the θ values are calculated at the grid points. Then, the four nearest neighboring grid points of the z grid are calculated. The function values at those points are used to interpolate the value at the grid point of the visualization grid.

0   **n$_{\mathbf{z}}$ − 1**

| | | | | | | |

| | | | | | | | | | | | | | | | |

0   **n$_\theta$ − 1**

Figure 3.5: The function values at the grid points of the fine θ visualization grid (bottom) are interpolated from the function values at the original z grid points (top)

Algorithm 3 Interpolation from GENE z grid to θ visualization grid

---

1: **procedure** INTERPOLATION_Z_THETA
2:     Create θ visualization grid with finer grid spacing
3:     Calculate θ positions of grid points in visualization grid
4:     Interpolate the function values from GENE z grid to θ positions

---

GENE y to ζ coordinate   The relation between the GENE y coordinate and the PEST ϕ angle is given by equation (2.49)

$$y = \sigma_{B_p} C_y (q\theta - \phi)\,. \tag{3.47}$$

In GENE-3D, this is done by creating the ϕ grid first. Then, for all the points of the visualization grid in x and θ, the ϕ grid is mapped to the GENE y grid. The four neighboring grid points of the GENE y grid are taken to interpolate the function value at the mapped position. For interpolation, the same third-order cubic Catmull-Rom splines are used as for the z-boundary exchange (see sec. 3.2.3).

Initially, this was implemented directly into GENE-3D. Later, to save computation time during the simulation, the visualization interpolation was moved to the python diagnostic tool.

3.2.5 The PETSc field solver

(a) Gyromatrix

(b) Field matrix

Figure 3.6: "Illustration of the sparsity pattern of (a) the gyromatrix, $G_{nk}(z,\mu,\sigma)$, and (b) the field matrix, $\mathcal{P}''_{nk}(z)$, for a W7-X case of resolution $k = n = n_x \times n_y = 64 \times 64$ (at a given z,μ,σ grid point). Here, the fraction of non-zero entries is 1.2% for the gyromatrix and 3.4% for the field matrix" Cf. [31].

Because of the geometric variations in the y-direction the field equation cannot be solved independently for different y points like in x-global GENE. Hence, for every z point in the simulation a linear equation of the form

$$\mathcal{P}''\Phi = \rho \tag{3.48}$$

has to be solved. $\mathcal{P}''$ is the field matrix implemented as `p_doubleprime` which is of dimension $(n_x \cdot n_y)^2$. Φ is the one dimensional `emfields` vector of length $n_x \cdot n_y$. ρ is the charge density, also a one dimensional vector of length $n_x \cdot n_y$. PETSc is used to store the sparse field matrix and the vectors. Details on how PETSC matrices and vectors are built can be found in Refs. [73, 74, 75]. A sparsity pattern of a gyromatrix and a field matrix from the simulation in Sec. 5.1.4 is shown in Fig. 3.6.

PETSc is also used to solve equation (3.48) in GENE-3D. Here, a brief outline of different options for linear equation solvers which interface to PETSc is given. Before this project, PETSc was only used to store matrices and vectors, and an explicit interface to the MUMPS library [82, 83] was implemented in GENE-3D. This was inflexible. Therefore, the solution of the linear equation is now done directly by PETSc, which internally can use different libraries, including MUMPS. Interestingly, using MUMPS through PETSc resulted in better performance than calling it directly.

There are two categories of different solvers: direct solvers like MUMPS, SuperLU [84, 85, 86], and PaStiX [87] and iterative solvers like conjugate gradient (CG) and GMRES that need to be coupled with a preconditioner to work.

Direct solvers   use LU decomposition to solve the linear system of equations. Instead of solving the system $Ax = b$ directly, the matrix A is decomposed into a lower triangular matrix L and an upper triangular matrix U:

$$A = LU \tag{3.49}$$

$$\rightarrow LUx = b. \tag{3.50}$$

Therefore the 2 equations

$$Ux = y \tag{3.51}$$

$$Ly = b \tag{3.52}$$

need to be solved which is easier given that L and U are triangular matrices. Computationally, the initial LU decomposition step is of order $\mathcal{O}(n^3)$ while all subsequent steps are of order $\mathcal{O}(n^2)$. The solution obtained from solvers using LU decomposition is accurate to machine precision as the system of equations is solved directly.

Iterative solvers   solve the system of equations by starting with an initial guess x' for x and using iterative algorithms to get closer to the correct answer. A cut-off accuracy ϵ is defined such that the iteration ends when

$$\left| Ax' - b \right| < \epsilon. \tag{3.53}$$

The accuracy of the solution x' then depends on the choice of ϵ. PETSc furthermore features multiple options for preconditioners, which can be used to transform A in such a way that the condition number decreases. It is then more suitable for the iterative solving algorithms, and hence the time to find a solution decreases.

The performance of different solvers   is tested throughout this project along with scalability and memory usage (see sec. 3.2.7). While iterative solvers like a CG solver with a Jacobi preconditioner can be faster up to a factor of 50 than a direct solver like SuperLU for some problems, it can be a factor of 10 slower for others. The performance of iterative solvers depends critically on how well the preconditioner suits the matrix A in equation (3.48). Which solver is computationally the most efficient ultimately depends on the problem size, resolution, and whether adiabatic or kinetic electrons are used. However, the memory usage is universal. As iterative solvers do not save the LU-

decomposed matrices, memory can be saved. For a test problem with $n_x = n_y = 128$, the iterative conjugate gradient algorithm used 30% less memory than SuperLU. For the flux-surface averaged system of linear equations, a direct solver is always used as the matrix sizes are small, and the direct solvers are better suited.

Hyper-diffusions

The centered difference schemes used to calculate derivatives in GENE-3D can introduce unphysical high-k modes. These can overshadow the actual physics in a simulation. One solution to this problem is adding numerical hyper-diffusion terms, which can damp unphysical high-k modes, see Ref. [88]. Explicit hyper-diffusion terms with an amplitude that can be set by the user have the advantage of full control over the level of damping compared to upwind schemes with intrinsic damping. However, this introduces free parameters, which is why convergence in the hyper-diffusion terms is benchmarked in Sec. 4.2. When using centered differences, there is always a trade-off between accuracy and computational costs. Large stencils increase the communication necessary between cores. However, low order schemes require lower stencil widths and, therefore, higher resolution. A good compromise for accuracy, computational cost, and communication needed is usually the fourth-order scheme that is used for first derivatives in GENE. The hyper-diffusion terms acting on the distribution function f typically used in GENE-3D use the fourth derivative with second-order stencils

$$\mathcal{H} = \eta \frac{-f(x_{i-2}) + 4f(x_{(i-1)}) - 6f(x_{(i)}) + 4f(x_{(i+2)}) - f(x_{(i+2)})}{16} \,. \tag{3.54}$$

The parameter η is an input parameter for GENE-3D called `hyp_*` where * can be any of the dimensions (x, y, z, v). Hyper-diffusion terms were implemented for the x and y direction in this project. A systematic study on how hyper-diffusion affects the simulation outcome of x-global GENE can be found in Refs. [88, 89].

3.2.6   Code structure

The idea behind the development of GENE-3D in this project is to develop a robust and well-performing code that is easy to read, debug, and modify for future projects. To this end, the old GENE-3D version is refactored, and large parts of the code are rewritten. In a two-step process, the code was reduced from 251 files consisting of 138691 lines to 88 files containing 28000 lines. This reduction decreased the compilation time from more than five minutes to less than one. The main reason for the reduction is the separation of the source code of GENE-3D from the other versions of GENE. Large parts of the code were only useful for the function and performance of the local version of GENE, x-global GENE, or y-global GENE. Also, different programming paradigms are reduced to a single

procedural programming paradigm. This has several advantages. Readability, especially for new users, is improved. Debugging a procedural code is more straightforward in an HPC application. The program flow becomes apparent. Also, after the reduction, the number of compiler warnings became manageable and was further reduced to zero. This makes it easier to spot essential warnings that otherwise might remain unseen.

Also, GENE-3D is now integrated into a production chain from geometric preprocessing with GVEC over GENE-3D to postprocessing tools, including ParaView visualizations (see sec. 3.3).

Valgrind Memory leaks   The code version before this project was leaking memory at numerous parts of the code. In some instances these memory leaks were inside of a loop. The tool Valgrind [90] helped in finding those memory leaks. After fixing them, GENE-3D can be considered memory leak free. This helps GENE-3D to be efficient in memory usage and it also helps the developer to be aware of the memory management of the code.

3.2.7   Performance aspects

Implementation of performance measurements   During this project, performance measurements are implemented, which are indexed by integers and only depend on the MPI walltime function. This has three main advantages:

- No external libraries are needed to perform measurements

- No measurable performance overhead of measurement

- Performance can be measured in every simulation

Hence, for every simulation run by GENE-3D, there is a detailed description of where the simulation time is spent. This has helped and will help to find performance bottlenecks and to quantify improvements. It is implemented by defining an enumerator, which is a numerated list of macros. This way, every measured function gets a distinct number which is passed into the `timemeasure_before` and `timemeasure_after` functions. Here, the difference in walltime between the two calls is measured, and the call count increased by 1. At the end of the simulation, the measured times for the different functions are printed in a human-readable form. The performance summary contains

- mean: average time spent in a given function averaged over all processors

- stddev: standard deviation around the mean value calculated over all processors

- min: minimum time spent on any processor

- max: maximum time spent on any processor

- #calls: the number of times the function was called.

The stddev, min, and max function help to detect load imbalance, which can be very detrimental to parallelization performance.

Scaling behavior   Extending GENE to a 3D-global code also increased the resolution and computing power requirements of the simulations. For instance, a highly resolved Wendelstein-7X simulation requires even linearly with adiabatic electrons a resolution in directions (x, y, z, $v_{||}$, μ) of 128 x 1024 x 128 x 48 x 10 $\approx$ 8 billion grid points. Parallelization and excellent scaling behavior are therefore paramount. Hence, parallelization in all (x, y, z, $v_{||}$, μ) directions as well as over the species label is implemented in GENE-3D. Scalability is usually measured with two methods: Strong scaling and weak scaling. In the case of strong scaling, the problem size remains fixed while the number of processors is increased. In weak scaling, the problem size per processor remains fixed while the system size is increased. Weak scaling is relevant when larger physical systems are simulated on an increasing number of processors.

Strong scaling   can be quantified by the speedup S_p which is defined as the time spent by one processor to solve a problem divided by the time p processors need to solve the same problem,

$$S_p = t_1/t_p \,. \tag{3.55}$$

Perfect (linear) scaling is achieved when the speedup is equal to the number of processors $S_p = p$. This perfect behavior is depicted in Fig. 3.7a as the dashed grey line. According to Amdahl's law [91], perfect scaling can never be achieved as parallelization always has overhead, and global MPI communication serializes the code. The sequential parts of code will inevitably dominate the computing time as parallelization is increased. Memory requirements usually limit the range in which strong scaling can be measured as a problem fitting into the memory of one processor will lead to a small problem size per processor when running on, e.g., 24576 processors where parallelization overhead will inevitably dominate. Therefore, instead of taking the simulation time on one processor as a baseline, the time on 5120 cores is used as the reference time t_1. A nonlinear simulation of W7-X is chosen as described in Sec. 5.1.4 to demonstrate the strong scaling behavior of GENE-3D. The resolutions are 240 x 256 x 128 x 48 x 10 in directions (x, y, z, $v_{||}$, μ) with number of processors $40 \times 4 \times 64 \times 2 \times 1$ as a base line. For the strong scaling benchmark the z-parallelization is changed from 64 to 32, 16 and 8 as depicted in Fig. 3.7a. Considering this is a real-life example of a simulation that is run with these exact parameters, the overall scaling behavior can be considered satisfactory. Usually, one would stop increasing the parallelization in strong scaling when a doubling in the number of processors only yields a performance gain of a factor of 1.5. GENE-3D is

above this limit even when going to the maximum number of cores available (20480) on a modern supercomputer like Cobra [92].

Weak scaling can be measured over a wide range of number of processors as the memory requirements stay relatively constant. In weak scaling, perfect linear scaling is achieved when the run time remains constant as the workload and the number of processors is increased proportionally. The efficiency of weak scaling is therefore measured as

$$\eta_w = t_1/t_N \,, \tag{3.56}$$

with t_N being the time N processors need to simulate a problem N times larger than the problem solved by one processor in t_1 time. As for the strong scaling test, the nonlinear Wendelstein 7-X simulation of Sec. 5.1.4 is taken as the benchmark parameter case. This time, both the resolution and the parallelization in z-direction are changed from $(\texttt{nz0}, \texttt{n\_procs\_z}) = (128, 64)$ down to $(\texttt{nz0}, \texttt{n\_procs\_z}) = (8, 4)$ while keeping the ratio of two points per processor constant. The results are shown in Fig. 3.7b. The efficiency curve stays relatively flat from 5120 to 20480 cores. To conclude, good parallelization behavior is found both for strong and weak scaling for a realistic case with the code version of august 2019. It is therefore possible to run GENE-3D on several $10,000$ cores making above mentioned simulations with eight billion grid points possible at an expense of multiple $100,000$ CPU hours (see Sec. 4.2.1).

(a) Strong scaling (b) Weak scaling
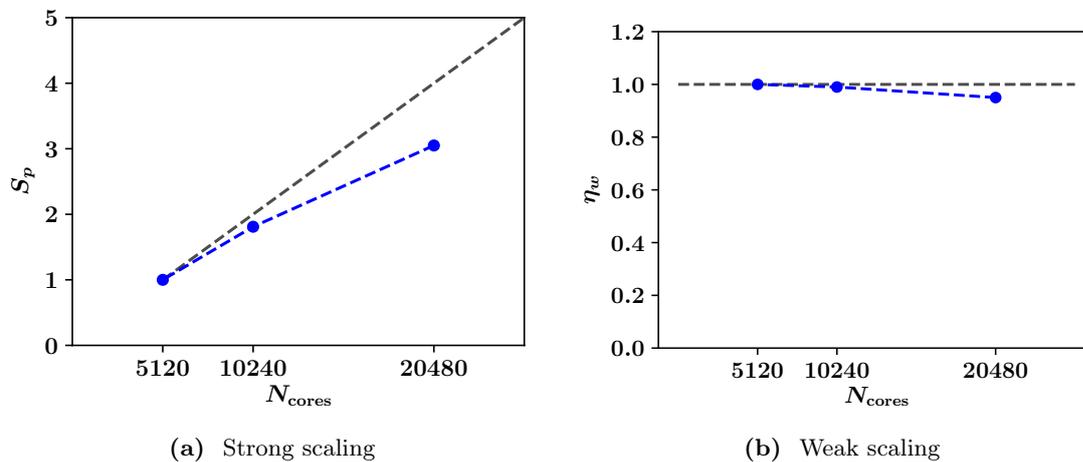
Figure 3.7: Scaling behavior of GENE-3D: the strong scaling (a) and weak scaling (b) behavior is promising for efficient runtime at high parallelizations. The parameter set chosen is the same as in Sec. 5.1.4. Hence, a realistic parameter set is chosen to show real-life scaling behavior.

The scaling behavior of different parts of the code can be analyzed individually thanks to the performance measurements. At various stages of this project,

the scaling behavior was measured and improved. After the twist and shift boundary (see Sec. 3.2.3) was implemented in real space, the y-coordinate could be parallelized, and the scaling behavior of the different dimensions could be compared. Fig. 3.8 shows the strong scaling behavior in all six phase space dimensions $x, y, z, v_\parallel, \mu$ and species in the labels of the parameter input file for the two main parts of the code `calc_rhs_only` and the `Fieldsolver` and the total time per time step. These plots are based on data created with the GENE-3D version of November 2018. The dashed grey line represents the optimal scaling behavior. The resolution of all simulations is $(n_x \cdot n_y \cdot n_z \cdot n_v \cdot n_\mu \cdot n_{\mathrm{spec}} = (128 \cdot 128 \cdot 128 \cdot 128 \cdot 64 \cdot 2))$. The plots show that the right-hand-side calculations scale well in all directions up to a low number of points per core. The x, y, z, and v direction do not scale perfectly as derivatives in these directions are calculated, and hence boundary exchanges are necessary.

However, the field solver shows a different scaling. For this test, an iterative solver is chosen as direct solvers scale even worse, and for adiabatic electron runs, the standard solver is an iterative one at the time of writing this thesis. The z-direction scales very well as the different points along the field line solve a different linear equation due to the gyrokinetic ordering. The x and y direction also scale well as the matrices are distributed in memory over the x and y dimension so that the matrix-vector multiplications of the iterative solvers can be parallelized. The velocity space and species dimension, however, barely scale at all as they solve the same system of linear equations. In theory, it should be possible to reach a scaling behavior similarly well to the x, and y direction as the information of the matrix entries is redundant, and the matrix-vector multiplications could be parallelized. This, however, is done internally in PETSc and is not under the control of a developer using PETSc as an external library. Although the strong scaling test is performed with an older GENE-3D version, the qualitative behavior has not changed at the time of writing this thesis.

Time Consumption   The previous paragraph shows that different parts of the code scale differently. While the right-hand side calculations scale well in all directions, the same is not true for the field solver. Hence, the time spent in the different parts of the code needs to be monitored to evaluate the parallelization efficiency. The more time is spent in the right-hand side computations, the better it is in terms of efficiency. If most of the time is spent in the field solver, further increasing the number of processors will not increase the performance of the simulation.

Table 5 summarizes the ratio of time spent in field solver / time spent in right-hand side computations for the four different simulation types run in Sec. 5.1. For adiabatic electron runs, the ratios are below 1, and hence more time is spent in the right-hand side computations. The linear kinetic electron case already takes 1.6 times more time in the field solver. For nonlinear kinetic electron runs, the ratio is 146, and the simulation
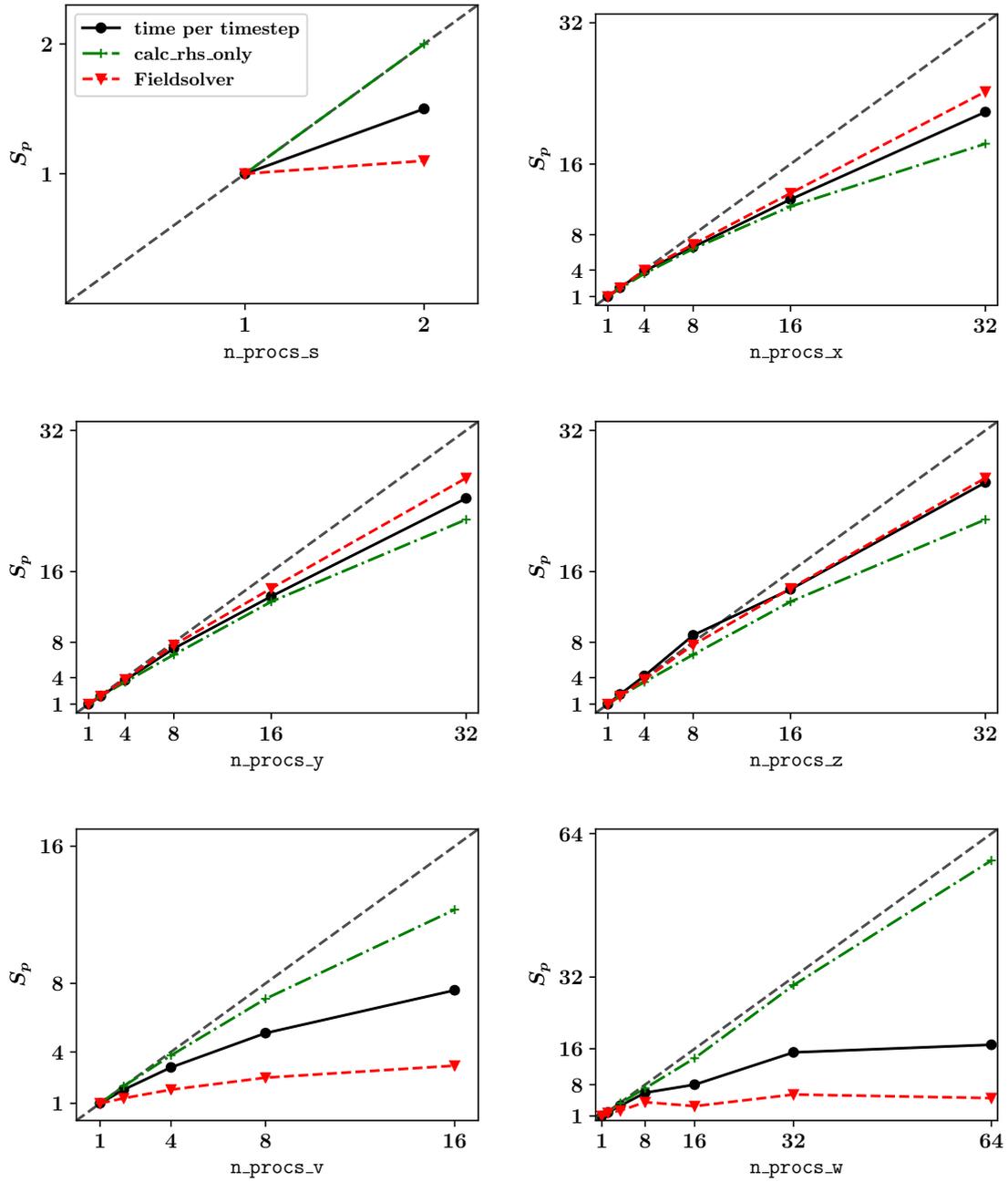
Figure 3.8: Strong scaling parallelization test in all dimensions.

spent all the time in the field solver. This makes the nonlinear kinetic electron simulation unfeasible. The difference between the linear and the nonlinear simulation is mainly more resolution in the combined x and y direction in the nonlinear case. The field matrix has a size of 46080^2 in the linear case and 61440^2 in the nonlinear case. More investigations are necessary to see why the relatively small difference in matrix size leads to such a vast difference in time spent in the field solver. Both kinetic electron simulations are run with the direct solver SuperLU as the preconditioners of the iterative solver do fit to the kinetic electron field solver matrices.

Memory scaling With increasing resolution and larger matrices also the memory demand increases. Fig. 3.9a shows how the memory needs of a GENE-3D simulation scale with the number of points in the directions $x, y, z, v_\parallel, \mu$. The base line resolution is chosen to be `nx0 = 20, ny0 = 8, nz0 = 8, nv0 = 8, nw0 = 8` in a geometry with concentric flux surfaces. The number of processors is set to 1 in all directions except for the z direction where `n_procs_z = 2` The memory usage is linear with increasing points in the directions $z, v_\parallel, \mu$. In these dimensions, a higher number of points just increases the size of the distribution function, and in the case of z and μ increases the number of gyromatrices which need to be stored. This is why increasing the number of points in $v_\parallel$ has the smallest effect on the memory footprint. For the x and y direction, the memory usage strongly depends on the number of points per gyroradius ρ^*. The memory usage for a box size of $L_x = 20\,\rho^*$ is higher than for a box size of $L_x = 150\,\rho^*$ for all `nx0` as the smaller box increases the number of points per gyroradius. This also increases the density of the gyro-matrices and field matrices as the base functions of more gyrocenters are needed to calculate the gyroaverage at a given gyrocenter. Also, the matrices get larger as the dimensions of all matrices in GENE-3D are $(\texttt{nx0} \cdot \texttt{ny0})^2$. The box size in y direction is $L_y = 20\,\rho^*$ for all simulations shown here. The simulations are performed with the SuperLU solver. The MUMPS solver uses for $L_x = 20\,\rho^*$ and $\texttt{nx0} = 80\,\rho^*$ 220 MB compared to 254 MB for the SuperLU solver.

Increased memory usage can be compensated for by increasing parallelization. Ideally, when doubling the number of processors, the memory usage per processor should be cut in half. This scaling is equivalent to the computation time being half when a code

Table 5: Ratio of time spent in field solver vs. time spent in right-hand side computations for simulations in Sec. 5.1. Nonlinear kinetic electron simulations are not feasible at the moment as all the time is spent in the field solver.

|	adiabatic	kinetic
linear	0.6	1.6
nonlinear	0.9	146
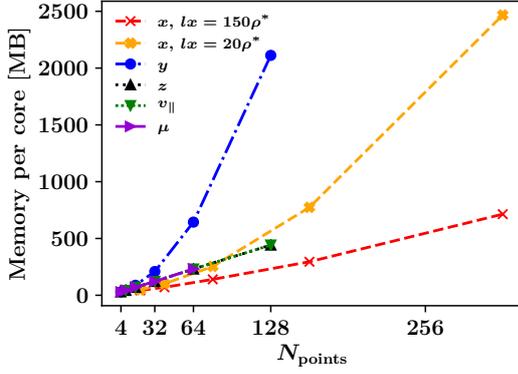
(a)
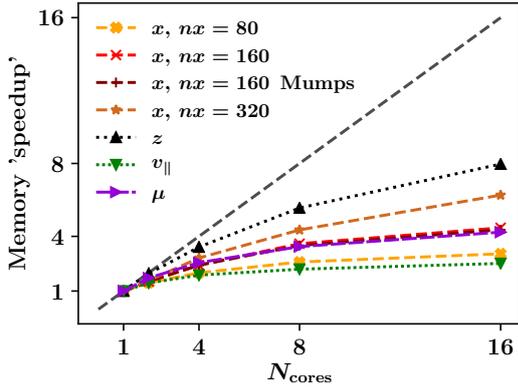
(b)
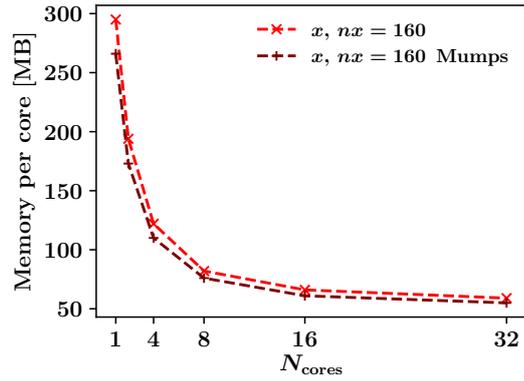
(c)

Figure 3.9: Memory scaling of GENE-3D. The memory usage is displayed for varying number of points in directions $x, y, z, v_{\parallel}, \mu$ in (a). The memory usage is linear for $z, v_{\parallel}, \mu$. The memory usage depends quadratically on the number of points in x and y-direction as the matrices become more dense with an increasing number of points per gyroradius ρ^*. The memory "speedup" is shown in (b). The scaling gets worse for a small number of points per core in all directions. The z-direction is the most efficient in splitting memory across processors, while the $v_{\parallel}$-direction is the least efficient. For scaling purposes, it does not matter whether SuperLU or MUMPS is used. The memory usage per core for an increasing number of cores in x-direction is shown in (c) for SuperLU and MUMPS. While MUMPS always uses less memory, the difference to SuperLU gets smaller for a larger number of processors.

displays perfect linear scaling. Hence, the memory "speedup" is plotted in Fig. 3.9b. The number of points in $z, v_{\parallel}$ and μ are set to 32 for this test. Parallelizing in z-direction is the most effective in terms of distributing memory as one gyro-matrix and field matrix is stored per point in z-direction. The $v_{\parallel}$-direction is the least effective as neither the gyro-matrices nor the field matrices depend on $v_{\parallel}$. The x-direction becomes more effective as the number of points per core increases. While the rows of PETSc matrices can be stored separately in the xy-communicator due to the PETSc "MPIAij" format, data still needs to be communicated and stored for matrix operations. The choice of the solver does not play a role in the memory "speedup". The μ direction also scales relatively well as also one gyromatrix is stored per point in μ-space. However, for the memory needs of the field matrix, the μ-parallelization does not help as the fields only depend on spatial coordinates x, y, z, and the velocity and species dimension are integrated over in the setup of the field matrix (see Eq. (3.28). Fig. 3.9c shows the memory usage per core for a simulation run both with MUMPS and SuperLU. While the memory usage is always less for the simulations using the MUMPS solver, the difference to SuperLU gets smaller with an increasing number of cores.

Optimization GENE-3D still lacks features of the x-global GENE version like electromagnetic effects, the parallel nonlinearity and collisions. These features are all demanding in performance, and the code needs to be further optimized. This paragraph outlines some ideas on how to improve the performance of GENE-3D. The main problem in terms of performance is the field solver. Direct LU solvers do not scale well enough to utilize present and future supercomputers fully. The preconditioner needed for iterative solvers, however, needs to be adapted to the problem at hand. For example, the standard Jacobi preconditioner included in PETSc usually fits very well to the field matrices of adiabatic electron simulations. In contrast, the same preconditioner slows down kinetic electron simulations to a point where direct solvers are more efficient even for highly parallelized simulations.

The modern standard approach to this type of problem is a multigrid (MG) solver. If n is the number of rows in the field matrix, then the multigrid solvers can solve the system in $\mathcal{O}(n)$ steps. This is done by coarsening the problem on multiple levels until the problem is small enough to solve it with a direct solver efficiently. On the finer levels, an iterative solver is used. Hence, preconditioners are still needed, and the algorithm needs to be adapted to the problem at hand. Implementing an MG solver could be an exciting future project as such an approach can gain performance and scalability. Multigrid solvers also interface to PETSc, so the codebase does not need to be changed significantly.

Another way to optimize the code further is porting the field solver to the GPU as future exascale computers will have a heterogeneous hardware architecture. GPUs

are optimized for matrix-vector multiplications, making them ideal for iterative linear equation solvers. While the GPU calculates the fields, the CPU could already calculate the right-hand side terms, which do not depend on the fields to parallelize the code further.

Another way of optimizing the code is to use lossy compression algorithms to reduce MPI communications costs and possibly also gain single-core performance as more data fits into the caches. Furthermore, block-structured grids have successfully been tested in the x-global GENE version [93, 94, 95], which helps to reduce the number of points necessary in velocity space. Block-structured grids could be adapted and utilized for GENE-3D.

Also, currently, the cache blocking mechanisms for the right-hand side computations are not adjusted ideally to GENE-3D. Here, single-core performance can be gained.

Conclusion

This section has described the development of GENE-3D into a production code ready for investigating plasma microturbulence in 3D geometries. The different development stages have been described and performance measurements have been presented showing the suitability of GENE-3D for present-day supercomputers. In summary, the foundation for the studies of the next sections has been laid.

3.3   The production chain around GENE-3D

The idea during the development of GENE-3D was to not only develop a code but an integrated code package with a working production chain from the geometric processing for all kinds of geometries to the gyrokinetic simulation with GENE-3D to the postprocessing and visualization of simulation results using Python [96] and Paraview [97]. This ensures that someone wanting to use GENE-3D also has access to all the software which GENE-3D is dependent on to work. Hence, GVEC [55] and the python diagnostic are included as git submodules into the GENE-3D repository. This integration is a substantial advantage over the y-global version of GENE, which needed the GIST interface to read in VMEC geometry files. As GIST is not publicly available, not everyone who has access to y-global GENE can use it.

In the following, the GVEC code and its interface to GENE-3D are described in Sec. 3.3.1. The python diagnostics used as postprocessing are outlined in Sec. 3.3.2. Lastly, the possibility of creating three-dimensional visualizations and videos through ParaView along with some example visualizations are shown in Sec. 3.3.3.

3.3.1   Geometry and interface to GVEC and VMEC

An MHD equilibrium is needed to supply GENE-3D with the geometric information of the stellarator or perturbed tokamak. For this purpose, an interface to the GVEC code is implemented in GENE-3D. The Galerkin Variational Equilibrium Code (GVEC) was recently developed by Florian Hindenlang at IPP Garching. GVEC is an ideal MHD solver that works similarly to the well established VMEC code [98, 99]. GVEC works on the assumption that a set of nested flux surfaces exist. One of the main advantages over VMEC is the ability to discretize the radial direction with Spline Finite Element polynomials of arbitrary degree with an arbitrary point distribution. This discretization allows resolving the entire radial domain accurately from the axis to the edge with a reasonable number of points.

In the representation of the flux surface geometry by the Fourier modes $[\cos(m\theta - n\phi), \sin(m\theta - n\phi)]$ GVEC works similarly to VMEC. Also here, m and n denote the poloidal and toroidal mode number. To optimize the mode spectrum [100] an additional periodic variable $\lambda(\rho_{tor}, \theta, \phi)$ is introduced. It distinguishes the geometrical poloidal angle θ from the straight field line angle $\theta^* = \theta + \lambda$. The equilibrium is then found by minimizing the total energy via a variational formulation

$$W(R, Z, \lambda) = \int_V \left(\frac{B_0^2}{2\mu_0} + \frac{p}{\gamma - 1} \right) dV \,, \tag{3.57}$$

with the magnetic permeability of free space μ_0, the pressure profile p and the specific heat ratio γ which is an input parameter to GVEC. For a given safety factor and pres-

sure profile, the constraints for the minimization procedure are the unknowns (R, Z, λ) describing closed nested flux surfaces. This is based on the assumption of a fixed geometry at the last closed flux surface with a given toroidal flux $\Psi_{\text{tor}}(\text{edge})$. An iterative preconditioned gradient descent algorithm is used to solve Eq. (3.57). The final result of the iteration is then written to a GVEC state file.

During the initialization of a GENE-3D simulation, the GVEC state file is then evaluated at the specific positions in space given by the number of points in (x, y, z) direction in GENE-3D. For every point in the GENE-3D grid GVEC calculates the value of the magnetic field B_0, the derivative of the absolute magnetic field $\nabla |B_0|$, and the gradient of the coordinates ∇s, $\nabla \theta$ and $\nabla \zeta$. All quantities provided by GVEC to GENE-3D are summarized here together with how they are used in GENE-3D

- Scalars

  - $\Psi_{\text{tor}}(\text{edge})$: toroidal flux at the edge needed for setup of radial grid (see (2.52)) and calculation of B_{ref} (see Sec. 3.1.1)

  - a: minor radius used as a normalization for many quantities. For stellarators $L_{\text{ref}} = a$ (see Sec. 3.1.1).

