A Finite Difference Code for 3D Plasma Edge Modeling

I n a u g u r a l d i s s e r t a t i o n

zur
Erlangung des akademischen Grades
doctor rerum naturalium (Dr. rer. nat.)
an der Mathematisch-Naturwissenschaftlichen Fakultät
der
Ernst-Moritz-Arndt-Universität Greifswald

vorgelegt von

Oleksandr Kalentev

geboren am 11.06.1979

in Kharkov, Ukraine

Greifswald, 03.07.2008

Dekan .....

1. Gutachter: .....

2. Gutachter: .....

Tag der Promotion: .....

I dedicate this work to my beloved wife Tetyana Ivanovska

for her boundless patient and love.

Abstract

The present work is concerned with developing a finite difference code for modeling the physics of the plasma edge in fusion devices. It allows the treatment of a complex 3D geometry and an ergodic structure of field lines in the plasma edge, like in W7-X and TEXTOR-DED, using a finite difference code approach. The code described in the present work requires only a magnetic field configuration data file and specified boundary conditions in order to generate solutions of the plasma transport equations. The concept of local magnetic coordinates is used to handle the different geometries consistently. The pre-processing steps of the general algorithm are grid construction, finding poloidal neighbours and calculation of metric coefficients. For this, a general tracing procedure was developed in a sequential and a parallel version to solve the system of ordinary differential equations for the different local magnetic coordinate systems (LCMS) needed for the discretization stencil.

The neighbours are found with a shadowing algorithm. To solve the coupled set of transport equations in the LCMS representation a solver for conduction-convection equations using a "semi-explicit" generalized upwind scheme for non-staggered grids was developed. For the parallel direction all terms are treated implicitly, whereas the rest of the terms are calculated explicitly. This allows to reduce the amount of memory needed for the calculations and also to use simple tridiagonal and cyclic tridiagonal solvers which are extremely fast. At the same time the procedure remains robust enough due to the fact that the dominant process of the parallel transport is calculated implicitly. The non-parallel terms are not expressed as a divergence of the flux. Instead, all derivatives are derived with the help of the "free point" method. This conduction-convection solver was successfully validated with analytical solutions and benchmarked with two other codes.

The key physics question addressed in the present work is the problem of power loads at the divertor plates, including the influence of ergodicity and 3D configuration of the field on the heat transport. The study of conductive heat transport was done as a first step towards the analysis of the heat flux pattern. A W7-X divertor case was studied as a reference case for the purely conductive problem. Also, a comparison between limiter and divertor scenarios for W7-X was done.

The effect of ergodicity was studied for TEXTOR-DED. A close relation between the structure of the magnetic field lines and the transport was discovered. The correspondence between the connection length and the heat transport was found. The effect of convective terms are discussed for a W7-X divertor case and compared with the purely conductive case. This allows to identify the importance of the convective effects for the power load patterns at the target plates. It was found that electrons do not change much, but just get larger values of power flux density due to the additional convective energy. Their heat flux density distribution is mostly determined by the field line lengths as in the purely conductive case. A more homogeneous pattern of the ion heat flux density due to the convective heat flux is shown compared with the purely conductive case, which showed more pronounced maxima. The coincidence of the maxima of electron heat flux, ion heat flux and particle flux densities in the target patterns demonstrates the importance of the convective heat fluxes. The full capability of this code will require further numerical developments: a domain-decomposition based parallel version will have to be created to satisfy the resolution requirements of the metrics and neutral dynamics.

Contents

Abstract . . . v

Contents . . . vii

1 Introduction . . . 1

2 Basics . . . 2
- 2.1 Magnetic confinement . . . 2
- 2.2 Transport along field lines . . . 5
- 2.3 Plasma wall interaction . . . 5
- 2.4 Braginskii equations . . . 7
- 2.5 Ergodic effects . . . 8
- 2.6 Local magnetic coordinate system . . . 11
- 2.7 State of the art . . . 12

3 Transport equations in LMCS representation . . . 14
- 3.1 Findif grid . . . 14
- 3.2 Metrics of the LMCS . . . 15
- 3.3 Findif transport equations . . . 19

4 Numerical setup . . . 22
- 4.1 Pre-processing . . . 22
  - 4.1.1 Mesh generation . . . 22
  - 4.1.2 Metrics calculation . . . 27
  - 4.1.3 Determining poloidal neighbours . . . 27
- 4.2 Finite difference solution of the conduction-convection problem . . . 30
  - 4.2.1 Discretization of parallel terms . . . 31
  - 4.2.2 Discretization of the non-parallel terms . . . 36
  - 4.2.3 Conduction-convection solver . . . 38
- 4.3 Validation of the conduction-convection solver . . . 42

5 Results . . . 47
- 5.1 Heat conduction . . . 47
  - 5.1.1 W7-X geometry . . . 48
  - 5.1.2 TEXTOR-DED geometry . . . 57
- 5.2 Extended model . . . 64
  - 5.2.1 1D solution for W7-X . . . 64
  - 5.2.2 3D W7-X case . . . 69

6 Effect of neutrals . . . 78
- 6.1 Neutral fluid model . . . 78
- 6.2 Transport equations including neutrals . . . 79
- 6.3 Implementation details . . . 81

6.4 1D solution 83

7 Conclusions 89

A Implicit algorithm of the Free Point Method 91

Bibliography 97

Acknowledgments 102

Chapter 1

Introduction

The present work is concerned with developing a finite difference code for modeling the physics of the plasma edge in fusion devices. We are interested in a quantitative understanding of the energy transport from the plasma core to the vessel wall in order to optimize steady state operation of magnetic fusion devices. One problem to consider is the broadening of the power deposition pattern on the wall structures. This broadening is necessary to avoid power loads that exceed the engineering limits. This introduces the need for the development of new concepts and tools for handling such situations. The present work allows the treatment of a complex 3D geometry and an ergodic structure of field lines in the plasma edge, like in W7-X and TEXTOR-DED.

In any transport code for modeling the physics in a magnetized plasma the following characteristics must be considered. First, the extreme anisotropy of energy transport caused by very strong electron heat conduction along the magnetic field lines. This parallel conduction is typically $10^4 - 10^7$ [1, 2] times higher than the transport across the field lines. Second, conduction along the field lines is mainly by Coulomb collisions and is strongly non-linear, with $\kappa_{\|} \propto T_e^{5/2}$. By contrast, conduction across the field lines is driven by turbulence in the plasma [3] and can be described as a diffusive process with $\chi_{\perp} \approx 1 m^2 s^{-1}$. Third, the geometry of the plasma may further complicate the application of the computational method, especially for 3D devices such as W7-X. Fourth, in the edge region of a fusion plasma, the field lines may have an ergodic geometry, meaning spatially chaotic trajectories.

The special advantage of a finite difference code in modeling plasma physics in fusion experiments is its applicability to any device regardless of anisotropy, ergodicity or geometry. Essentially, the code described in the present work requires only a magnetic field configuration data file and specified boundary conditions in order to generate solutions of the plasma transport equations.

A key question addressed in the present work is the problem of power loads at the divertor plates. The influence of conduction and convection will be discussed and comparison with experimental results for TEXTOR-DED will be presented.

The present work is structured as follows. Chapter 2 introduces the basics of magnetic confinement and edge physics followed by a summary of the current state of the art in 3D edge modeling. Chapter 3 introduces the specific transport equations used in this work. A particular coordinate system suitable for the task we want to solve is introduced. The way of constructing the grid in such a coordinate system, as well as metric coefficients for it, is discussed. Chapter 4 gives a full description of the framework including pre-processing procedures and the main algorithm. Different discretization schemes are presented and validation of the main solver is done. Results obtained with this code are gathered in chapter 5. Firstly, for the pure conduction problem results for W7-X and TEXTOR-DED geometries are discussed. After that, the extended model including convection is applied to W7-X with the main emphasis on the target loads and the impact of the convection to that. In chapter 6 a neutral fluid model is introduced to include the influence of the neutral dynamics. The results for 1D calculations are shown. Finally conclusions are presented.

Chapter 2

Basics

2.1 Magnetic confinement

One of the greatest challenges in this century is the limitation of resources, especially energy. Fossil fuel, especially oil, is mostly used for this purpose at the moment. The growing demand of the energy resources and at the same time the depletion of the oil and gas sources leads more and more to the need of finding alternative sources of the energy. Fusion energy is one possible option. The aim of fusion is to provide an energy source using the same principle as in stars. A fusion reactor is designed to fuse the hydrogen isotopes deuterium (D or ${}^{2}_{1}H$) and tritium (T or ${}^{3}_{1}H$) into helium, where an additional neutron is created. The major technology which supposed to be used for the future fusion reactors is magnetic confinement. In a magnetic confinement fusion reactor a plasma is confined by powerful magnetic fields in order to create the conditions necessary for fusion reactions to occur in the plasma core and release energy [4]. This energy can be converted to heat and used to drive steam turbines to generate electricity. The most promising designs for the magnetic confinement concept [5] are tokamaks and stellarators which are toroidal devices. Fig.2.1 shows a generalized tokamak configuration.

In the tokamak the central transformer coil adds a helical twist to the magnetic field lines.

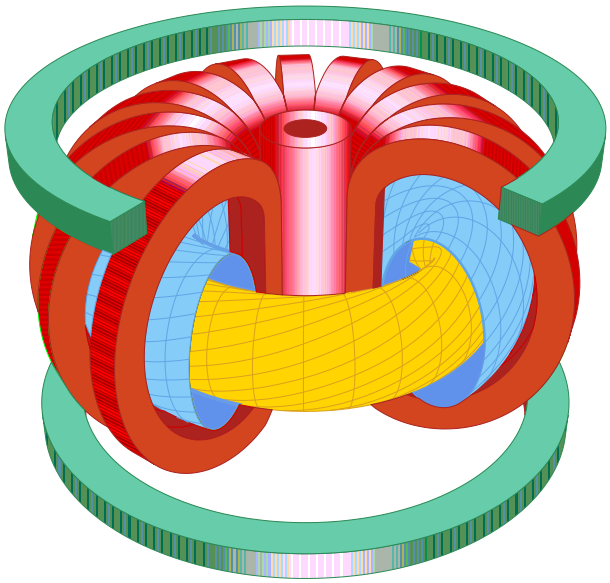

Figure 2.1: *In a tokamak the plasma cross-section remains a constant shape at all toroidal positions, therefore the tokamak may be treated as a 2D device. A central transformer coil drives a toroidal current in the plasma.*

This short-circuits the creation of electric fields and an outward drift which would destabilize the plasma [6, 7]. The current flowing in the plasma boosts the core temperature by Ohmic resistance, though only at relatively low temperatures. Higher temperatures are achieved by additional heating methods such as Neutral Beam Injection (NBI), and Ion and Electron Cyclotron Resonance Heating (ICRH, ECRH). The use of the transformer principle for creation a

toroidal current in the plasma means that the tokamak operates as a pulsed device [5].

A successful fusion power plant must operate in a steady state mode over long time intervals. Therefore the plasma must be kept clean and stable. In a steady state fusion experiment, fuelling and pumping of the helium "ash" (produced during the fusion reaction) has to be realized. The "ash" must be removed from the core to prevent contamination of the plasma [8]. Further contamination may come from the reactor walls and the divertor plates as a result of plasma erosion. If the impurities in the plasma core exceed a critical threshold, only a few percent, then the fusion reactions will stop [9]. Fig.2.2 shows two cross-sections of a tokamak plasma and two basic concepts of power and particle removal.

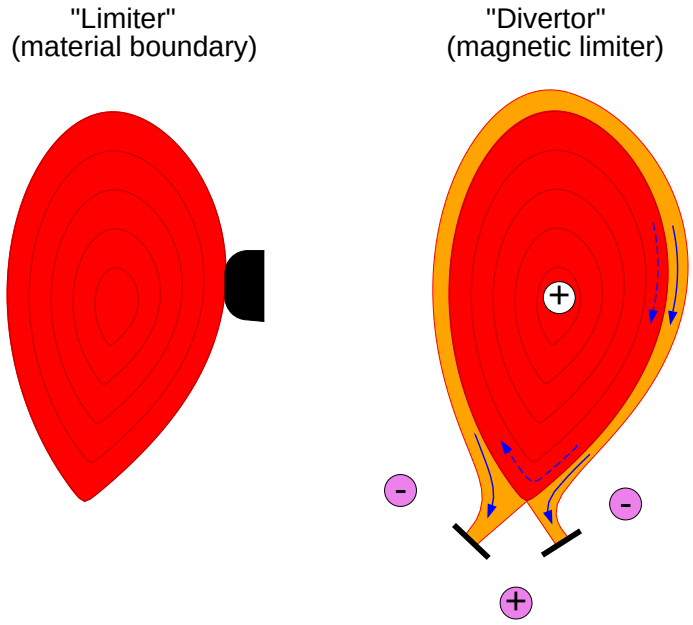

Figure 2.2: *(left) A limiter intersecting the plasma edge. (right) Additional poloidal coils diverting the plasma edge onto 2 divertor plates at the bottom.*

In the first concept a limiter structure directly impinges on the hot plasma core. The advantage of this concept is its simplicity and its direct influence on the core. However, direct contact with the core can produce high power loads on the limiter structure, resulting in severe erosion and impurity problems. In the second concept, additional coils create an x-point in the separatrix (the boundary between closed and open field lines). In this way, the outer part of the plasma is scraped off and diverted onto the divertor plates [10]. This so-called Scrape-Off-Layer (SOL) is the region of open field lines in the plasma edge which intersect the divertor plates. It plays a critical role in maintaining the purity and stability of the plasma. The purpose of such plasma edge diversion is to move the interaction zone between the divertor plates and the plasma, away from the plasma core. This offers the possibility of good impurity control [11] and also the reduction of the heat flux to the divertor plates due to radiation losses. The divertor concept is now the standard solution for all reactor designs [12].

Fig. 2.3 shows the W7-X stellarator configuration. In a stellarator there is no central transformer coil and practically no plasma current. The lack of a central transformer coil allows the possibility of steady state (non-pulsed) operation [13]. The short-circuiting of electric fields is achieved by the complex shape of the plasma.

In W7-X, which is an optimized stellarator [14], we make use of the intrinsic islands which naturally divert the exhaust power fluxes from the core into the islands. By placing target plates

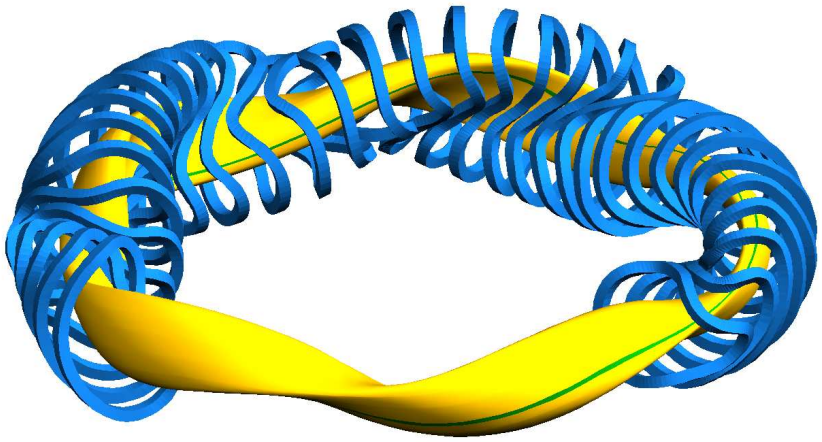

Figure 2.3: *In the W7-X stellarator the cross-section of the plasma, shown in yellow, changes with toroidal position, therefore a stellarator can only be treated as a fully 3D device. The plasma is shaped by a series of non-planar field coils shown in blue.*

at the interaction zones we can introduce a similar concept of plasma wall interaction control away from the core, like in the tokamak divertors. Therefore, such islands are called island divertors, and because of the 3-dimensionality of the plasma in W7-X these divertors have a more complex shape. See Fig.2.4.

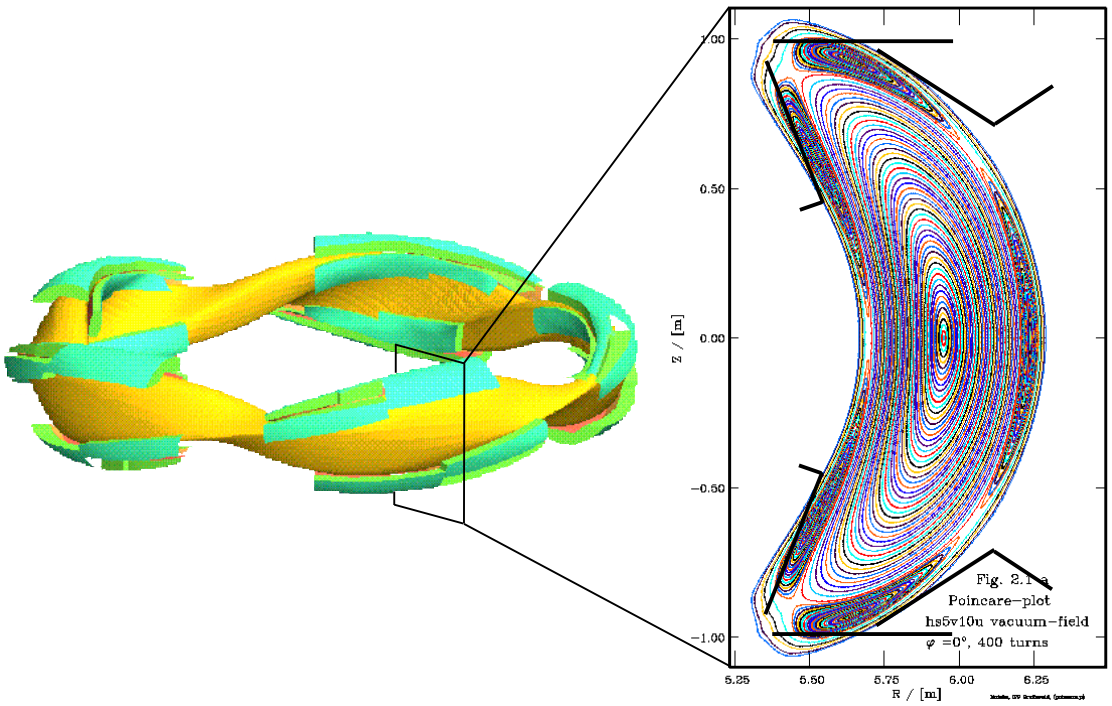

Figure 2.4: *Possible island divertor configuration in W7-X. The divertor structures are shown in green. At right is a plasma cross-section showing how the divertor plates intersect the plasma at toroidal position* $\phi = 0°$.

In the stellarator concept, there is the possibility to operate at higher plasma density without

the danger of disruptions because the stellarator is internally current-free. Also, one can use the 3-dimensionality for impurity and neutral screening. Potential problems are the danger of producing local instabilities even earlier than in the tokamak divertor, and the more complex baffling and pumping due to the plasma geometry.

2.2 Transport along field lines

The dominant mode of heat transfer in the magnetized plasma is electron heat conduction along magnetic field lines which is governed by Coulomb collisions [15]. The electron mean-free path length $\lambda_e = v\tau_e = \sqrt{3}v_{th}\tau_e$ for a test particle with energy $mv^2/2 = \frac{3}{2}kT$ and a thermal velocity $v_{th} = (kT/m)^{1/2}$ is:

$$\lambda_e \quad = \quad 1.5 \cdot 10^{16} \times \frac{T_e^2}{n}, \tag{2.1}$$

where T is temperature in electron Volts (eV) and the other quantities are in MKS-units. The electron mean-free path length is an average distance that an electron "runs", until it changes its direction by 90 degrees. This change happens due to the multiple collisions that make the electron change its direction by some small angles. According to the Rutherford scattering formula the mean-free path scales with v^4. The ion-ion and electron-ion mean-free paths are practically identical to the electron-electron mean-free path, within a factor of $1/\sqrt{2}$. The electron heat conduction along the magnetic field is:

$$q_{e,\|} \quad = \quad -\kappa_{e,\|}\nabla_\| T_e \quad = \quad -\chi_{e,\|} n_e \nabla_\| T_e, \tag{2.2}$$

where q_e and T_e are the electron heat flux and temperature respectively. The heat diffusivity is given by:

$$\chi_{e,\|} \quad \approx \quad 1.8 \cdot 10^3 \times T_e^{5/2} \tag{2.3}$$

This defines a very strong non-linearity in the heat conduction.

2.3 Plasma wall interaction

A plasma in contact with a boundary forms a sheath layer at the boundary. This is due to the electron velocity being much higher than the ion velocity (mass difference) [15, 16]. The sheath layer can be divided into different regions (sheath and pre-sheath) which can be characterized qualitatively by the properties of the plasma in this region [17] (see Fig.2.5). The recycling neutrals get ionized at a distance λ_{ion} in the pre-sheath where ions are accelerated to sound speed at the sheath edge x_S, as described by the Bohm criterion [1, 2].

For fluid models the sheath effects are introduced as effective boundary conditions at the sheath edge, that is, the acceleration of the plasma ions to velocities which satisfy the Bohm condition [18]:

$$v_{i,potential} \quad \geq \quad \sqrt{\frac{kT_e}{m_i}}, \tag{2.4}$$

and an effective boundary condition for the heat flux:

$$Q_{e,x}(x_s heath) \quad = \quad \delta_e T_{e,sheath} \Gamma_{e,x}, \tag{2.5}$$

taking into account the electron reflection properties of the sheath and introducing a kinetic factor of about $\delta_e = 5$.

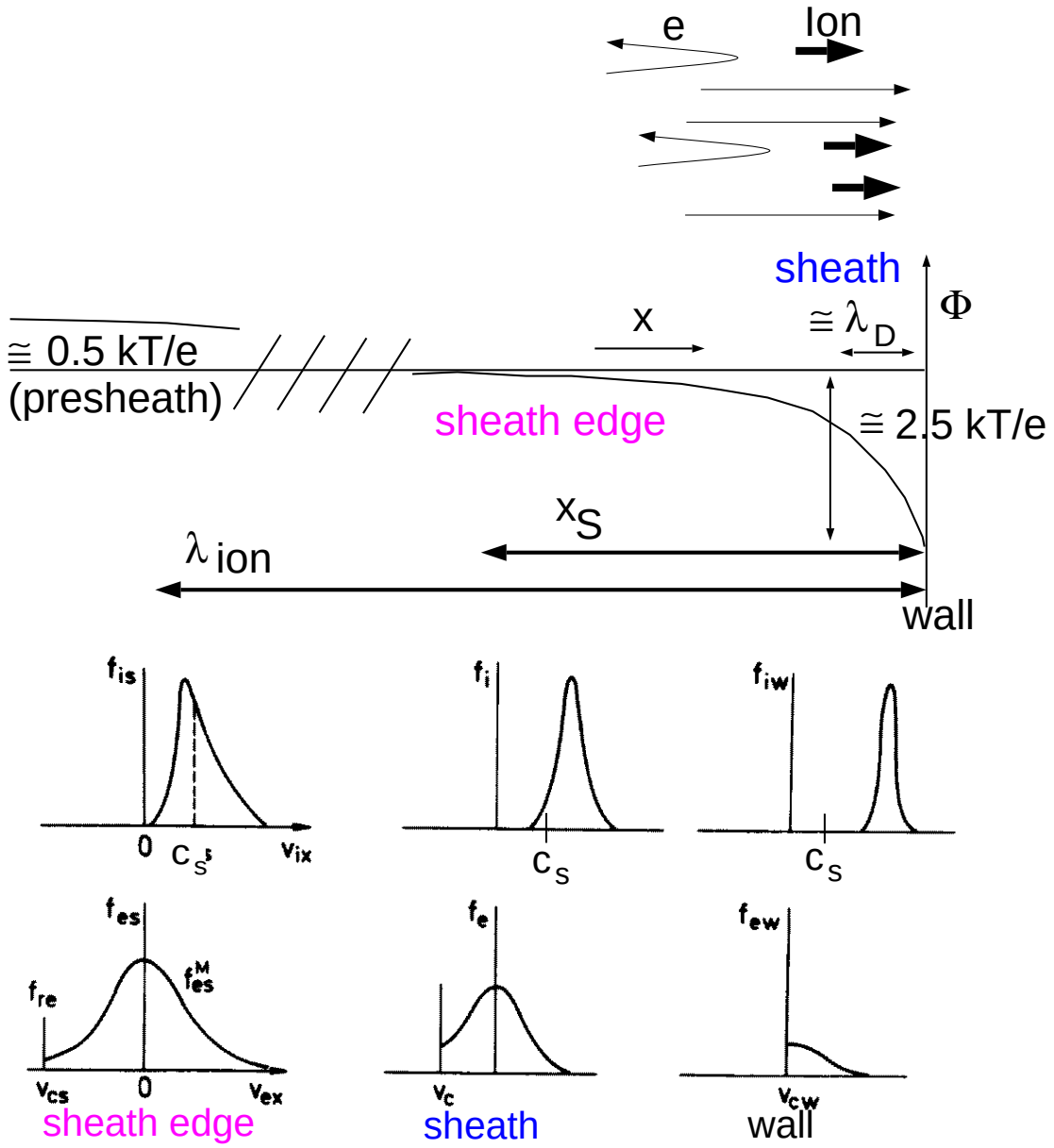

Figure 2.5: *Schematics of the sheath potential in front of the wall without magnetic field (top). The relevant lengths scales are shown: Debye-length λ_D, sheath edge position x_S and ionization length for neutrals λ_{ion}. At the bottom, the velocity distribution functions for ions f_i and electrons f_e are shown at the sheath edge, in the sheath, and at the wall. The electrons have a cut-off Maxwellian (with a cut-off velocity $v_c s$ at the sheath edge). At the wall no electrons are going back (half Maxwellian). At the sheath edge the ions already have a non-Maxwellian distribution function as they are accelerated to c_S or larger because of the Bohm criterion. (Figure from R. Chodura)*

The general edge problem is complex because of the interaction with neutrals and impurities and other phenomena which have to be included [19]. The description of plasma wall interaction needs many disciplines:

1. ***Magnetohydrodynamic equilibrium studies*** are a necessary prerequisite in edge physics. From plasma fluid theory we obtain the transport equations which govern edge transport.
2. ***Kinetic theory*** describes the sheath forming in front of a plasma-facing structure. This defines the boundary conditions for fluid model transport codes. The classical transport of neutrals strongly determines the operational parameters in divertor plates.
3. ***Turbulence and anomalous radial transport.*** Due to these effects, scaling laws or direct coupling to turbulence codes is necessary for good modeling. Drift-wave turbulence in full geometry has the potential for a full predictive capability for the edge transport. Anomalous transport is still the largest uncertainty in edge physics.
4. ***Atomic and molecular processes*** strongly affect the plasma through ionization, charge exchange and recombination.
5. ***Materials science*** and plasma wall interaction involve the study of sputtering, reflection and recombination.
6. ***Numerics.*** Transport codes for heat conduction/convection problems may use various numerical methods, but they must allow for physical characteristics such as strong anisotropy, ergodicity and complex geometry.