  - $\Psi'_{\text{tor}}(\text{edge})$: toroidal flux derivative needed for calculation of the sign of the poloidal magnetic flux at the edge in Eq. (2.48)

  - $q(\text{edge})$: safety factor profile at the edge needed for calculation of the sign of the poloidal magnetic flux at the edge in Eq. (2.48)

  - n_0: number of field periods used in the setup of the grid in y-direction (see Sec. 3.2.3)

- Radial Profiles

  - $q(x)$: needed for the boundary exchange in z direction as the phase shift is q dependent (3.44) and for the geometric factor C_{xy} appearing e.g. in the definition of B_0 in Eq. (2.53).

  - $p(x)$: needed to determine plasma β

  - dp/dx: The pressure gradient needed for electromagnetic simulations (not considered in this thesis)

- 3D quantities

  - $\mathbf{B}_0(x, y, z)$: the magnetic field playing a role virtually everywhere in the equations and in the code

  - ∇x, $\nabla \theta$, $\nabla \zeta$: needed for the calculation of g^{ij}, dB/du ($u = (x, y, z)$) and $J(x, y, z)$ in (2.58)

- $\nabla |B_0|$: needed for the calculation of dB/du

From these quantities GENE-3D calculates the relevant quantities necessary for the simulation:

- g^{ij}: the geometric factors used e.g. to calculate the curvature (Table 4) and in the setup of the gyromatrix (Eq. (3.10))

- dB/du: derivatives of the magnetic field used e.g. for the curvature (Table 4) and the trapping term in Eq. (3.4)

- $J(x, y, z)$: the Jacobian also used in a lot of parts of the code e.g. in the normalized Vlasov (3.4) equation and for the flux surface average (2.43)

- K_x and K_y: curvature terms in Vlasov equation (3.4) (see Table 4)

- β: The plasma β defined as the ratio of thermal pressure to magnetic pressure needed for electromagnetic simulations (not considered in this thesis)

and saves them to a separate file containing all relevant information in the GENE grid so if another simulation uses the same MHD equilibrium, grid and box sizes GVEC does not have to be run again.

3.3.2   Diagnostics

An extensive amount of time was spent developing postprocessing tools for GENE-3D output. One of the questions that need to be answered when plotting data is which color map to choose. The choice was made in favor of Matplotlibs colormap "viridis". It has the main advantages of accurately representing the data by being perceptually uniform. Meaning that it is linear in color space, so it is easy to estimate numerical values from the resulting image. It is still a decent colormap when printed in black and white and is also accessible to colorblind viewers. For further explanations about how this colormap was developed and a brief introduction into color theory, see Ref. [101].

Slice diagnostic   The slice diagnostic is developed to visualize slices of three-dimensional quantities like the electrostatic field potential $\Phi(x, y, z)$ in different planes. To this end, one of the three coordinates is fixed, and the other two are plotted in a plane with the field amplitude being color-coded. Examples of this diagnostic in this thesis are can be found in sections 5.1 and 5.2 e.g. in Fig. 5.10. Here, the diagnostic is used to visualize linear mode structures and nonlinear turbulent structures.

Growth rate diagnostic   Along with the mode structure, one of the main characteristics of a linear simulation is the growth rate. As linear simulations do not have any saturation mechanism, the modes grow uninhibitedly. A rescaling mechanism is built into GENE-3D to keep the numbers in the accurate floating-point range. Here, the entire distribution is divided by the maximum value of the distribution function if one of the moments of the distribution function has a maximum value larger than 10^{30}.

This gives the possibility to determine the growth rate in two different ways. For both, the maximum field value is used at all timesteps that write out the entire field data. In the first mechanism, the difference of the maximum field value between two time steps is divided by the time between the two field outputs.

$$\gamma_{\text{flying}}(t_i) = \frac{\max \Phi(t_i) - \max \Phi(t_{i-1})}{t_i - t_{i-1}} \ . \tag{3.58}$$

In the diagnostic this is called "flying" γ as it is a growth rate local in time which fluctuates strongly over time. The points in time before and after a rescaling are neglected for this diagnostic as they would lead to negative growth rates. The second way is a least squares fit of all the maximum field values in the time frame between rescaling points. This gives one growth rate value per rescaling time window and is hence subject to less fluctuation as it averages over a longer time frame. Fig. 4.13 shows an example of this diagnostic and shows the difference in the level of fluctuation between the two methods. A converged growth rate can be determined when both methods yield the same result. The growth rate diagnostic is used for all linear simulations in this thesis.

Frequency diagnostic   Another important diagnostic to quantify linear simulations is the frequency. The frequency diagnostic described here is the same as the one used by XGC and is developed for the benchmark described in Sec. 4.2.1. The algorithm works as follows.

- The position of the maximum field value is determined at the last time step.

- The time trace of the field value at this position is determined for every time window between two rescalings

- The field values are corrected by the average growth rate of the respective time window. The result is shown for the W-7X benchmark simulation with $\kappa_T = 4.0$ and $\eta_x = 1.0$, $\eta_y = 2.0$ in Fig. 3.10a

Then two different algorithms are used. A simple algorithm determines the time between two maxima of the corrected field. This is again done for the "flying" value and averaged over the entire rescaling range. The second algorithm is based on Fourier transforming the corrected field value. As the signal usually is not periodic, the Fourier frequency

is determined for all windows of the corrected field values from the complete interval between two rescalings down to only half of the time window. The final result then is the average of the maxima of the frequency spectra. Fig. 3.10b shows both the FFT algorithm and the maxima counting algorithm yield the same result if averaged over the rescaling time window. However, the accuracy of both these methods depends crucially on the output frequency of the field values. If the field value is not written out often enough, both diagnostics will yield lower frequencies due to aliasing effects. The maxima counting diagnostic usually yields the correct result for longer as the output frequency is lowered. The frequency is used for the benchmarks in sections 4.2.1, 4.2.2 and the

(a)   (b)
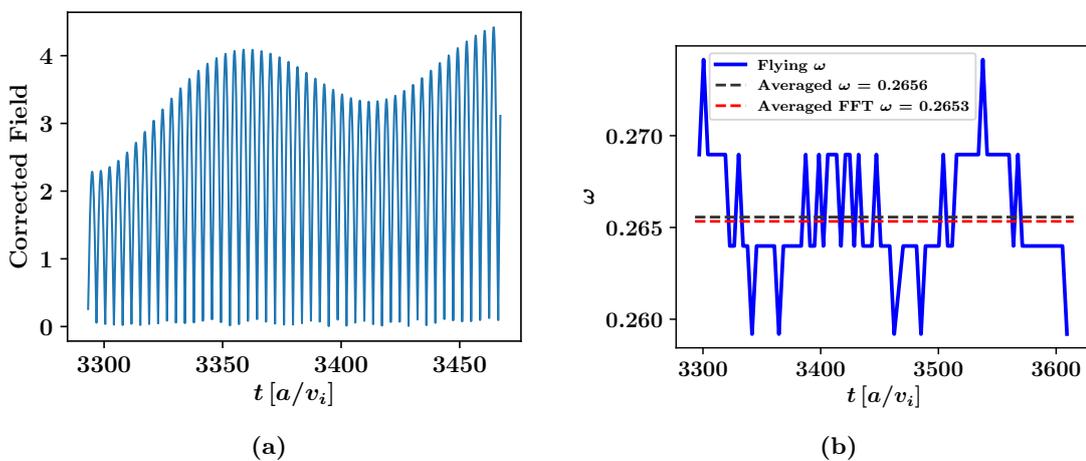
Figure 3.10: Frequency diagnostic. (a) shows the field value in a time frame between two rescalings corrected by the average growth rate in that time range. (b) shows the frequency determined both by determining the time between maxima of the corrected field and by Fourier transforming the corrected field signal. Both algorithms yield the result.

simulations in Sec. 5.2.

PEST grid diagnostics   The interpolation algorithm described in Sec. 3.2.4 is used to visualize the simulation results not only in the GENE coordinate system but also in PEST coordinates to compare to EUTERPE and XGC. Several diagnostics are based on the PEST grid. The slice diagnostic in the PEST grid allows looking at mode structures and turbulent structures. It is used in Sec. 4.2.1. Also, the poloidal and toroidal mode numbers n and m can be determined by Fourier transforming the field data for linear simulations and the density or heat flux for nonlinear simulations. This diagnostic is used in sections 4.2.1, 4.2.2, 5.1 and 5.2. Fig. 5.9 shows the mode spectrum for Wendelstein 7-X simulations. Additionally, the data in the PEST grid is used for the 3D ParaView visualization described in Sec. 3.3.3.

3.3.3   3D ParaView visualisation

The interface to GVEC and the interpolation to the PEST grid allow create 3D plots the GENE-3D simulation results using ParaView. These plots can help to gain insight about the geometric details of the machine that is being simulated. Also, plots and movies can be created of simulation data like particle densities or heat fluxes to get an intuition for the localization of linear modes and turbulent structures. Examples of such plots are given in this work in Fig. 1.5, Fig. 4.7, Fig. 4.14, and Fig. 5.31.

Summary

The GVEC code which serves as geometric preprocessing to GENE-3D has been described. Also the postprocessing tools have been described which serve to extract relevant information from GENE-3D simulations and to quantify and visualize it. The combination of these tools results in a production chain which is integrated into a git repository. Hence, everything necessary to run and analyze GENE-3D simulations is available to the user.

4   Code verification and benchmarks

In this chapter, verification studies between GENE-3D and other codes are presented. Sec. 4.1 compares GENE-3D with the radially global version of GENE for linear simulations with adiabatic and kinetic electrons in Sec. 4.1.1 and for turbulence simulations in Sec. 4.1.2. Part of these simulations are published in Ref. [31]. Sec. 4.2 presents a series of benchmarks carried out between GENE-3D, EUTERPE, and XGC. Here, extensive convergence tests are performed, and the influence of the hyper-diffusion terms are investigated. Finally, Sec. 4.3 studies the influence of numerical precision both on the numeric outcome of the simulations as well as their performance.

4.1   Benchmark of GENE-3D with the radially global version of GENE

4.1.1   Linear benchmarks

Linear ITG modes with adiabatic electrons in tokamak geometry   The first test performed is a well benchmarked [102] ITG mode case in a tokamak geometry with adiabatic electrons and $\rho^* = 1/180$.

(a)   (b)
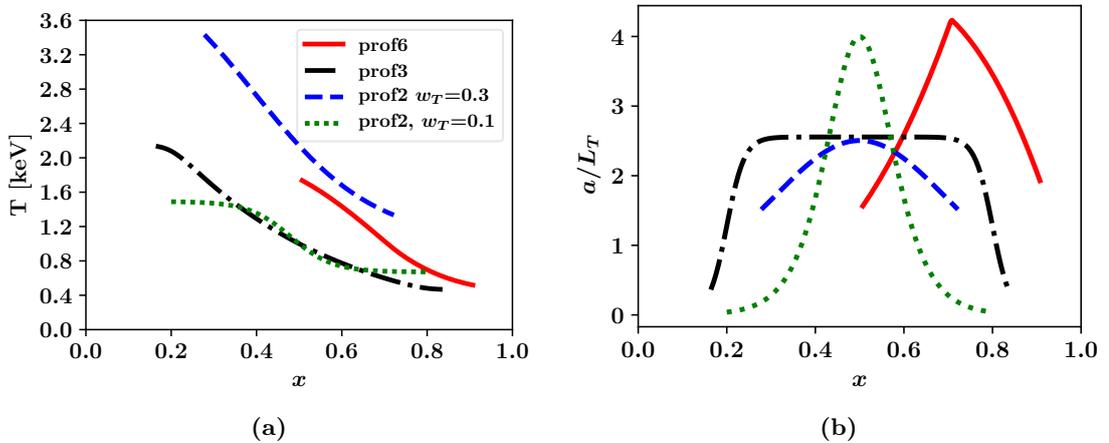
Figure 4.1: Temperature and temperature gradient profiles used for the benchmarks. The narrow profile 2_n with a width of $w_T = 0.1$ is used in the ρ^* scan in Sec. 5.2. The wider profile 2_w with a width of $w_T = 0.3$ is used in Sec. 4.1.1. Profile 3 is used in the nonlinear benchmark in Sec. 4.1.2 while Profile 6 is used in the W7-X benchmark in Sec. 4.2.1.

The choice of an axisymmetric geometry allows to compare the results of GENE-3D to the radially global GENE version. The simulations are performed in a simple tokamak geometry with circular, concentric flux-surfaces (see Fig. 4.2) and a safety factor profile defined as

$$q(x) = 0.86 - 0.16(x/a) + 2.52(x/a)^2 \,, \tag{4.1}$$
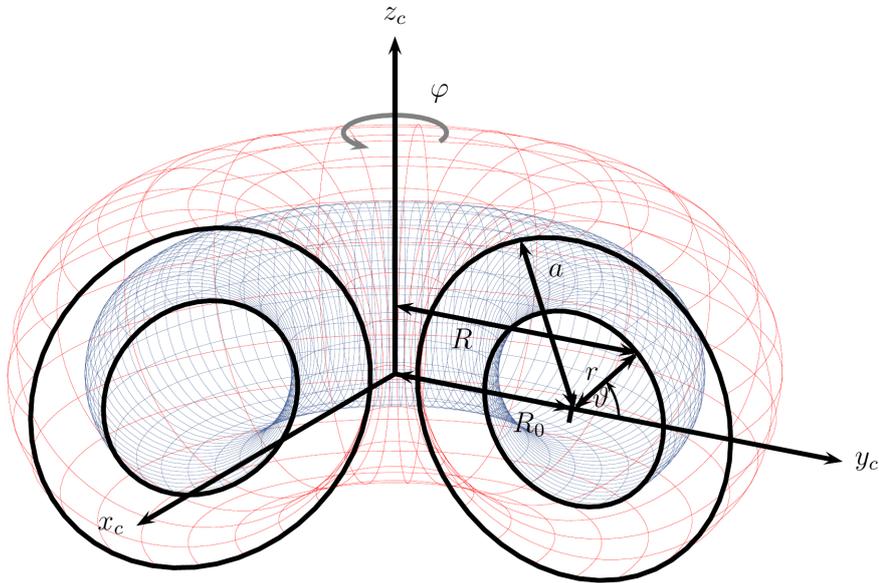
Figure 4.2: Visualization of the geometry with circular concentric flux surfaces with the minor radius a and the major radius R_0 Cf. [61].
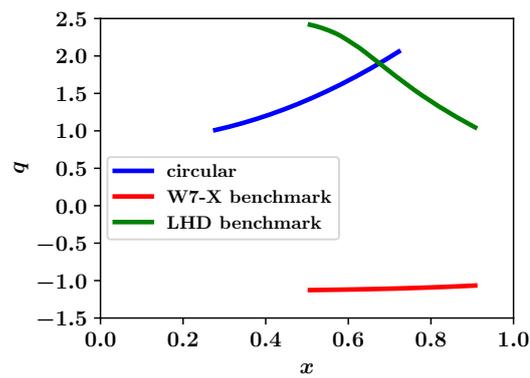
Figure 4.3: Safety factor q-profile for circular geometry of Sec. 4.1.1 and W7-X geometry of Sec. 4.2.1.

(see Fig. 4.3). The temperature profiles are the same for both ions and electrons,

$$T_{i,e}(x)/T_{\text{ref}} = \exp\left[-\kappa_T w_T \frac{a}{R_0} \tanh\left(\frac{x-x_0}{w_T a}\right)\right], \qquad (4.2)$$

where w_T is the characteristic profile width and x_0 is the gradient peak position (see Fig. 4.1). The density profile follows the same functional form with κ_n and w_n instead of κ_T and w_T. Here, a and R_0 indicate the minor and major radius of the machine. While the y-direction in radially global GENE is discretized in Fourier space (see Ref. [30]) and the distribution function and fields are therefore saved as complex numbers these quantities are all represented in real space in GENE-3D. Hence, different numerical schemes are used for the calculation of derivatives and the terms on the right-hand side of equation (3.4).

The radially global GENE version was benchmarked for this case against EUTERPE, ORB5, and GYSELA in Ref. [102] with overall excellent agreement. The results depicted in Fig. 4.4 show very good agreement between GENE-3D and x-global GENE. Thus, the axisymmetric limit can be well reproduced in GENE-3D. The converged resolutions are 320 x 16 x 30 x 32 x 32 in directions (x, y, z, $v_{||}$, μ) with normalized box lengths $(L_x, L_{v_{||}}, L_\mu) = (150, 3, 9)$ and maximum temperature gradient $\kappa_T = 6.96$ and maximum density gradient $\kappa_n = 2.23$.
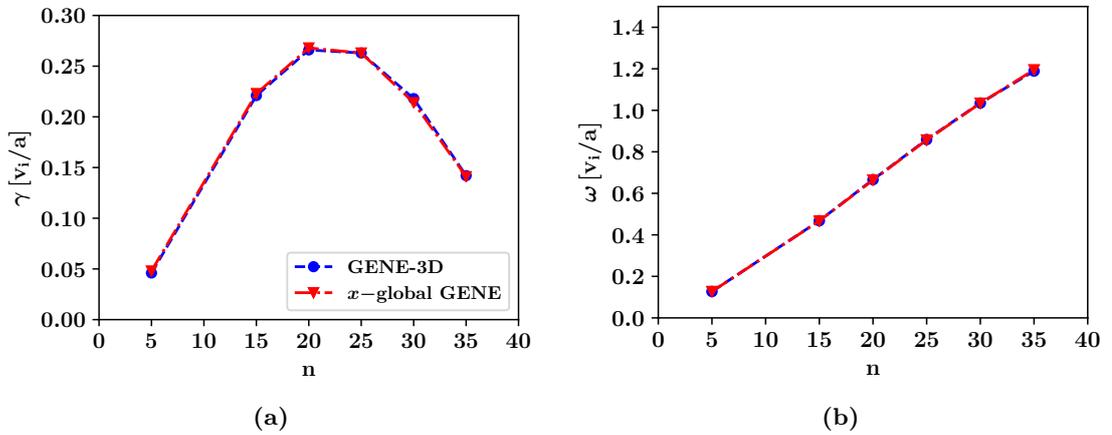
Figure 4.4: Growth rates γ and frequencies ω for ITG modes in a circular geometry with adiabatic electrons. The agreement between radially global GENE and GENE-3D is virtually perfect. Hence, the differences in algorithms do not affect the results.

Linear ITG modes with kinetic electrons and TEMs in tokamak geometry
Here, we retain electrons as a fully gyrokinetic species while still simulating in the axisymmetric geometry like before. This way, the behavior of ITG modes and trapped electron modes (TEMs) is investigated and compared to the results of x-global GENE.

The converged GENE-3D resolutions are $2560 \times 8 \times 30 \times 64 \times 32$ in $(x, y, z, v_{||}, \mu)$ directions, with normalized box lengths $(L_x, L_{v_{||}}, L_\mu) = (80, 3, 9)$. The simulations are run with $\kappa_T = 6.96$, $\kappa_n = 2.23$, $w_T = w_n = 0.3$, and $x_0 = 0.5$ for both species and $\rho^* = 1/180$. Fig. 4.5 compares the growth rates (a) and frequencies (b) of the most unstable mode for different toroidal mode numbers n. Two different branches can be observed. For toroidal mode numbers smaller or equal to 35 ($n \leq 35$) the modes have a positive frequency ω, which corresponds to a poloidal direction of propagation in the ion diamagnetic drift direction. Hence, the modes can be identified as ITG modes. For higher toroidal mode numbers, a negative frequency branch is found. A negative frequency corresponds to a propagation in the electron diamagnetic drift direction. Therefore, the modes can be identified as TEMs. Very good agreement of frequencies and growth rates between GENE-3D and x-global GENE is found for both branches including the transition point from ITGs to TEMs.
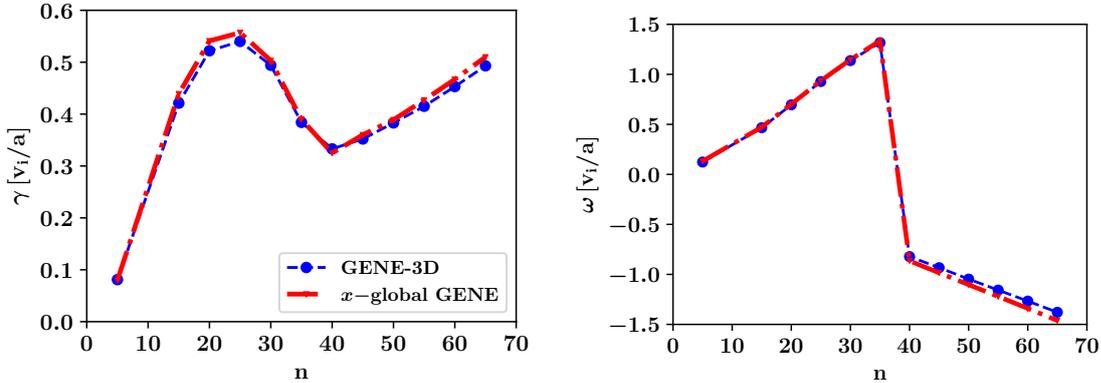
Figure 4.5: "Linear benchmark between GENE-3D and the global tokamak version of GENE for ITG modes with kinetic electrons and TEMs a circular tokamak geometry: (a) Growth rates γ and (b) real frequencies ω_r as a function of the toroidal mode number n" Cf. [31].

4.1.2   ITG turbulence with adiabatic electrons in tokamak geometry

Here, ITG turbulence with adiabatic electrons in a circular axisymmetric geometry with $\rho^* = 1/180$ is studied both with GENE-3D and x-global GENE. The temperature and density profiles are defined as

$$T(x)/T_{\text{ref}}, n(x)/n_{\text{ref}} = \left[\frac{\cosh\left(\frac{x - x_0 + \Delta_{T,n}}{w_{T,n}}\right)}{\cosh\left(\frac{x - x_0 - \Delta_{T,n}}{w_{T,n}}\right)} \right]^{-0.5\kappa_{T,n} w_{T,n} a/R_0}, \tag{4.3}$$

with $\kappa_T = 7.1$, $\kappa_n = 2.2$, $w_T = 0.04$, $x_0 = 0.5$, and $\Delta T = \Delta n = 0.3$ (see Fig. 4.1). The safety factor profile is given by

$$q(x) = 0.85 - 0.01(x/a) + 2.28(x/a)^2 - 0.09(x/a)^3 + 0.22(x/a)^4 \,, \qquad (4.4)$$

(see Fig. 4.3). A Krook-type heat source with $\kappa_H = 0.035$ is used to maintain the profiles and reach a quasi-steady state.

The simulation is performed with $120 \times 256 \times 16 \times 64 \times 24$ grid points in $(x, y, z, v_{\parallel}, \mu)$ directions, with normalized box lengths $(L_x, L_y, L_{v_\parallel}, L_\mu) = (120, 132, 4, 16)$. The time traces of the volume-averaged (electrostatic) ion heat flux Q_σ^{es} are depicted in Fig. 4.6 in GyroBohm (GB) units, which is defined as

$$Q_i^{es}/Q_{\mathrm{GB}} = \frac{1}{2m_\sigma} \int f_{1,i}(\mathbf{x}, \mathbf{y}) \, \mathbf{v}_{E_1} \cdot \nabla x \, d\mathbf{v} \,, \qquad (4.5)$$

where $f_{1i}(\mathbf{x}, \mathbf{y})$ is the perturbed part of the ion *particle* distribution function and $Q_{\mathrm{GB}} = n_{\mathrm{ref}} \, T_{\mathrm{ref}} \, v_i \, (\rho_i / L_{\mathrm{ref}})^2$. (For the circular geometry considered here, $L_{\mathrm{ref}} = R_0$, while for the W7-X simulations shown below, $L_{\mathrm{ref}} = a$.) The time averaged heat flux, computed over the second half of the saturated state (indicated with a straight solid line in Fig. 4.6), is essentially the same for both codes: $Q_i^{es}/Q_{GB} = 22.4 \pm 6.2$ for GENE-3D and $Q_i^{es}/Q_{GB} = 21.5 \pm 5.7$ for x-global GENE. The uncertainty is calculated as the standard deviation of the heat flux in the time interval used for averaging. The comparison can therefore be considered successful.
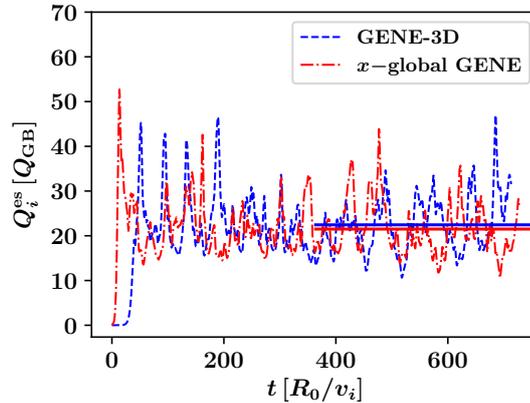
Figure 4.6: Nonlinear benchmark between GENE-3D (blue) and x-global GENE (red) for ITG turbulence with adiabatic electrons in tokamak geometry. Shown are the time traces of the volume averaged heat flux in units of Q_{GB}. Solid lines indicate time averaged values Cf. [31].

Conclusion

In this section benchmarks between GENE-3D and the x-global GENE version have been performed in a tokamak geometry. The tests have considered linear physics employing both adiabatic and kinetic electrons as well as nonlinear dynamics employing kinetic electrons. All benchmarks have successfully shown that GENE-3D works correctly in the axisymmetric limit.

4.2 Inter-code comparison between EUTERPE, GENE-3D and XGC

This section summarizes the results of a series of GENE-3D simulations performed for benchmark purposes with EUTERPE and XGC for ITG-driven modes in a Wendelstein 7-X equilibrium in Sec. 4.2.1 and a LHD equilibrium in Sec. 4.2.2. In the first step of this inter-code comparison endeavor, the simulations are linear, adiabatic, and electrostatic.

4.2.1 Linear ITG modes with adiabatic electrons in a Wendelstein 7-X geometry

This benchmark is done for the Wendelstein 7-X geometry created from the VMEC file "wout_w7x2_rz300p128.nc" and visualized in Fig. 4.7 For this benchmark the temperature gradient κ_T is varied between 1.0 and 5.0. The VMEC file used for the benchmark is wout_w7x2_rz300p128.nc.
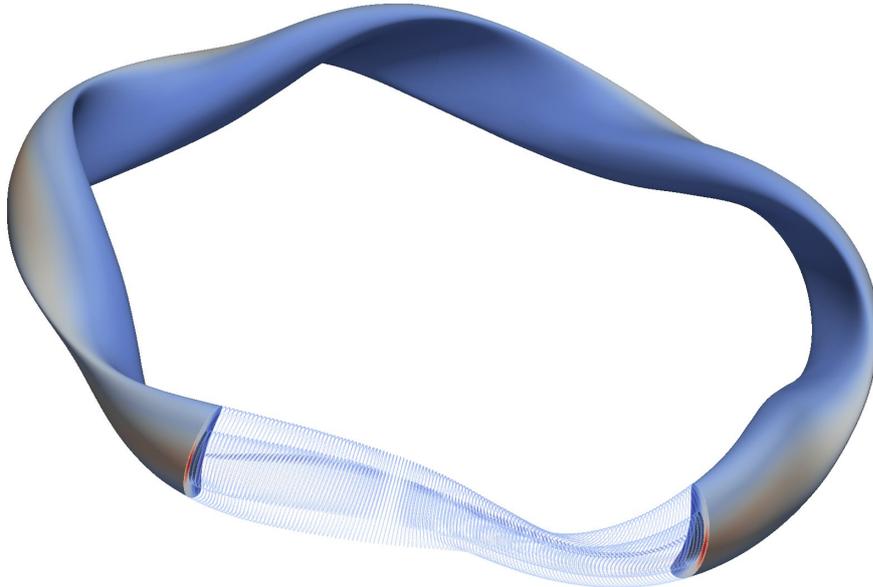
Figure 4.7: Visualization of the Wendelstein 7-X geometry using the ParaView diagnostic Cf. [31].

Convergence tests   In order to ensure the correctness of the GENE-3D results extensive convergence tests are performed in resolution, box sizes and hyper-diffusion terms before running the actual benchmark. The reference parameter set is outlined in the following:

parameter change from reference parameter file	growth rate γ
reference	0.190
$\mathtt{lx\_a} = 0.5$	0.191
$\mathtt{nx0} = 256$	0.192
$\mathtt{nx0} = 256, \mathtt{ny0} = 512$	0.191
$\mathtt{ny0} = 1024$	0.204 ($k_y = 4$ mode)
$\mathtt{nv0} = 96$	0.191
$\mathtt{nv0} = 96, \eta_x = 2, \eta_y = 2$	0.163
$\mathtt{nv0} = 96, \eta_x = 2, \eta_y = 2, \eta_z = -1$	0.161
$\mathtt{nv0} = 96, \mathtt{lv} = 4$	0.191
$\mathtt{nv0} = 96, \mathtt{lv} = 5$	0.191
$\mathtt{nw0} = 30$	0.191

Table 6: Convergence test for Wendelstein 7-X benchmark. Only high-k modes and hyper-diffusion terms have an effect on the growth rate.

```
! resolutions          ! box sizes            ! hyper-diffusions
n_spec =    1          lv    = 3.28633        η_x = 0.0
nx0    =  128          lw    = 10.8000        η_y = 0.0
ny0    =  384          x0    = 0.70710        η_z = 2.0
nz0    =  120          lx_a  = 0.4            η_v = 2.0
nv0    =   48
nw0    =   10          .                      .
```

Convergence tests are performed for the case of $\kappa_T = 3.0$. The results are summarized in Table 6. For very high resolutions in y-direction high-k_y modes are observed. The other two codes are looking at the lower-k_y mode. Hence, the resolution cannot be set to high values in order to limit the maximum k_y present in the system. Sec. 5.2 investigates and discusses high-k_y modes in more detail. In this convergence study, the hyper-diffusion terms (see Eq. (3.54)) are the only parameters significantly affecting the result. Hence, another series of convergence tests are performed to investigate the effects of hyper-diffusion in all applicable dimensions. In a previous study in a different parameter regime $\eta_y = 2.0$ was found to be a reasonable value for the y-direction and therefore chosen for the subsequent tests (see fig 4.8).
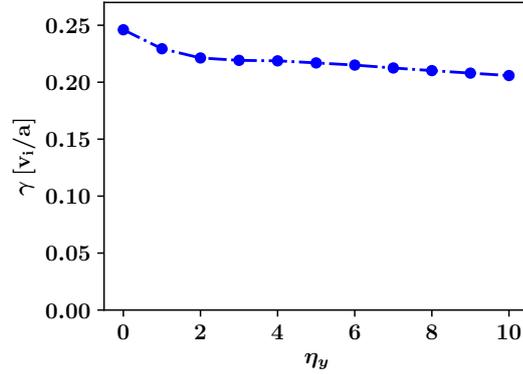
Figure 4.8: Hyper-diffusion scan in y direction from a previous study with a different parameter setup. After an initial jump from no hyper-diffusion to $\eta_y = 2$ the growth rate plateaus.

Hyper-diffusion in x-direction   The scan in the hyper-diffusion amplitude in x is presented in Fig. 4.9. For hyper-diffusion in x direction a small jump in growth rate from $\eta_x = 0$ to $\eta_x = 1$ is observed. However, without hyper-diffusion, the simulations converge on much longer time scales (see Fig. 4.9b). After $4000\,a/v_i$ the mode structure actually gets closer (from $k_x = -0.7$ to $k_x = -0.6$) to the ones observed for finite hyper-diffusion amplitudes. Conversely, the wavenumber for the simulation at $\eta_x = 1.0$ also converges at $k_x = -0.6$ when the resolution in x-direction is doubled. This is contrary to the observation in Table 6 where the resolution did not matter for $\eta_x = 0$. Also, in this scan doubling the resolution for the $\eta_x = 0$ case changes neither mode structure nor growth rate. This can be explained by the fact that hyper-diffusion is most influential on the highest k modes. By doubling the resolution, also the wavenumber of the highest k mode is doubled. Therefore, there have been efforts in the past to come up with models to have a resolution-dependent hyper-diffusion amplitude [89, 88]. A value of around $\eta_x = 1.0$ seems reasonable for future simulations as the value is high enough to get relatively quick convergence but not too high to over-damp the fundamental physics in the system.
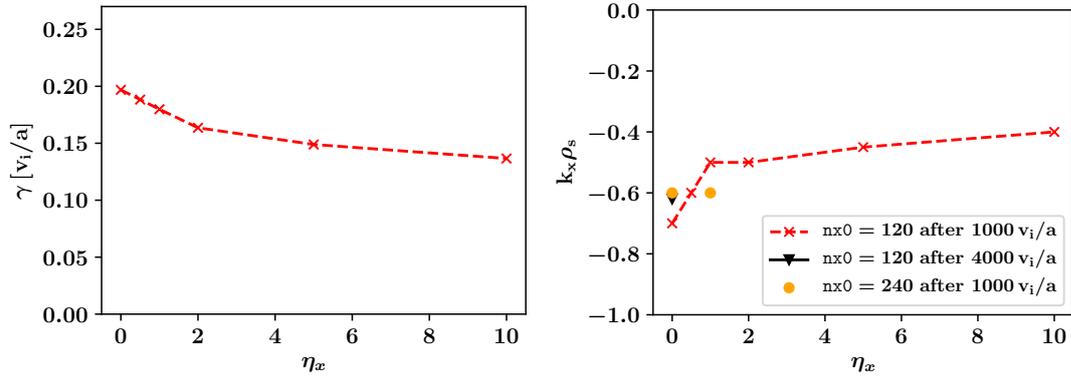
(a) After an initial jump from no hyper-diffusion to $\eta_x = 1$ the growth rate plateaus. Doubling the resolution at $\eta_x = 1$ only has a small effect of around 2% on the growth rate.

(b) Also the mode structure can change due to hyper-diffusion. After the jump to $\eta_x = 1$ the mode structure remains relatively constant. When run for 4 times longer k_x moves up from $k_x = -0.7$ to $k_x = -0.6$. This long time scale to convergence is only observed for hyper-diffusion amplitudes close or equal to zero. Conversely, doubling the resolution in x-direction for the $\eta_x = 1$ simulation lowers k_x from $k_x = -0.5$ to $k_x = -0.6$.

Figure 4.9: Hyper-diffusion scan in x direction.

Hyper-diffusion in v- and z-direction Next are the hyper-diffusions in v and z direction. Fortunately, these free parameters have no influence whatsoever on the growth rate or mode structure, as shown in figures 4.10a and 4.10b.
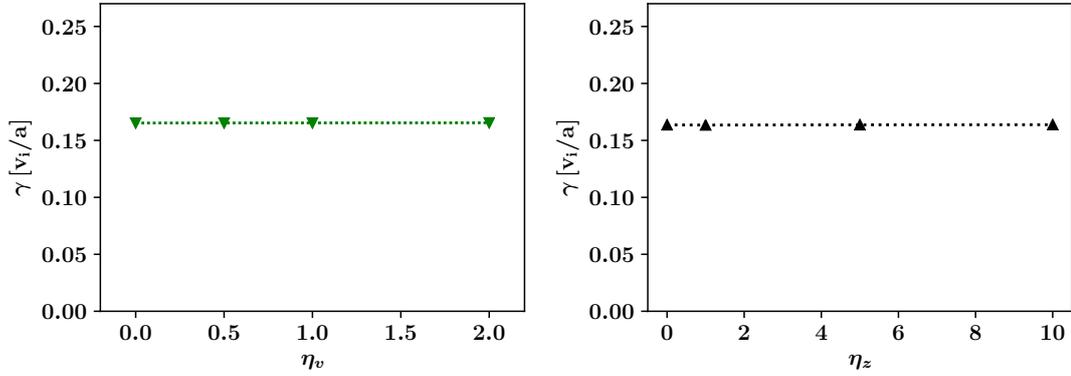
(a) Hyper-diffusion scan in v direction: The hyper-diffusion amplitude in v direction has no significant effect on the simulations.

(b) Hyper-diffusion scan in z direction: The hyper-diffusion amplitude in z direction has no significant effect on the simulations.

Figure 4.10: Hyper-diffusion scan in v and z direction: Neither one has a significant effect on growth rate or mode structure.

Hyper-diffusion in y-direction With the knowledge of sensible values for $\eta_x = 1$, $\eta_v = 0.2$, $\eta_z = 2.0$ the y-direction is rechecked. The results are shown in figures 4.11 and 4.11b. Similarly to the first scan, the main differences occur between $\eta_y = 0$ and $\eta_y = 2$. Hence, the choice of $\eta_y = 2$ is confirmed.
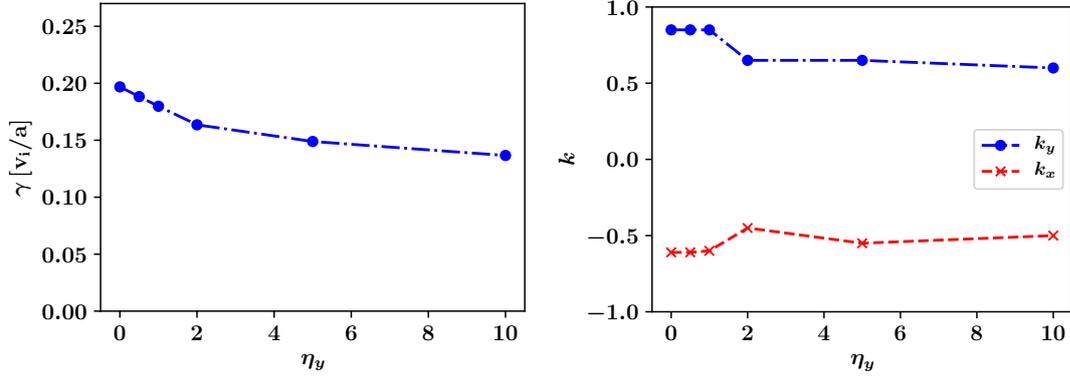
(a) Second hyper-diffusion scan in y direction: After an initial jump in growth rate from $\eta_y = 0$ to $\eta_y = 2$ the decline in growth rate saturates similarly to the first time (Fig. 4.8)

(b) Second hyper-diffusion scan in y direction: As for hyper-diffusion in x the hyper-diffusion in y also affects the mode structure. However, after an initial change from $\eta_y = 0$ to $\eta_y = 2$ the changes are marginal.

Figure 4.11: Hyper-diffusion scan in y direction.

Hence, the final values for the hyper-diffusions are $\eta_x = 1$, $\eta_y = 2$, $\eta_z = 2.0$, $\eta_v = 0.2$. As the hyper-diffusion amplitudes are still free parameters and there is no rigorous rule on how to set them, error bars are added, representing the possible changes in results due to different hyper-diffusion amplitudes in Fig. 4.12. The minimum hyper-diffusion set is $\eta_x = 0$, $\eta_y = 0$, $\eta_z = 2.0$, $\eta_v = 0.2$ while the maximum set is $\eta_x = 2.0$, $\eta_y = 2.0$, $\eta_z = 2.0$, $\eta_v = 0.2$.

Final results   With the convergence tests done, this paragraph shows the final results of the benchmark between EUTERPE, GENE-3D, and XGC. Jörg Riemann provided the EUTERPE data points in private communication while Michael Cole provided the XGC data. The linear growth rates, frequencies, and mode numbers are shown in Fig. 4.12. For the growth rate, γ good agreement is found between all three codes for all values of κ_T considered here. For the frequency, ω XGC and GENE-3D agree very well, while the EUTERPE results differ. This might be due to different algorithms used for determining the frequency. GENE-3D and XGC use the algorithm described in Sec. 3.3.2. The algorithm used by EUTERPE is not public at the time of writing this thesis. The mode numbers m and n again agree very well between XGC and GENE-3D, while the EUTERPE mode numbers are roughly 10% higher. The different hyper-diffusion amplitudes only have a small effect on the final result for growth rate and mode number, while the effect is practically zero for the frequency. Hence, for future simulations the effort of running simulations multiple times does not have to be repeated as the influence

of hyper-diffusion on the final result is on the same order of magnitude as the fluctuations within a simulation (see Fig. 4.13).
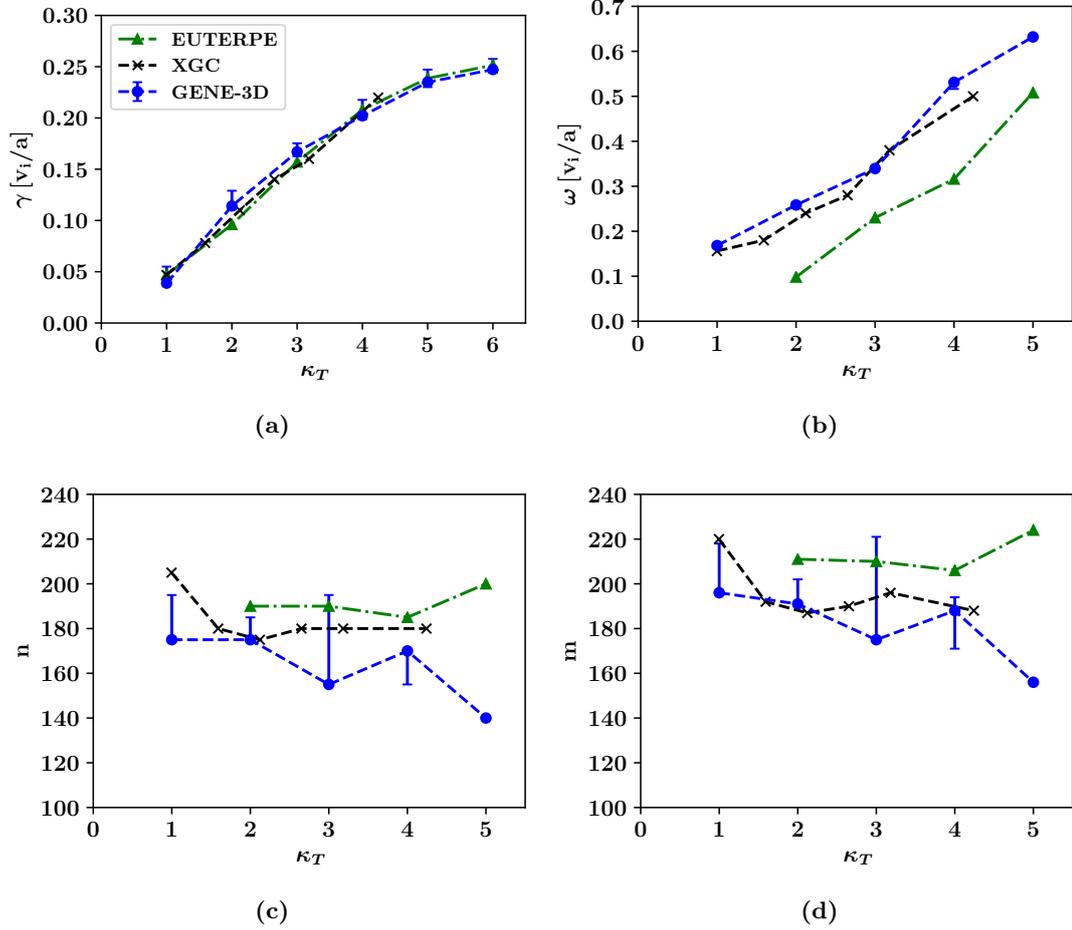
Figure 4.12: Linear benchmark between GENE-3D (blue circles) and EUTERPE (green triangles) for ITG modes with adiabatic electrons in a W7-X geometry. Depicted are a) growth rates γ, b) frequencies ω, and c,d) Mode numbers n,m, as a function of the logarithmic temperature gradient κ_T. The error bars denote the difference in results due to different hyper-diffusion amplitudes.
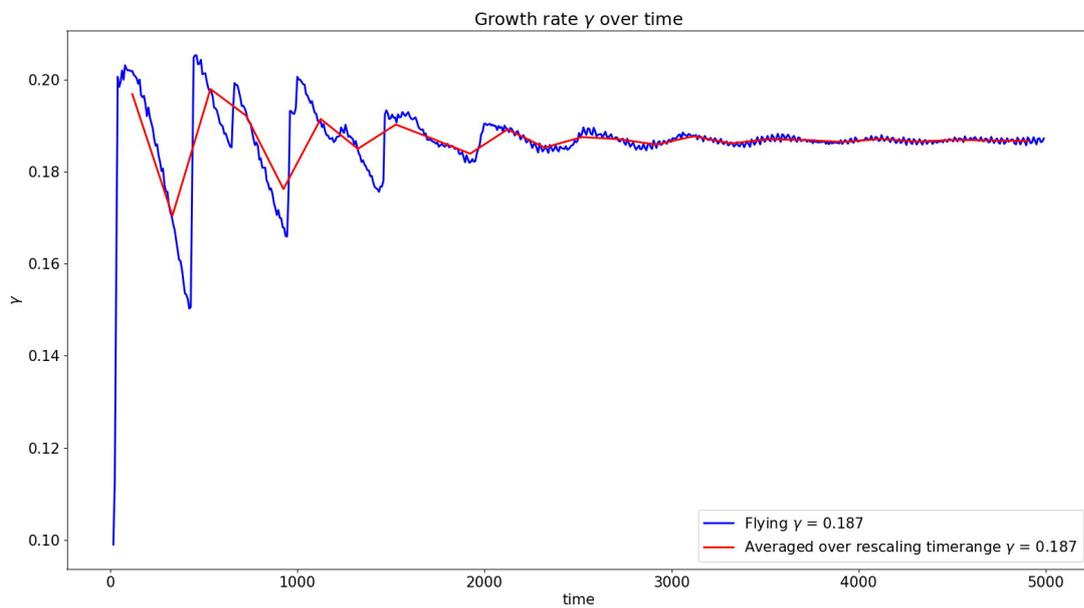
Figure 4.13: Growth rate diagnostic over time. The "flying" γ (blue) is calculated as the growth rate between two write-outs of the field diagnostic. The "averaged over rescaling time range" uses a least squares fit algorithm to calculate the growth rate between two rescaling points of the simulation. The result converges slowly over time. Simulations have to be run long enough to get an accurate result, or the result should be presented with an error bar.

4.2.2   Linear ITG modes with adiabatic electrons in an LHD geometry

In this subsection, a benchmark between EUTERPE and GENE-3D is presented for linear ITG modes in an LHD geometry employing adiabatic electrons. The temperature and density profiles used are profile 6 (see Fig. 4.1), same as in the W7-X benchmark in the previous subsection. The temperature gradient is varied between $\kappa_T = 1.5$ and $\kappa_T = 3.5$. The VMEC file used in this benchmark is `wout_lhd15_rz300p128.txt`. The geometry is visualized in Fig. 4.14. Before the GENE-3D results are compared to the EUTERPE results, convergence tests are performed for GENE-3D to ensure the correctness of the results.
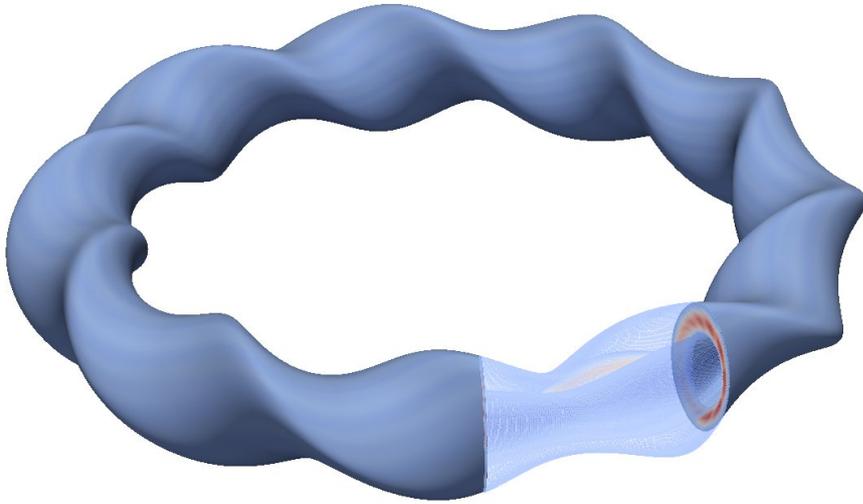
Figure 4.14: Visualization of the LHD geometry used in the benchmark with EUTERPE.

Convergence tests   The base parameter file is as outlined in Table 7.

! resolutions		! box sizes		! hyper-diffusions
n_spec =	1	lv	= 3.29	$\eta_x = 1.0$
nx0	= 128	lw	= 10.8	$\eta_y = 1.0$
ny0	= 400	x0	= 0.70710	$\eta_z = 2.0$
nz0	= 128	lx_a	= 0.4	$\eta_v = 2.0$
nv0	= 48			
nw0	= 10	.		.

Table 7: Reference parameters for LHD convergence test.

The extensive convergence studies for the hyper-diffusion terms in the previous subsection are not repeated here. Rather, the values outlined in Table 7 are used. The tested resolutions and the growth rates of the simulations are summarized in Table 8.

parameter change from reference parameter file	growth rate γ	mode number n
reference	0.198	60
nx0 = 192	0.198	60
nz0 = 192	0.198	60
nw0 = 20	0.195	60
nv0 = 96	0.198	60
ny0=200, nv0=92, nw0=32, lv=5, lw=25	0.198	60
nx0=240, ny0=120, nz0=256, nv0=64, nw0=32, lv=4, lw=16, lx_a=0.5	0.198	60

Table 8: Parameters of convergence test for LHD benchmark.

As none of the different parameter sets seemed to have an effect neither on growth rate nor on mode number the final parameter set is chosen to be the same as the reference parameter set with the only difference of $ny0 = 200$ to save computation time.

Final results Fig. 4.15 shows the final results of the benchmark between EUTERPE and GENE-3D. Overall good agreement is found for the growth rates with the exception of a 10% difference at $\kappa_T = 3.5$. For the frequencies, the values of EUTERPE and GENE-3D do not agree, same as for the W7-X benchmark in the previous subsection. This is probably due to different algorithms for determining the frequency. The algorithm of GENE-3D is the same as the one used by XGC. It is described in Sec. 3.3.2. At the time of writing this thesis, the algorithm employed by EUTERPE is not public. For the mode numbers again, good agreement is found. While a difference of 10 in mode number might seem like a lot, the fact that for LHD $n_0 = 10$ puts this into perspective. Mode numbers are calculated as $n = j \cdot n_0$ with j being and integer number and hence a difference of 10 means it is the next mode.
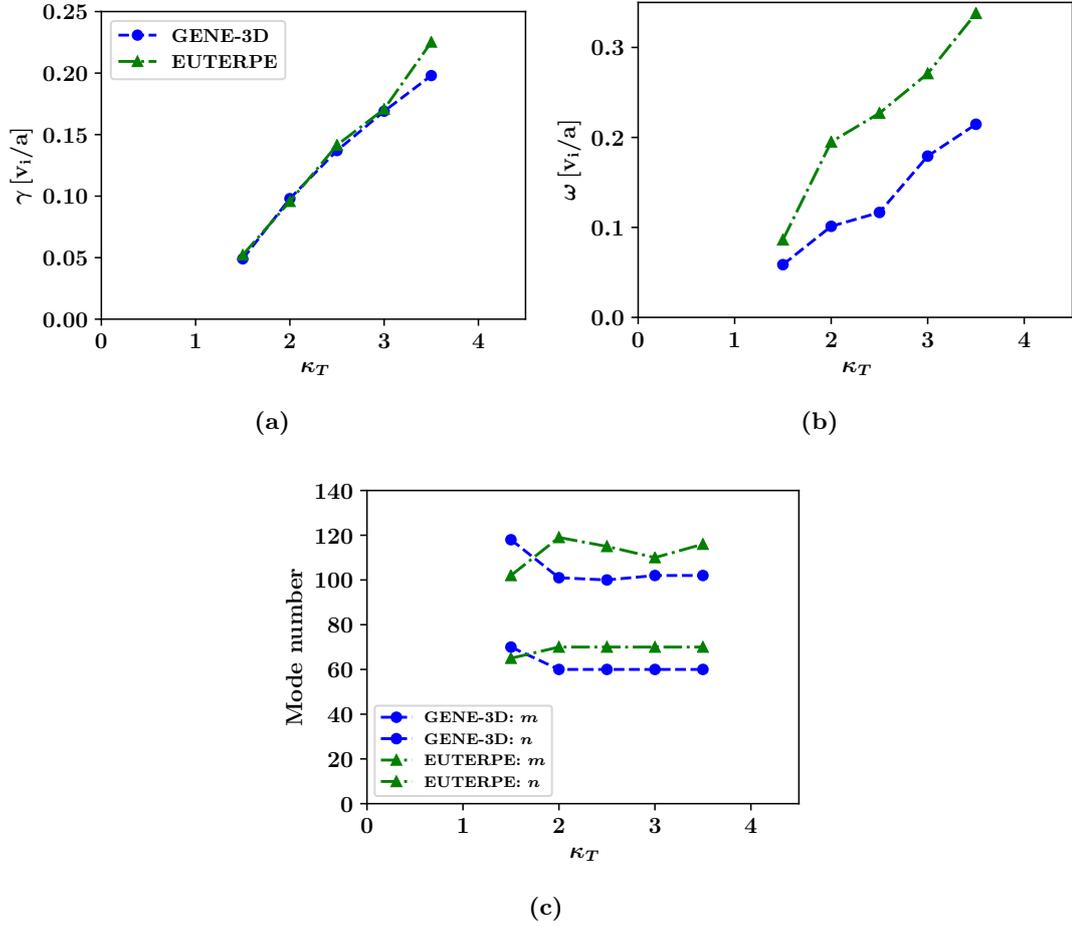
Figure 4.15: Linear benchmark between GENE-3D (blue circles) and EUTERPE (green triangles) for ITG modes with adiabatic electrons in a LHD geometry. Depicted are a) growth rates γ, b) frequencies ω, and c) Mode numbers m,n, as a function of the logarithmic temperature gradient κ_T.

Conclusion

In this section linear benchmarks have been carried out between GENE-3D, EUTERPE, and XGC in both a Wendelstein 7-X and an LHD geometry. With exceptions for differences due to different algorithms used to determine the frequency the codes agree for growth rate, frequency and mode numbers. Thus, GENE-3D can be considered fully operational for linear and nonlinear studies of 3D geometries with one or multiple species.

4.3   Influence of numerical precision

Numerical precision can not only determine the accuracy to which variables can be stored but also greatly influence performance. When switching from double to single precision the floating point variable size is halved. Not only does this reduce the memory requirements by 50% but also loading data from the RAM through the Cache cascade can theoretically be twice as fast. Furthermore, CPUs like the Intel Skylake architecture have vectorization units with a fixed register size. Hence, instruction sets like the Advanced Vector eXtensions (AVX)-512 can process twice as many variables simultaneously. Parallel performance can also be improved as less data is communicated.

Here, GENE-3D is run in both double and single precision and tested for performance and correctness of the physical results.

The time traces of the electrostatic heat flux of a single precision nonlinear simulation (green curve) and a double-precision run (blue curve) are depicted in Fig. 4.16. The same physical case is considered here as the one described in Sec. 4.1.2. In terms of their
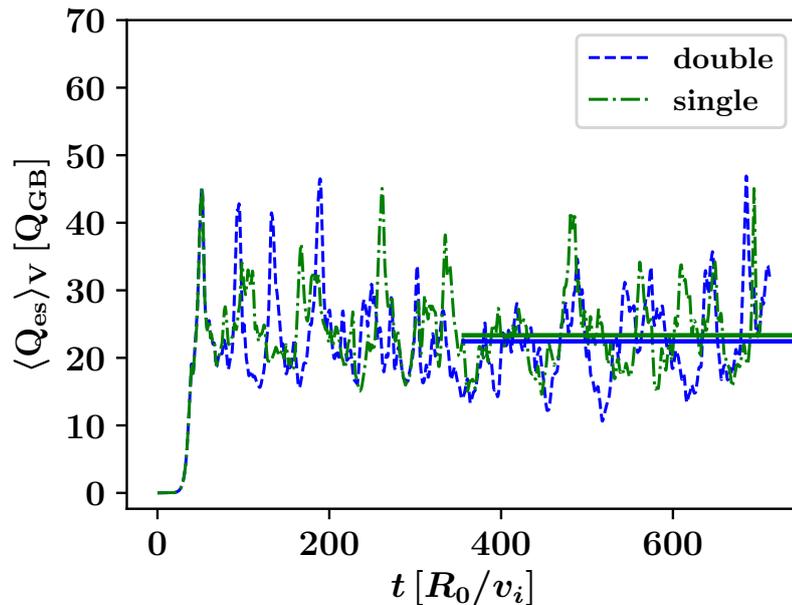
Figure 4.16: Time trace of the heat transport for GENE-3D in double precision (blue) and in single precision (green). The results statistically equivalent Cf. [31].

statistics, the simulations are equivalent, $Q_{single}^{es} = 23.3 \pm 5.4$ and $Q_{double}^{es} = 22.4 \pm 6.2$ (in Q_{GB} units). Hence, single precision is sufficient for the parameters considered here. Also in other cases examined during this project the results remained the same. Performance measurements are implemented to test run time, load imbalance, and call count for each part of GENE-3D (see Sec. 3.2.7). Therefore, the simulations can be compared in terms of their performance.

Using single precision, the average time to calculate an entire timestep takes 65% of the time it takes using double precision. Within one timestep, the $v_\parallel$ integration which consists mainly of MPI communication is reduced to 53%, the PETSc field solver which is a mix of computation and communication is reduced to 70%, and the right-hand-side computation is reduced to 72% of the runtime in double precision.

Concluding, the total performance is increased by a factor of 1.5 while the physical results are not affected greatly. This result motivates an ongoing project looking into lossy data compression techniques to further improve the performance of GENE-3D.

5   Impact of geometry variations on turbulent transport in Wendelstein 7-X

GENE-3D can - for the first time in the GENE family - simultaneously take three dimensional variations of the magnetic field geometry into account. This allows testing the impact of geometric variations on the linear stability of modes and nonlinear transport of heat. To this end, Sec. 5.1 examines the geometric variations due to three different coil optimization strategies. Here, the three magnetohydrodynamic (MHD) equilibria are studied for their geometric details. The effect of their differences onto linear stability with adiabatic and kinetic electrons and finally for nonlinear transport of heat is tested.

Furthermore, Sec. 5.2 investigates the system size effects. In this context, the differences between the flux surface (y-global) GENE version and GENE-3D are tested. Also, the geometrical drive due to bad curvature regions is studied linearly, and comparisons between linear and nonlinear simulations are performed.

5.1   Towards a stellarator optimized for turbulent transport

As neoclassical transport is suppressed in present day stellarators like Wendelstein 7-X turbulent transport becomes more important [103]. Optimizing future devices also in regard to turbulent transport is, therefore, crucial to improving plasma confinement. As the first step in this direction, this section investigates the influence of geometric variations of the original high-mirror Wendelstein 7-X configuration due to different coil optimization strategies on the simulation results of GENE-3D. The corresponding gyrokinetic results obtained in this thesis project and presented in this chapter have also been partially published in Ref. [104]. The three different equilibria are the target magnetic field, which is produced by a smooth current distribution and equilibria produced by two different coil configurations from the coil optimization tool ONSET [43] and its stochastic extension [53].

The "target" magnetic field   is derived from the original W7-X plasma boundary that became the high-mirror configuration of the first version of the W7-X configuration space [105]. The main plasma quality criterion is the field error defined as $\epsilon_f = \mathbf{B} \cdot \mathbf{n}$ with the magnetic field vector $\mathbf{B}$ and the normal on the plasma boundary $\mathbf{n}$. The target magnetic field can be considered "ideal" from a coil optimization point of view because of its negligible field error. Since a current distribution produces it, it cannot be realized by a finite number of coils and consequently lacks the corresponding ripple of the magnetic surfaces.

The "reference" magnetic field   stems from the coil configuration optimized with ONSET. A magnetic field realized by a coil configuration will always deviate from a

target magnetic field due to the discrete nature of the coils. Fig. 5.1 illustrates the actual coil geometry of the Wendelstein 7-X geometry and the plasma volume of a GENE-3D simulation. Geometric constraints in the manufacturing and assembly can also lead to deviations in the magnetic field present in the experiments. Traditional stellarator coil optimization hence aims at producing a configuration of magnetic field coils which generate a magnetic field as close to the target magnetic field as possible. Quality criteria measure the difference between the target magnetic field and the magnetic field produced by the coils. Also the geometric properties of the coil configuration are measured by quality criteria. The deviations from the design values are captured in the penalty function, and its minimization is the main target of coil optimization. In ONSET, the magnetic configuration produced by the coils is itself optimized for the performance criteria and the "target" configuration is only taken as a starting point.
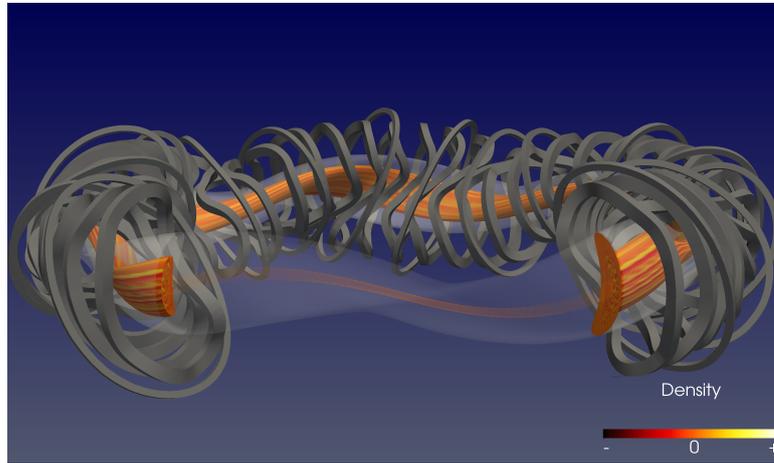
Figure 5.1: Actual coil geometry of Wendelstein 7-X as realized in the experiments and the resulting magnetic field geometry. The snapshot is taken from a nonlinear GENE-3D simulation using the ParaView diagnostic.

The "stochastic" magnetic field is produced by the stochastic extension of ON-SET. Instead of optimizing a single coil configuration, the stochastic version of ONSET optimizes the average of a cloud of coil configurations such that the reduction of the quality of the magnetic field caused by coil displacements can be addressed. This way, the construction tolerances can be loosened, and the time and cost of production of a new stellarator can be lowered. This is useful as recent projects like W7-X [47] and NCSX [48] have been influenced negatively by their own strict tolerance requirements (see Sec. 1.6). It has been shown that the stochastic version of ONSET can not only produce coil configurations that are more resilient against coil displacements but also reproduce the target magnetic field more accurately [53]. Further details about the origins of the different equilibria can be found in the first two sections of Ref. [104].

These three different equilibria are compared in terms of geometric details in Sec. 5.1.1, mode structure and linear growth rates for adiabatic electrons in Sec. 5.1.2, for kinetic electrons in Sec. 5.1.3 and for turbulent transport in Sec. 5.1.4.

5.1.1   Geometric details of the three different equilibria

Before the differences in the output of GENE-3D are investigated, this subsection looks in detail into the differences in geometric quantities serving as input parameters for GENE-3D. The VMEC equilibria provided by J. F. Lobsien are used as input parameters to GVEC, where the total energy is further reduced iteratively (see Sec. 3.3.1). After 15000 iterations, the "target" case is converged, and the file used for the simulations in the following subsections is called `lobsien_orig_w7x_State_0000_00015000.dat`. It can be found in the `gvec_State_files` subfolder of GENE-3D. Similarly, the "reference" case produced by ONSET converged after 20000 iterations and is called `lobsien_-ref_w7x_State_0002_00020000.dat`. The "stochastic" case produced by the stochastic version of ONSET converged after 20000 iterations and is called `lobsien_8000_w7x_-State_0002_00020000.dat`. These three geometry files are then discretized for GENE-3D on a grid with $240 \times 256 \times 128$ points in (x, y, z) direction.   The radial profiles for
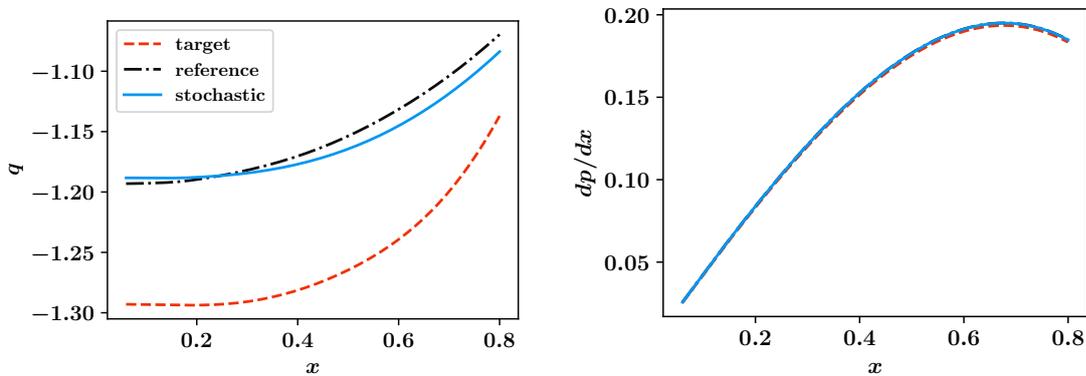
Figure 5.2: The q-profiles show visible differences between the three equilibria while the radial derivative of the pressure profile is quasi identical.

	target	reference	stochastic
a	0.533	0.541	0.528
ρ^*	$3.02 \cdot 10^{-3}$	$2.98 \cdot 10^{-3}$	$3.05 \cdot 10^{-3}$
C_y	0.337	0.369	0.366

Table 9: Scalar geometric values differing between the three equilibria.

safety factor q and radial derivative of the pressure are shown in Fig. 5.2. The q-profile

of the equilibria created by ONSET vary by up to 10% from the target equilibrium. The dp/dx profiles are all the same.