2.4 Braginskii equations

The basic description for plasma transport on open field lines relies on a fluid picture due to the high collisionality. One can derive the fluid equations for the plasma, multiplying the kinetic equation by the combination of the velocity components, that correspond to the appropriate moment and then integrating the kinetic equation. This scheme was originally presented by S.I.Braginkii [20]. The complete set of the fluid equations is presented below.
The continuity equations for ions and electrons look as follows:

$$\begin{aligned} &\frac{\partial n_i}{\partial t} + \vec{\nabla} \cdot \left(n_i \vec{V}_i \right) = S_{n_i}; \\ &\frac{\partial n_e}{\partial t} + \vec{\nabla} \cdot \left(n_e \vec{V}_e \right) = S_{n_e}. \end{aligned} \tag{2.6}$$

The total momentum equation for ions and electrons is:

$$\begin{aligned} &\frac{\partial}{\partial t} \left(m_i n_i \vec{V}_i \right) + \vec{\nabla} \cdot \left(m_i n_i \vec{V}_i \vec{V}_i \right) = \\ &- \vec{\nabla} p - \vec{\nabla} \overleftrightarrow{\Pi}_i + \vec{j} \times \vec{B} + \vec{S}_{m_i \vec{V}_i}, \end{aligned} \tag{2.7}$$

where total electric current density $\vec{j}$ and total pressure p are:

$$\begin{aligned} &\vec{j} = e \left(n_i \vec{V}_i - n_e \vec{V}_e \right); \\ &p = n_e T_e + n_i T_i. \end{aligned}$$

The electron energy equation (simplified) and the total ion energy equation look as follows:

$$\frac{\partial}{\partial t}\left(\frac{3}{2}n_eT_e\right)+\vec{\nabla}\cdot\left(\frac{5}{2}n_eT_e\vec{V}_e+\vec{q}_e\right)= \\ -en_e\vec{E}\vec{V}_e+\vec{R}\cdot\vec{V}_e+Q_{ei}+S_{E_e}; \tag{2.8}$$

$$\frac{\partial}{\partial t}\left(\frac{3}{2}n_iT_i+\frac{m_in_i}{2}\vec{V}_i^2\right)+\vec{\nabla}\cdot\left[\left(\frac{5}{2}n_iT_i+\frac{m_in_i}{2}\vec{V}_i^2\right)\vec{V}_i+\overleftrightarrow{\Pi}_i\cdot\vec{V}_i+\vec{q}_i\right]= \\ \left(en_i\vec{E}-\vec{R}\right)\cdot\vec{V}_i-Q_{ei}+S_{E_i}, \tag{2.9}$$

where the energy fluxes for electrons and ions are:

$$\vec{q}_e=-\kappa_{||}^e\vec{\nabla}_{||}T_e-\kappa_{\perp}^e\vec{\nabla}_{\perp}T_e+\kappa_{\wedge}^e\frac{\vec{B}}{B}\times\vec{\nabla}_{\perp}T_e \\ -0.71\frac{T_e}{e}\vec{j}_{||}-\frac{3}{2}\frac{T_e}{e\omega_e\tau_eB}\vec{B}\times\vec{j}_{\perp}, \\ \vec{q}_i=-\kappa_{||}^i\vec{\nabla}_{||}T_i-\kappa_{\perp}^i\vec{\nabla}_{\perp}T_i+\kappa_{\wedge}^i\frac{\vec{B}}{B}\times\vec{\nabla}_{\perp}T_i,$$

and the energy exchange term between electrons and ions is:

$$Q_{ei}=\frac{3m_e}{m_i}\frac{n_e}{\tau_e}\left(T_i-T_e\right). \tag{2.10}$$

2.5 Ergodic effects

In some configurations or devices, a spatially chaotic structure of field lines [21] can exist. In W7-X this ergodicity is created by intrinsic plasma effects (finite beta). In TEXTOR-DED, such a structure is introduced by additional toroidal coils.
The plasma core is a non-ergodic region in which the field lines form a tightly packed series of nested flux surfaces, and there is an ordered temperature gradient across the surfaces. In the plasma edge some field lines are ergodic which means that they fill out volumes, and neighboring field lines diverge from each other exponentially, on average. To measure the ergodicity a so-called Kolmogorov length was introduced:

$$L_k=\frac{S}{log\left(\frac{\delta_0}{\delta_1}\right)} \tag{2.11}$$

From the formula above and the Fig. 2.6 One can observe, that Kolmogorov length, traversing along two magnetic lines, measures how far one magnetic line will "run" from the other one. This is the fundamental characteristic, that should be taken into account while resolving ergodic systems. Note that in the SOL, the open field lines have varying lengths. Those that are shorter than the Kolmogorov length do not exhibit ergodicity, and their structure is laminar.

In terms of field line trajectories, ergodicity means that as a field line travels around a torus it undergoes excursions in the radial direction. This results in a mixing of the strong parallel transport with radial transport. Thus, the radial transport is enhanced and there is a flattening of the temperature profile across the edge region. This broadening of the temperature profile was the origin of the idea to use ergodic perturbations to relax the power load problem. However, on open field lines (connected with the target plates) this idea fails, because they are usually shorter than the Kolmogorov length. Nevertheless, as was shown in [22], a broadening of the heat deposition zone still occurs due to the cascading of energy from long to short field

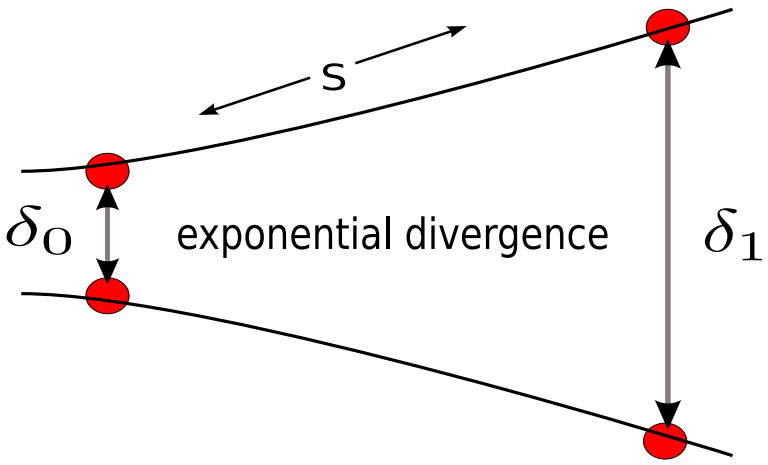

Figure 2.6: *Explanation of the meaning of the Kolmogorov length.*

lines and the modulation of radial fluxes in the ergodic region. Nowadays, the introduction of additional perturbation coils is mostly motivated to get tools for ELM (edge localized modes) control, like in DIII-D. Fig.2.7 illustrate the idea of the ergodicity usage for for the temperature profile broadening. Three different regions are shown schematically. The lowest is the region correspondent to the plasma core with where closed flux surfaces can be observed. In the top on the picture the region with open field line is shown. This part, as it is clear from the picture, is not affected by the perturbation and remains laminar. And in the middle one can see the ergodic layer, where field lines are mixtured chaotically and thus the huge parallel conduction contributes to the radial direction, which is schematically shown in the middle of the picture. The temperature profile shown on the right demonstrate, its behaviour in different above mentioned regions.

An attempt to describe the effect of the ergodicity on plasma transport was done by Rechester and Rosenbluth (see [23]). They tried to express the impact of the parallel transport to the radial one that occurs due to the ergodization in terms of the diffusion coefficient. The estimate looks like:

$$\chi_r^{(RR)} = D_{fl} D_{||e} \left[L_k ln \left(\frac{1}{r k_\theta} \left(\frac{\chi_{||e}}{\chi_{\perp e}} \right)^{\frac{1}{2}} \right) \right]^{-1}, \tag{2.12}$$

where D_{fl} is the field line diffusion coefficient, $k_\theta = m/r$ is the characteristic perpendicular wave number, L_k is the Kolmogorov length, r is the minor radius and m is the poloidal number of the perturbation; $\chi_{||e}$, $\chi_{\perp e}$ are the ordinary transport coefficients arising from the underlying plasma transport. According to this estimate the ergodization leads to a flattering of the electron temperature profile in the correspondent region. Based on the field line diffusion coefficient, D_{fl}, one can calculate the diffusion of each population of free streaming particles along the field lines [24]. The diffusion is then simply $D_{fl} v_{th}$ where v_{th} is the typical velocity of each species, namely the thermal velocity. The ratio of electron to ion transport is therefore $\sqrt{m_i/m_e}$. Using this order of magnitude, one readily expects a large plasma response for "electronic" fields, such as the electron temperature, the toroidal component or the radial electric field, and a relatively small direct response for the "ionic" fields, ion temperature, density and plasma momentum.
As discussed before the original idea of the transport of power in the stochastic layer has been treated as a diffusive process [23], which gives a significantly enlarged effective cross-field transport for the electron energy. As a result a widening of the contact zone between plasma and wall [25] should be observed. Subsequent experiments at different devices (Tore Supra [26], TEXTOR [27] and DIII-D [28]) proved that the heat and particle deposition patterns are strongly

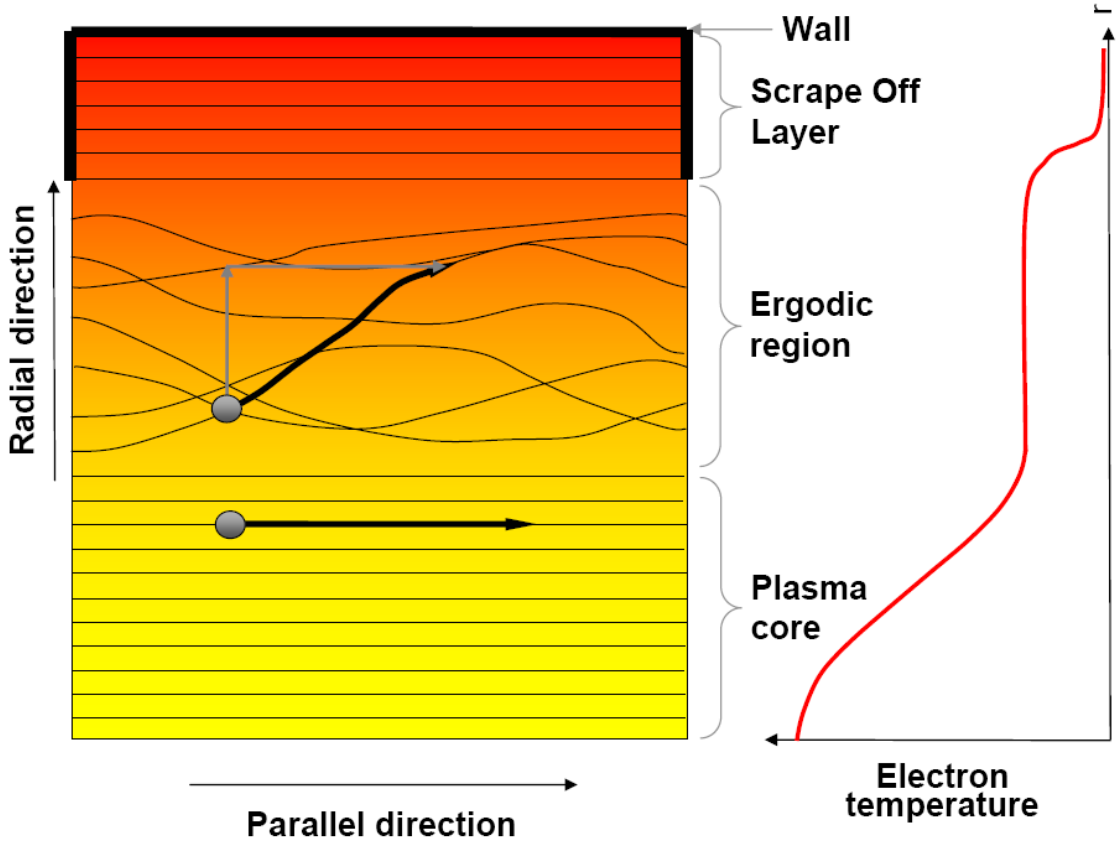

Figure 2.7: *Effect of ergodicity on transport in the edge region.*

structured. This is a result of the heterogeneous topology of the open chaotic region, which consists of magnetic field lines with dramatically different connection lengths [29].