Table 9 summarizes differences in scalar geometric quantities. As a result of different C_y and ρ^* also the box sizes L_y in normalized GENE-3D quantities are different (see Eq. (3.43)). In order to compare the equilibria the y-direction is renormalized by calculating $\alpha = y/(C_y/\rho^*)$. This way, α is an angular coordinate ranging from 0 to $2\pi/n_0$. With this renormalization the magnetic field value of the different equilibria can be compared on the same grid (see Fig. 5.3).
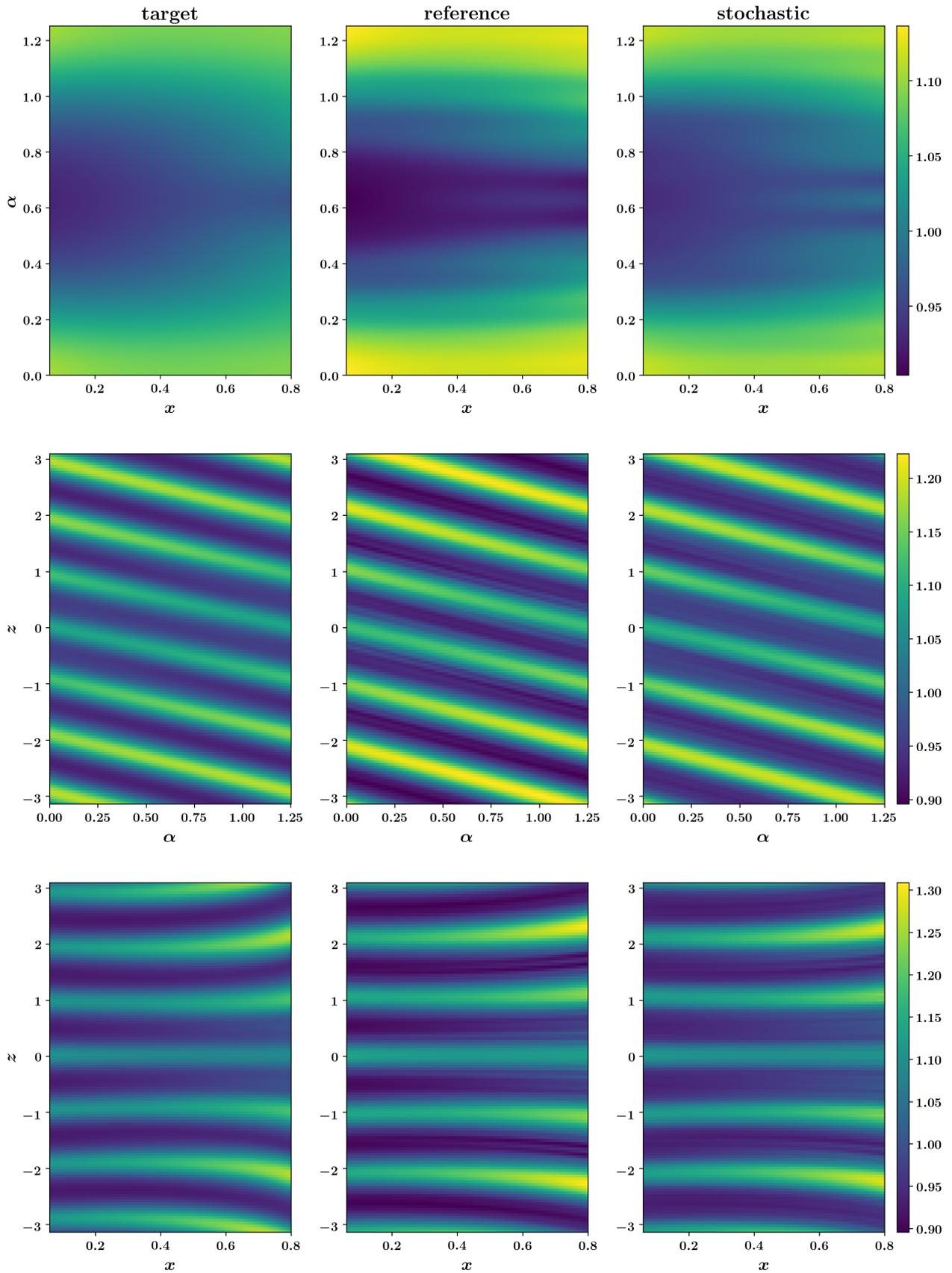
Figure 5.3: Two-dimensional cuts of the background magnetic field B_0 in units of B_{ref} for the three equilibria. The first row shows cuts in x, α direction at $z = 0.0$. The second row shows cuts in the α, z-direction for $x = 0.43$. The third row shows cuts in x, z-direction for $\alpha = 0.0$. The positions are chosen at the peak positions of the turbulent eddies in Fig. 5.19 The differences between the equilibria are small.

The first row compares the magnetic field in the x, α-plane for $z = 0.0$. The general structure is similar for all three equilibria with the highest amplitudes for the reference case.

The second row shows the magnetic field in the α, z-plane for $x = 0.43$. Here, the wavelength of the structure in z direction is shorter for the target case ($\lambda_{\text{target}} = 0.99$) than for the reference and stochastic case ($\lambda_{\text{reference}} = 1.10$). Again, the reference case has the highest amplitudes of the three cases.

The third row shows the magnetic field in the x, z-plane for $\alpha = 0$. Again, the wavelength in z direction is shorter for the target case than for the other two. Also, in the x, z-plane, the reference case has the highest amplitudes.

Overall, however, the differences are relatively small from qualitative visual inspection.

One dimensional cuts of the magnetic field are plotted in Fig. 5.4 to get a better understanding for the quantitative differences. In the leftmost plot the magnetic field is
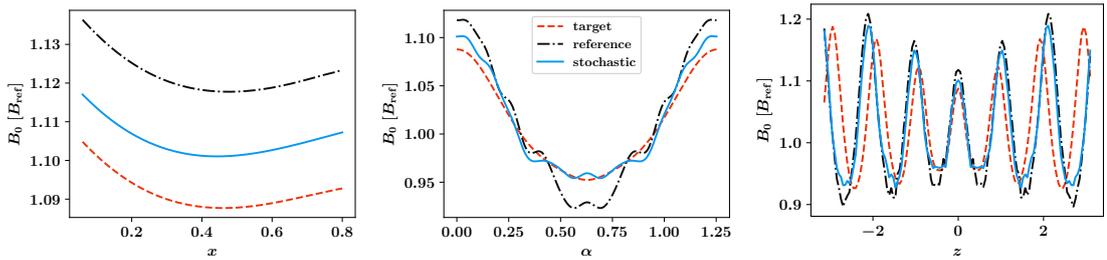
Figure 5.4: One dimensional cuts in x (for $\alpha = 0.0; z = 0.0$), α (for $x = 0.43; z = 0.0$) and z (for $x = 0.43; \alpha = 0.0$) direction.

plotted over the radial coordinate x for $\alpha = 0.0$ and $z = 0.0$. The lines are all shaped similarly with a maximum of 3% of difference between the reference and the target equilibrium.

The plot in the middle shows the magnetic field over the α-coordinate for $x = 0.43$ and $z = 0$. Also here, the trend is similar between all three equilibria with a maximum difference of 5% of the field value in the middle of the α domain.

The plot on the right shows the magnetic field over the z-coordinate for $x = 0.43$ and $\alpha = 0.0$. Again, the wavelength of the structure in z direction is shorter for the target case than for the other two cases. For the stability of a plasma, the extent and location of good and bad curvature regions are important [106].

A bad curvature region is generally classified when $K_y < 0$ (see Table 4). Hence, K_y is plotted in all three planes in Figs. 5.5, 5.6, 5.7. Fig. 5.5 shows K_y in the x, z plane both for $\alpha = 0$ (upper row) which is the center of the bad curvature region in Fig. 5.6

and for an average over all α (second row). The white lines depict the $K_y = 0$ lines separating the good and bad curvature regions. For $\alpha = 0$, the bad curvature region is periodic in z-direction with different radial extent. The extent is the longest at around $z = 0$. Hence, one would expect non-slab-like modes to grow strongest around $z = 0$ for $x < 0.5$. The reference and the stochastic case differ from the target case mainly by having rougher edges of the regions. For the α averaged x, z cut there is only a small bad curvature region around $z = 0$ for $x < 0.2$. Also, here the smallest K_y values are all around $z = 0$ for all x. Note, that the peaked temperature gradient profile (profile 2) mainly used in the following subsections peaks around $x = 0.5$ (see Fig. 5.8) for the linear simulations and around $x = 0.43$ for the nonlinear ones. One expects the modes to be localized in the regions around the temperature gradient peak and especially ITG modes in the bad curvature regions [106]. Fig. 5.6 shows K_y in the x, α plane both for $z = 0$ (first row) which is the center of the bad curvature regions in Fig. 5.5 and averaged along the field line (second row). For $z = 0$ the main region is around $\alpha = 0$ at $x < 0.45$ with a second region at the outer radial boundary around $\alpha = \pi/5 = 0.63$. Here, there are some differences between the equilibria. The region is smallest for the target case, whereas, for the reference case, the bad curvature region extends until $x = 0.6$. This might lead to differences in the simulations between the three equilibria. For the z averaged case, there are no bad curvature regions with the smallest values being at the outer radial boundary of the box. Fig. 5.7 shows K_y in the α, z plane both for $x = 0.43$ (first row) where the temperature profile of the nonlinear simulations is peaked and averaged over all flux surfaces (second row). The bad curvature regions are again centered around $z = 0$ with different extents over the z direction and spanning over all α for the three equilibria. In the x averaged case, the largest region is, however, around $\alpha = 0$. Note that the wave length of the K_y structures in z-direction $\lambda_{K_y} = 1.25$ is larger than for the magnetic field $\lambda_B = 0.95$. Concluding from these observations the modes in the following subsections will most likely peak around $(x, \alpha, z) = (< 0.5, 0.0, 0.0)$. However, the temperature gradient profile also influences the location and amplitude of modes. Therefore, to compare the influence of the temperature profiles to the influence of the geometric variations, the linear analysis is done with two different temperature profiles. In summary, the reference and stochastic cases are overall very similar, while there are small differences to the target configuration. The expectation for the simulations of the next subsections is, therefore, that the stochastic and reference case behave similarly, while there might be deviations to the target case.

5.1.2   Linear simulations with adiabatic electrons

In this subsection, the linear mode structure and growth rates are compared for two different temperature and density profiles. The first temperature profile is the peaked
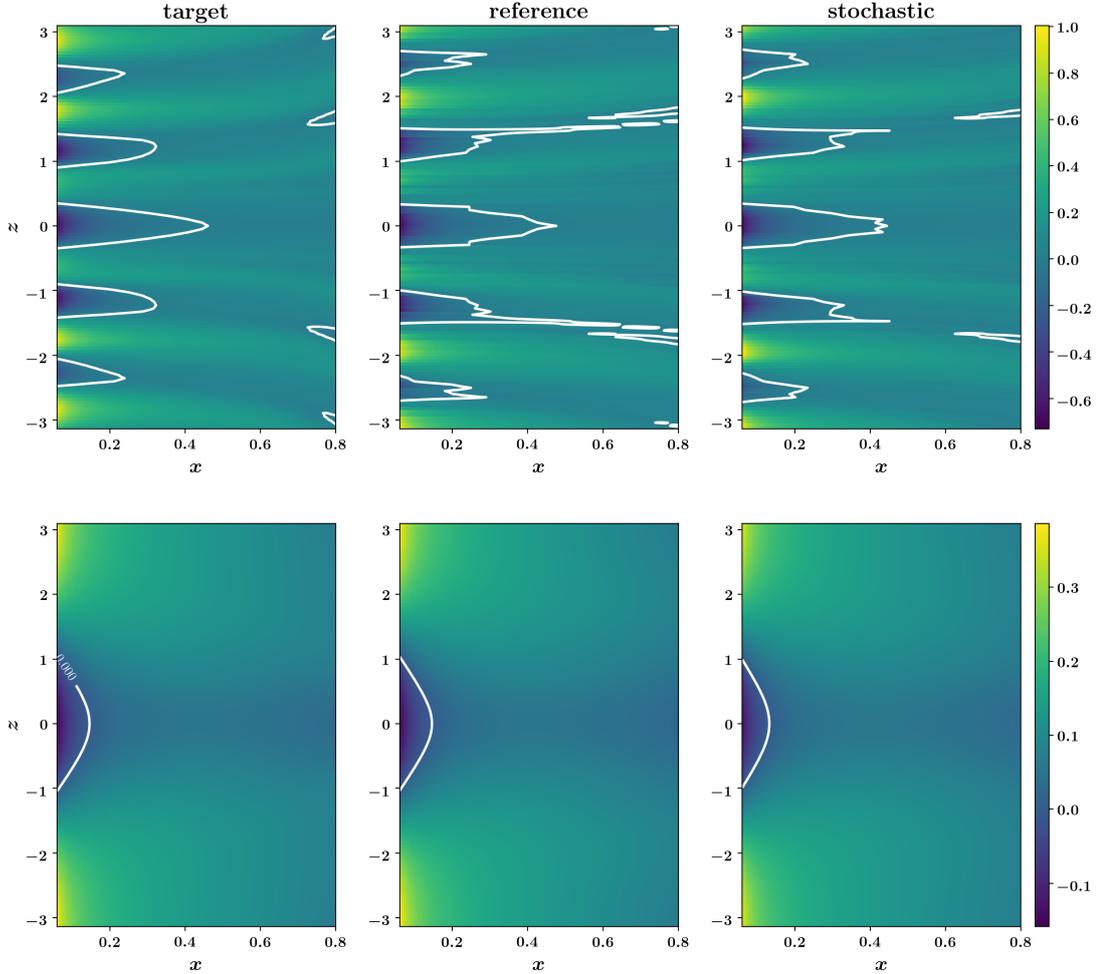
Figure 5.5: Two-dimensional x, z cuts of the K_y value. The first row shows cuts at $\alpha = 0$ which is the center of the bad curvature regions in Fig. 5.6. Here, the bad curvature regions are located mainly at small x. The largest region for all three equilibria is around $z = 0$. The regions of the reference and stochastic case have rougher edges than the "smooth" target case. The second row shows the x, z cut averaged over all α. Note, that the colorbar is not the same for the first and second row as otherwise the structures in the averaged case would not be visible. In the α averaged case the equilibria look very similar with only a small bad curvature region for $x < 0.2$ around $z = 0$. The smallest K_y values are around the $z = 0$ line.
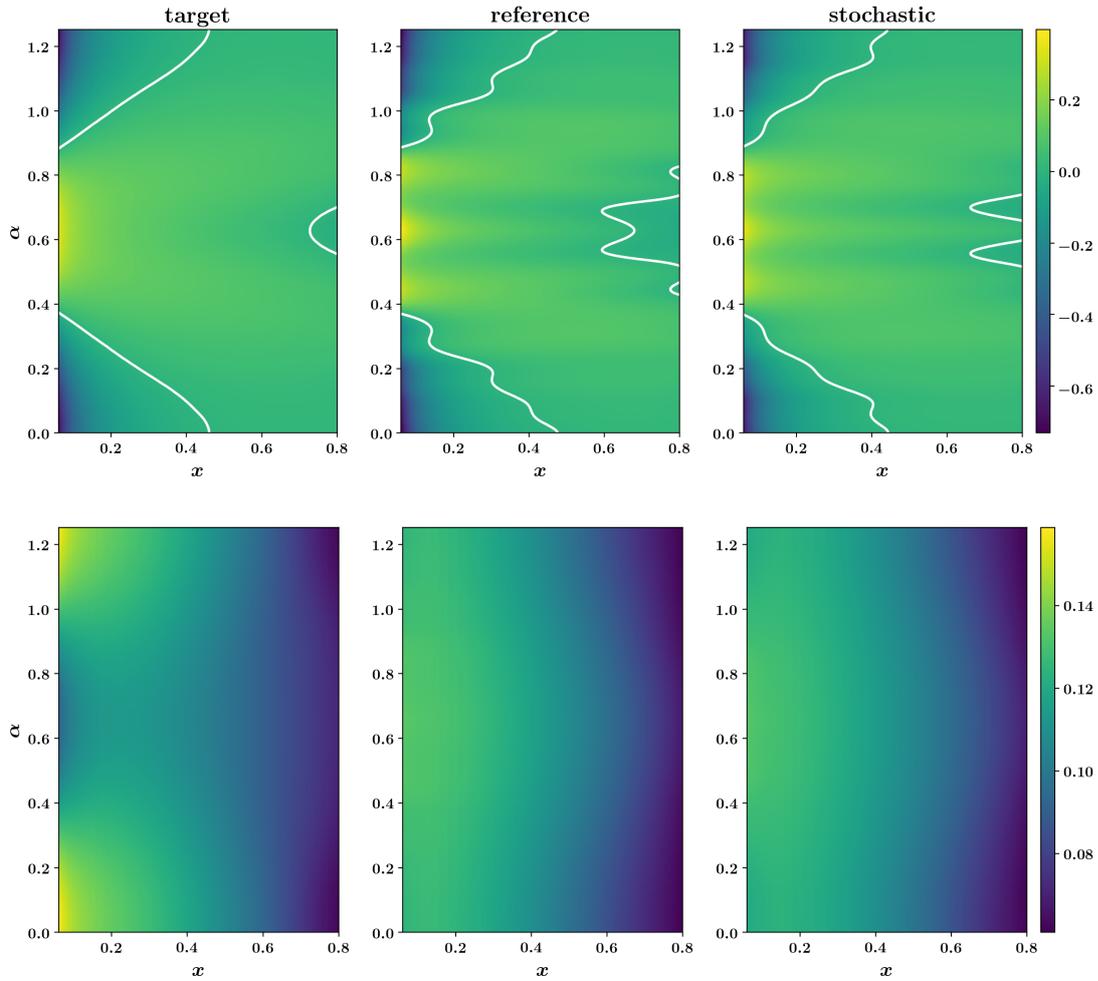
Figure 5.6: Two-dimensional x, α cuts of the K_y value. The first row shows cuts at $z = 0$ which is the center of the bad curvature regions in Fig. 5.5. Here, the bad curvature regions are centered around $\alpha = 0$ for $x < 0.45$. Another small region appears at the radial outer boundary of the box around $\alpha = \pi/5 = 0.63$. This second region has different shapes for the three equilibria. The second row shows the x, α cuts averaged over the z coordinate. Here, no bad curvature regions appear.
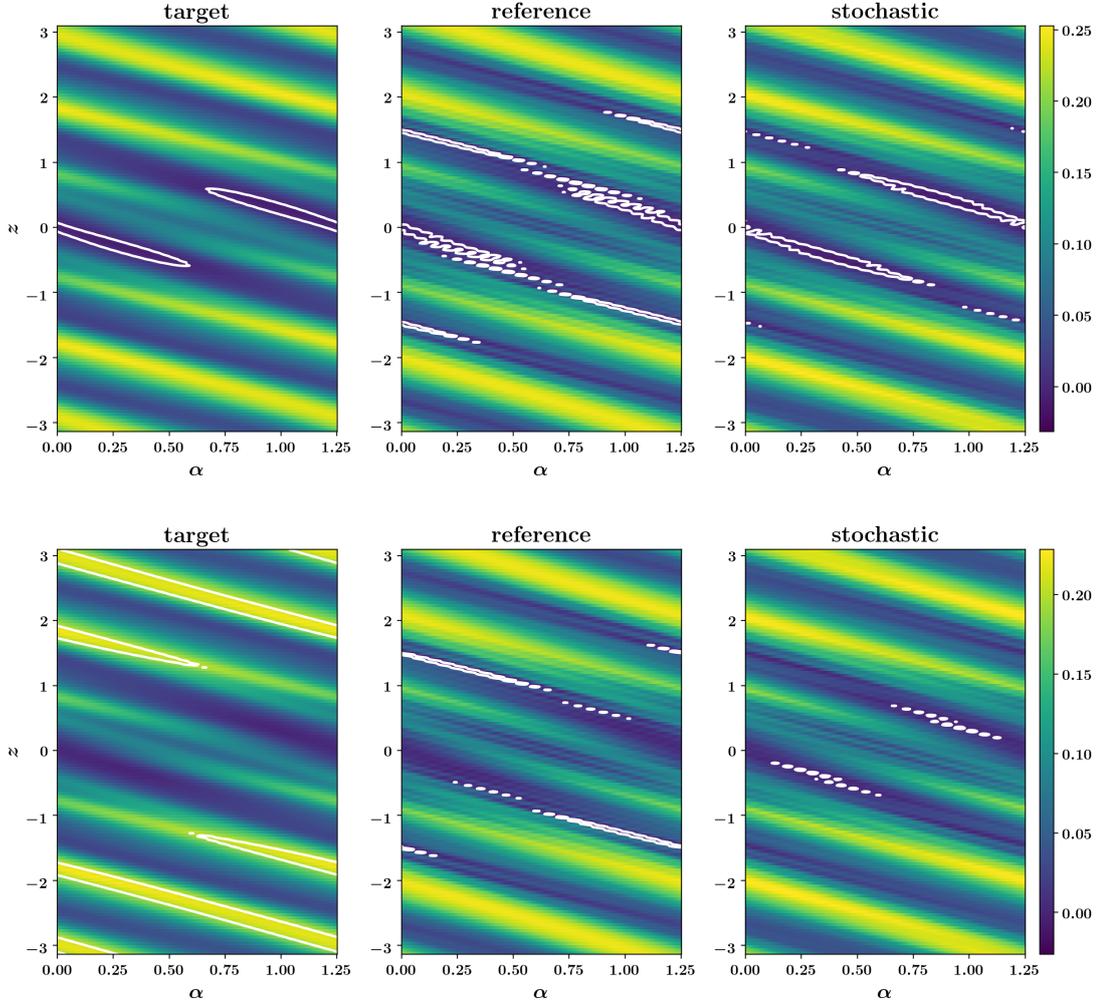
Figure 5.7: Two-dimensional α, z cuts of the K_y value. The first row shows cuts at $x = 0.43$ which is the center of the radial box and the location of the peak of the temperature gradient of the nonlinear simulations in Sec. 5.1.4. Here, the bad curvature regions are around $z = 0$ for all α for the target case. In the reference and stochastic case these regions are distorted and span to smaller and larger z-values than the target case. The second row shows the α, z cuts averaged over all x. Here, again, the bad curvature span over all α. However, for the regions further away from $z = 0$ the structures are centered around $\alpha = 0$. Again, the structures are smoother in the target case.

gradient profile "profile 2" described in Sec. 4.1.1. The name derives from the naming convention in the `profile.F90` GENE-3D source file. The maximum temperature gradient is set to $\kappa_T = 4.0$ with a width of $\omega_T = 0.1$ and a maximum density gradient of $\kappa_n = 1.0$ and a width of $\omega_n = 0.3$. The second profile is the one used in section 4.3 of Ref. [104] to test for neoclassical transport.

The temperature profile 7 is derived from the pressure profile described in Ref. [105], which was also used in the final design of Wendelstein 7-X. With a density profile determined in the experiments of Wendelstein 7-X, the temperature is then derived from the relationship $p = n \cdot T$ with the pressure p, density n, and temperature T. In principle, the self-consistent derivation of the density and temperature profiles needs to be done in an iterative process taking the magnetic field produced by the bootstrap current into account, which influences the pressure profile. Since the bootstrap current in W7-X is small compared to tokamaks, this method would not change the final temperature and density profiles drastically and is therefore neglected. These profiles are called "profile 7" in the following analogously to the name in the `profile.F90` GENE-3D source file. The analytic formulas for temperature, density, and their logarithmic gradients are

$$T_i(x)/T_{\text{ref}} = + \frac{5}{7} \cdot \frac{7.0 - 11.0 \cdot x^2 + 4.0 \cdot x^4}{0.15 + 0.85 \cdot (1.0 - x^{15})^2 - 0.15 \cdot (1.0 - \exp(-x^2/0.8^2))}$$

$$\Omega_T(x) = - \frac{a}{5} \cdot \frac{0.85 \cdot (-x^{15} + 1.0)^2 + 0.15 \cdot \exp(-1.56 \cdot x^2)}{0.57 \cdot x^4 - 1.57 \cdot x^2 + 1.0} \cdot$$

$$\left(5.0 \cdot \frac{2.29 \cdot x^3 - 3.14 \cdot x}{0.85 \cdot (-x^{15} + 1.0)^2 + 0.15 \cdot \exp(-1.56 \cdot x^2)} \right.$$

$$+ 5.0 \cdot (25.5 \cdot x^{14} \cdot (-x^{15} + 1.0) + 0.47 \cdot x \cdot \exp(-1.56 \cdot x^2)) \cdot \qquad (5.1)$$

$$\left. \frac{0.57 \cdot x^4 - 1.57 \cdot x^2 + 1.0}{(0.85 \cdot (-x^{15} + 1.0)^2 + 0.15 \cdot \exp(-1.56 \cdot x^2))^2} \right)$$

$$n_i(x)/n_{\text{ref}} = + 1.46 \cdot (0.15 + 0.85 \cdot (1.0 - x^{15})^2 - 0.15 \cdot (1.0 - \exp(-x^2/0.8^2)))$$

$$\Omega_n(x) = - a \cdot \frac{-37.2 \cdot x^{14} \cdot (-x^{15} + 1.0) - 0.68 \cdot x \cdot \exp(-1.56 \cdot x^2))}{1.24 \cdot (-x^{15} + 1.0)^2 + 0.22 \cdot \exp(-1.56 \cdot x^2)} \cdot$$

The Jupyter notebook with the calculation of the derivatives and zero crossings and respective plots can be found in the GENE-3D subfolder `tools/python/plot_analytic_-profiles`. In order to only have positive temperature gradients Ω_T, the simulation box

extends until $x = 0.935$ for profile 7. Due to the peak positions of the temperature and density gradients, the simulation box for the simulations using profile 7 is centered around $x_0 = \sqrt{0.5} = 0.707$, with a width of `lx_a = 0.458` to end at $x = 0.935$. The simulations using the peaked gradient profile (profile 2) are centered around $x_0 = 0.5$ with a radial width of `lx_a = 0.6`. The profiles are plotted for the radial range of the simulations in Fig. 5.8. Unfortunately, the density gradient diverges towards the edge.
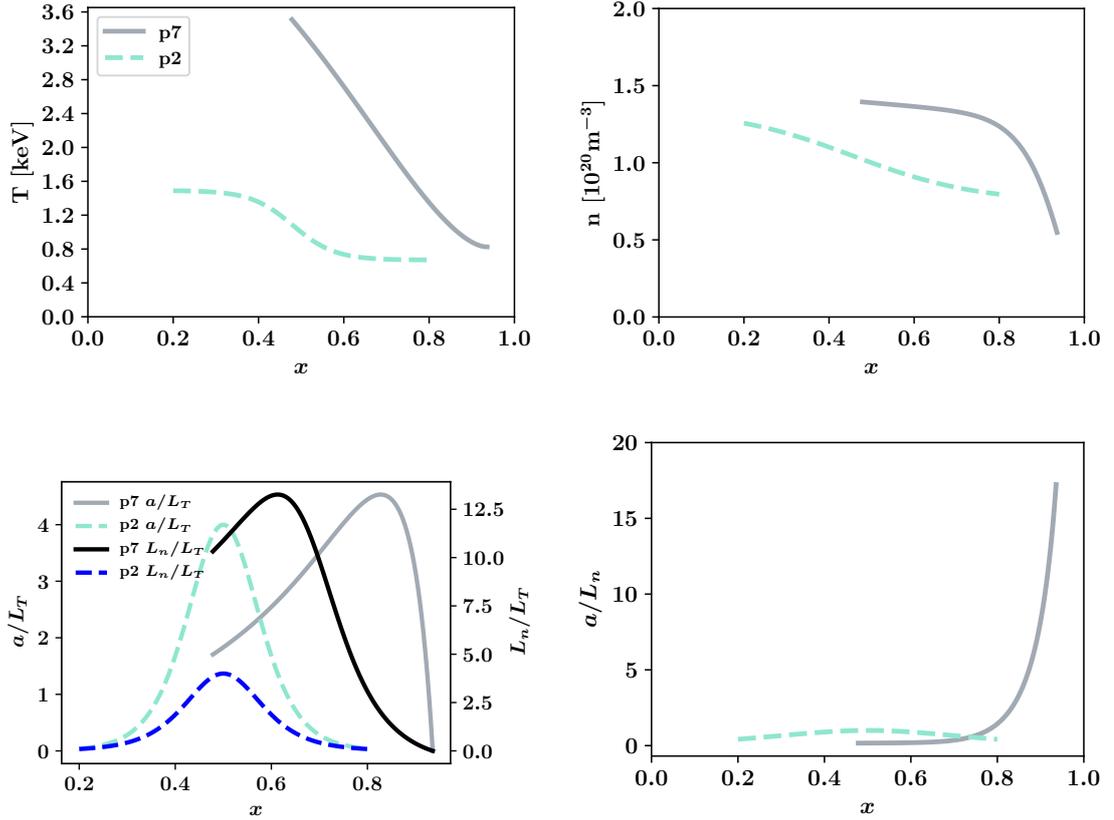
Figure 5.8: Temperature, density and gradient profiles for the geometric variation tests of Wendelstein 7-X. The density gradient of profile 7 diverges towards the edge.

GENE-3D generally assumes Dirichlet type boundary conditions at the edges of the radial simulation box where profiles are assumed to be zero. Therefore, there can be unphysical profile variations for profiles with high values at the edges. However, the ratio of L_n/L_T which is a main driver for ITG modes [107] depicted in Fig. 5.8 (c) converges to zero toward the outer boundary. Different locations and heights of the peaks of the gradient profiles can give rise to different modes in GENE-3D.

Results

The most unstable modes drifting in the ion diamagnetic directions are shown in Fig. 5.9. For the "reference" and "stochastic" simulations using profile 7 (solid lines), the modes peak at the same toroidal mode number ($n = 110$). The dominant mode in the target magnetic field simulation peaks at $n = 105$. A smaller mode number indicates that the mode varies on a larger scale. However, large scale instabilities are more relevant in nonlinear simulations as they, generally, lead to more transport of heat and particles. In a simple quasi-linear model, the nonlinear transport of heat Q is approximated by $Q \propto \gamma/k_y^2$ with the linear growth rate γ and the wavenumber k_y [108]. The growth rate of the target magnetic field simulation $\gamma = 0.147$ is larger compared to $\gamma = 0.137$ for the other two cases. The growth rates in this section are all given in normalized form and the unit v_i/a is omitted to improve readability. As a reminder, $v_i = \sqrt{T_{i0}/m_i}$ is the ion thermal velocity and a denotes the minor radius.
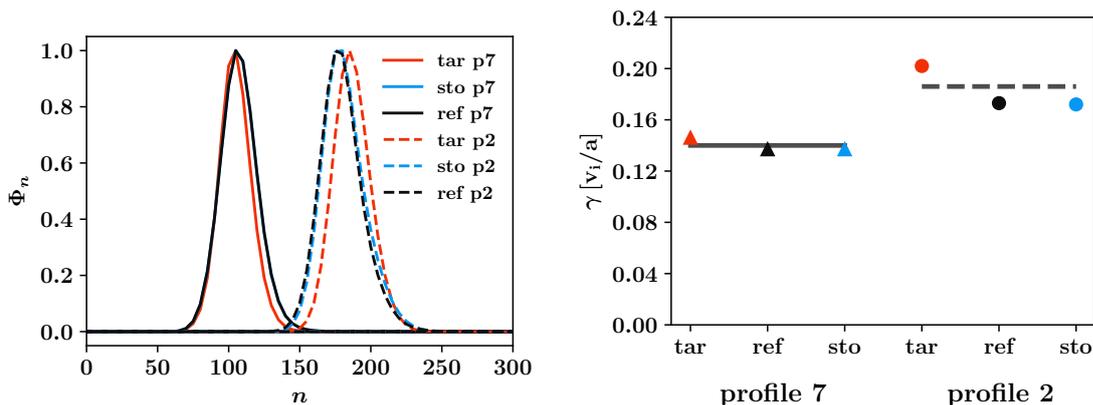
Figure 5.9: Most unstable modes in linear adiabatic electron simulations for the three different magnetic equilibria 'target' (black), 'reference' (blue) and 'stochastic' (red). For all three equilibria two different temperature and density profiles are used (see Fig. 5.8): profile 7 (triangles & solid lines) and profile 2 (circles & dashed lines). The average growth rate for profile 2 is $\gamma = 0.140$ whereas the average growth rate for profile 7 is $\gamma = 0.186$.

The most unstable modes in the simulations employing the reference and stochastic geometries and profile type 2 (dashed lines) again peak at the same toroidal mode number ($n = 175$) with almost the same growth rates ($\gamma \approx 0.17$). The dominant mode in the target magnetic field simulation peaks at $n = 185$ with a higher growth rate of $\gamma = 0.20$.

In summary, the mode number and growth rates are almost identical for the reference and the stochastic case and slightly different for the target case. This is in line with the observations in the previous subsection, where the reference and stochastic cases show almost identical geometric properties.

The differences in growth rate between the profile 2 and profile 7 simulations can be

attributed to the differences in peak value and width of both the temperature gradient curve and the L_n/L_T curve depicted in Fig. 5.8. Both curves, which represent the main drivers for ITG modes, have higher peak values and stay at high values for a wider radial range for profile 7 compared to profile 2.

The mode structure in all three planes   is depicted in Fig. 5.10 for profile 2 and in Fig. 5.11 for profile 7 to allow a more detailed look at the simulations. The GENE coordinate system is chosen over the PEST coordinate system as structures appear larger in the former, making visual inspection and comparisons easier. The differences in mode structure are minimal between the three equilibria. Also, the peak positions of the modes are all very similar (see Table 10). The differences due to different temperature profiles are larger than the differences due to geometry. The mode structure is more localized in the simulations using profile 2 due to the more localized temperature gradient profile.