The inner ergodic layer is composed of stochastic field lines with connection length (L_c) of many poloidal turns, which form a "woven" structure. Here the significant increase of the field line diffusion coefficient results in the enhancement of the heat radial transport. Experimental examples of such flattening are found in Tore Supra, Text and JFT-2M [30]. The 3D transport modeling of the heat conduction equation in the ergodic region of Tore Supra with the ERGOT code reproduced such flattening [31].

In the outermost layer, the interaction of the near field with field lines forms the flux tubes, which have very short wall-to-wall connection length (i.e. one or two poloidal turns). This area is known in literature as a laminar region. Here it is expected that the parallel convection is the main transport mechanism rather then radial diffusion [32, 33]. There is no clear boundary between the ergodic and laminar region, while the flux tubes formed by the long connection length field lines are adjacent to the flux bundles of laminar field lines. It is typically assumed that these field lines have connection length much shorter than Kolmogorv length ($L_c \ll L_k$). This different areas have been identified experimentally on TEXTOR [34, 33], where it has been shown that the topology of the perturbed volume is three dimensional. The resulting heat deposition pattern reflects the complicated structure of the perturbed volume. It has been shown in [35, 36] that the connection length and the radial penetration of the magnetic field lines defines the amount of power deposited on the target structures. The maximum of the heat flux density corresponds to the field lines with long connection length, however those shallow penetrating seem to be strongly affected by the collisionality [37] in contrary to the field lines

with deep penetration, which connect outermost existing island chain to the divertor surface. These field lines are strongly linked to the topology of the homo- [28] or heteroclinic tangles [35].

Although the Rechester Rosenbluth formula (2.12) is very useful for understanding the basic influence of the ergodicity on transport, it can be only used for a rough estimate of that impact and can not help to see the structure of the target loads and temperature profiles.

2.6 Local magnetic coordinate system

The basic concept that allows to investigate the properties of the SOL plasma affected by the ergodic effects is the local magnetic coordinate concept. Its idea comes from the properties of the plasma. The characteristics of the plasma make the other "standard" coordinate systems inapplicable. In this section we will describe these properties and give reasons for preference of the local magnetic coordinate system (LMCS).
The problem we want to attack is characterized by following features:

1. strong parallel heat conduction ($10^6 - 10^7$ times greater) with respect to the perpendicular one;

2. unavoidable existence of the 3D effects in the scrape-off layer plasma;

3. ergodic behaviour of the magnetic field in stellarators and modern tokamaks.

The first property gives us a demand for the separation of the parallel fluxes from the perpendicular ones, as we need to avoid their contribution to the perpendicular flows. Otherwise, even a small error in the flux component calculations will cause a huge increase of the perpendicular heat transport. The second property obviously requires a 3D mesh. The ergodicity effects constrain us in use of a unique coordinate system for the whole domain. This property appears to be the strongest one and makes such coordinates as Boozer coordinates [38], [39] inapplicable for the current task.
The main idea of the local magnetic coordinate system is to use a set of the overlapped non-orthogonal, curvilinear coordinate systems. In each coordinate system in the set the coordinate curves (see [40]) of one of the coordinate are aligned with magnetic field lines in all points of the sub-domain. This allows one to separate the huge parallel flows from perpendicular ones and, thus, to avoid a numerical diffusion which can be significantly big for such an asymmetric system. One should keep in mind that the contribution from the radial transport to the parallel one still exists, but it is considered to be negligible. The other two coordinates can be chosen arbitrary, but the most natural choice is to take them lying on the radial cut plains. As the magnetic field lines in ergodic region intersect the radial cuts with a tip each coordinate system in the set LMCSs is in general a nonorthogonal curvilinear coordinate system.
It is particularly important to point out that the length of the unique LMCS should be less then the above mentioned Kolmogorov length, see Eq. (2.11). The restriction on the length of the unique LMCS, arises from the following fact. If this length is taken greater than Kolmogorov length, then the magnetic lines will mix with each other and, thus, it will be impossible to construct a consistent coordinate system.

Fig. 2.8 shows 3 consecutive cuts in a toroidal geometry. There are two eigenvectors showing orthogonal poloidal coordinates x_1 and x_2 on the central cut. So the coordinate curves of coordinates 1 and 2 are just orthogonal lines here. The third eigenvector is locally tangential to the magnetic field line. Following the field line we see, that coordinate curves 1 and 2 change so that on the forward and backward cuts we have the non-orthogonal eigenvectors tangential to the respective coordinate curves.

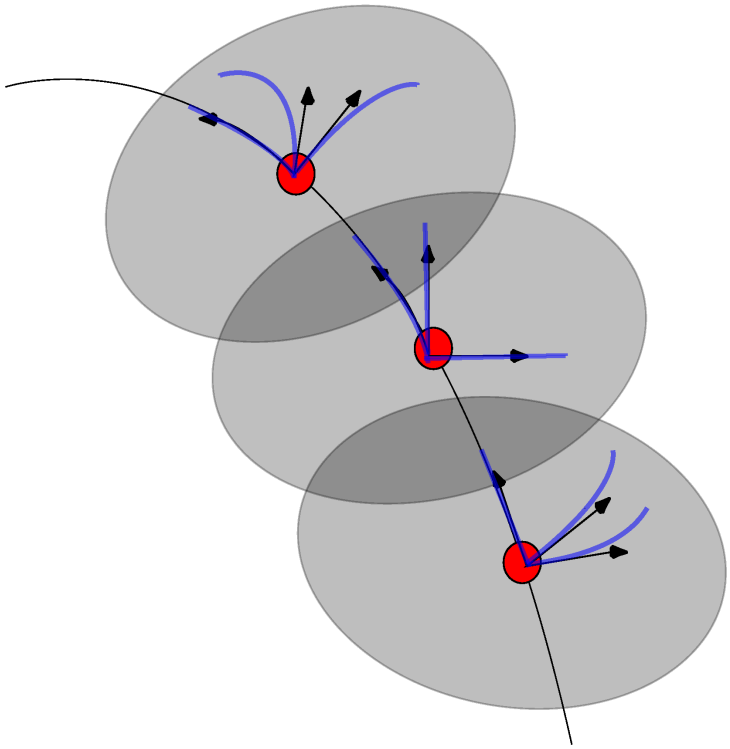

Figure 2.8: *Schematics of the local magnetic coordinate system. Three toroidal cuts is shown. The central cut is the reference one where two non-aligned with the field lines coordinate curves coincide with the Cartesian coordinate curves. On the other two cuts coordinate curves performs arbitrary shaped.*

2.7 State of the art

In recent years a number of codes have been developed for the numerical solution of the plasma fluid transport equations in tokamak and stellarator 3D configurations. The motivation is the need for tools which can handle strong anisotropy in complex 3D topologies including flux surfaces, islands, ergodic regions and complicated surfaces of plasma-surface interaction. Presently, the most advanced codes in this field of computational physics are the BoRiS [41], EMC3-EIRENE [42, 43], and E3D [44, 45].

BoRiS is a 3D Scrape-Off-Layer transport code for solving a system of coupled partial differential equations using a finite volume method. It has been developed for 3D edge modeling in W7-X but is also applicable to other devices without ergodicity. The code is characterized by interpolation for mixed convection-diffusion, generalized magnetic coordinates and the Newton method. The development of BoRiS was influenced by experience with 2D codes like B2-Eirene [46] and UEDGE [47, 48].

The EMC3-EIRENE code uses a Monte-Carlo technique in real space using a field-aligned local orthogonal vector basis which reduces the diffusion tensor to a diagonal form. The parallel and perpendicular transport are separated by integrating the parallel transport along the field lines. Presently the code is being applied to several machines, e.g. W7-AS, W7-X and TEXTOR-DED.

The E3D code uses a multiple coordinate system approach (MCSA) with local magnetic coordinates and the appropriate full metric tensor. The local or piecewise treatment of field lines allowed in the Monte Carlo method makes it applicable for modeling in ergodic regions. Also, the grid used in the Monte-Carlo approach can be refined at point resolution. Disadvantages of the

Monte Carlo approach are the computational expense and the noise in the solutions which make it difficult to identify a steady state. E3D was developed for modeling in TEXTOR-DED[49] but has also been applied to W7-X and DIII-D.

A general finite volume method requires a continuum of the computational grid which is only available for certain magnetic configurations with closed flux surfaces. By comparison, using Monte Carlo methods, transport in ergodic regions can be treated locally which allows greater flexibility in the mesh construction. Regardless of the numerical method used, the strong anisotropy of the transport requires a complete separation of parallel and radial transport terms. This may be achieved by a clear identification of the parallel direction such as the magnetic coordinates used in BoRiS, the local magnetic coordinates used in E3D, and the finite flux tube coordinates used in EMC3. The finite difference approach has a number of advantages in comparison to other methods, mentioned above. For example, it allows to avoid such problems as the volume connections and vessel walls transfer into magnetic coordinate system, which are encountered in a finite volume case (BoRiS). Its advantage over Monte-Carlo approach is that the finite difference scheme does not produce noisy results that need smoothing.

The main idea of the finite difference method is to discrete the parallel (to the magnetic field) and radial transport components separately. This separation is carried out by applying a specific grid generation procedure. The mesh is formed of the points that result from the intersection of the magnetic field lines and a predefined number of the toroidal cuts. We repeat this tracing procedure for a number of lines in the domain of interest and end up with a set of Poincarè plots, that contain additional information about the toroidal point connections. To obtain a complete discretization of the derivatives one needs to use the information about the neighbours on each toroidal cut.

Chapter 3

Transport equations in LMCS representation

3.1 Findif grid

The choice of the coordinate system determines the grid and also the way of its construction. As it was stressed in Chap. 2 the goal of using the LMCS is to separate the parallel transport from the radial one. This separation is carried out by applying a specific grid generation procedure. The mesh is formed of points that result from the intersection of the magnetic field lines and a predefined number of the toroidal cuts (see Fig. 3.1). We repeat this tracing procedure for a number of lines in the domain of interest and end up with a set of Poincarè plots, that contain additional information about the toroidal point connections.

To give a concrete way of the grid construction first we have to describe mathematically a

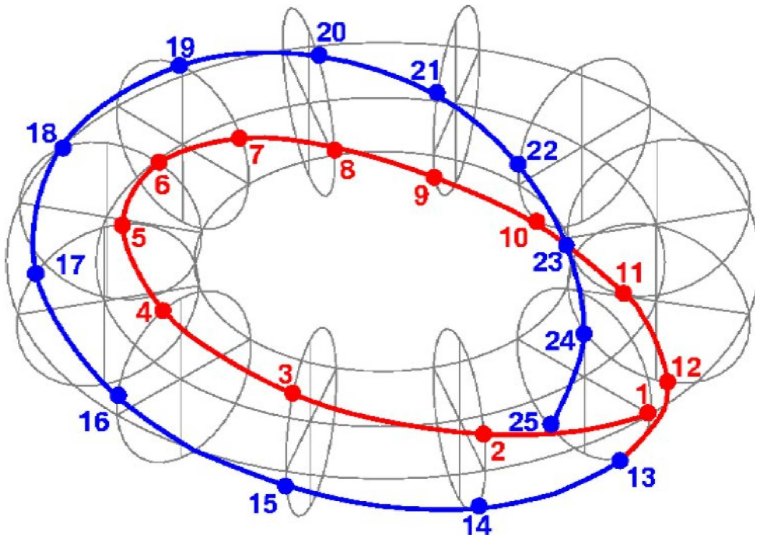

Figure 3.1: *Construction of the mesh procedure. Tracing and matching points of intersection with poloidal cuts.*

magnetic field line. Following [40] we can write down the equation of a magnetic field line in the component form.

$$\frac{B}{dl} = \frac{B^1}{du^1} = \frac{B^2}{du^2} = \frac{B^3}{du^3} \tag{3.1}$$

Substituting the components u^i by the cylindrical coordinates r, z, ϕ we obtain a set of two ordinary differential equations:

$$\begin{aligned} \frac{dr}{d\phi} &= \frac{B^r}{B^\phi} \\ \frac{dz}{d\phi} &= \frac{B^z}{B^\phi} \end{aligned} \tag{3.2}$$

Following the field line means solving the set of equations (3.2) starting from some initial point. The coordinates of the starting point serves as initial conditions for the ODEs. Solving the system Eqs. (3.2) for the next toroidal cut with $\varphi = \varphi_0 + \delta\varphi$, where φ_0 is the toroidal angle of the initial point and $\delta\varphi$ is the angle distance from the initial cut to the next one, we obtain coordinates r, z on the next cut and thus determine the point lying on the field line started from the initial point.
There are two possible kind of magnetic field lines, which exist in the SOL. First is the infinite line either forming the closed flux surface or traversing in the ergodic layer. Second is an open field line, which starts and ends on the wall. The second type of the field lines gives no problem for the grid construction. The first, on the other hand, as they have an infinite length have to be treated specially using a so-called closing criteria. After some turns a line can come again close to the starting point. This fact is used for "closing" the line, i.e. the line "closes" at its starting point instead of making an infinite number of turns, that cover the flux surface or some part of the ergodic layer. In the grid construction only those starting points are accepted satisfying this criterion.
Of course, this artificial closing of the line introduces some error and it is intuitively understandable, that this error decreases with more turns. To estimate the introduced error such a criteria has been derived:

$$\Delta << S_{||}\sqrt{\frac{\chi_{||}}{\chi_{\perp}}}, \tag{3.3}$$

where $\chi_{||}$ and $\chi_{\perp}$ are the parallel and perpendicular conductivities, $S_{||}$ is a length of the line and Δ is the distance between the start and final point on the same toroidal cut. Except this there are other constrains on the mesh that have to be fulfilled. The mesh should be dense enough to resolve all gradients of the variables (temperatures, densities, etc.) in both toroidal and radial directions. As we have to know the solution in advance to evaluate the mesh density, we can not use such a criteria for the real mesh production. To construct a mesh which will be dense enough we follow another considerations. For toroidal resolution we know, that we have to resolve the variation of all the metric coefficients and Jacobian along the field line. So this serves as a criteria for this direction. As for the radial resolution at the moment we do not have an additional 'a priori' criteria and, therefore, we just try to construct a reasonably radially resolved mesh.

3.2 Metrics of the LMCS

The finite difference representation of the equations to be solved (see Chap. 4) needs a set of overlapping coordinate systems (see Chap. 2). All the inner points in the grid can be considered belonging to three different coordinate systems: connected with the point itself, connected with the previous point, connected with the next point. Points at the ends of the open field lines can be observed only within two coordinate systems. So in general there are 27 metric quantities connected with each point. The following subsection describes the procedure for constructing these values (a description of this procedure for turbulence computations can be found in [50]).

Our goal is to construct a set of coordinates (v^1, v^2, τ), which are aligned to an arbitrary magnetic field configuration. In this coordinate system the coordinate τ locates the position along the field line, whereas two others v^1, v^2 determine it's form. These two coordinates are known as Clebsch-type ones. For such a coordinate system the contravariant components of the magnetic field line should be zero e.g. the following demand should be satisfied:

$$B^1 = B^2 = 0 \tag{3.4}$$

First we take a poloidal cut at some $\phi = \phi_0$ with a Cartesian grid on it defined by curves $r = const$ and $z = const$. This determines initial coordinates v^1 and v^2. Each point (r, z) on the

cut serves as a starting point for a magnetic field line and thus $r = const$ and $z = const$ lines in the $\phi = phi_0$ can be considered as a "magnetic surfaces" intersecting the reference cut. It have to be mentioned, that these surfaces do not coincide with the flux surfaces. The magnetic field line passed through each grid point (r_i, z_i, ϕ_0) can be envisaged as the intersection of two "magnetic surfaces". These two surfaces can be described with two equations: $v^1(\vec{R}) = const$ and $v^2(\vec{R}) = const$. On the reference cut ($\phi = \phi_0$) these relationships will degenerate to $r = const$ and $z = const$, respectively. It should be stressed here, that the choice of the Clebsch-type coordinates v^1 and v^2 are not unique. It is possible to define another kind of "magnetic surfaces", but in our setup, coordinates described here are the most suitable.
To have a complete basis the third coordinate should be introduced. The main property of this coordinate is such, that it's covariant basis vector is tangential to the field line. The natural choice of such a coordinate would be the arc length l. However, the most convenient way of describing such a coordinate in our case is a dimensionless coordinate τ. The τ coordinate is connected with the arc length through the following expression:

$$l'(\tau) = \frac{B}{B^\phi}, \tag{3.5}$$

This relation means, that the τ is not a physical parameter for the representation of the field line. The reason for this choice of the parametrization is that τ represents in it an angle-like coordinate. Further we will prove, that this coordinate coincides with the toroidal angle ϕ shifted by some constant value. The field line equation in the cylindrical coordinate system can be written in the form:

$$\frac{B}{dl} = \frac{B^\phi}{d\phi} = \frac{B^r}{dr} = \frac{B^z}{dz}, \tag{3.6}$$

This leads to the expression:

$$\frac{d\phi}{dl} = \frac{B^\phi}{B}, \tag{3.7}$$

Substituting dl from (3.5) we obtain the relation

$$d\tau = d\phi \tag{3.8}$$

Using the fact, that for the reference cut we have $\tau(0) = \phi_0$ as initial condition and integrating we end up with the relationship:

$$\tau = \phi - \phi_0 \tag{3.9}$$

The coordinate system constructed in such a way satisfies the Eq. (3.4). And for the third component of the field line we derive:

$$B^\tau = \vec{B} \cdot \vec{\nabla}\tau = \vec{B} \cdot \vec{\nabla}\phi = B^\phi \tag{3.10}$$

After the definition of the coordinate system we will describe the procedure for constructing the metric coefficients. First let us denote (following the notation in [50]) cylindrical coordinates (r, z, ϕ) as (y^1, y^2, y^3) and τ as v^3.
By definition, the metric coefficients for the transformation from cylindrical coordinate system to our coordinate system are

$$g^{kl} = \sum_{i,j=1}^{3} g_c^{ij} \frac{\partial v^k}{\partial y^i} \frac{\partial v^l}{\partial y^j}, \tag{3.11}$$

where

$$g_c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{r^2} \end{bmatrix} \tag{3.12}$$

From Eq. (3.11) it is obvious, that in order to obtain the metric coefficients we have to calculate the derivatives $\frac{\partial v^k}{\partial y^i}$. Let us denote them as:

$$C_i^k \equiv \frac{\partial v^k}{\partial y^i} \quad (i,k=1,2,3), \tag{3.13}$$

For determining this quantities we will start from rewriting the Eq. (3.4) as

$$\vec{B}\cdot\vec{\nabla}v^l = 0 \quad (l=1,2) \tag{3.14}$$

or using our notation of C_k^l

$$\sum_{k=1}^{3} B_c^k C_k^l = 0 \quad (l=1,2) \tag{3.15}$$

where B_c^k (for $k=1,2,3$) are contravariant cylindrical components of the magnetic field. Now, we differentiate Eq. (3.15) with respect to y^j ($j=1,2,3$) and obtain

$$\sum_{k=1}^{3} B_c^k \frac{\partial C_k^l}{\partial y^j} + \sum_{k=1}^{3} C_k^l \frac{\partial B_c^k}{\partial y^j} \quad (l=1,2), \tag{3.16}$$

and taking into account the relation (which just gives us a constrain on the class of functions used for setting up the coordinates v^1, v^2, namely $v^l \in C^2$)

$$\frac{\partial C_k^l}{\partial y^j} \equiv \frac{\partial C_j^l}{\partial y^k} \tag{3.17}$$

we end up with an expression

$$\sum_{k=1}^{3} B_c^k \frac{\partial C_j^l}{\partial y^k} = -\sum_{k=1}^{3} C_k^l \frac{\partial B_c^k}{\partial y^j} \quad (l=1,2), \tag{3.18}$$

Now, we will proceed further obtaining a system of ordinary differential equations from the system of partial differential Eqs. (3.18). This will be done with the help of the field-line equation written in cylindrical coordinates. From the Eq. (3.6) we take:

$$\begin{aligned} \frac{dr}{dl} &= \frac{B^r}{B} \\ \frac{dz}{dl} &= \frac{B^z}{B} \end{aligned} \tag{3.19}$$

Now we can write down the field-line Eq. (3.19) in terms of τ parametrization (here we also come back to the notation of B_c^k as cylindrical components of the magnetic field and y^k as cylindrical coordinates itself):

$$\frac{dy^k}{d\tau} = l'(\tau)\frac{B_c^k}{B}. \tag{3.20}$$

To go further we can figure out that the full derivatives over C_j^l quantities with respect to τ can be expressed as:

$$\frac{d}{d\tau}C_j^l = \sum_{k=1}^{3} \frac{\partial C_j^l}{\partial y^k}\frac{dy^k}{d\tau} \tag{3.21}$$

The expression (3.21) can be used to construct the $\frac{d}{d\tau}$ over the C_j^l in equation (3.18). Using in addition the Eq. (3.5) we come to the final form:

$$\frac{d}{d\tau}C_j^l = -\sum_{k=1}^{3}\frac{C_k^l}{B^\phi}\frac{\partial B_c^k}{\partial y^j} \tag{3.22}$$

To be able to solve Eqs. (3.22) we have to impose additionally the initial conditions. As we started from the surface $\phi = \phi_0$ with coordinate lines $v^1 = const$, $v^2 = const$ that coincide with the $r = const$, $z = const$ (the isolines in cylindrical coordinates) we have

$$C_n^m(\tau = 0) = \delta_n^m \quad (m, n = 1, 2). \tag{3.23}$$

Due to the relation between coordinates τ and ϕ initial condition for the C_3^3 is

$$C_3^3 = 1 \tag{3.24}$$

The two quantities C_3^1 and C_3^2 are derived from the Eq. (3.15). The exact expression is

$$\begin{aligned} C_3^1\,(\tau = 0) &= -\frac{B^r}{B^\phi} \\ C_3^2\,(\tau = 0) &= -\frac{B^z}{B^\phi} \end{aligned} \tag{3.25}$$

After solving the above system of ordinary differential equations with the initial conditions (3.22)- (3.25) we get all needed g^{ij} quantities.
The Jacobian is another important quantity, that should be calculated. This can be done both from the coefficients g^{ij} and through the procedure similar to that was described for the metric coefficients.
We start to write down the Jacobian after applying a chain rule in the form:

$$J = J_c\left|\frac{\partial(y^1, y^2, y^3)}{\partial(v^1, v^2, \tau)},\right| \tag{3.26}$$

where J_c stands for the Jacobian for cylindrical system and as well known

$$J_c = r, \tag{3.27}$$

So, to compute Jacobian we need to obtain similar quantities as for the metric coefficients. We denote

$$D_j^i \equiv \frac{\partial y^i}{\partial v^j} \quad (i, j = 1, 2, 3),. \tag{3.28}$$

Now we connect new quantities D_j^i with C_j^i through the equality

$$\sum_{k=1}^{3} D_k^l C_m^k = \delta_m^l. \tag{3.29}$$

Then we differentiate the identical equation (3.29) with respect to τ and get

$$\sum_{k=1}^{3} D_k^l \frac{dC_m^k}{d\tau} + \sum_{k=1}^{3} C_m^k \frac{dD_k^l}{d\tau} = 0. \tag{3.30}$$

Finally we can use equation (3.22) for substituting the $\frac{d}{d\tau}C_m^k$ and get the result

$$\frac{d}{d\tau}D_j^l = \sum_{k=1}^{3}\frac{D_j^k}{B^\phi}\frac{\partial B_c^l}{\partial y^k} \quad (l,j=1,2). \tag{3.31}$$

And for initial conditions we have

$$\begin{aligned} D_1^3 &= D_2^3 = 0 \\ D_3^3 &= 1 \\ D_3^1 &= 1 \\ D_1^3 &= \frac{B^r}{B^\phi} \\ D_3^2 &= \frac{B^z}{B^\phi} \end{aligned} \tag{3.32}$$

The above was obtained in a similar way to what was done for C_n^m. We used the equation (3.29) and take into account initial conditions on the $\phi = \phi_0$ surface which look like

$$D_n^m(\tau = 0) = \delta_n^m \quad (m,n=1,2). \tag{3.33}$$

3.3 Findif transport equations

To formulate the set of fluid equations for plasma transport we use a number of additional assumptions:

1. Plasma is quasineutral. This allows us to put $\vec{j} = 0$ and not to solve the electron momentum equation separately;
2. Moreover, we assume that the plasma is locally neutral, hence, we put $n_e = n_i$, so it suffices to solve only one density equation for the plasma as a whole;
3. Anomalous radial transport is assumed, which is much stronger than the classical terms. Thus, we keep all radial transport only diffusive and neglect the classical terms for it;
4. Viscosity tensor is simplified, so that we neglect all cross-terms in it.