Concluding,   every observed quantity is either the same or similar for the three equilibria. As this behavior is the same for both temperature profiles the simulations in the following subsections use profile type 2 leading to lower computational cost of the simulations.
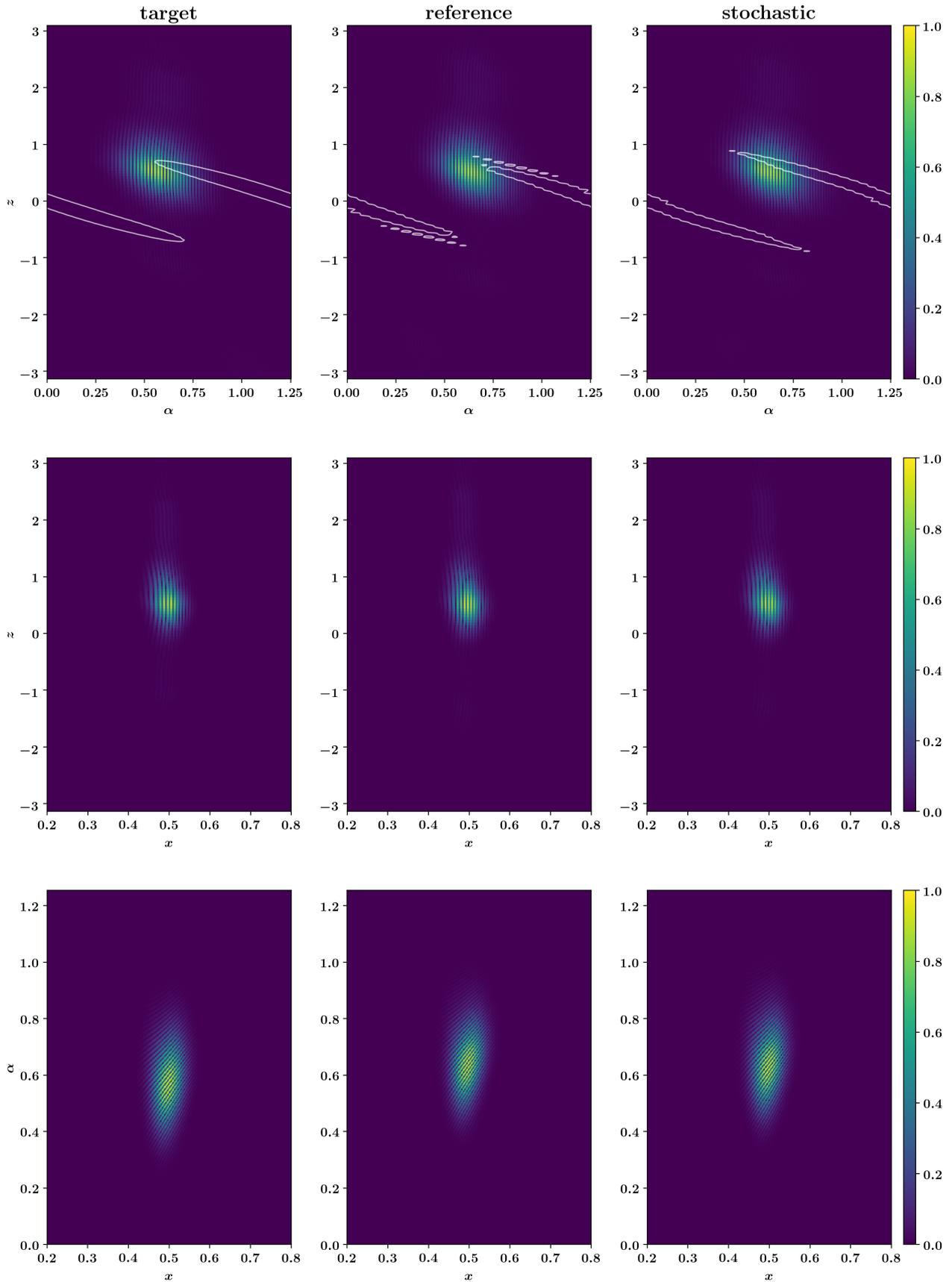
Figure 5.10: Mode structures in the (α, z) (top), (z, x) (middle) and (α, x) (bottom) planes for target geometry (left), reference geometry (middle) and stochastic geometry (right) for profile type 2 and adiabatic electrons. The cuts are done where the field value peaks in the third coordinate (see Table 10). Color coded are the normalized field values. The normalization is done such that the maximum field value corresponds to 1.
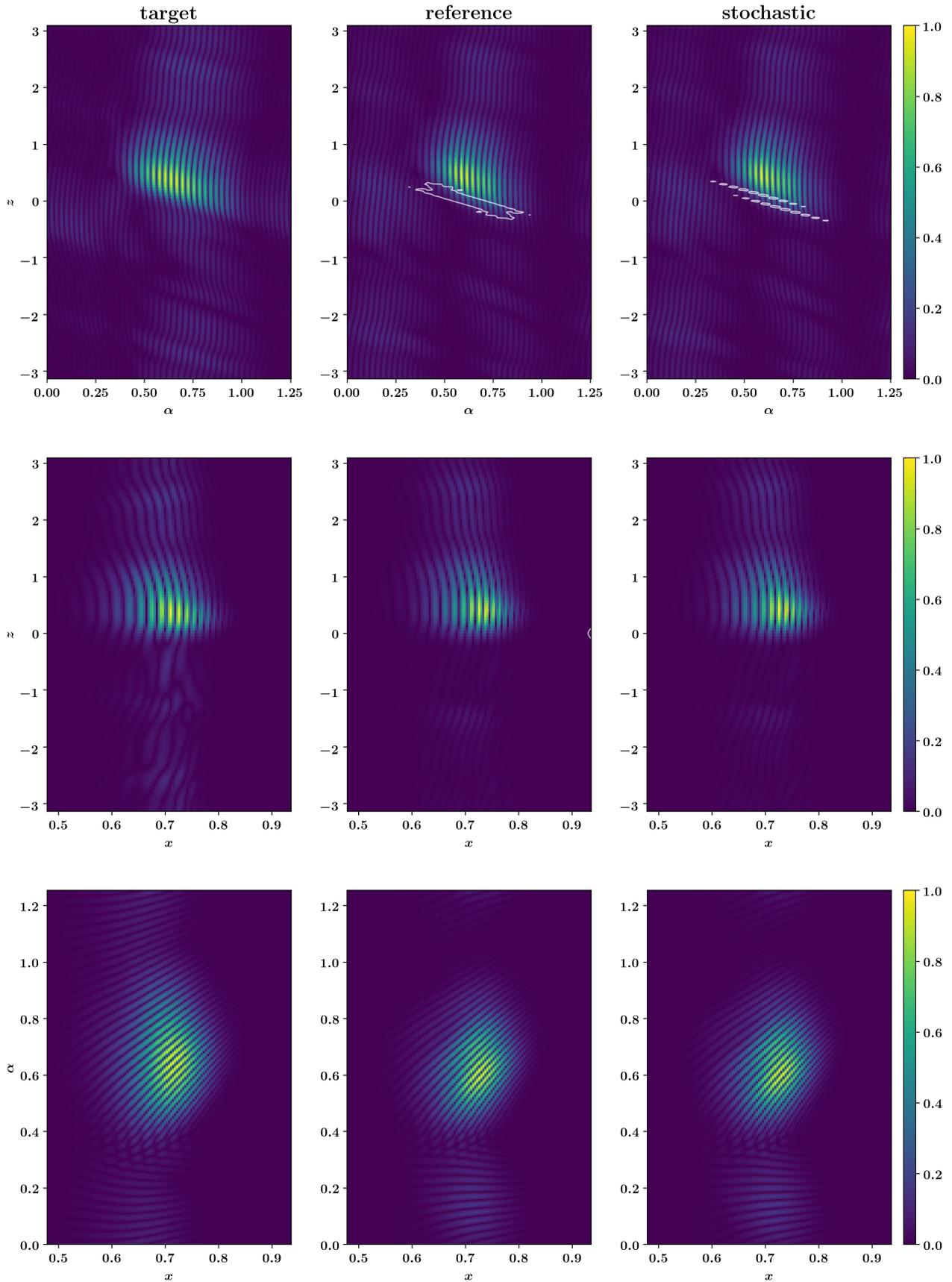
Figure 5.11: Mode structures in the (α, z) (top), (z, x) (middle) and (α, x) (bottom) planes for target geometry (left), reference geometry (middle) and stochastic geometry (right) for profile type 7 and adiabatic electrons. The cuts are done where the field value peaks in the third coordinate (see Table 10). Color coded are the normalized field values. The normalization is done such that the maximum field value corresponds to 1.

	x	α	z
target adiabatic profile 2	0.50	0.57	0.49
reference adiabatic profile 2	0.50	0.64	0.49
stochastic adiabatic profile 2	0.50	0.62	0.49
stochastic adiabatic profile 7	0.72	0.21	0.39
target kinetic $\kappa_{T_e} = \kappa_{T_i}$	0.49	1.11	0.10
reference kinetic $\kappa_{T_e} = \kappa_{T_i}$	0.49	1.09	0.15
stochastic kinetic $\kappa_{T_e} = \kappa_{T_i}$	0.49	1.08	0.15
stochastic kinetic $\kappa_{T_e} = 0$	0.51	0.83	0.29

Table 10: Position of the peak value of the electrostatic field potential for the different simulations in the GENE coordinate system. Note that the field does not necessarily peak precisely at the peak of the temperature gradient profile. The cuts in Fig. 5.10, Fig. 5.11 and Fig. 5.14 are taken at these positions.

5.1.3   Linear simulations with kinetic electrons

Adding a kinetic electron species can change the outcome of a simulation. In principle, the dominant mode in an adiabatic electron simulation can be an ITG mode while a kinetic electron simulation run with the same parameters can show a dominant electron mode such as an ETG or a TEM. This is especially true for stellarator geometries with multiple magnetic wells which drive TEMs [106]. Therefore, the tests of Sec. 5.1.2 are repeated with kinetic electrons. The same density and temperature gradients are assumed for the electrons and the ions, $\kappa_T = 4.0$ and $\kappa_n = 1.0$. Also, the resolutions are the same as in the previous section. Hence, the only difference between the simulations is the explicit heavy electron species in the system with $m_e/m_i = 1/100$ (m_e: electron mass, m_I: ion mass). The choice of heavy electrons allows keeping the resolution requirements low. For the realistic mass ratio of $m_e/m_i = 1/1836$, the scale separation between electron scales and ion scales would require the resolutions in all dimensions to be much higher. E.g. for the kinetic electron benchmark in a tokamak geometry in Sec. 4.1.1 the number of points in radial direction needed to be `nx0 = 2580` compared to the 240 points used here.

Results

Figure 5.12 shows that the growth rates and toroidal mode numbers are increased in the kinetic electron simulations compared to adiabatic electron simulations. The dominant mode is still rotating in the ion diamagnetic drift direction. Hence, it is still an ITG mode, just amplified by the electrons, and shifted to smaller scales. The up-shift of the mode number due to the kinetic electrons is smaller than the change of mode number
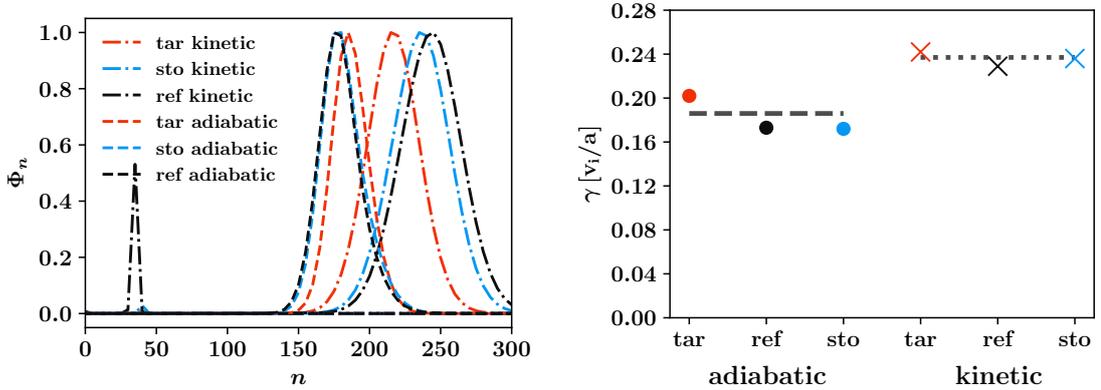
Figure 5.12: Most unstable modes in linear simulations of adiabatic (circles & dashed lines) and kinetic electrons (crosses & dotted lines) using profile 2 for the three different magnetic equilibria 'target' (black), 'reference' (blue) and 'stochastic' (red). The small additional peak for the reference case at $n = 40$ is due to the simulation not being fully converged in time. The average growth rate for adiabatic electron simulations is $\gamma = 0.186$ whereas the average growth rate for the kinetic electron simulations is $\gamma = 0.237$.

due to different temperature and density profiles. This result indicates that an explicit electron species in the simulations does not strongly affect the underlying instability for the parameters and profiles investigated here. Table 12) summarizes the mode numbers and growth rates for this study. Consistent with the adiabatic electron results, the differences between the three equilibria are small. Therefore, also kinetic electron simulations do not strongly react to differences in geometry.

A test with a flat temperature profile for the electrons $\kappa_{T_e} = 0$ is performed with a flat temperature profile using the reference geometry. Here, convergence in time takes a lot longer than for the previous case. For the last continuation of this simulation, the growth rate diagnostic is illustrated in Fig. 5.13. The growth rate of $\gamma = 0.086$ is lower than for the adiabatic case ($\gamma = 0.174$) and the kinetic case with $\kappa_{T_e} = \kappa_{T_i}$ ($\gamma = 0.226$). This is in line with the findings in Ref. [106], where the electrons are stabilizing the ITG modes for W7-X when the electrons are not driven by an electron temperature gradient. Here, further studies with kinetic electrons are needed to investigate this effect. At the time of writing of this thesis, kinetic electron simulations are too expensive to do systematic studies, also in terms of convergence. Sec. 3.2.7 covers ideas to further improve code performance of GENE-3D to alleviate this issue.

The mode structures for the $\kappa_{T_e} = 0$ (rightmost column) and the $\kappa_{T_e} = \kappa_{T_i}$ (first three columns) case are compared in Fig. 5.14. The simulation with a non-zero electron temperature gradient shows a more localized mode structure than the simulation with a zero electron temperature gradient. Compared to the adiabatic electron simulation,

the mode structure of the kinetic electron simulation displays some differences. In the α, z-plane, the mode in the kinetic electron simulation is shifted towards higher values of α compared to the adiabatic electron case. This is in line with the bad curvature regions as they span over the entire α-domain. Also, the structure is smaller, and the subdominant structures are more pronounced for the kinetic electron simulation than for the adiabatic electron simulation. Also, in the x, z-plane, the subdominant structures are more pronounced for the kinetic electron simulation than for the adiabatic electron simulation. The main structure is in the same position in both cases. In the x, α plane is again shifted towards higher values of α, whereas the mode is slightly more extended in x-direction for the kinetic electron simulation. The positions of the cuts are taken at the peak field value positions summarized in Table 10).
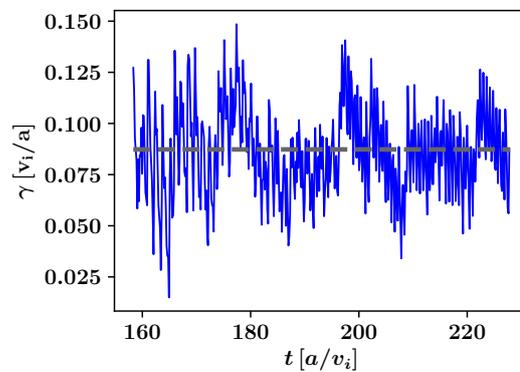
Figure 5.13: Growth rate γ over time for the kinetic electron $\kappa_{T_e} = 0$ simulation. The growth rate fluctuates strongly and does not appear to converge. The average growth rate is $\gamma = 0.086$.
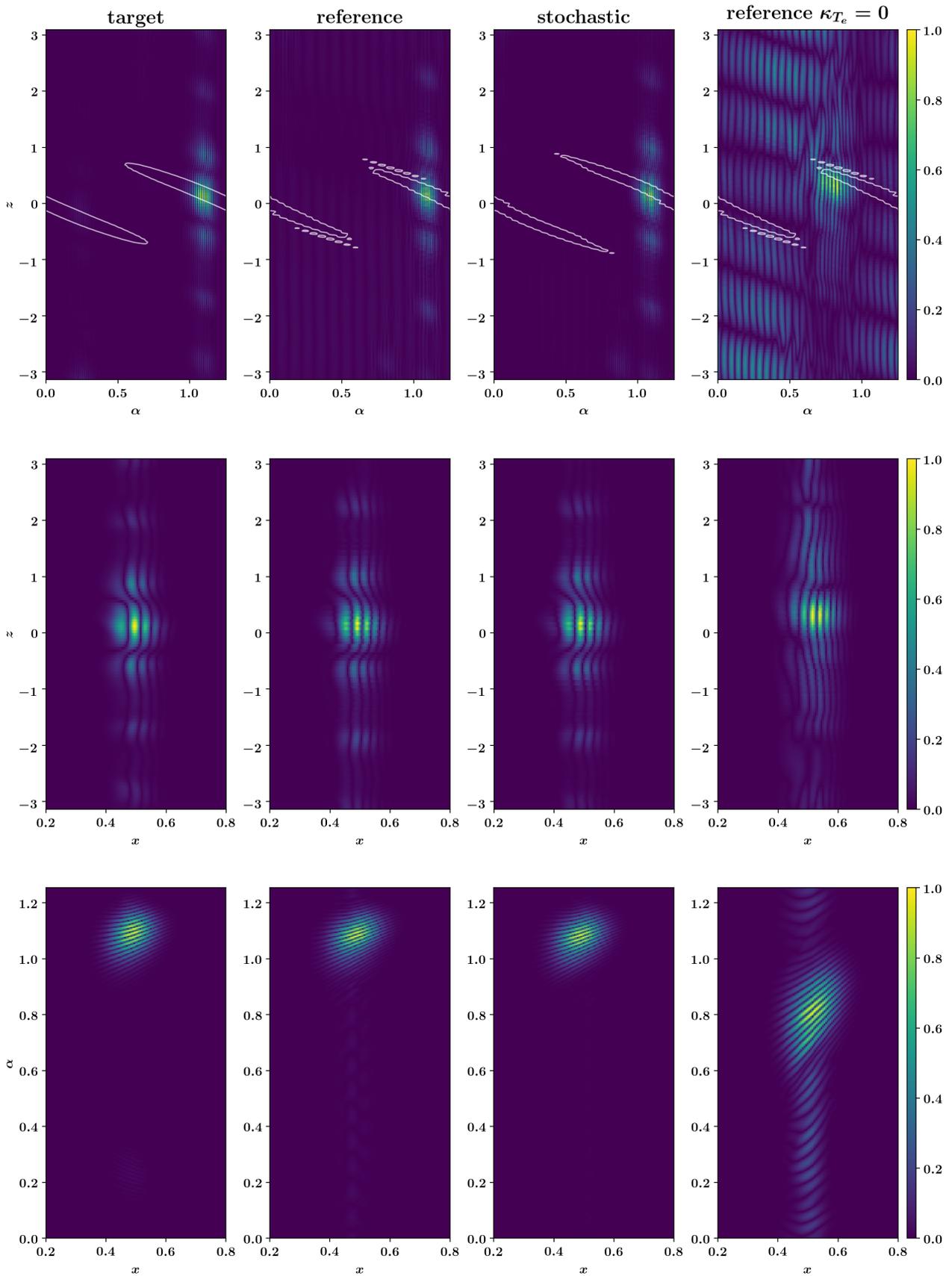
Figure 5.14: Mode structures for kinetic electron simulations with $\kappa_{T_e} = \kappa_{T_i} = 4.0$ (first three columns) and $\kappa_{T_e} = 0$ (rightmost column). The mode structure is more localized for the kinetic electron case with a non-zero electron temperature gradient.

5.1.4 Nonlinear simulations with adiabatic electrons

Here, nonlinear simulations are performed with adiabatic electrons to measure, among other things, the volume-averaged (electrostatic) ion heat flux. A Krook-type heat source with $\kappa_H = 0.06$ is added to keep the temperature gradient profiles constant during the simulations.

Convergence for box sizes and resolutions is tested for the stochastic equilibrium. All simulations in this section use profile 2. However, as the minimum and maximum radial values of $x_{\min}$ and $x_{\max}$ are changed during the convergence test, so does the peak position of the temperature gradient profile. It is always in the middle of the radial simulation box at $x_0 = (x_{\min} + x_{\max})/2$. In total, four different parameter sets are used. The differences are outlined in Table 11.

The first try is using the same parameter set and box sizes as in the linear simulations before. This, however, gives rise to heat transport close to the inner boundary at $x = 0.2$, where Dirichlet boundaries artificially damp possible physical effects (see Sec. 3.2.3). This transport of heat around $x = 0.2$ is due to geometrical drive by the bad curvature regions. Sec. 5.2 looks at this effect in detail. Therefore, the second try sets the inner boundary at $x_{\min} = 0.001$. This, in turn, leads to a numerically unstable simulation. Hence, the following simulations are performed with $x_{\min} = 0.06$.

The total volume averaged heat flux over time is illustrated in Fig. 5.15a (a). Note that the volume average includes the buffer zones where a damping term is added to the right-hand side of the Vlasov equation (3.4) in order to avoid unphysical behavior at the radial edges of the simulation box (see Ref. [30]). Also, the part of the gyro-ring outside the simulation box is set to zero. This affects the physics inside the simulation domain through the gyroaverage operator and the field matrix. Averaging over the total volume is still justified as long as it is done the same way for all simulations. The first try simulation (green dash-dotted line) has a lower total heat flux than the following simulations.

The time and flux surface averaged heat flux over the radial coordinate is illustrated in Fig. 5.15b. Here, the first try simulation has a steep increase at the boundary, which is an indication for the boundary being too close to the turbulent structures. The following simulations show a gradual decrease to zero heat flux at the inner boundary, which is a good indicator for a converged box size. At the outer boundary, the convergence to zero appears smoother for the fourth try (stochastic, red dotted line) simulation with an outer boundary at $x_{max} = 0.8$ instead of $x_{max} = 0.7$ for the third try (blue dashed line). The effect on the total heat transport in Fig. 5.15a seems to be small, but the larger simulation box is chosen to err on the side of caution.

The n-spectrum visualized in Fig. 5.15c justifies using less resolution in y direction as for the first try simulation the heat flux reaches zero before $n = 500$. Here Q_{es} is multiplied with k_y to improve visibility of the curve at higher n, and k_y is converted to n on the x-axis to compare to the linear results from before.

The source term depicted in Fig. 5.15d confirms the observations of Fig. 5.15b to span the radial box from $x = 0.06$ to $x = 0.80$ as a smooth convergence to zero is observed before the boundaries are reached.

	first try	second try	third try	fourth try (stochastic)
nx0	120	240	160	240
ny0	384	384	256	256
$x_{\min}$	0.2	0.001	0.06	0.06
$x_{\max}$	0.8	0.801	0.70	0.80

Table 11: Differences in parameters in the convergence tests.

Results Fig. 5.16 (a) shows the volume averaged nonlinear heat flux $\langle Q_{es} \rangle_V$ for the three equilibria. The first part of a nonlinear simulation is considered the linear phase as it usually displays uninhibited growth similar to purely linear simulations. In Fig. 5.16 (a) the time evolution is almost identical for the first 250 time units which confirms the linear results of the previous section. After the linear phase, nonlinear effects start to take over and similarity between two simulations can only be determined statistically. The qualitative features of the time traces of the heat flux are similar as strong fluctuations are observed for all three equilibria. The fluctuations persist over all simulated periods of time. Quantitatively, the time-averaged heat fluxes of the three equilibria are similar, as well as their level of fluctuation (see Table 12). The level of fluctuation is calculated as one standard deviation of the time trace in the interval used for averaging. Fig. 5.16 (b) shows the time and flux surface averaged heat flux over the radial coordinate x. The shapes and peak positions are very similar with slightly different peak values in accordance with the volume-averaged heat flux determined in Fig. 5.16 (a). Therefore, also the distribution of heat flux over the radial coordinate is not affected strongly by the differences in geometry.

The fluctuations of the heat flux in time are further investigated by plotting the heat flux spectrum for different points in time corresponding to different stages of the fluctuation in the reference simulation in Fig. 5.16c. The toroidal mode number spectrum of a single simulation can differ depending on the level of heat flux at a given

(a) Total volume averaged heat flux $\langle Q_{es} \rangle_V$ over time.

(b) Time and flux surface averaged heat flux $\langle Q_{es} \rangle_{FS,t}$ over radial coordinate x.

(c) Spectrum of heat flux times wave number $Q_{es}(k_y) \cdot k_y$ over mode number n.

(d) Source term $M22$ over radial coordinate x.
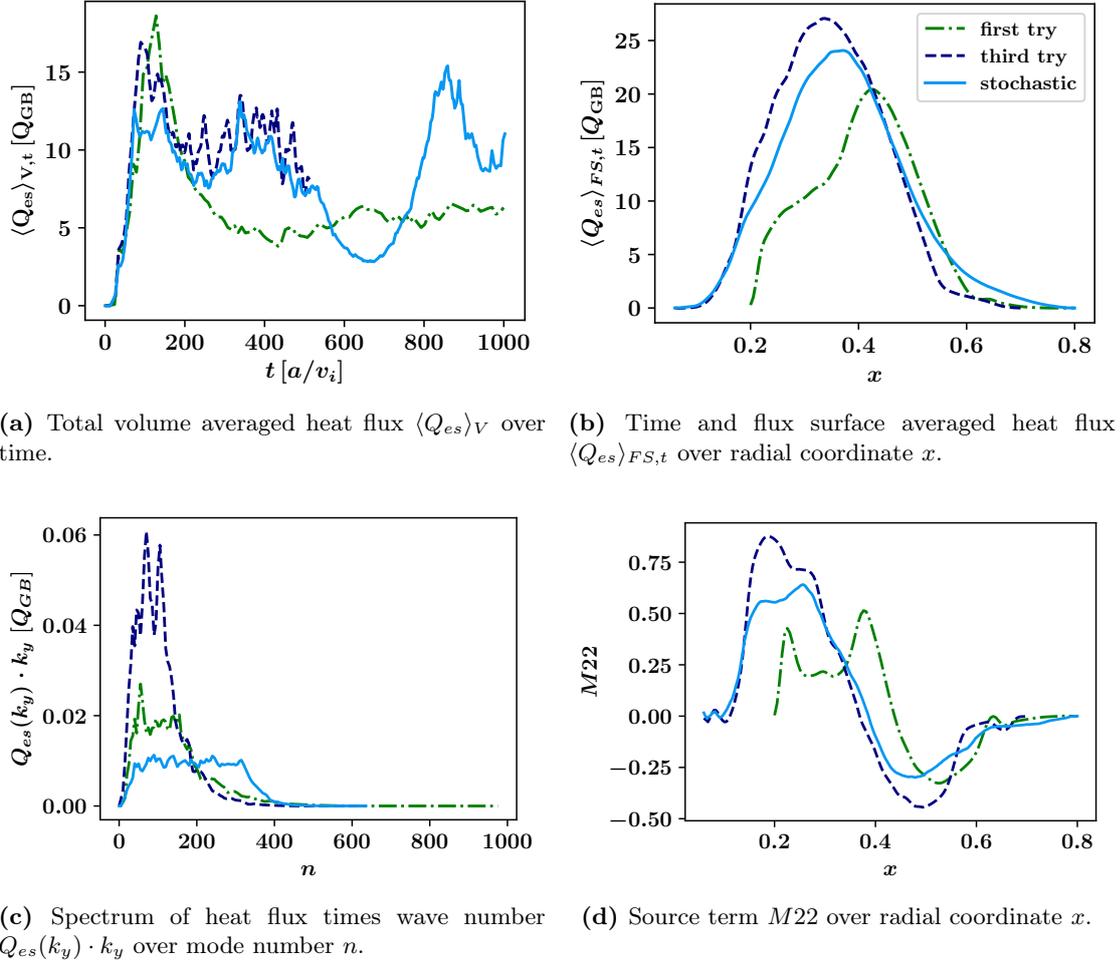
Figure 5.15: Convergence tests for the nonlinear simulations. Changed between the simulations are resolutions and box sizes in x and y direction (see Table 11). Figure (a) shows that the first try simulation has significantly lower heat flux than the other two simulations. Figure (b) gives a possible explanation as the heat flux increases steeply at the boundary of the simulation box at $x = 0.2$. Therefore, for the following simulations, the box was expanded. Figure (c) justifies to use less resolution in y direction as the heat flux spectrum is zero for $n > 500$. Here Q_{es} is multiplied with k_y to improve visibility of the curve at higher n. Figure (d) confirms that the radial box needs to span from $x = 0.06$ to $x = 0.80$ as the sources should be close to zero near the boundaries.

time. The smaller the total heat flux the wider and flatter the toroidal mode number spectrum becomes.
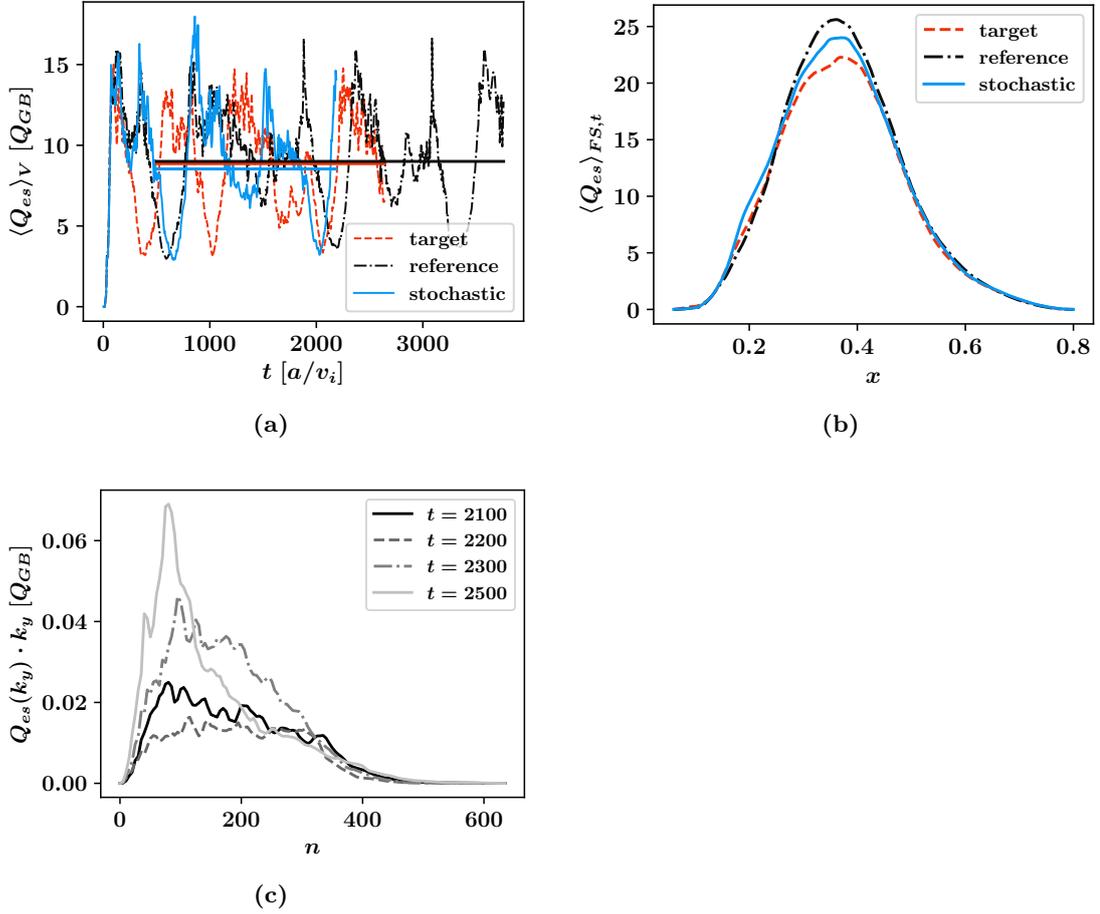
Figure 5.16: (a) Total volume averaged heat flux over time for the three equilibria, (b) time and flux surface averaged heat flux over x and (c) toroidal mode number spectrum for different times of the reference simulation. Linear phase, qualitative- and quantitative features and averages over time are all very similar. The spectrum differs at different stages of the fluctuations.

The toroidal mode number spectra of the three simulations are therefore compared in Fig. 5.17 both at the minimum level of transport (lower left) and at the peak of heat flux in time (upper left). The times at which the minimum and maximum are taken are summarized in Table 12. The heat flux spectrum at the times of maximum transport of all three simulations is similar. At the times of minimum transport, however, there are some differences. The main one is a peak at $n \approx 350$ for the reference case. This peak disappears once the averaging is not done over all flux surfaces but only in the regime between $x = 0.2$ and $x = 0.6$. The "reference cut" curve illustrates this and suggests that the (unphysical) Dirichlet type boundaries might affect the simulation at the edges.

Therefore, for nonlinear simulations, the radial boxes should always be chosen as large as possible to minimize the effect of the boundary conditions.

Most interesting is the differences in spectra between the maximum and the minimum heat transport. At the time of minimum transport, the spectrum is flatter and broader compared to the relatively peaked spectrum at maximum heat transport. This behavior is consistent among all three equilibria.

The radial distribution of heat flux   is shown in Fig. 5.17 at the time of maximum heat flux (upper right) and minimum heat flux (lower right). While at the time of maximum heat flux, there is one plateau of high heat flux between $x = 0.3$ and $x = 0.4$, there is a double peak structure for the time of minimum heat flux. The first smaller peak is at $x = 0.27$, and the second peak is at $x = 0.43$, where the temperature profile peaks. The first peak might be caused by the bad curvature region close to the axis for low x. Also here, the three equilibria do not differ much. Therefore, also the toroidal mode number spectrum and the radial distribution of heat flux is not strongly affected by the changes in geometry.

The flux surface averaged radial distribution of heat transport   changes over time. It is plotted as a function of time in Fig. 5.18. At times of low heat flux the radial distribution of heat flux becomes smaller and centers around the peak of the temperature profile at $x = 0.43$ with small "islands" of transport around $x = 0.25$ consistent with the previous observations. This behavior is universal for all three equilibria.