These assumptions and, thus, the equations are similar to the ones used in the standard B2 model [51, 52]. Taking these considerations into account we write down the "Findif" equations. So, the plasma density equation has the following form:

$$\begin{aligned} &\frac{\partial n_i}{\partial t} + \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[h^3 V_{i||}n_i - D_{n_i\perp}\left(g^{33}\frac{\partial n_i}{\partial x^3} - (h^3)^2\frac{\partial n_i}{\partial x^3}\right)\right] - \\ &\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}D_{n_i\perp}\left(g^{31}\frac{\partial n_i}{\partial x^1} + g^{32}\frac{\partial n_i}{\partial x^2}\right) - \\ &\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}D_{n_i\perp}\left(g^{11}\frac{\partial n_i}{\partial x^1} + g^{12}\frac{\partial n_i}{\partial x^2} + g^{13}\frac{\partial n_i}{\partial x^3}\right) - \\ &\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}D_{n_i\perp}\left(g^{21}\frac{\partial n_i}{\partial x^1} + g^{22}\frac{\partial n_i}{\partial x^2} + g^{23}\frac{\partial n_i}{\partial x^3}\right) = 0; \end{aligned} \tag{3.34}$$

To achieve a stable solution we use only the parallel component of the pressure for the momentum equations. We take the parallel velocity as a primary variable and, thus, the time

derivative over the density will be a part of the sources. Parallel velocity equation looks as follows:

$$
\begin{aligned}
& mn_i \frac{\partial}{\partial t}\left(V_{i||}\right) + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[mn_i V_{i||}(h^3 V_{i||} + V_{i\perp}^3) - \left[\eta_i^A g^{33} + (\eta_i^0 - \eta_i^A)(h^3)^2\right]\frac{\partial V_{i||}}{\partial x^3}\right] - \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\eta_i^A\left(g^{31}\frac{\partial V_{i||}}{\partial x^1} + g^{32}\frac{\partial V_{i||}}{\partial x^2}\right)\right] + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left[mn_i V_{i||}V_{i\perp}^1 - \eta_i^A\left(g^{11}\frac{\partial V_{i||}}{\partial x^1} + g^{12}\frac{\partial V_{i||}}{\partial x^2} + g^{13}\frac{\partial V_{i||}}{\partial x^3}\right)\right] + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left[mn_i V_{i||}V_{i\perp}^2 - \eta_i^A\left(g^{21}\frac{\partial V_{i||}}{\partial x^1} + g^{22}\frac{\partial V_{i||}}{\partial x^2} + g^{23}\frac{\partial V_{i||}}{\partial x^3}\right)\right] \\
& = -g^{33}\frac{\partial p_i}{\partial x^3} - mV_{i||}\frac{\partial n_i}{\partial t}.
\end{aligned}
\tag{3.35}
$$

We take the temperatures for the heat balance equations as the primary variables. That is why, the time derivative over the density is represented as sources.

In the electron temperature equation we consider only the parallel component of the electric field, which gives us the $V_{i||}\dot{\nabla}_{||}p_e$ term. So, the electron temperature equation looks as follows:

$$
\begin{aligned}
& \frac{3}{2}n_i\frac{\partial}{\partial t}\left(T_e\right) + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\frac{5}{2}n_i T_e(h^3 V_{i||} + V_{i\perp}^3) - \left[\kappa_e^A g^{33} + (\kappa_e^0 - \kappa_e^A)(h^3)^2\right]\frac{\partial T_e}{\partial x^3}\right] - \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\kappa_e^A\left(g^{31}\frac{\partial T_e}{\partial x^1} + g^{32}\frac{\partial T_e}{\partial x^2}\right)\right] + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left[\frac{5}{2}n_i T_e V_{i\perp}^1 - \kappa_e^A\left(g^{11}\frac{\partial T_e}{\partial x^1} + g^{12}\frac{\partial T_e}{\partial x^2} + g^{13}\frac{\partial T_e}{\partial x^3}\right)\right] + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left[\frac{5}{2}n_i T_e V_{i\perp}^2 - \kappa_e^A\left(g^{21}\frac{\partial T_e}{\partial x^1} + g^{22}\frac{\partial T_e}{\partial x^2} + g^{23}\frac{\partial T_e}{\partial x^3}\right)\right] \\
& = (V_{i||}h^3 + V_{i\perp}^3)g^{33}\frac{\partial p_e}{\partial x^3} - \frac{3}{2}T_e\frac{\partial n_i}{\partial t} - k(T_e - T_i);
\end{aligned}
\tag{3.36}
$$

And ion temperature equation looks:

$$
\begin{aligned}
&\frac{3}{2}n_i\frac{\partial}{\partial t}\left(T_i\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\frac{5}{2}n_iT_i(h^3V_{i||}+V_{i\perp}^3)-\left[\kappa_i^Ag^{33}+(\kappa_i^0-\kappa_i^A)(h^3)^2\right]\frac{\partial T_i}{\partial x^3}\right]-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\kappa_i^A\left(g^{31}\frac{\partial T_i}{\partial x^1}+g^{32}\frac{\partial T_i}{\partial x^2}\right)\right]+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left[\frac{5}{2}n_iT_iV_{i\perp}^1-\kappa_i^A\left(g^{11}\frac{\partial T_i}{\partial x^1}+g^{12}\frac{\partial T_i}{\partial x^2}+g^{13}\frac{\partial T_i}{\partial x^3}\right)\right]+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left[\frac{5}{2}n_iT_iV_{i\perp}^2-\kappa_i^A\left(g^{21}\frac{\partial T_i}{\partial x^1}+g^{22}\frac{\partial T_i}{\partial x^2}+g^{23}\frac{\partial T_i}{\partial x^3}\right)\right]\\
&=\frac{3}{2}T_i\frac{\partial n_i}{\partial t}-\frac{\partial}{\partial t}\left(\frac{1}{2}m_in_iV_{i||}^2\right)-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left(\frac{1}{2}m_in_iV_{i\perp}^1(V_{i||})^2\right)-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left(\frac{1}{2}m_in_iV_{i\perp}^2(V_{i||})^2\right)-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left(\frac{1}{2}m_in_i\left(V_{i\perp}^3+h^3V_{i||}\right)(V_{i||})^2\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}g^{11}\left(\frac{1}{2}\eta_i^A\frac{\partial(V_{i||})^2}{\partial x^1}\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}g^{22}\left(\frac{1}{2}\eta_i^A\frac{\partial(V_{i||})^2}{\partial x^2}\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\frac{1}{2}\left[\eta_i^Ag^{33}+(\eta_i^0-\eta_i^A)(h^3)^2\right]\frac{\partial(V_{i||})^2}{\partial x^3}-\\
&(V_{i||}h^3+V_{i\perp}^3)g^{33}\frac{\partial p_e}{\partial x^3}+k(T_e-T_i);
\end{aligned}
\tag{3.37}
$$

In equations (3.35)-(3.37) $V_{i\perp}$ is taken as:

$$
\begin{aligned}
V_{i\perp}^3&=-D_{i\perp}\left[(g^{33}-(h^3)^2)\frac{\partial n_i}{\partial x^3}+g^{31}\frac{\partial n_i}{\partial x^1}+g^{32}\frac{\partial n_i}{\partial x^2}\right]/n_i\\
V_{i\perp}^2&=-D_{i\perp}\left(g^{21}\frac{\partial n_i}{\partial x^1}+g^{22}\frac{\partial n_i}{\partial x^2}+g^{23}\frac{\partial n_i}{\partial x^3}\right)/n_i\\
V_{i\perp}^1&=-D_{i\perp}\left(g^{11}\frac{\partial n_i}{\partial x^1}+g^{12}\frac{\partial n_i}{\partial x^2}+g^{13}\frac{\partial n_i}{\partial x^3}\right)/n_i
\end{aligned}
\tag{3.38}
$$

Due to the choice of the coordinate system (see Chap. 2) the metric coefficients, which characterize geometry enter in all equations. The full metric tensor has 9 values:

$$
\begin{bmatrix}
g^{11} & g^{12} & g^{13}\\
g^{21} & g^{22} & g^{33}\\
g^{31} & g^{32} & g^{33}
\end{bmatrix}
$$

The next chapter is dedicated to the numerical solution of the equations described in current chapter. The codes, that realized the preliminary procedures such as: construction the grid, finding poloidal neighbours and metric coefficients calculation as well as the main code "Findif" will be presented.

Chapter 4

Numerical setup

4.1 Pre-processing

Before starting the main code which solves the set of transport equations a number of the pre-processing procedures have to be done, namely the grid construction, finding poloidal neighbours and calculation of metric coefficients as discussed before.

4.1.1 Mesh generation

The first step is constructing the grid. As it was mentioned in the previous chapter (see Ch. 3) to do this we have to solve the set of Eqs. (3.2). This gives us the possibility to follow the field line, started from some starting point and match the points of intersection between the line and toroidal cuts forming the grid. The field line tracing code described in the present work uses a 4th order Runge-Kutta integrator with adaptive step described in [53] to solve the above equation. The detection of the collision of the field line with the wall on each step of the tracing was implemented. The code needs the wall geometry for determining collisions. The tracing procedure is done by a separate code "gmesh". Two more informations are needed, namely a correct list of starting points and a magnetic field (can be a 2D and a 3D equilibrium). An important parameter, that the user should set is the discriminating radius for neighbour points. The main idea of the algorithm is to provide a rather big amount of starting points covering the region of interest. After tracing a line a discarding procedure discriminates all starting points, that lie in a circle of prescribed radius over each point of the mesh on each plot. This allows one to easily manipulate the point density of the final mesh and, thus, its quality and speed of calculations. Unfortunately, the procedure of constructing the mesh is not completely automatic because it needs a list with the starting points, but with such an approach this list can be easily generated by another code, specific for a concrete device. For example, for W7-X it has been done in 4 steps.

1. Determining suitable points for constructing the outermost and the innermost fluxes.
2. Finding (roughly) the position of the point on $Z = 0$ line from which the island starts on a bean-like shape poloidal cut.
3. Generating a list of starting points on the $Z = 0$ line on all five bean-like shape cuts of the W7-X.
4. Constructing a mesh stencil with the help of tracing routine without checking collisions with boundaries etc. e.g. make a fix number of turns.

After this we can use the stencil mesh for producing different meshes with different density of points on the cut so as with different number of cuts. On each of the tracing step a collision with the wall is checked. These checks can be switched on or off individually for each point in the starting point list. This possibility is used in two cases: first for constructing of the stencil mesh and second for inner most and outer most flux surfaces, which are used as boundaries. Details of the procedure implementation can be found in the algorithm (1) and (2).

Algorithm 1 Mesh generation (field line tracing) procedure

Require: *StartingPoints* array, magnetic field and walls (2D segments or 3D triangle strips), parameters
  $MeshPoints \Leftarrow NULL$
  for all $pi \in StartingPoints$ **do**
    if pi is valid **then**
      $TracedPoints \Leftarrow NULL$
      $FieldLineType \Leftarrow 0$
      $TracingDirection \Leftarrow 1$
      $status = Trace(pi, TracingDirection, TracedPoints, FieldLineType)$ {Tracing procedure fills $TracedPoints$ array in, recognizes the type of the field line, returns the operation termination status}
      $DelRadius \Leftarrow 0$
      if $status = 1$ **then**
        $DelRadius \Leftarrow R$ {R is an input parameter}
      else
        $DelRadius \Leftarrow R * \epsilon$ {ϵ is a very small value}
      end if
      for all $pj \in TracedPoints$ **do**
        $Delete(StartingPoints, pj, DelRadius)$ { Delete all the starting points that lie in $DelRadius$ from pj }
      end for
      if $FieldLineType = OPEN$ **then**
        $SaveToMesh(FieldLineType, TracedPoints)$ {Save the type of the field line and the traced points}
      else if $FieldLineType = CLOSED$ **then**
        $AddFakePoints(TracedPoints)$ {Add some fake points to the innermost and the outermost flux surfaces}
        $SaveToMesh(FieldLineType, TracedPoints)$
      else if $FieldLineType = LONG$ **then**
        print `"ERROR: Line is too long!!!"`
      else if $FieldLineType = SHORT$ **then**
        print `"ERROR: Line is too short!!!"`
      end if
    end if
  end for
  return $MeshPoints$

In addition to this non-parallel version a parallel version has been developed. It has been done to be able to deal with very big meshes which can potentially appear for complex 3D devices. Due to the fact, that the tracing procedure itself can be done completely independent, it is possible to distribute all starting points among the processors. The only place where the exchange of the data is needed is then the eliminating procedure. At the same time the tracing procedure (2) remains the same for both cases. Fig. (4.1) shows an example of the "bean-like" and Fig. (4.2) a "triangle-like" Poincarè plot for the zero beta W7-X case.