The mode structure   is depicted at the time of maximum transport of heat in Fig. 5.19 at the peak position of the temperature gradient ($x = 0.43$). The white lines depict the zero crossings of the x-averaged K_y field between which the bad curvature regions lie. In the α, z-plane, the heat flux is located inside the bad curvature regions for all three equilibria. The primary mode structures are the same with small differences due to the different shapes of the bad curvature regions. The structure in the x, z-plane is limited radially by the temperature gradient profile and along the field line by the bad curvature region. In the z-averaged x, α-plane, there are no bad curvature regions. Hence, the structure is expanded over the entire α range.
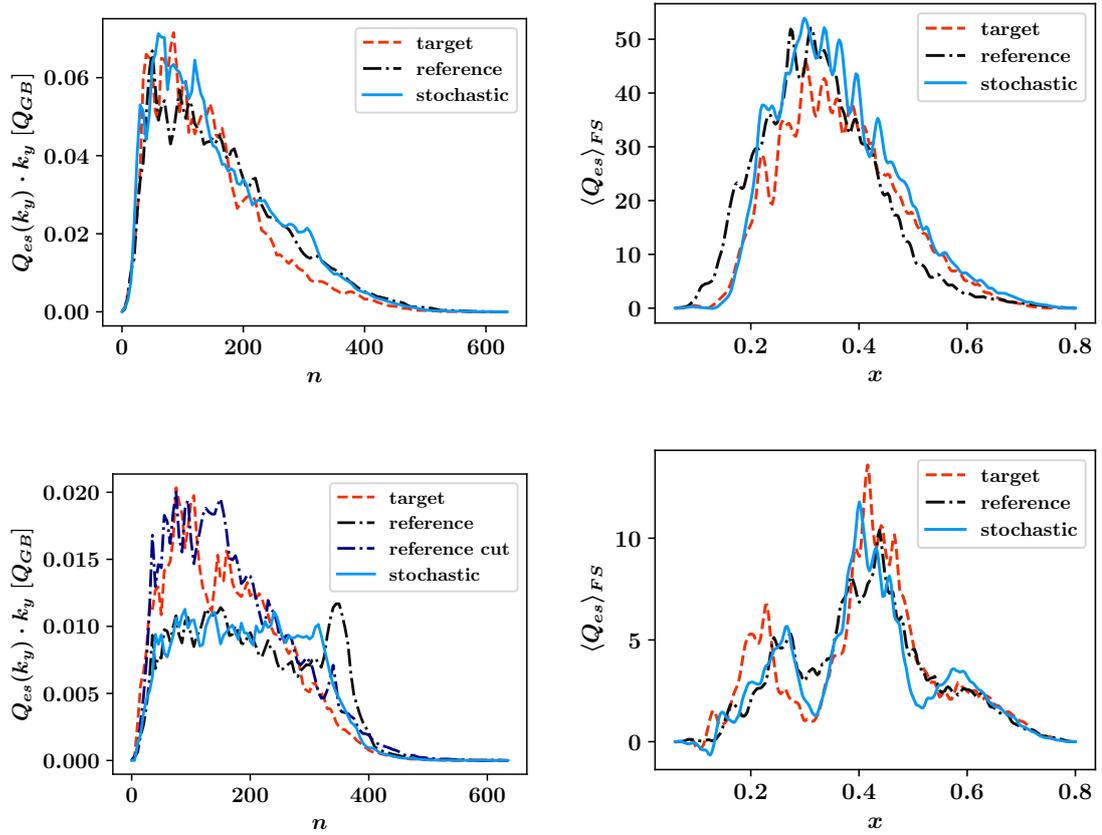
Figure 5.17: Toroidal mode number spectrum of heat flux (left) and flux surface averaged heat flux over x (right) at time of maximum heat flux (first row) and minimum heat flux (second row).
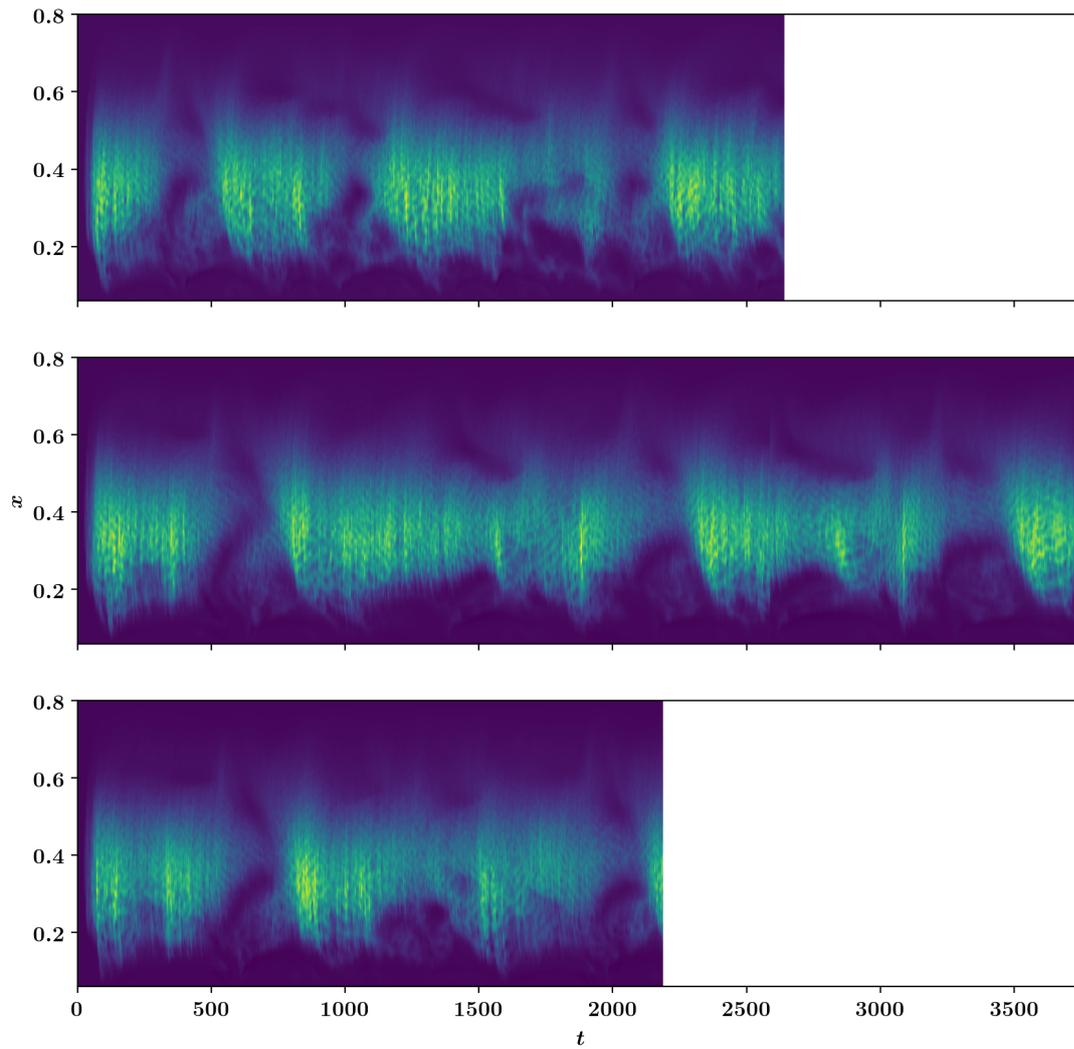
Figure 5.18: Heat flux over radial coordinate x and time for target equilibrium (upper) reference equilibrium (middle) and stochastic equilibrium (lower). When the volume averaged heat flux dips in Fig. 5.16 the radial extension of the flux surface averaged heat flux is compressed to the region around $x = 0.45$. This behavior is the same for all three equilibria.
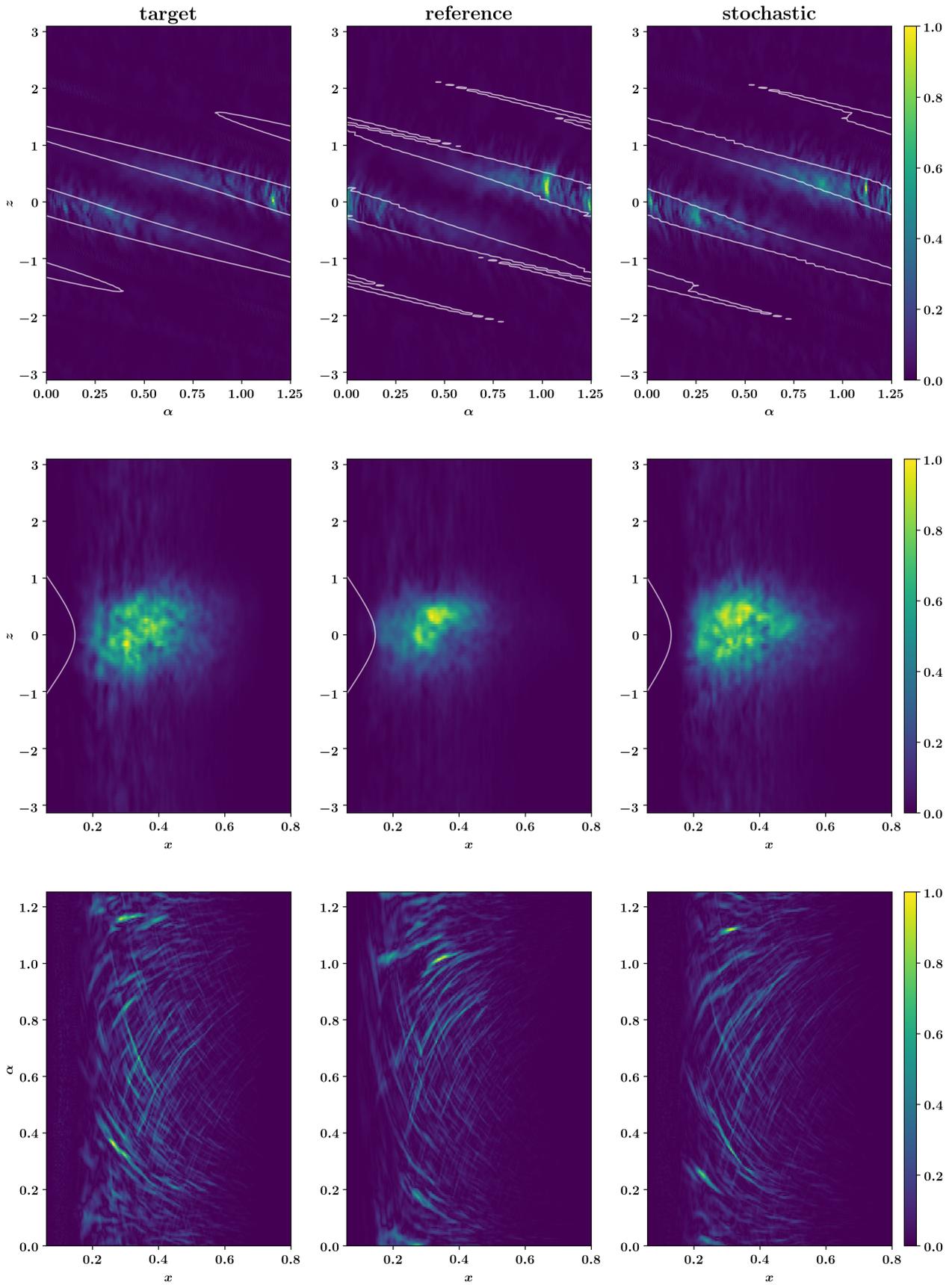
Figure 5.19: Heat flux Q_{es} plotted in three planes for all three equilibria.

	target	reference	stochastic
adiabatic linear			
profile 2: γ	0.202	0.173	0.172
profile 2: n	185	175	175
profile 7: γ	0.146	0.137	0.137
profile 7: n	105	110	110
kinetic linear			
$\kappa_{T_e} = \kappa_{T_i}$: γ	0.242	0.229	0.236
$\kappa_{T_e} = \kappa_{T_i}$: n	215	245	235
$\kappa_{T_e} = 0$: γ			0.086
$\kappa_{T_e} = 0$: n			65
adiabatic nonlinear			
Q_{es} $[Q_{GB}]$: mean	8.86	9.00	8.53
Q_{es} $[Q_{GB}]$: standard deviation	2.97	3.02	3.45
time of minimum heat flux	1022	590	660
time of maximum heat flux	2252	1882	859

Table 12: Summary of the results of this section. Overall, the differences between the three equilibria are small.

5.1.5   Conclusion

In this section, three different equilibria of Wendelstein 7-X have been investigated for differences in geometry and their effect on linear growth rates and mode numbers both for adiabatic electrons and kinetic electrons as well as the heat flux in turbulence simulations. The small differences in geometry only lead to small differences in results in all observed quantities for the chosen temperature and density profiles. This is a good sign both for GENE-3D not to be sensitive to small changes in input parameters but also for the engineers of future devices. The origin of these three equilibria are different coil optimization strategies. There is a tolerance for the shape of the magnetic field in which the plasma confinement quality does not suffer in regards to turbulent transport, judging from the results of this section.

Linearly, a shift to smaller scales from $n = 180$ to $n = 240$, along with higher growth rates are observed when adding an explicit electron species to the simulations. However, the main characteristics like the frequency remain the same, and hence even with kinetic electrons considered, the ITG mode is still the dominant mode in the system for the chosen temperature and density profiles. Here, future investigations could focus on kinetic electron simulations with realistic mass ratio and with different ratios of ion temperature gradient and electron temperature gradient to investigate whether this

observation is universal. Also, even with reduced mass ratio, higher resolutions are required to resolve ETG modes which have been shown not to be relevant in a flux-tube W7-X geometry [39]. An investigation of ETG modes in a 3D W7-X geometry could be envisioned for the future.

The nonlinear simulations show fluctuations in the volume-averaged heat flux over time. These fluctuations coincide with a widening of the toroidal mode number spectrum of the heat flux. At times of high heat flux, the spectrum is peaked between mode numbers $n = 50 \sim 200$ while at times of low heat flux, the spectrum has a plateau between $n = 50 \sim 350$. Also, the radial extent of the heat flux varies as it becomes more localized at times of low heat flux. This effect is universal for all geometries considered here and also shows for the high mirror configuration used in sections 4.2.1 and 5.2. Future investigations could further investigate the cause of this effect. In comparison, the width of the peak of the mode number spectrum in the linear cases is $\Delta n = 40$.

The goal should be to run collisional nonlinear electromagnetic multi-scale simulations with realistic mass ratio and temperature and density profiles close to the ones used in the experiment. These simulations could further show how ion-temperature gradient driven modes interact with trapped electron modes, electron temperature gradient driven modes, and other types of modes. For this goal, the performance of the code needs to be further improved. Ideas for future optimizations are outlined in Sec. 3.2.7.

5.2   Finite size effects on linear stability and turbulent transport

One of the first applications of the $x-$global version of GENE was to test the effect of varying the ratio of the characteristic gyroradius ρ^* to the machine size on the growth rate [61]. Here, the value of ρ^* is varied for fixed temperature and density profiles. The idea is, as the gyroradius increases, the mode becomes larger in comparison to the width of the temperature and density gradient peak. Because of this, the mode is only driven by some effective gradient. Hence, for larger values of ρ^* the growth rate decreases (see Fig. 5.20). This effect is observed to increase for decreasing profile widths $\Delta T, \Delta n$. For small values of ρ^*, the growth rate converges towards the local
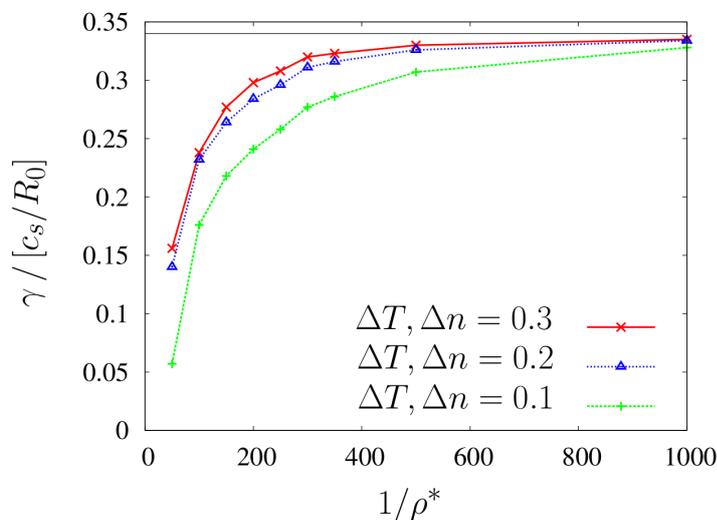
Figure 5.20: Growth rate of an ITG mode with adiabatic electrons at $k_y\rho_i \approx 0.3$ as function of the inverse ρ^* value using the peaked temperature and density gradient profiles (profile 2) for different profile widths. "Here, the radial simulation box length is kept fixed with respect to the gyroradius and the number of grid points is set to $(64 \times 16 \times 48 \times 16)$ in the $(x, z, v_\parallel, \mu)$ directions. The local code result using the maximum gradients is shown as a thin, black line." Cf. [61].

result, which is driven by the value of the gradient at the most unstable flux surface for the entire radial domain. In this subsection, this test is repeated with GENE-3D for a Wendelstein 7-X equilibrium. The VMEC equilibrium is the same as in Sec. 4.2.1 "wout_w7x2_rz300p128.nc". Sec. 5.2.1 investigates the linear growth rates both for a fully 3D geometry and for a reduced (y-global) model, which only takes one flux surface into account. Sec. 5.2.2 then looks more closely at the differences between the reduced model and the 3D geometries by keeping both the temperature gradient and the density gradient fixed over the radial simulation domain to isolate geometric effects. Lastly, Sec. 5.2.3 examines the finite ρ^* effects on nonlinear simulation results. Here, the geometric drive and the temperature gradient drive found in the linear sections are compared.

$1/\rho^*$	nx0	ny0	$k_x^{\min}$	$k_x^{\max}$	$k_y^{\min}$	$k_y^{\max}$	$n_{\max}$	nx0/L_x	ny0/L_y
70	240	384	0.224	53	0.161	61.8	1920	9.84	9.84
140	240	384	0.112	26.9	0.081	30.9	1920	4.92	4.92
180	240	384	0.087	20.9	0.063	24.0	1920	3.83	3.83
250	240	384	0.063	15.1	0.045	17.3	1920	2.76	2.76
280	240	384	0.056	13.5	0.040	15.5	1920	2.46	2.46
363	240	384	0.043	10.4	0.031	11.9	1920	1.90	1.90
500	240	384	0.031	7.5	0.023	8.7	1920	1.38	1.38
560	240	384	0.028	6.7	0.020	7.7	1920	1.23	1.23

Table 13: Parameter setup for the "open" simulations with a fixed number of points in x and y, nx0 = 240, ny0 = 384.

$1/\rho^*$	nx0	ny0	$k_x^{\min}$	$k_x^{\max}$	$k_y^{\min}$	$k_y^{\max}$	$n_{\max}$	nx0/Lx_a	ny0/L_y
70	31	37	0.224	7	0.161	6	185	1.11	0.95
140	62	75	0.112	7	0.081	6	375	1.11	0.95
180	80	96	0.087	7	0.063	6	480	1.11	0.95
250	111	133	0.063	7	0.045	6	655	1.11	0.95
280	125	149	0.056	7	0.040	6	745	1.11	0.95
363	162	193	0.043	7	0.031	6	965	1.11	0.95
500	223	266	0.031	7	0.023	6	1330	1.11	0.95
560	250	298	0.028	7	0.020	6	1490	1.11	0.95

Table 14: Parameter setup for the "closed" simulations with a fixed maximum $k_x^{\max} = 7.0$ and $k_y^{\max} = 6.0$. The number of points in x and y are scaled such that $k_u^{\min} \cdot$ nu0 $= k_u^{\max}$ with $u = (x, y)$
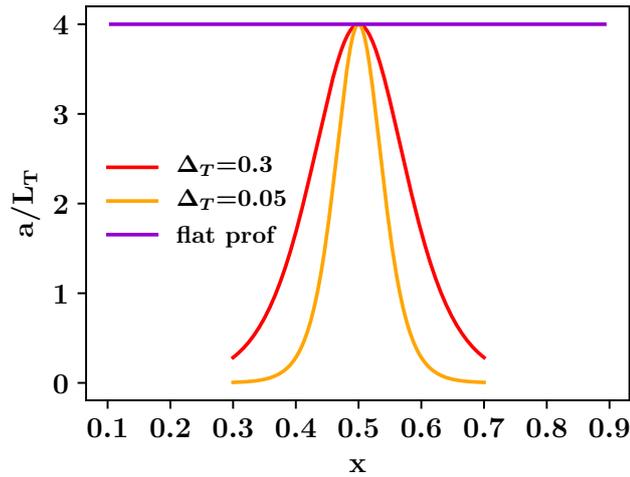
Figure 5.21: Logarithmic temperature gradient profiles used for the linear ρ^*-scan simulations. The peaked profile (profile 2) is plotted for two different widths. The flat profile is implicitly used by all y-global simulations and explicitly used by the 3D simulation in 5.2.2.

5.2.1 Linear ρ^*-scan

The test is done for three types of parameter sets. The first idea is to leave the resolution in x, and y-direction fixed and vary ρ^*. Here, the converged resolutions of the previous section are used, which were performed for the "native" $1/\rho^* = 363$. Hence, all simulations with a smaller $1/\rho^*$ are even higher resolved, as summarized in Table 13. The two simulations with a higher $1/\rho^*$ are still well resolved as the number of points per gyroradius is greater than 1, which is usually considered a rule of thumb limit. Keeping the resolution fixed for varying ρ^* leads to different values of $k_x^{\max}$ and $k_y^{\max}$. The temperature profile used is the peaked profile (profile 2) with a width of $\Delta_T = 0.3$ depicted in Fig. 5.21. In the simulations performed this way, modes with $k_y \rho_i > 2.5$ dominate the system for small values of $1/\rho^*$ as $k_y^{\max}$ increases with ρ^*. This is illustrated in Fig. 5.22d. The modes at high k_y have higher growth rates than their low k_y counterparts (see Fig. 5.22a). The high k_y modes also appear for higher values of $1/\rho^*$ if the number of points in y is chosen high enough. This was demonstrated in Sec. 4.2.1 in Table 6. This means that high k_y modes will dominate the linear physics of the system if they are resolved. This effect can be observed both for simulations with the full $3D$ geometry as well as for the reduced model with just a single flux surface (see Fig. 5.22). For the reduced model, only one simulation is run for the open case to save computational resources. This simulation considers the flux surface at $x_0 = 0.5$ for $1/\rho^* = 140$ with the same parameters as the 3D simulation.

Linear simulations with a limited maximum wavenumber

Because of the high k-modes the next idea is to fix the maximum wavenumber in x and y direction in an attempt to have comparable values of $k_y \rho_i$ and $k_x \rho_i$ for all values of ρ^*. This "closed" test for the ρ^* scan limits the maximum wave numbers to $k_x^{\max} \rho_i = 7.0$ and $k_y^{\max} \rho_i = 6.0$. The resolved wavenumbers are less than a quarter of these values as the $k_{\max}$ values represent the maximum wavenumber if one wavelength is resolved by one point, which in real space is obviously not enough. The parameters for this test are outlined in Table 14. The results are depicted as the navy blue dotted line in Fig. 5.22 and Fig. 5.23. Here, similar to the tokamak result a deviation of the growth rates for $1/\rho^* < 250$ between the single flux surface geometry (y-global, brown dashed line) and the 3D geometry is observed. While the y-global growth rates remain almost the same over the entire ρ^*-range, the growth rates of the 3D simulations reduce to less than half their value from $1/\rho^* = 560$ to $1/\rho^* = 70$.

The growth rates of the y-global simulations run at $x_0 = 0.50$ are sometimes smaller than the growth rates of the 3D simulations. This indicates that despite the temperature gradient peaking at $x = 0.5$, the most unstable flux surface is another.

(a) Growth Rate

(b) Frequency

(c) Wave number in x-direction

(d) Wave number in y-direction

(e) Toroidal wave number

(f) Poloidal wave number
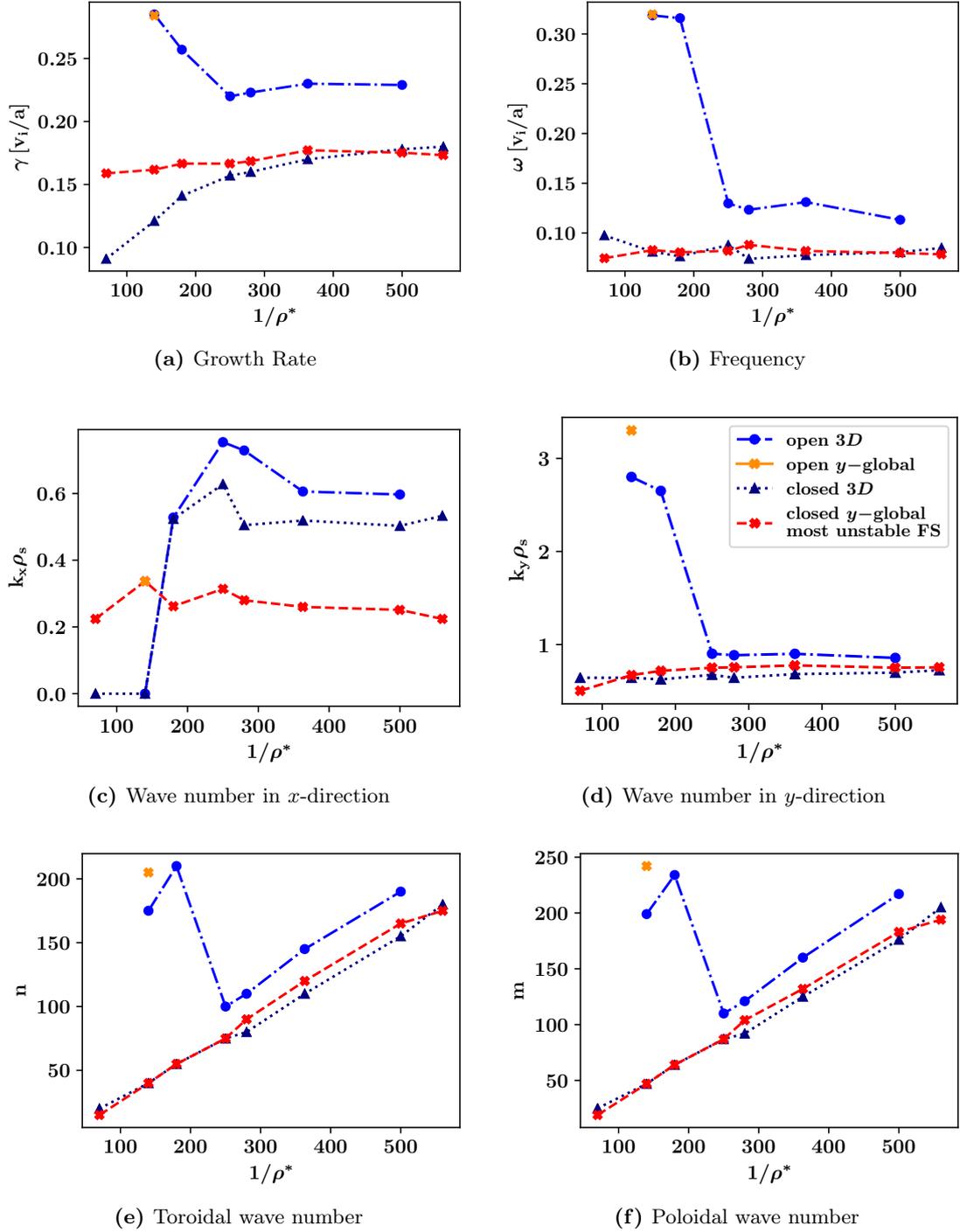
Figure 5.22: Scan over $1/\rho^*$ for "open" and "closed" parameter set as well as for single flux surface geometries and full 3D geometries. The red line represents y-global simulations performed at the most unstable flux surface. The values for growth rate and frequency are renormalized to be comparable to the other simulations performed at $x_0 = 0.5$.

(a) Growth Rate

(b) Frequency

(c) Wave number in x-direction

(d) Wave number in y-direction

(e) Toroidal wave number

(f) Poloidal wave number
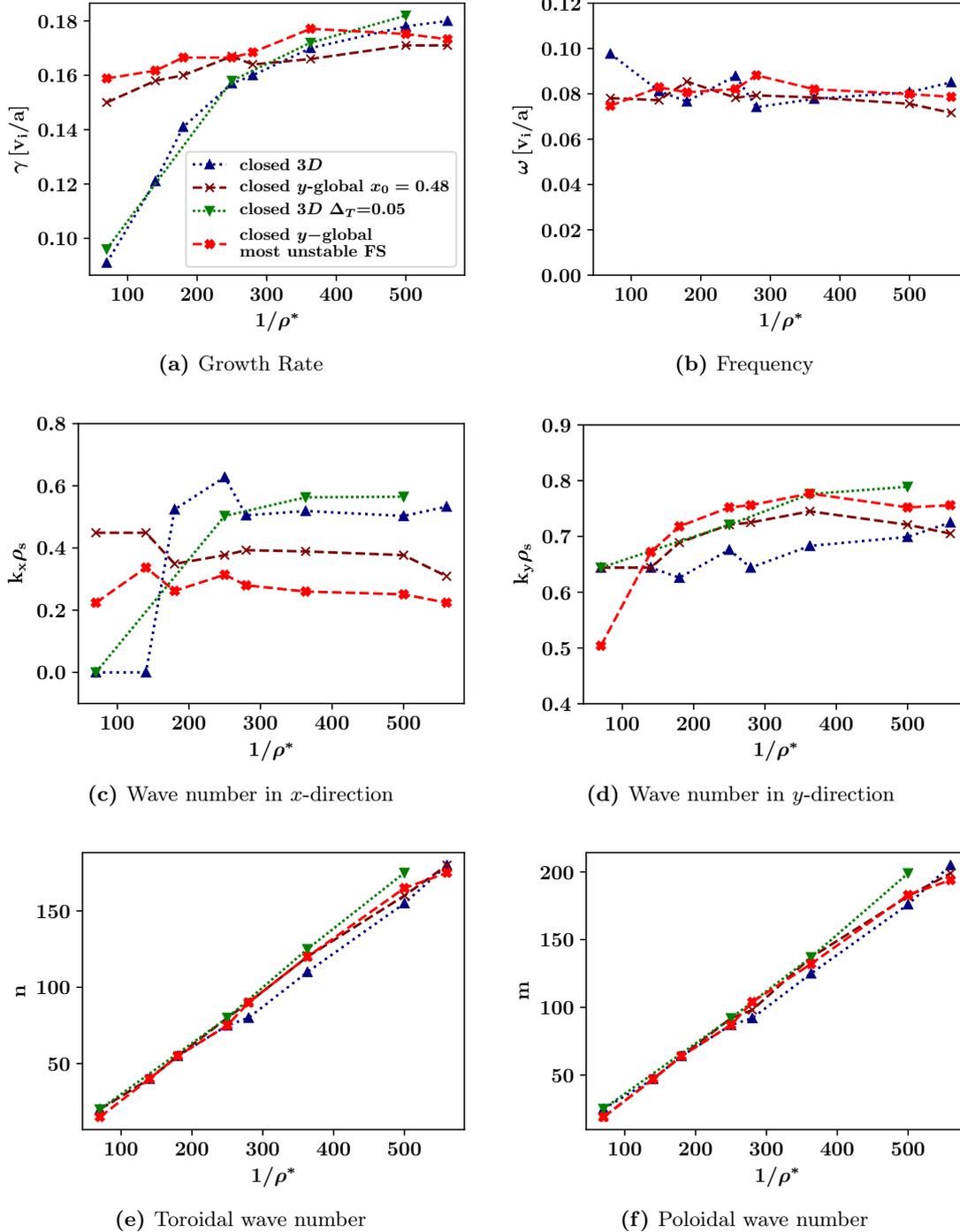
Figure 5.23: Scan over $1/\rho^*$ for "closed" parameter set for single flux surface geometries and full 3D geometries.
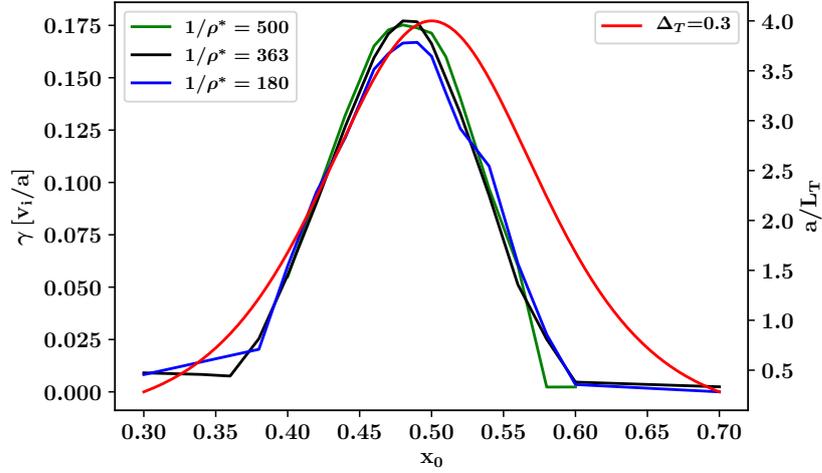
Figure 5.24: Growth rate over flux surface label x_0 for the (y-global) flux surface simulations. The most unstable flux surface for both $1/\rho^* = 180$ and $1/\rho^* = 363$ is not at $x_0 = 0.5$ where the temperature gradient peaks but at $x_0 = 0.48$.

Hence, a radial scan over the flux surfaces is performed and visualized in Fig. 5.24. For $1/\rho^* = 180$ and $1/\rho^* = 363$ the most unstable flux surface is at $x_0 = 0.48$ while it is at $x_0 = 0.47$ for $1/\rho^* = 500$. The red line in Fig. 5.23 depicts the y-global simulations at the most unstable flux surface, which have higher growth rates than the y-global simulations performed at $x_0 = 0.5$ and higher than the 3D results.

The dominant wavenumber of the instability in k_y is approximately the same over the entire ρ^* range $k_y \rho_i \approx 0.7$ for almost all simulations as illustrated in Fig. 5.23d. The poloidal mode number m can be determined by hand by counting the number of maxima of the mode around the poloidal plane in a plot like Fig. 5.25. The poloidal mode number increases with decreasing ρ^* as $k_y^{\min}$ decreases with ρ^* (see (3.41)). This result is also observed for the tokamak geometry investigated in Ref. [61] which is depicted in Fig. 5.25.

Comparing the 3D simulations with the y-global simulations for the Wendelstein 7-X case, the single flux surface geometry (y-global, brown and red dashed lines) yields similar results for all quantities except for the growth rate and the wave number in k_x. The difference in radial structure makes sense, given that only the full 3D simulation takes radial profile and geometry variations into account. The frequency, wave number in y-direction, and mode numbers are comparable.

A test with very narrow temperature and density profiles is performed ($\Delta_T = \Delta_n = 0.05$ compared to $\Delta_T = 0.3, \Delta_n = 0.1$, see Fig. 5.21) for the full 3D geometry with a fixed maximum k_x and k_y to test the dependence on the profile width. The results
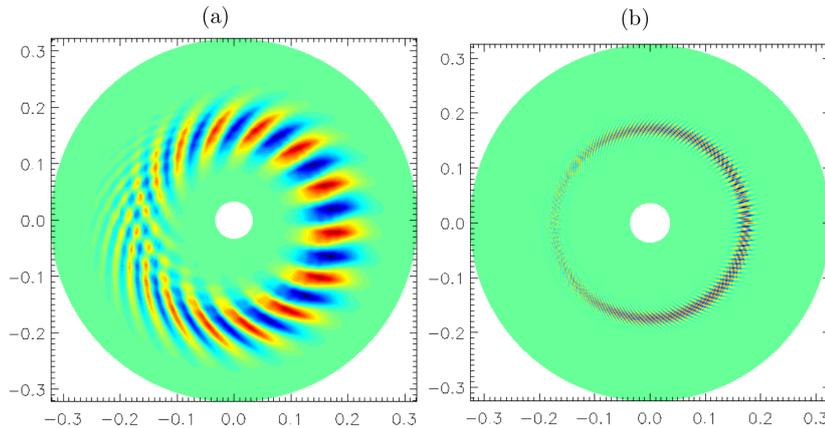
Figure 5.25: "Poloidal cross-section of the electrostatic potential for (a) $\rho^* = 1/100$ and (b) $\rho^* = 1/1000$." Temperature and density profiles are kept constant as well as the value of $k_y\rho_i = 0.3$ which is possible in the x-global version of GENE. As $k_y\rho_i$ is kept constant while ρ_i changes the mode poloidal number m increases for decreasing ρ^* Cf. [61].

are depicted as the green densely dotted line in Fig. 5.23. The results are similar to the previous tests with a limited maximum wave number. This leads to the conclusion that the modes do not need a large radial extent to grow at the maximum growth rate. The small deviations can be attributed to a slightly different hyper-diffusion value in y-direction of η_y=1.0 for the narrow profile compared to η_y=2.0 for the wider profile. The hyper-diffusion value can have a direct effect on the simulation results as tested in Sec. 4.2.1 as they damp higher mode numbers stronger than lower ones. This is illustrated in Fig. 5.23d, where the k_y wave numbers of the narrow profile are higher than the wavenumbers of the wide profile for all values of ρ^*.

Time to convergence of the growth rate   is a difference between the 3D simulations and the y-global simulations as the 3D simulations take longer to reach a steady state. One example is plotted in Fig. 5.26. While the closed 3D simulation is not converged even after 5000 time units, the y-global simulation reaches convergence after 2000 time units. Growth rates usually converge fast when one mode clearly dominates the system. When there are multiple modes on different flux surfaces or different k_y-modes with similar growth rates, convergence can take a long time. Hence, this result might indicate that geometrical effects are relevant here, which are not captured by a single flux surface geometry. Fig. 5.26 shows the most extreme example, but it is true for all simulations considered that convergence takes longer for the 3D geometries than for single flux surface geometries. Note, that other 3D simulations like the ones displayed in Fig. 5.26 do converge.
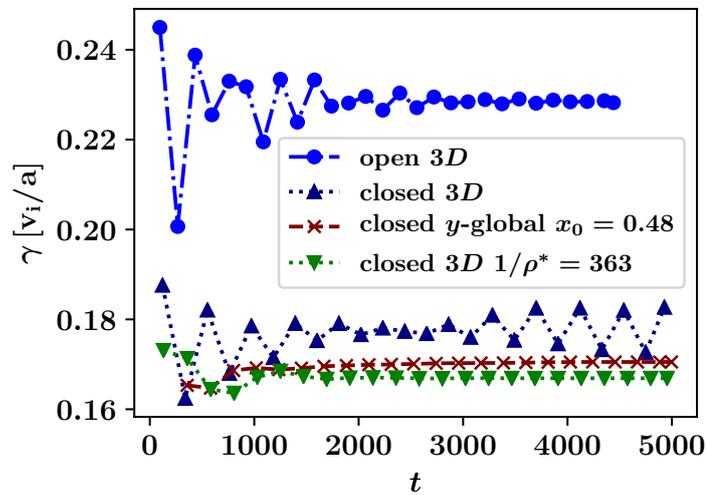
Figure 5.26: Time trace of the growth rate diagnostic with $1/\rho^* = 500$ for the open and closed 3D simulations and single flux surface geometry (y-global). The closed 3D case with $1/\rho^* = 500$ does not converge in the presented time window. While this is the most extreme example, it is true for all cases considered that when using the same parameters, the 3D simulations take longer to converge than the flux surface geometry simulations. Note, that other closed simulations like the one presented here with $1/\rho^* = 363$ do converge.

5.2.2 Comparison of flux surface geometry and 3D geometry

In the previous subsection, it was found that the most unstable flux surface is not necessarily where the temperature gradient is highest. Hence, one of the geometrical drive terms in Eq.(2.27) must provide an additional drive for the modes to grow stronger. The temperature and density profiles, as well as their gradients, are artificially set to their values at the gradient peak positions ($x = 0.5$) for the entire radial domain to isolate the influence of the geometric terms. This is illustrated as the flat profile in Fig. 5.21.

As a consequence, every flux surface gets the same drive from the ITG-term and the density gradient term. The only differences are now the geometrical quantities of the different flux surfaces. In the (y-global) flux-surface simulations, the geometrical quantities from one flux surface are assumed to be the same for all other flux surfaces. Therefore, a scan over different radial positions x_0 is performed to compare with the 3D simulation. The results are plotted in Fig. 5.27a.

The growth rate for the 3D simulation is $\gamma = 0.29$ (blue dotted line), which is higher than any of the "closed" 3D simulations with the peaked profile (profile 2) in Fig. 5.23. The square of the flux surface averaged electrostatic field potential $\langle \Phi^2 \rangle_{\mathrm{FS}}$ is plotted as the dodger blue line. It is normalized to fit into the plot dimensions only to show the localization of the mode. The mode peaks at $x = 0.157$, making this flux surface the most unstable from a geometric point of view. However, as the radial extent of the simulation box ends at $x = 0.1$, the Dirichlet boundary condition (see sec. 3.2.3) might already affect the mode structure through the gyroaverage. In the field-aligned coordinate system employed in GENE-3D (see sec. 2.4), the coordinate system diverges at the magnetic axis at $x = 0.0$. Hence, resolving the area close to the axis can lead to numerical instabilities as discussed in sec. 5.1.4.

The $y-$global simulations (red dashed line) show the most unstable flux surface to be at $x_0 = 0.15$. Only the simulation run at this flux surface can reproduce the 3D result for growth rate (Fig. 5.27a), frequency (Fig. 5.27b) and mode numbers (Fig. 5.27e). This explains why for the simulations visualized in Fig. 5.23 the y-global growth rates are higher at $x_0 = 0.48$ than at $x_0 = 0.50$. The most unstable flux surfaces are below $x = 0.5$. Fig. 5.27c might give an explanation as to why the modes prefer to grow around $x = 0.157$. Here, the curvature term K_y is plotted over x and z. The values of K_y are averaged over the y-dimension as the mode has a large extent over the y-domain (see Fig. 5.27d). The mode is localized where K_y is most negative and extends over a wide range in z. Regions of negative K_y are considered bad curvature regions where especially

ITG modes grow (see sec. 1.3). This result shows again that the simulation box in radial direction should be chosen as large as possible to capture all geometric effects.
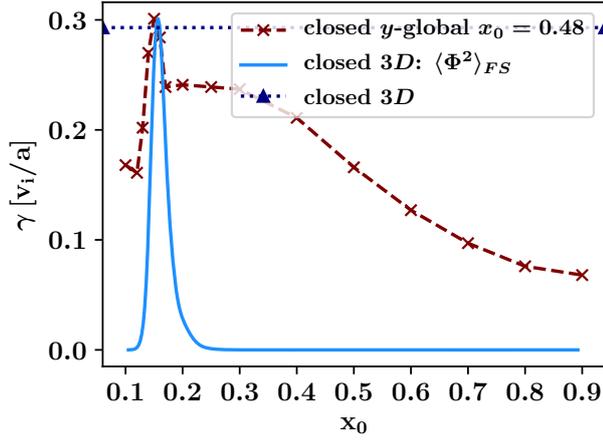
(a) Growth Rates of the 3D (dotted blue line) and y-global (red crosses) simulations. The dodger blue line depicts $\langle\Phi^2\rangle_{FS}$ normalized to fit into the plot to show the mode position for the 3D simulation.
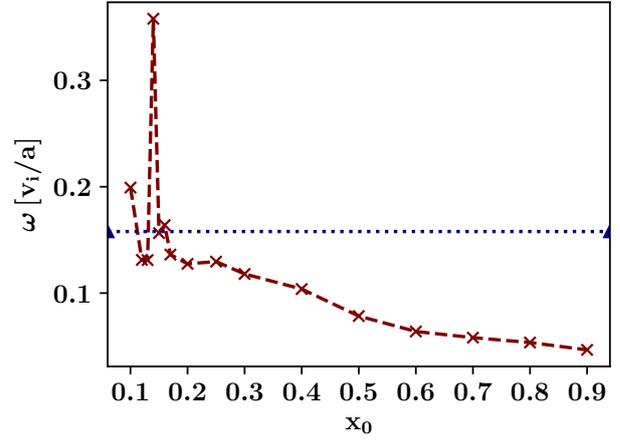

(b) Frequency of the flux surface scan.
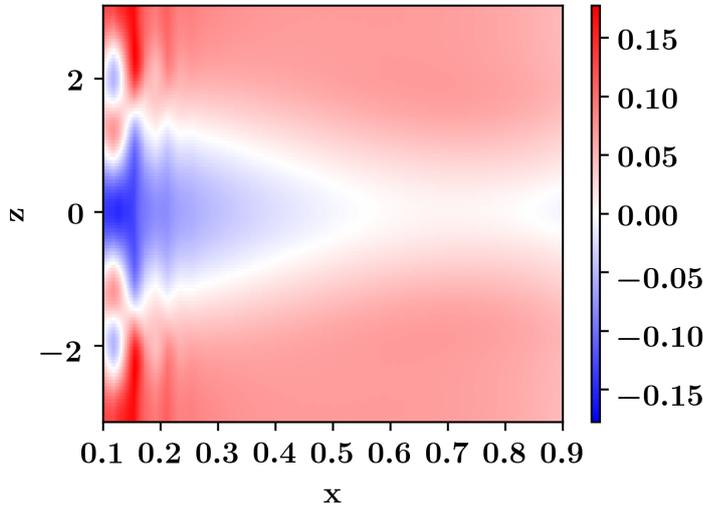
(c) Curvature term K_y plotted over x and z averaged over the entire y-range. This plot can be compared to the bottom row plots of Fig. 5.5. The W-7X geometry employed in this section has larger bad curvature regions than the ones in sec. 5.1.
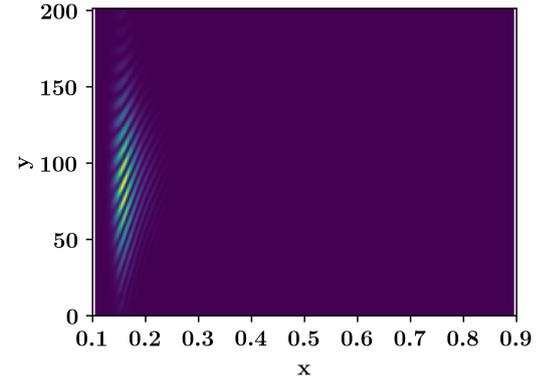
(d) Square of the electrostatic field potential Φ^2 averaged over z and plotted over x and y. The mode has a large extent in y-direction.
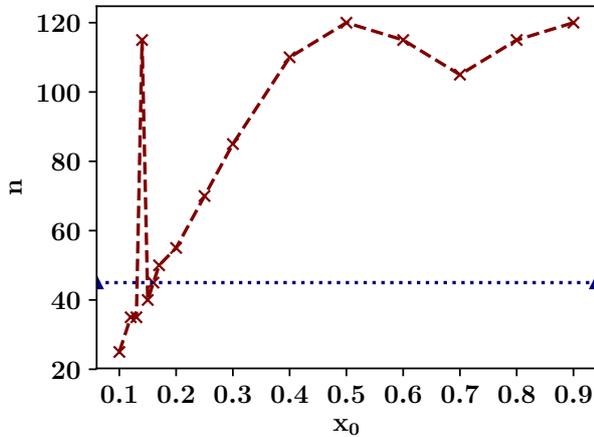
(e) Toroidal mode number n over x_0.

Figure 5.27: Simulation of a 3D geometry ranging from $x = 0.1$ until $x = 0.9$ with constant temperature gradient and density gradient. The mode peaks around $x = 0.15$ where the curvature is most negative. For the (y-global) flux surface simulations the growth rate, frequency and mode number of the 3D simulation can only be reproduced if the simulation is run exactly at the peak position of the 3D simulation.

5.2.3   Finite size effects in nonlinear Wendelstein 7-X simulations

The contribution of modes to the transport of heat reduces with the mode number [108]. Hence, high-k modes matter less in nonlinear simulations. The previous subsection showed that there are many modes with similar growth rates and different wave numbers present in a linear simulation of a 3D Wendelstein 7-X geometry. At the end of a converged linear simulation, however, only one mode is visible, while others with even just slightly lower growth rates do not appear anymore. In a nonlinear simulation, modes are coupled and can exchange energy. Due to saturation effects, the modes do not grow indefinitely, and turbulent eddies form. This subsection, therefore, investigates the effect of varying the ratio of characteristic gyroradius to the machine size on the nonlinear transport of heat. Due to constraints in computational resources, the tests of the previous subsection are only repeated for the values of $1/\rho^* = (180, 363, 500)$. Other than the added nonlinearity and the necessary sources with a value of `ck_heat` $= 0.06$, the chosen resolutions and other parameters are chosen to be identical to the linear "open" simulations in the previous subsection to make direct comparisons between the linear and nonlinear simulations possible.
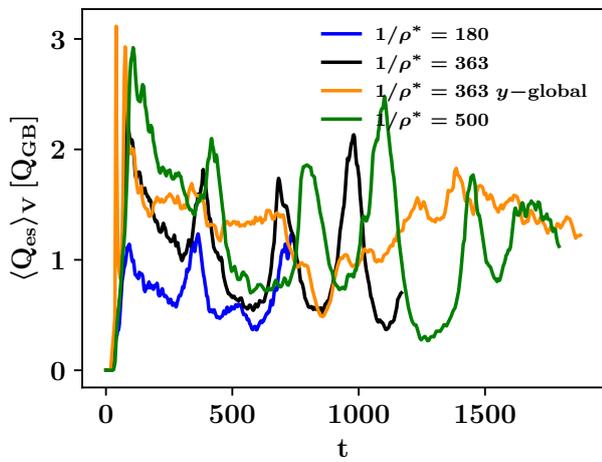
(a) Time trace of the volume averaged heat flux. The y-global heat flux is scaled down by a factor of 10 to fit into the plot range. The strong fluctuations are present for all 3D simulations but not for the y-global one.
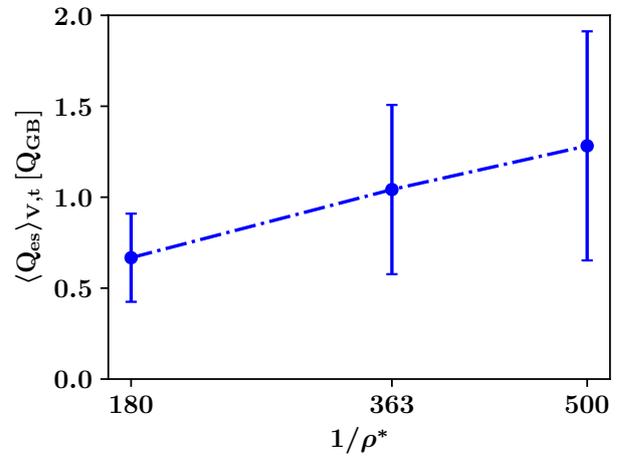
(b) Volume and time averaged heat flux $\langle Q_{\mathrm{es}} \rangle_{V,t}$ over inverse characteristic gyroradius $1/\rho^*$. The time average is performed over the entire time range. The values for the error bars are calculated as the standard deviation over the time window of $t = 0$ until $t = 740$ where the $1/\rho^* = 180$ simulation ends. Note, that this is not an "error" in the classic sense as the fluctuations are most likely a physical effect. Similarly to the linear growth rates, the heat flux increases with $1/\rho^*$. The fluctuation amplitude also increases with $1/\rho^*$.

Figure 5.28

The volume and time-averaged heat flux is presented in Fig. 5.28b. The heat flux increases with increasing $1/\rho^*$. This is in accordance with the results found for tokamaks in Ref. [109] where the fluxes in nonlinear x-global GENE simulations in a tokamak geometry follow a similar trend as the linear growth rates in Fig. 5.20.

Comparing 3D simulations with single flux surface simulations

The volume and time-averaged heat flux in the y-global simulation run at $x_0 = 0.5$ with the same parameters as the $1/\rho^* = 363$ 3D simulation yields a heat flux 12 times higher than the 3D result. This is illustrated in Fig. 5.28a, where the orange curve (y-global) needed to be scaled down by a factor of 10 to fit into the same plot with the 3D results. Here, the strong fluctuations present for all 3D simulations performed for the high-mirror configuration of Wendelstein 7-X are not visible for the single flux surface geometry. This leads to the conclusion that the fluctuations, also observed in Sec. 5.1, are a 3D geometric effect.

Any volume averaged quantities are not comparable between the 3D and the y-global simulations. This is because also for the flux surface simulations, a volume average is performed, including the radial dimension. The y-global simulations, however, only represent a single flux surface, which in reality does not have a radial extent. This difficulty is visualized in Fig. 5.29a. Here, the quadratic time and flux averaged transport of heat is plotted over x for all three nonlinear 3D simulations and the y-global simulation where the temperature gradient for the 3D simulations is plotted in red as a reference. While the 3D simulations all show a peaked structure in the middle, the flux surface simulation yields a heat flux with a wide radial extent as the temperature gradient profile is assumed to be constant over the radial domain.

The radial localization of the turbulent structures

The temperature gradient peaks at $x = 0.5$, and hence the ITG drive is strongest there. However, for all three 3D simulations, the quadratic heat flux peaks at a lower x. This is consistent with the observations of the previous subsection where the flux surface simulations had the strongest drive at $x_0 = 0.47 \sim 0.48$ despite the logarithmic temperature gradient peaking at $x = 0.5$. This can be explained by Fig. 5.27c, where the bad curvature region and with it the geometric drive is highest at $x = 0.15$. Which drive term is more important for a fixed geometry depends on the shape and peak of the temperature gradient curve. Also, a dependence on $1/\rho^*$ is observed as the main peak moves to lower x with decreasing $1/\rho^*$. The second peak at $x = 0.15$ is most pronounced for the $1/\rho^* = 180$ simulation. This can only be explained by the geometrical drive as the temperature gradient is practically zero in that region, as visualized in Fig. 5.29c. One explanation for this could be that for the smaller values of $1/\rho^*$ the (low k) linear
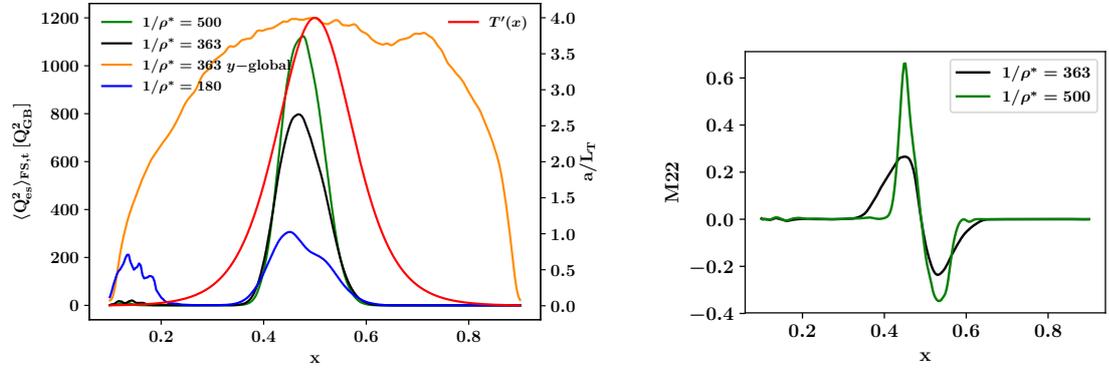
(a) Heat flux Q_{es} over inverse reference gyroradius ρ^*. The heat flux lowers for smaller $1/\rho^*$ compared to the width of the peak of the temperature profile. The y-global simulation does not show a peaked heat flux signature as the temperature gradient is flat. The "dome"-structure is due to the buffer zones in radial direction. The y-global heat flux had to be renormalized to fit into the plot.

(b) Time averaged source term $M22$. The sources are most active in the center of the simulation box to keep the logarithmic temperature gradient profile constant.
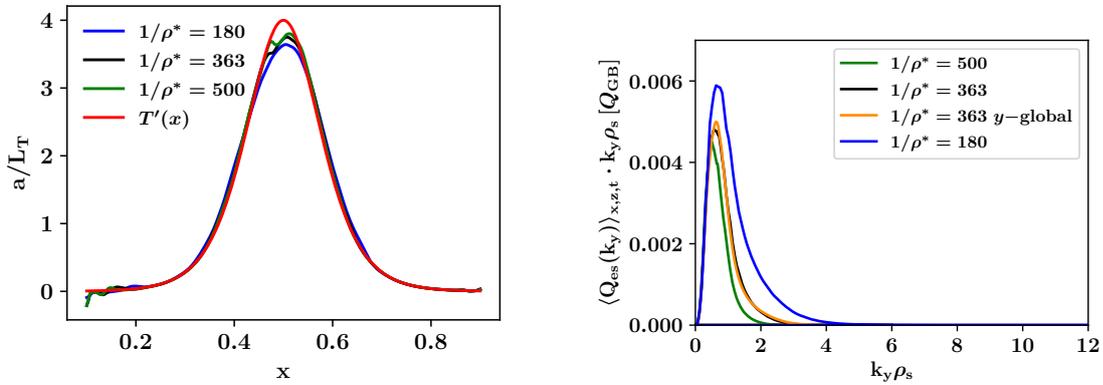
(c) Time averaged logarithmic temperature gradient of the simulations vs the prescribed profile. Small deviations to the prescribed profile (red) happen at the peak position and close to the inner boundary.

(d) Heat flux spectrum over k_y. The peak values are at $x_0 = (0.360.410.44)$ for $1/\rho^* = (500, 363, 180)$. The y-global result accurately reproduces the spectrum of the 3D simulation. The right flank is more extended for smaller values of $1/\rho^*$. The y-global heat flux had to be renormalized to fit into the plot.

Figure 5.29

modes, which underly the nonlinear simulations have a larger extent as illustrated in Fig. 5.30a. Hence, they "feel" less of the ITG drive in the center of the box, which is the same result found for the tokamak example in Ref. [61]. This is also visible as the width of the peak in the center of the simulation box in Fig. 5.29a. With increasing ρ^*, the peak widens so that the average of the temperature gradient curve over the peak width decreases. Hence, the effective ITG drive decreases, and with it, the heat flux due to the ITG term. However, due to their size, they expand more into the bad curvature region depicted in Fig. 5.27c, and hence the geometric drive becomes more important. The "dome"-structure of the y-global simulation is due to the Krook terms in buffer zones damping the fluctuations at the boundaries of the radial direction. Usually, they would be switched off for y-global simulations as periodic boundary conditions in the radial direction are employed. The terms are switched on also for the y-global simulation to test how far the damping reaches into the simulation domain. The Krook buffer is set to act on 5% of the radial domain at the inner and the outer boundary. However, only the volume between $x = 0.4 - 0.6$ is unaffected by the boundary treatment for a simulation with $x_{\min} = 0.1$ and $x_{\max} = 0.9$. Hence, the boundary also affects the geometric drive at $x = 0.15$ in the 3D simulations. A different boundary condition might be required to investigate the geometric effects close to the inner boundary in more detail in the future.

The spectrum in y-direction

The k_y-spectrum in Fig. 5.29d shows a higher value of k_y for the peak in heat flux for lower $1/\rho^*$. The peak values are at $k_y = (0.36, 0.41, 0.44)$ for $1/\rho^* = (500, 363, 180)$. This might again be due to the higher relative resolutions in combination with the hyper-diffusion acting strongest on the highest k-modes. Given that the linear simulations are run with the same hyper-diffusion values, a direct comparison between the linear and nonlinear simulations is possible. Interestingly, the wavenumber spectrum of the y-global flux surface simulation accurately represents the 3D result. The wavenumbers in y-direction in the nonlinear simulations are even lower than the ones of the "closed" linear simulations where the maximum value of k_y is limited. This confirms again the result that high-k modes matter less when considering the nonlinear transport of heat.

Visual comparisons

Poloidal cuts at $\phi = 0$ of the open and closed linear simulations and nonlinear simulations for $1/\rho^* = 180$ (a) and $1/\rho^* = 500$ (b) are presented in Fig. 5.30. In Fig. 5.30a the difference in mode number between the open and closed simulation becomes apparent. In the closed example the structures are larger for $1/\rho^* = 180$ than for $1/\rho^* = 500$ which is in line with the tokamak result presented in Fig. 5.25. For the open cases the poloidal mode numbers m are almost the same: $m_{180} = 234$ and $m_{500} = 217$. Note however, that

this is not a universal feature as also the $1/\rho^* = 500$ simulation would display high-k_y modes and therefore also high n, m modes if the resolution in y was chosen high enough (see Sec. 4.2.1).

In the nonlinear case (third column), one can see that the electrostatic field potential Φ does not show any fine structures like in the linear case. Both for linear and nonlinear simulations, the snapshot is taken at the last timestep at $\phi = 0$. The time-averaged heat flux Q_{es} is plotted in the fourth column. Here, the turbulent eddies also widen for the larger value of ρ^*, although the simulations are run with the same resolutions as the "open" simulations. This shows that the small scale high-k modes present in the "open" simulations do not dominate the nonlinear physics of the system. Also, the heat transport shows high values around $x = 0.15$ for $1/\rho^* = 180$, which are not as pronounced but still visible for $1/\rho^* = 500$. Hence, the linear physics described in the previous subsection is still visible in the nonlinear simulations, and the geometric drive also matters here.

Also the localizations of the linear modes of the open (a), closed (b) and the nonlinear time averaged heat flux at $x = 0.5$ (c) for $1/\rho^* = 180$ are plotted in Fig. 5.31. Also here, the nonlinear simulation does not follow the strong localization of the high-k mode but is rather spread out, similar to the mode structure of the "closed" simulation.

(a) $1/\rho^* = 180$

(b) $1/\rho^* = 500$
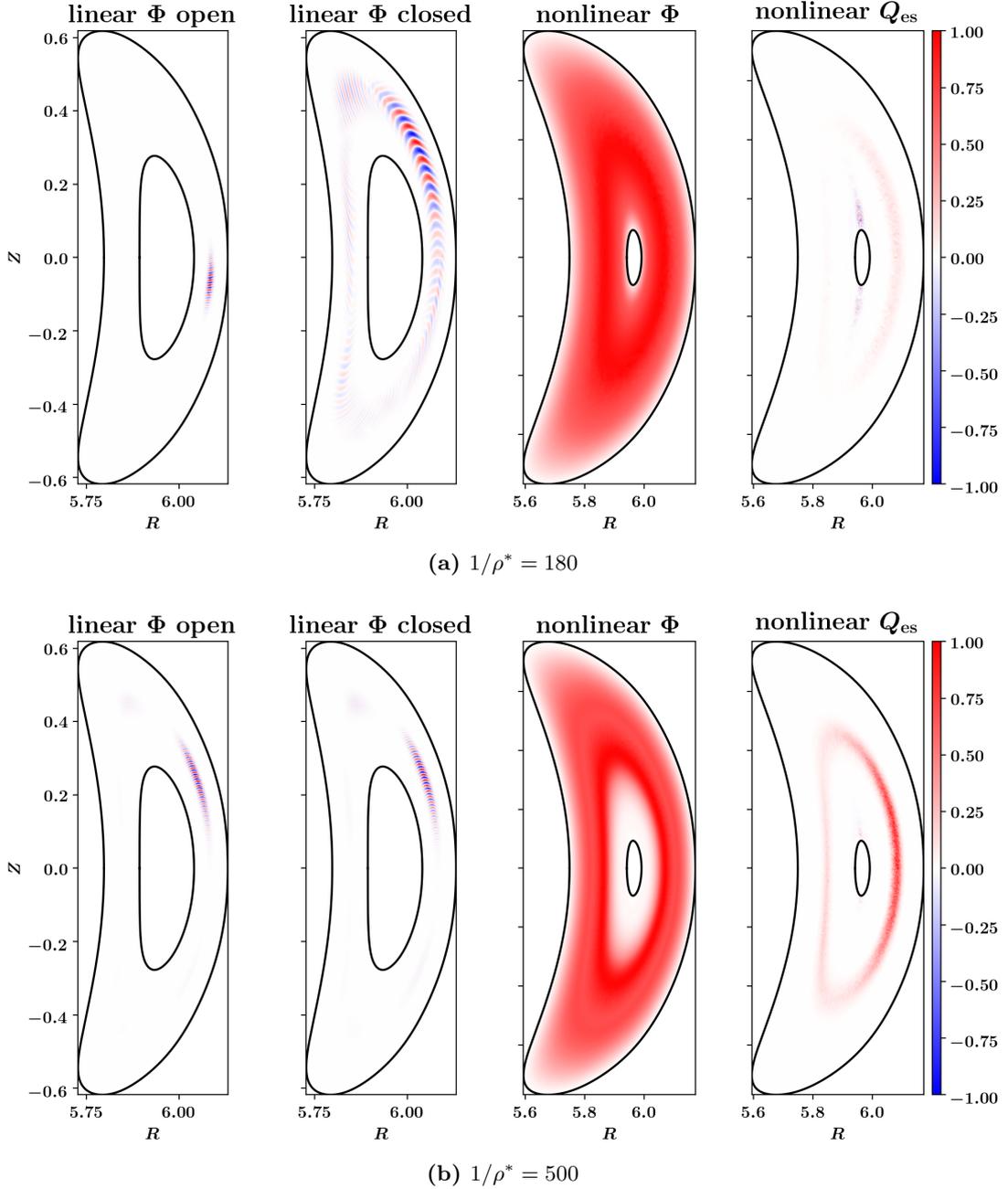
Figure 5.30: Poloidal cuts at $\phi = 0$ with the color coded electrostatic field potential fluctuation Φ and transport of heat Q_{es} normalized to range from -1 to 1 for $1/\rho^* = 180$ (a) and $1/\rho^* = 500$ (b). In (a), the differences between the "open" and "closed" simulations become apparent. The low-k mode of the closed simulation is less localized than the high-k mode of the open simulation. Comparing the closed simulation of (a) and (b) a larger radial extent of the mode is observed in (a) same as in the tokamak example in Fig. 5.25. Comparing with the nonlinear simulation, the potential does not show any fine structures. The heat flux shows features both of the linear low-k modes and the geometric drive at low x.