Algorithm 2 $Trace(pi, TracingDirection, TracedPoints, FieldLineType)$

Require: Walls information, $MaxIterCount$, $NumbeOfCuts$, $MinNumberOfPointsInLine$

if pi lies between the Cuts **then**
  Trace to the nearest cut
  $pi \Leftarrow PointOnCut$
end if
$\partial\phi = \frac{2\pi}{NumbeOfCuts}$
for $i = 1$ to $MaxIterCount$ **do**
  $ResultPoint \Leftarrow NULL$
  $ResultPoint.\phi \Leftarrow pi.phi + \partial\phi$
  $IntegratorStatus = ODEIntegrator(pi, ResultPoint)$ {Find the coordinates of $ResultPoint$ solving the differential equations $\frac{\partial r}{\partial \phi} = F_1(r, \phi, z)$, $\frac{\partial z}{\partial \phi} = F_2(r, \phi, z)$}
  add $ResultPoint$ to $TracedPoints$
  if $IntegratorStatus = WALL$ AND $TracingDirection = +1$ **then**
    $TracingDirection \Leftarrow -1$
    $TraceStatus = Trace(pi, TracingDirection, TracedPoints, FieldLineType)$ {Recursive call of $Trace$ procedure in the opposite (negative) direction}
    if $TraceStatus = 1$ **then**
      if $TracedPoints.Size < MinNumberOfPointsInLine$ **then**
        $FieldLineType \Leftarrow SHORT$
        return 0
      end if
      $Sort(TracedPoints)$
      return 1
    else
      return 0
    end if
  else if $IntegratorStatus = WALL$ AND $TracingDirection = -1$ **then**
    return 1
  else if $IntegratorStatus = OK$ AND $TracingDirection = +1$ **then**
    if $CheckLineClosed(pi, ResultPoint) = 1$ **then**
      $FieldLineType \Leftarrow CLOSED$
      return 1
    end if
  end if
end for
$FieldLineType \Leftarrow LONG$
return 0

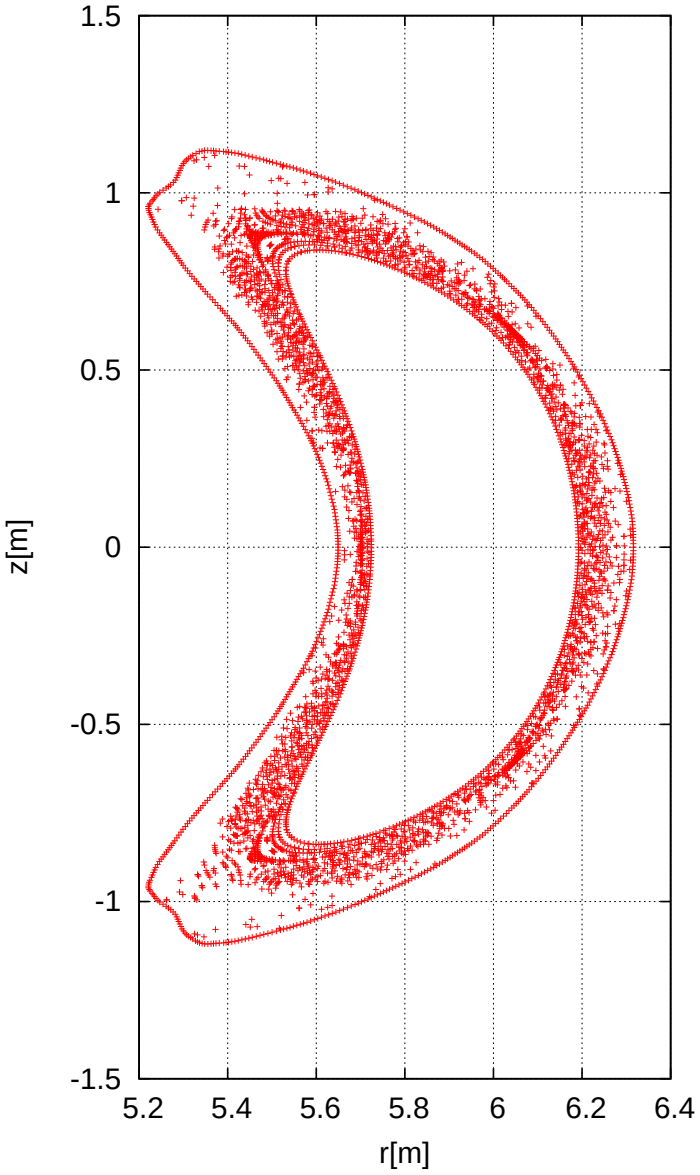

Figure 4.1: *W7-X bean-like shape Poincarè plot.*

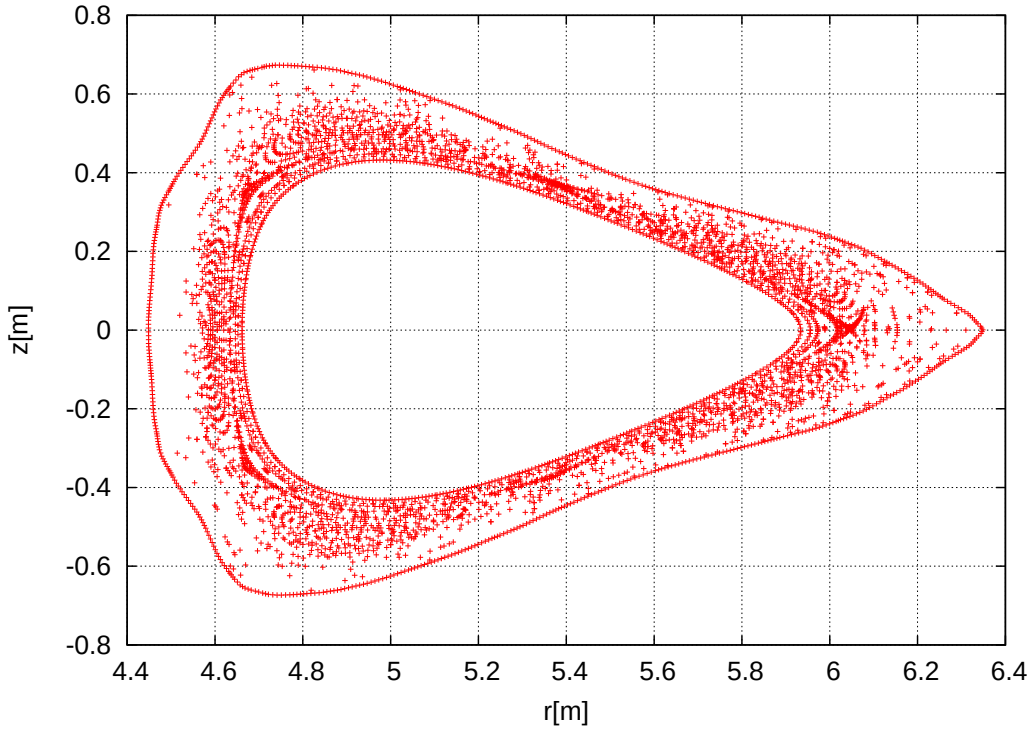

Figure 4.2: *W7-X triangle-like shape Poincarè plot.*

4.1.2 Metrics calculation

To compute metrics we have to solve the set of Eqs. (3.22), (3.25) and Eqs. (3.31), (3.33) derived in chapter 3. This is done by the separate code named "tracer". The code is suited for a great variety of tasks and machines. It produces metric information not only for this code, but also for turbulence codes GS2 [54] and GENE [55]. Nowadays "tracer" is a part of the GENE framework.

The code consists of a few subsystems, which can be individually adapted to the work needs. The input part, that reads the list of starting points and the output part, that writes the metric coefficients are task-dependent. The type of the input and output files can be changed through input parameters in the configuration file. The part, which provides magnetic field information, is device-dependent. Its behaviour can also be controlled through the configuration file. At the moment, the list of supported devices includes: W7-X, NCSX, DIII-D, TOK, NSTX, AUG, JET and TEXTOR. There are two main modes, in which this subsystem can work. First is "normal mode", in which the code uses the magnetic field information defined in the rectangular mesh nodes and second is a "VMEC" mode, which allows to use directly information provided by the "VMEC" [56] code. The core routine which traces along the magnetic field line and calculates all metrics information is independent of both the machine- and the task-type modes. This routine sequentially takes starting points, and depending on its type, traces along the magnetic line from that point. The direction, distance of the trace and the number of intermediate output points are defined by the type of the starting point. Due to the fact, that calculations of the metric coefficients for individual starting points are independent from each other, an effective parallelization was done. For the parallel version the whole set of the starting points was divided between processes and then all calculations were done independently. The output in the parallel version of the code is also realized independently on each process into the "netCDF" file format. Afterwards, it is possible to merge all outputs from all processors into one file with the help of special utility written for that purpose.

4.1.3 Determining poloidal neighbours

To construct a discrete form of the transport equations we are solving (Eqs. (3.34)-(3.37)), we need the neighbours information on each of the point of the mesh both in toroidal and poloidal directions. Due to the tracing procedure that is used for constructing the mesh we already have a list of toroidal neighbours. Finding neighbours on the Poincaré plot is a separate task that is done as a pre-processing step.

In the previous version of the code (see [57]) this was done with the help of a Delaunay triangulation procedure (see [58]). Delaunay triangulation covers the plane with a number of scattered points with a set of non-overlapping triangles, where the vertices lie in these points. An important property of the resulting set of triangles is that these triangles tend to form equilateral triangles. The drawback of the procedure is that the simple realization of the algorithm is very time consuming and one can not control the number of the neighbours for each point which it will produce. Also in the case of a very non-homogeneous mesh one can get very narrow triangles which can lead to large numerical errors. The use of the Delaunay triangulation was caused by the way of discretization of the equations in the previous version of the code. It was necessary to use an information about triangles as the code constructed fluxes in the center of each of them and then took the divergence of those fluxes.

In the present code we use a direct calculation of the derivatives in the mesh points and thus we don't need an information about triangles. Instead of this we use some fixed number of nearby points. The task in this case is to find as equally spatially distributed neighbours as possible. It is clear that the most preferable points are those with the shortest distances to the reference point. So it should be a trade-off between how far the neighbours are away and how uniform they are distributed.

The main idea of the neighbour-finding algorithm in this work follows [59]. It was proposed to use a "shadow" from points that were already chosen to discriminate other points. Taken this idea as a basis we constructed our own algorithm for solving the problem.

There are three parameters: number of neighbours for each point, far cut-off border, and near cut-off border. The far cut-off border is needed for speed up the code. It should be big enough to cover sufficiently large number and well distributed neighbours. The near cut-off border is used to discard the points which lie too close to the reference point. Taking all the points one by one the code finds neighbours for them independently from each other.