(a) Linear "open" simulation with $1/\rho^* = 180$. Color coded is the electrostatic field potential at $x = 0.5$.
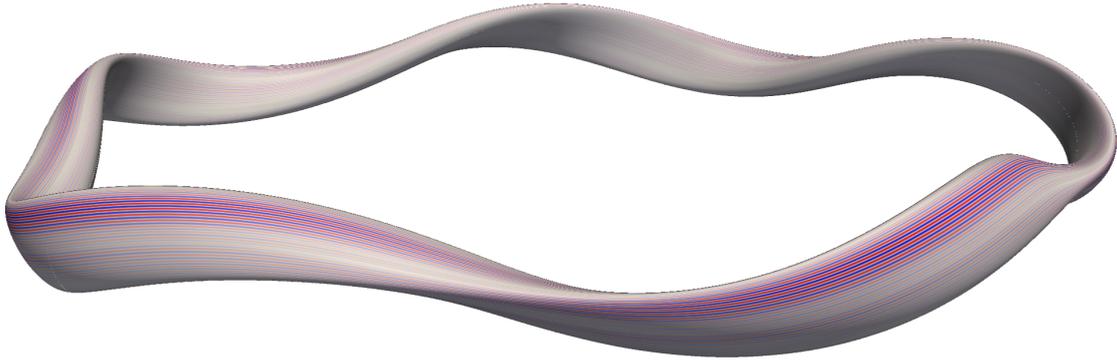
(b) Linear "closed" simulation with $1/\rho^* = 180$. Color coded is the electrostatic field potential at $x = 0.5$.
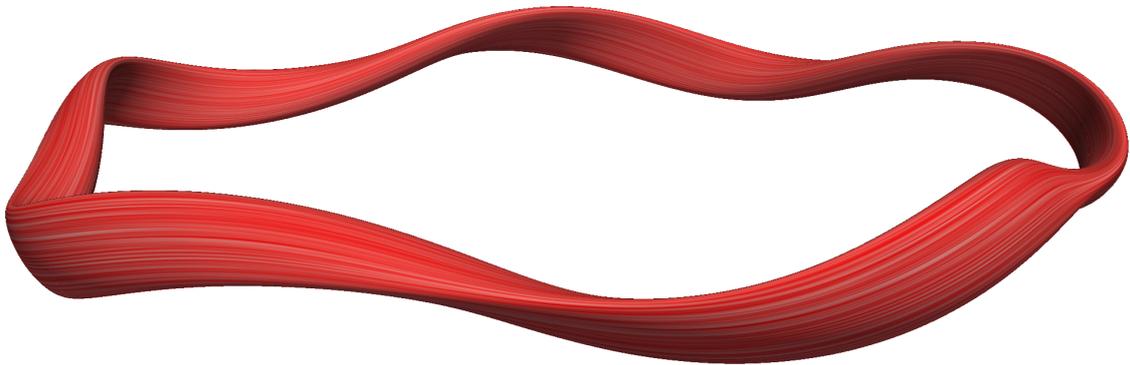
(c) Nonlinear "open" simulation with $1/\rho^* = 180$. Color coded is the electrostatic field potential at $x = 0.5$ at the last timestep.
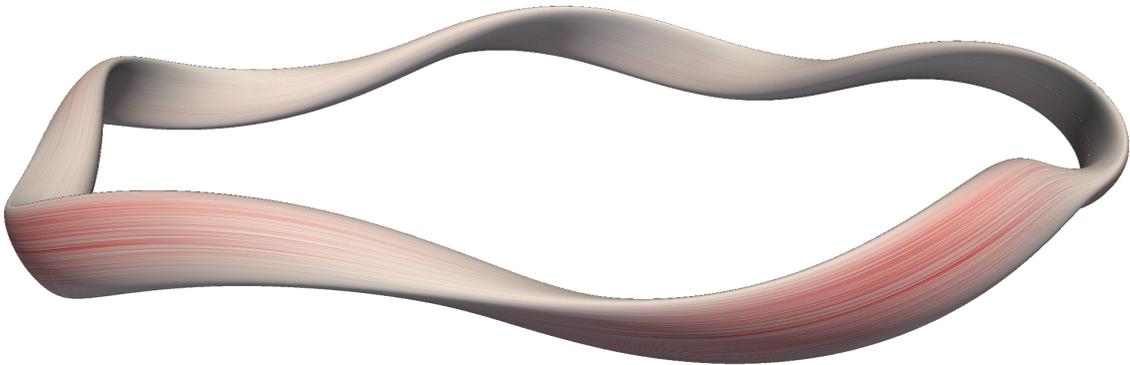
(d) Nonlinear "open" simulation with $1/\rho^* = 180$. Color coded is the nonlinear heat flux $\langle Q_{\mathrm{es}} \rangle_t$ at $x = 0.5$ averaged over time.

Figure 5.31: Surface plots of the linear open (a), linear "closed" (b) and nonlinear "open" simulations. The low k_y-mode in the closed simulation is less localized than the high-k_y modes in the open simulation. The structures in the nonlinear heat flux Q resemble more the low-k_y modes of the closed simulation as in nonlinear simulations low-k_y structures contribute more to the transport of heat (see Ref. [108]).

5.2.4   Conclusion

In this subsection, system size effects are investigated for linear growth rate, frequency, and mode numbers as well as for nonlinear transport of heat. Here a strong dependence on the number of grid points in y-direction is found. In principle, to investigate micro instabilities, resolutions need to be chosen such that further increasing the number of grid points does not change the outcome of the simulations. In the case of linear simulations of Wendelstein 7-X, increasing the resolution in y-direction leads to high k modes dominating the system. Systematically studying these modes is currently not feasible as for large values of $1/\rho^*$ more than 1000 grid points in y-direction are necessary to resolve them. While the high-k modes are physical, they do not play a significant role in nonlinear simulations. Hence, the linear ρ^* scaling test is performed with the number of grid points chosen, such that there is a maximum wavenumber k_y^{max} to focus the study on the low k modes. With this setup, it is found that the linear growth rates show a $1/\rho^*$ dependence similar to the tokamak results. For Wendelstein 7-X, the growth rates start to deviate from the local result for values of $1/\rho^* < 250$. This result is found to be independent of the width of the temperature profile but might be different for different profile types.

Furthermore, modes are found to grow in the regions of bad curvature when the temperature gradient and density gradient are set to constant values and the geometric drive terms in the Vlasov equation (3.4) are isolated. Here, one simulation considering the entire 3D geometry is enough by covering the entire radial x range while the reduced model needs to be run at many radial positions x_0 to recover the 3D result. Only the flux surface simulation run at the radial position of the peak value of the 3D simulation yields the correct growth rate, frequency, and mode number. Without a 3D code, the most unstable flux surface is, however, not known a priori.

Lastly, also the nonlinear transport of heat is also found to increase with increasing $1/\rho^*$. However, no gyroBohm breaking, which is expected from tokamak simulations, is found in the investigated parameter regimes. Here, further studies need to be conducted with different temperature profiles and possibly a more extensive range of $1/\rho^*$-values. The nonlinear turbulent structures are dominated by structures with even lower k_y than the linear simulations with a limited maximum k. However, also the geometric drive found in the linear simulations is relevant in nonlinear simulations and becomes apparent for smaller $1/\rho^*$ where the temperature gradient drive is not as strong due to the size of the structures compared to the temperature gradient curve.

6   Summary and Outlook

In the course of this work, the gyrokinetic code GENE-3D has been developed to describe plasma microturbulence in geometries with three-dimensional variations. The code has then been utilized to investigate the influence of geometric variations on turbulent transport in the context of stellarator optimization. The following gives a summary of the most critical findings of each section. Furthermore, an outlook for future code development and application is given.

6.1   Summary

Gyrokinetic theory and application to GENE-3D

In this section, the equations underlying GENE-3D have been briefly reviewed. Here, the assumptions and simplifications leading to the gyrokinetic model have been outlined. Then the gyrokinetic Vlasov equation and Poisson equation have been discussed. Emphasis has been put on the coordinate system necessary to describe non-axisymmetric toroidal magnetic equilibria. This section has created the basis for understanding the remainder of this thesis.

The development of the global stellarator version of GENE

Previous versions of GENE could only take geometric variations in two dimensions into account. To enable the investigation of 3D geometric effects in stellarators, almost all parts of the code needed to be rewritten. The gyromatrices and field matrices have been adapted for GENE-3D before the start of this Ph.D. project. Other parts of the code, like the treatment of boundary conditions, have been developed throughout this project. Also, to describe dependencies in the toroidal direction, a real-space implementation of the y-coordinate in GENE-3D has been essential. As large grids are required to capture all geometric details, the y-direction has then been parallelized. Obtaining the electrostatic field potential from the distribution function by solving the field equation is computationally the most expensive part of GENE-3D. Dedicated efforts have been made during this project to identify efficient solvers and an interface to the PETSc linear equation solver library has been implemented to improve the performance of GENE-3D. Also, new diagnostics have been developed throughout this project to compare the simulation results to codes like EUTERPE and XGC which employ a different coordinate system than GENE-3D.

Code verification and benchmarks

In this section, extensive benchmarks have been performed to ensure the correctness of GENE-3D. To this end, first benchmarks between GENE-3D and the tokamak version

of GENE have been performed for linear and nonlinear comparisons in an axisymmetric geometry, both with adiabatic and kinetic electrons. Here, both codes yield the same results for linear growth rates and frequencies and nonlinear transport of heat. Linear benchmarks between EUTERPE, GENE-3D, and XGC in a Wendelstein 7-X and an LHD geometry have been carried out to test the accurate description of 3D geometries. The results of the three codes agree for growth rates, mode numbers, and frequencies. GENE-3D can thus be considered fully operational for linear and nonlinear investigations of 3D geometries with one or multiple particle species.

Impact of geometry variations on turbulent transport in Wendelstein 7-X

In this section, GENE-3D has been utilized in the context of optimizing stellarators for turbulent transport. Three MHD equilibria for Wendelstein 7-X stemming from different coil optimization strategies are investigated for linear stability and nonlinear transport of heat. This way, the sensitivity of GENE-3D to changes in the magnetic geometry is tested. It is found that - for the considered parameter regimes - geometric variations do not have a substantial effect on turbulent transport. This result indicates that future stellarators can be planned with higher construction tolerances to reduce the time and cost of production.

Furthermore, GENE-3D is utilized to analyze finite-size effects in Wendelstein 7-X, and single flux surface simulations are compared to 3D simulations. While the flux surface simulations generally yield the correct spectrum in y-direction, the radial structures of the 3D simulations cannot be reproduced. For the linear ρ^*-scaling, results similar to the ones obtained with the tokamak version of GENE for axisymmetric geometries have been found if the maximum wave numbers are limited. Without this limitation, high-k modes have been found which dominate the linear physics of the system. In the nonlinear simulations performed in this section the structures have even lower wave numbers than in the linear simulations where the maximum wave number is limited. Hence, the high k-modes found in linear simulations do not play a significant role for the nonlinear dynamics.

The nonlinear heat flux increases with increasing $1/\rho^*$. However, contrary to the tokamak results in previous publications no strong gyroBohm breaking is found. Here, more studies with different geometries and temperature profiles are needed.

Lastly, geometric drive terms have been found in linear simulations, which in simulations considering the full 3D geometry affect the nonlinear radial transport of heat in regions with almost no drive from the temperature gradient. This drive is strong in regions of bad curvature. These results have demonstrated the importance of 3D simulations as the geometrical drive is only visible when taking the corresponding flux surfaces into account.

6.2   Outlook

Current possibilities

The development of GENE-3D opens many opportunities for future research projects. As even the most modern stellarator Wendelstein 7-X only has a minor radius of 0.5m compared to ITER (2m), the validity of local models can be questioned. With the current version of GENE-3D, already stellarators and tokamaks with resonant magnetic perturbations can be investigated for ion scale turbulence. Hence, GENE-3D can be used to investigate the importance of zonal flows in 3D stellarator geometries and the level of the Dimits shift compared to tokamak geometries. Zonal flows act as a secondary instability which damp the primary instabilities present in linear simulations. Unfortunately, when the temperature gradient is further increased, the zonal flow structures also break up due to tertiary instabilities. The resulting up-shift of the critical temperature gradient is called Dimits shift.

Also, more studies on the geometric drive encountered for Wendelstein 7-X in Sec. 5 could be performed to further investigate its origin. Here, comparisons with other stellarator geometries might be interesting.

Future development of GENE-3D

Currently, developments are ongoing to extend GENE-3D to take electromagnetic effects into account. Also, a collision term is implemented while simulations have not been feasible as collisions add further resolution requirements and constraints on the time step, making the simulations more expensive. For the goal of a numerical stellarator, GENE-3D needs further performance improvements to capture more physical effects while keeping the cost of the simulations low. To this end Sec. 3.2.7 has outlined some ideas for future performance optimizations. One of the most critical steps to take is the implementation of a multigrid field solver with a preconditioner adapted to the matrices present in GENE-3D. Future supercomputers will most likely implement a heterogeneous computing architecture with a combination of CPUs and GPUs. Hence porting the field solving process to the GPU might further increase parallelism. Also, GPUs are highly optimized for matrix-vector multiplications, which is essential for iterative linear equation solvers. Another way to improve performance is by using the block-structured grids which have been successfully tested for the tokamak version of GENE [93, 94, 95].

Future application of GENE-3D

These optimizations could pave the way to more sophisticated physical scenarios, including nonlinear simulations employing kinetic electrons and taking electromagnetic effects into account. Using single flux surface simulations, ETG-driven turbulence has

been found to play minor role for turbulent transport in Wendelstein 7-X compared to ITG turbulence. These investigations could be repeated taking the full 3D geometry into account. If ETG turbulence is found to be important in 3D simulations, multiscale interactions between ion scale turbulence and electron scale turbulence could then be investigated. Also, temperature, density, and pressure profiles derived from experiments could serve as input both to GVEC and GENE-3D to simulate scenarios closer to the experiments. In combination with other tools like ONSET, GENE-3D could then be utilized to optimize the next generation of fusion experiments for turbulent transport. Turbulence optimization will be a necessary step towards confinement times long enough for a commercial application of a fusion reactor.

References

[1] P. Gerland, A. E. Raftery, H. Ševčíková, et al. World population stabilization unlikely this century. *Science*, 346:234–237, 2014.

[2] Statistical review of world energy (2019). https://www.bp.com/content/dam/bp/business-sites/en/global/corporate/pdfs/energy-economics/statistical-review/bp-stats-review-2019-full-report.pdf, 2019.

[3] W. C. Wang, Y. L. Yung, A. A. Lacis, et al. Greenhouse effects due to man-made perturbations of trace gases. *Science*, 194:685–690, 1976.

[4] S. H. Schneider. The greenhouse effect: Science and policy. *Science*, 243:771–781, 1989.

[5] H. P. Peters, G. Albrecht, L. Hennen, and U. Stegelmann. Die Reaktionen der Bevölkerung auf die Ereignisse in Tschernobyl - Ergebnisse einer Befragung. Technical report, Forschungszentrum Jülich, Jülich, 1987.

[6] G. Steinhauser, A. Brandl, and T. E. Johnson. Comparison of the chernobyl and fukushima nuclear accidents: A review of the environmental impacts. *Science of The Total Environment*, 470-471:800 – 817, 2014.

[7] R. Saidur, N.A. Rahim, M.R. Islam, and K.H. Solangi. Environmental impact of wind energy. *Renewable and Sustainable Energy Reviews*, 15:2423 – 2430, 2011.

[8] Predictions for the pv industry in 2019. https://cdn.ihs.com/www/pdf/0219/IHS-Markit-Predictions-for-the-PV-industry-2019.pdf, 2019.

[9] J. D. Lawson. Some criteria for a power producing thermonuclear reactor. *Proceedings of the Physical Society. Section B*, 70:6–10, 1957.

[10] Wesson J. A. *Tokamaks*. Oxford University Press, 2011.

[11] R. Dux. Plasmaphysik und Fusionsforschung, Teil I: Plasmaphysik. Universität Augsburg, 2001.

[12] ITER - the way to new energy. https://www.iter.org/, 2020.

[13] T. Klinger, T. Andreeva, S. Bozhenkov, et al. Overview of first Wendelstein 7-X high-performance operation. *Nucl. Fusion*, 59:112004, 2019.

[14] T. Sunn Pedersen, R. König, M. Krychowiak, et al. First results from divertor operation inWendelstein 7-X. *Plasma Physics and Controlled Fusion*, 61:014035, 2018.

[15] S.A. Henneberg, M. Drevlak, C. Nührenberg, et al. Properties of a new quasi-axisymmetric configuration. *Nuclear Fusion*, 59:026014, 2019.

[16] Asdex upgrade fusion experiment. https://www.ipp.mpg.de/987491/AUG_engl.pdf, 2020.

[17] Robert Arnoux. Taking the big leap. https://www.iter.org/newsline/179/731, 2011.

[18] M. Maurer. On the Limitation of the Dimits Shift in Plasma Microturbulence. Bachelor thesis, Freie Universität Berlin, 2011.

References

[19] J. Hüser. Modellierung und gyrokinetische Simulation des Dimits Shifts. Diploma thesis, Universität Münster, 2010.

[20] X. Garbet, Y. Idomura, L. Villard, and T.H. Watanabe. Gyrokinetic simulations of turbulent transport. *Nuclear Fusion*, 50:043002, 2010.

[21] F. Jenko, W. Dorland, M. Kotschenreuther, and B. N. Rogers. Electron temperature gradient driven turbulence. *Physics of Plasmas*, 7:1904–1910, 2000.

[22] W. W. Lee. Gyrokinetic approach in particle simulation. *The Physics of Fluids*, 26:556–562, 1983.

[23] C. S. Chang, S. Ku, P. H. Diamond, et al. Compressed ion temperature gradient turbulence in diverted tokamak edge. *Physics of Plasmas*, 16:056108, 2009.

[24] G. Jost, Tran T. M., K. Appert, et al. Global linear gyrokinetic pic simulations in 3D magnetic configurations. *Proc. Joint Varenna-Lausanne International Workshop Theory of Fusion Plasmas*, 2000.

[25] E. Sanchez, R. Kleiber, R. Hatzky, et al. Linear and nonlinear simulations using the EUTERPE gyrokinetic code. *IEEE Transactions on Plasma Science*, 38:2119–2128, 2010.

[26] A. Bottino, A. G. Peeters, R. Hatzky, et al. Nonlinear low noise particle-in-cell simulations of electron temperature gradient driven turbulence. *Physics of Plasmas*, 14:010701, 2007.

[27] S. Jolliet, A. Bottino, P. Angelino, et al. A global collisionless pic code in magnetic coordinates. *Computer Physics Communications*, 177:409 – 425, 2007.

[28] 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*, 12:122305, 2005.

[29] J. Büchner. Vlasov-code simulation. *Advance methods for space simulations*, pages 23–46, 2007. cited By 4.

[30] T. Görler, X. Lapillonne, S. Brunner, et al. The global version of the gyrokinetic turbulence code gene. *Journal of Computational Physics*, 230:7053 – 7071, 2011.

[31] M. Maurer, A. Bañón Navarro, T. Dannert, et al. GENE-3D: A global gyrokinetic turbulence code for stellarators. *Journal of Computational Physics*, 420:109694, 2020.

[32] F. Jenko and the GENE development team. The gene code, 2019. http://genecode.org.

[33] M. D. J. Cole, R. Hager, T. Moritaka, et al. Verification of the global gyrokinetic stellarator code xgc-s for linear ion temperature gradient driven modes. *Physics of Plasmas*, 26:082501, 2019.

[34] P. Xanthopoulos, H. E. Mynick, P. Helander, et al. Controlling turbulence in present and future stellarators. *Phys. Rev. Lett.*, 113:155001, 2014.

[35] F. Albert. Study of ITG Turbulence in the presence of magnetic perturbations in ASDEX Upgrade Tokamak using GENE-3D. Master thesis, Universität Stuttgart, 2019.

[36] A.E. Costley. On the fusion triple product and fusion power gain of tokamak pilot plants and reactors. *Nuclear Fusion*, 56:066003, 2016.

[37] F. Merz. *Gyrokinetic Simulation of Multimode Plasma Turbulence*. PhD thesis, Westfälische Wilhelms-Universität Münster, 2008.

[38] J. H. E. Proll, P. Helander, J. W. Connor, and G. G. Plunk. Resilience of quasi-isodynamic stellarators against trapped-particle instabilities. *Phys. Rev. Lett.*, 108:245002, 2012.

[39] G. G. Plunk, P. Xanthopoulos, G. M. Weir, et al. Stellarators resist turbulent transport on the electron larmor scale. *Phys. Rev. Lett.*, 122:035002, 2019.

[40] H. Lewy R. Courant, K. Friedrichs. Über die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, 1928.

[41] J. F. Lobsien. *Tools for designing the next generation of stellarators*. PhD thesis, Universität Greifswald, 2009.

[42] T. Andreeva. Vacuum magnetic configurations of Wendelstein 7-X. http://edoc.mpg.de/282107, 2002. IPP III/270.

[43] M. Drevlak. Optimization of heterogenous magnet systems. http://www-fusion.ciemat.es/sw2005/conferences/madison99/Abstract/Drevlak.pdf, 1999. IPP-EURATOM Association.

[44] A. Reiman, G. Fu, S. Hirshman, et al. Physics design of a high-bbeta quasi-axisymmetric stellarator. *Plasma Physics and Controlled Fusion*, 41:B273–B283, 1999.

[45] T. Pedersen, M. Otte, S. Lazerson, and P. Helander. Confirmation of the topology of theWendelstein 7-X magnetic field to better than 1:100,000. *Nature Communications*, 7:13493, 2016.

[46] T. Rummel, K. Risse, H. Viebke, et al. Accuracy of the construction of the superconducting coils forWendelstein 7-X. *IEEE Transactions on Applied Superconductivity*, 14:1394–1398, 2004.

[47] H.-S. Bosch, R. Brakel, T. Braeuer, et al. Final integration, commissioning and start of theWendelstein 7-X stellarator operation. *Nuclear Fusion*, 57:116015, 2017.

[48] R.L. Orbach. Statement about the future of the princeton plasma physics laboratory. *Under Secretary for Science and Director, Office of Science, U.S. Department of Energy*, 2008.

[49] J. P. Kremer. *The creation and first studies of electron plasmas in the Columbia non-neutral torus*. PhD thesis, Columbia University, 2007.

[50] T. Sunn Pedersen, J. P. Kremer, R. G. Lefrancois, et al. Construction and initial operation of the columbia nonneutral torus. *Fusion Science and Technology*, 50:372–381, 2006.

[51] K. C. Hammond, A. Anichowski, P. W. Brenner, et al. Experimental and numerical study of error fields in the CNT stellarator. *Plasma Physics and Controlled Fusion*, 58:074002, 2016.

[52] T. Sunn Pedersen, J. P. Kremer, R. G. Lefrancois, et al. Experimental demonstration of a compact stellarator magnetic trap using four circular coils. *Physics of Plasmas*, 13:012502, 2006.

References

[53] J.-F. Lobsien, M. Drevlak, T. Sunn Pedersen, and W7-X Team. Stellarator coil optimization towards higher engineering tolerances. *Nuclear Fusion*, 58:106013, 2018.

[54] A. Dinklage, C. D. Beidler, P. Helander, et al. Magnetic configuration effects on theWendelstein 7-X stellarator. *Nature Physics*, 14:855–860, 2018.

[55] F. Hindenlang and O. Maj and E. Strumberger and M. Rampp and E. Sonnendrücker. GVEC: A newly developed 3D ideal MHD Galerkin Variational Equilibrium Code. Presentation given in 'Simons Collaboration on Hidden Symmetries and Fusion Energy', 2019. https://hiddensymmetries.princeton.edu/meetings/simons-hour-video-talks.

[56] E. A. Frieman and Liu Chen. Nonlinear gyrokinetic equations for low-frequency electromagnetic waves in general plasma equilibria. *Phys. Fluids*, 25:502–508, 1982.

[57] A. J. Brizard and T. S. Hahm. Foundations of nonlinear gyrokinetic theory. *Rev. Mod. Phys.*, 79:421–468, 2007.

[58] J. A. Krommes. The gyrokinetic description of microturbulence in magnetized plasmas. *Annual Review of Fluid Mechanics*, 44:175–201, 2012.

[59] G. G. Howes, S. C. Cowley, W. Dorland, et al. Astrophysical gyrokinetics: Basic equations and linear theory. *The Astrophysical Journal*, 651:590–614, 2006.

[60] T. Dannert. *Gyrokinetische Simulation von Plasmaturbulenz mit gefangenen Teilchen und elektromagnetischen Effekten*. PhD thesis, Technische Universität München, 2005.

[61] T. Görler. *Multiscale Effects in Plasma Microturbulence*. PhD thesis, Universität Ulm, 2009.

[62] N. Tronko and C. Chandre. Second-order nonlinear gyrokinetic theory: from the particle to the gyrocentre. *Journal of Plasma Physics*, 84:925840301, 2018.

[63] N. Tronko, A. Bottino, T. Görler, et al. Verification of gyrokinetic codes: Theoretical background and applications. *Physics of Plasmas*, 24:056115, 2017.

[64] Alain J. Brizard. On the validity of the guiding-center approximation in the presence of strong magnetic gradients. *Physics of Plasmas*, 24:042115, 2017.

[65] M. Oberparleiter, F. Jenko, D. Told, et al. Interaction between neoclassical effects and ion temperature gradient turbulence in gradient- and flux-driven gyrokinetic simulations. *Physics of Plasmas*, 23:042509, 2016.

[66] A. Di Siena. *Implementation and investigation of the impact of different background distributions in gyrokinetic plasma turbulence studies*. PhD thesis, Universität Ulm, 2019.

[67] W.D. D'haeseleer, W.N.G. Hitchon, J.D. Callen, and J.L. Shohet. *Flux Coordinates and Magnetic Field Structure, A Guide to a Fundamental Tool of Plasma Theory*. Springer-Verlag, New York, 1991.

[68] M. A. Beer, S. C. Cowley, and G. W. Hammett. Field-aligned coordinates for nonlinear simulations of tokamak turbulence. *Physics of Plasmas*, 2:2687–2700, 1995.

[69] Meng Li, Boris N. Breizman, and Linjin Zheng. Canonical straight field line magnetic flux coordinates for tokamaks. *Journal of Computational Physics*, 326:334 – 341, 2016.

[70] Allen H. Boozer. Plasma equilibrium with rational magnetic surfaces. *The Physics of Fluids*, 24:1999–2003, 1981.

[71] D. Told. *Gyrokinetic microturbulence in transport barriers*. PhD thesis, Universität Ulm, 2012.

[72] S. Brunner, T. Dannert, X. Lapillonne, et al. Developer's manual for the global version of the gene code, 2020.

[73] S. Balay, S. Abhyankar, M. F. Adams, et al. PETSc Web page. http://www.mcs.anl.gov/petsc, 2018.

[74] S. Balay, S. Abhyankar, M. F. Adams, et al. PETSc users manual. https://www.mcs.anl.gov/petsc/petsc-current/docs/manual.pdf, 2018.

[75] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith. Efficient management of parallelism in object oriented numerical software libraries. In *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhauser Press, 1997.

[76] C. Runge. Ueber die numerische auflösung von differentialgleichungen. *Mathematische Annalen*, 1895.

[77] W. Kutta. Beitrag zur naeherungsweisen integration totaler differentialgleichungen. *Z. Math. Phys.*, 46:435–453, 1901.

[78] W. H. Press. Book-Review - Numerical Recipes in Pascal - the Art of Scientific Computing. *Science*, 248:234, 1990.

[79] A Segun and M Abramowitz. Handbook of mathematical functions. *Appl. Math. Ser*, 55, 1965.

[80] E. Catmull and R. Rom. A class of local interpolating splines. In *Computer Aided Geometric Design*, pages 317 – 326. Academic Press, 1974.

[81] M. F. Martin, M. Landreman, P. Xanthopoulos, et al. The parallel boundary condition for turbulence simulations in low magnetic shear devices. *Plasma Physics and Controlled Fusion*, 60:95008, 2018.

[82] P. R. Amestoy, I. S. Duff, J. Koster, and J.-Y. L'Excellent. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23:15–41, 2001.

[83] P. R. Amestoy, A. Guermouche, J.-Y. L'Excellent, and S. Pralet. Hybrid scheduling for the parallel solution of linear systems. *Parallel Computing*, 32:136–156, 2006.

[84] Xiaoye S. Li. An overview of SuperLU: Algorithms, implementation, and user interface. *ACM Transactions on Mathematical Software*, 31:302–325, 2005.

[85] X.S. Li, J.W. Demmel, J.R. Gilbert, et al. SuperLU Users' Guide. Technical Report LBNL-44289, Lawrence Berkeley National Laboratory, 1999.

[86] Xiaoye S. Li and James W. Demmel. SuperLU_DIST: A scalable distributed-memory sparse direct solver for unsymmetric linear systems. *ACM Trans. Mathematical Software*, 29:110–140, 2003.

References

[87] P. Hénon, P. Ramet, and J. Roman. PaStiX: A High-Performance Parallel Direct Solver for Sparse Symmetric Definite Systems. *Parallel Computing*, 28:301–321, 2002.

[88] M.J. Pueschel, T. Dannert, and F. Jenko. On the role of numerical dissipation in gyrokinetic vlasov simulations of plasma microturbulence. *Computer Physics Communications*, 181:1428 – 1437, 2010.

[89] M. J. Pueschel. *Electromagnetic Effects in Gyrokinetic Simulations of Plasma Turbulence.* PhD thesis, Westfälische Wilhelms-Universität Münster, 2009.

[90] The Valgrind Developers. Valgrind Website. http://valgrind.org/, 2019.

[91] G. M. Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. *IEEE Solid-State Circuits Society Newsletter*, 12:19–20, 2007.

[92] Cobra MPCDF Website. https://www.mpcdf.mpg.de/services/computing/cobra, 2019.

[93] D. Jarema. *Efficient eulerian gyrokinetic simulations with block-structured grids.* PhD thesis, Technische Universität München, 2016.

[94] D. Jarema, H. J. Bungartz, T. Görler, et al. Block-structured grids for eulerian gyrokinetic simulations. *Computer Physics Communications*, 198:105 – 117, 2016.

[95] D. Jarema, H.J. Bungartz, T. Görler, et al. Block-structured grids in full velocity space for eulerian gyrokinetic simulations. *Computer Physics Communications*, 215:49 – 62, 2017.

[96] Python Software Foundation. Python language reference, version 3.6, 2020. http://www.python.org.

[97] Ayachit U. *The ParaView Guide.* Kitware, 2015.

[98] S. P. Hirshman and J. C. Whitson. Steepest-descent moment method for three-dimensional magnetohydrodynamic equilibria. *The Physics of Fluids*, 26:3553–3568, 1983.

[99] S.P Hirshman and O Betancourt. Preconditioned descent algorithm for rapid calculations of magnetohydrodynamic equilibria. *Journal of Computational Physics*, 96:99 – 109, 1991.

[100] S. P. Hirshman and H. K. Meier. Optimized fourier representations for three-dimensional magnetic surfaces. *The Physics of Fluids*, 28:1387–1391, 1985.

[101] Simon Bance. Viridis color map. http://academic.bancey.com/the-new-default-colormap-for-matplotlib-is-called-vidiris-and-its-great/, 2019.

[102] T. Görler, N. Tronko, W. A. Hornsby, et al. Intercode comparison of gyrokinetic global electromagnetic modes. *Physics of Plasmas*, 23:072503, 2016.

[103] J. M. Canik, D. T. Anderson, F. S. B. Anderson, et al. Experimental demonstration of improved neoclassical transport with quasihelical symmetry. *Phys. Rev. Lett.*, 98:085002, 2007.

[104] J.F. Lobsien, M. Drevlak, F. Jenko, M. Maurer, et al. Physics analysis of results of stochastic and classic stellarator coil optimization. *submitted to Nuclear Fusion*, 2019.

[105] Carolin Nührenberg. Global ideal magnetohydrodynamic stability analysis for the configurational space of wendelstein 7–x. *Physics of Plasmas*, 3:2401–2410, 1996.

[106] J. H. E. Proll, P. Xanthopoulos, and P. Helander. Collisionless microinstabilities in stellarators. II. Numerical simulations. *Physics of Plasmas*, 20:122506, 2013.

[107] F. Romanelli. Ion temperature-gradient-driven modes and anomalous ion transport in tokamaks. *Physics of Fluids B: Plasma Physics*, 1:1018–1025, 1989.

[108] F Jenko, T Dannert, and C Angioni. Heat and particle transport in a tokamak: advances in nonlinear gyrokinetics. *Plasma Physics and Controlled Fusion*, 47:B195–B206, 2005.

[109] B. F. McMillan, X. Lapillonne, S. Brunner, et al. System size effects on gyrokinetic turbulence. *Phys. Rev. Lett.*, 105:155001, 2010.

Acknowledgements

List of publications

1. M. Maurer, A. Bañón Navarro, T. Dannert, M. Restelli, F. Hindenlang, T. Görler, D. Told, D. Jarema, G. Merlo, F. Jenko "GENE-3D: A global gyrokinetic turbulence code for stellarators", Journal of Computational Physics (JCP) 2020

2. J. Lobsien, M. Drevlak, F. Jenko, M. Maurer, A. Bañón Navarro, C. Nuehrenberg, T. Pedersen, H. Smith, Y. Turkin "Physics analysis of results of stochastic and classic stellarator coil optimization", Nuclear Fusion 2020

3. A. Bañón Navarro, G. Merlo, G. Plunk, P. Xanthopoulos, A. von Stechow, A. Di Siena, M. Maurer, F. Hindenlang, F. Wilms, F. Jenko "Global gyrokinetic simulations of ITG turbulence in the magnetic configuration space of the Wendelstein 7-X stellarator", Plasma Physics and Controlled Fusion 2020

4. P. Crandall, D. Jarema, H. Doerk, Q. Pan, G. Merlo, T. Görler, A. Bañón Navarro, D. Told, M. Maurer, F. Jenko "Multi-Species Collisions for Delta-F Gyrokinetic Simulations: Implementation and Verification with GENE", Computer Physics Communications 2020

Declaration Of Authorship

I hereby declare that I have authored this thesis independently and that I have cited all sources used in the references. Passages and figures which have been extracted from publications or elsewhere are marked as such.

Maurice Maurer   Garching, April 21, 2020