The neighbours procedure is done in a separate code ("nbh"). In this code neighbours for each of the point are found within several steps. As a first step, only points that lie between two borders are taken. Then they are sorted according to their distance to the reference point. After that the initial shadowing procedure is done. During this procedure points from the sorted list are taken one by one and, if they are not already discarded, a shadow circle is constructed around it. Then the rest of the points are marked to be shadowed or not. If some point occurs to be shadowed it is labeled as discarded and the procedure continues. The procedure is presented schematically in Fig. 4.3. For the radius of the shadow circle we use the following considerations:

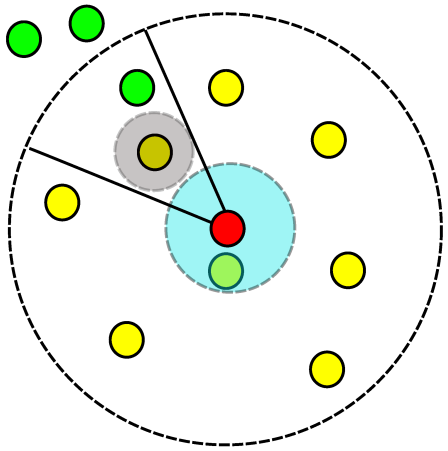

Figure 4.3: *Schematics of the shadowing procedure.The reference point for which the set of neighbours should be found is red. Two cut-off border line are painted with a dashed line. The inner cut-off region is light blue. The shadow circle around the point is grey. Discarded points are green.*

if we assume that all our neighbours (e.g. eight) are equally distributed around the reference point, then the rest of the region should be completely shadowed by them (see Fig. 4.4). In this case we can calculate the radius of the shadow circle as:

$$R_{sh} = R_0 sin\left(\frac{2\pi}{8 \cdot 2}\right) \tag{4.1}$$

Where R_{sh} is a radius of a shadow circle, R_0 is the distance to the reference point.
So for the case of spatially distributed points we can take the shadowing radius as $R_{sh} = R_0 sin\left(\frac{\pi}{n_N}\right)$, where n_N is the number of needed neighbours (8 in the example above). This formula is only valid for the case of points equally distributed in the space around the reference point, which is usually not true for practical cases. To achieve a better result the formula have to be corrected in order to take into account unequal distribution of neighbours in all directions. To do this we use two passes of the shadowing procedure. In the first pass we assume, that all points are distributed within an angle of $\pi/4$, which in fact is an underestimate (usually it varies

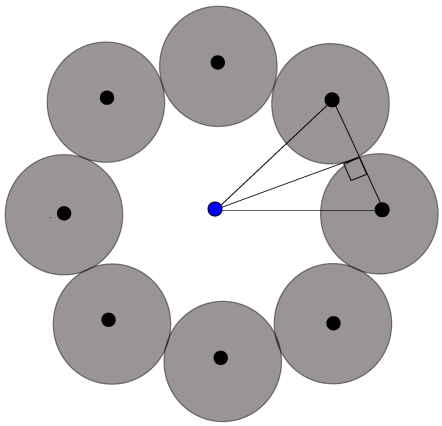

Figure 4.4: *The figure demonstrates the consideration, that was used for obtaining the radius of the shadow circle. Eight points around the reference point are distributed equally in space. Therefore they lie on the circle circumscribed about the reference point and form a regular octagon. The radius of the shadow circle should be the half of the side of the octagon.*

from π to 2π). Then, for each neighbour point i we calculate the shadow radius as:

$$R^i_{sh} = R^i_0 sin\left(\frac{\pi\tilde{k}}{n_N}\right), \tag{4.2}$$

where $\tilde{k} = 0.25$ according to our "a priori" assumption. We take such a small coefficient for this step in order to obtain more neighbours than needed. After that we correct the coefficient taken for the calculation of the shadowing radius in such a way:

$$k = \tilde{k}\frac{\tilde{n}_N}{n_N}, \tag{4.3}$$

where $\tilde{n}_N$ is the number of neighbours found after initial shadowing procedure. The corrected coefficient k then is used for the final shadowing procedure. In the final shadowing stage we take all the points from the list of points that was found in the region between two cut-off borders. As the first point is the nearest, we take it as it is. After that the code calculates the penalty factors for the rest of the points. The penalty factor is constructed in such a way, that it is proportional to the distance from the point to the reference point and inversely related to the distance to the center of the summary shadow region. If the point lies inside the shadow region it's penalty factor grows much faster with decreasing the distance to the center of the shadow region than if the point lies outside the region. Then, the code takes as a next neighbour the point with the smallest penalty factor. After that, the shadowing region is recalculated in order to include the shadowing from the newly added point and the procedure continues. When there are few separate parts of the shadowing region the penalty factor is calculated as a sum of the factors with respect to each of the shadowing sectors. The procedure described above is executed until the requested number of points are chosen. The described algorithm can be found in Alg. 3. An example of the resulting neighbours on a part of the W7-X mesh is shown in Fig. 4.5.

Algorithm 3 Finding neighbours of a specific point. Return $ListOfNeighbours$.

Require: Point p, List of grid points $GridPts$, $NumberOfNeighbours$, Initial radius R_0, $NearBorder$, $FarBorder$, empty of partially filled $ListOfNeighbours$
Find in $GridPts$ all the point between $(NearBorder, FarBorder)$
$Pts \Leftarrow$ found points
$Sort(Pts)$
Execute initial shadowing:
for all $pi \in Pts$ **do**
  if pi is not discarded **then**
    Draw a shadow circle with radius R_0 around it
    for $pj = pi + 1$ to $Pts.Last$ **do**
      if pj is shadowed **then**
        $pj \Leftarrow$ discard
      end if
    end for
  end if
end for
Update R_0
Remove discarded points from Pts
Execute final shadowing:
Add $Pts[1]$ to $ListOfNeighbours$
$p1 \Leftarrow Pts[1]$; Remove 1 point from Pts;
Calculate $ShadowAngle$
while $ListOfNeighbours.Size < NumberOfNeighbours$ **do**
  $MinPenalty \Leftarrow 10000000$; pn
  for all $pi \in Pts$ **do**
    Calculate $PenaltyFactor \Leftarrow F(pi, p1, ShadowAngle)$
    if $PenaltyFactor < MinPenalty$ **then**
      $pn \Leftarrow pi$;
      $MinPenalty \Leftarrow PenaltyFactor$
    end if
  end for
  Add pn to $ListOfNeighbours$
end while
return $ListOfNeighbours$

4.2 Finite difference solution of the conduction-convection problem

All transport equations ((3.34) – (3.37)) can be rewritten in a generalized form:

$$A_t \frac{\partial f}{\partial t} + \underbrace{\frac{1}{\sqrt{g}} \frac{\partial}{\partial x^i} \sqrt{g} \left(V^i f - D^{ij} \frac{\partial f}{\partial x^j} \right)}_{L \circ f} + \nu f = Q, \tag{4.4}$$

where D^{ij} is given by

$$D^{ij} = D_\perp g^{ij} + \left(D_{||} - D_\perp \right) h^i h^j \tag{4.5}$$

and it acts as a generalized diffusion matrix; V^i is a generalised velocity; A_t is a coefficient possibly dependent on other quantities like density, velocity etc.; ν is an implicit and Q an

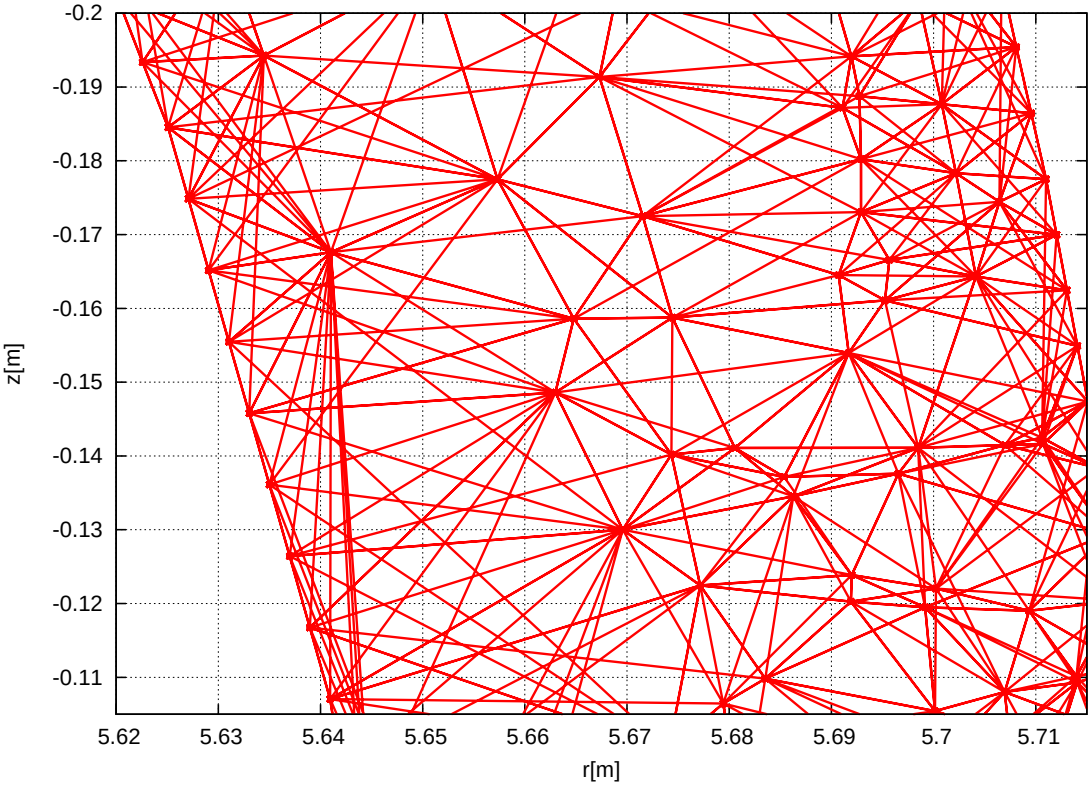

Figure 4.5: *A part of the W7-X bean-like shape Poincarè plot with neighbours on it.*

explicit part of the source. Here, a summation over repeated subscripts i, j is assumed. Hence, to solve the coupled set of transport equations it is enough to develop a solver for a generalized form of conduction-convection equations and assign appropriate values to generalized diffusion coefficients, velocity, A_t coefficient and sources for each equation.
Different discretization schemes for parallel and "mixed" terms were applied. Along the field line we use a simple central-difference scheme. For poloidal and "mixed" terms this is impossible because of the unstructured character of the mesh on the cut. That is why we are forced to use other methods, like a least-square method (see appendix refch:appendix).
We take from Eq. ((4.4)) operator L acting on f, and split it into two parts:

$$L = L_{||} + L_{\angle}, \tag{4.6}$$

where $L_{||}$ is an L with $(i, j) = (3, 3)$ and $L_{\angle}$ denotes the rest combination of indexes for the L. In this section we are interested in discretizing the $L_{||}$ term.

4.2.1 Discretization of parallel terms

Upwind scheme

First let's construct the simplest scheme for the parallel term. To do this we have to consider the $L_{||} \circ f$. This is a parallel part of the divergence of the parallel flux:

$$L_{||} \circ f = \nabla_{||} F_{||} = \frac{1}{\sqrt{g}} \frac{\partial}{\partial x^3} \left(\sqrt{g} \left(V\Phi - D \frac{\partial f}{\partial x^3} \right) \right) \tag{4.7}$$

To discretize it, let us denote $\triangle = (\delta x_w^+ + \delta x_e^-)^{-1}$. After applying the central difference scheme to equation (4.7) we have

$$\nabla_{||} F_{||} = \frac{1}{\sqrt{g}} \triangle \left(\sqrt{g_e} F_e - \sqrt{g_w} F_w \right), \tag{4.8}$$

where F_w and F_e are defined due to the up-wind scheme as follows:

$$\begin{cases} F_w = V_w f_W - D_w \dfrac{f_P - f_W}{\delta x_w} & \text{if } V_w > 0 \\ F_w = V_w f_P - D_w \dfrac{f_P - f_W}{\delta x_w} & otherwise \end{cases} \tag{4.9}$$

$$\begin{cases} F_e = V_e f_P - D_e \dfrac{f_E - f_P}{\delta x_e} & \text{if } V_e > 0 \\ F_e = V_e f_E - D_e \dfrac{f_E - f_P}{\delta x_e} & otherwise \end{cases} \tag{4.10}$$

Here we use the notation from [60] where P means the reference point, W – west, E – east, w – west interface point and e – east interface point.
To express both cases of F_w and F_e simultaneously we can rewrite equations (4.9) and (4.10) in the following form:

$$F_w = \frac{1}{2} f_W \left(V_w + |V_w| \right) + \frac{1}{2} f_P \left(V_w - |V_w| \right) - D_w \frac{f_P - f_W}{\delta x_w} \tag{4.11}$$

$$F_e = -\frac{1}{2} f_E \left(|V_e| - V_e \right) + \frac{1}{2} f_P \left(|V_e| + V_e \right) - D_e \frac{f_E - f_P}{\delta x_e} \tag{4.12}$$

And the complete discretized form of equation (4.7) is

$$\begin{aligned} \nabla_{||} F_{||} = \frac{1}{\sqrt{g_P}} \triangle \left(\sqrt{g_e} F_e - \sqrt{g_w} F_w \right) \equiv \\ f_E \left[\sqrt{g_e} \left(-\frac{1}{2} \left(|V_e| - V_e \right) - \frac{D_e}{\delta x_e} \right) \right] + \\ f_P \left[\sqrt{g_e} \left(\frac{1}{2} \left(|V_e| + V_e \right) + \frac{D_e}{\delta x_e} \right) - \sqrt{g_w} \left(\frac{1}{2} \left(V_w - |V_w| \right) - \frac{D_w}{\delta x_w} \right) \right] + \\ f_W \left[-\sqrt{g_w} \left(\frac{1}{2} \left(|V_w| + V_w \right) + \frac{D_w}{\delta x_w} \right) \right] \end{aligned} \tag{4.13}$$

Generalized upwind scheme

The method described above is of course not the only one for discretization such an equation. It is possible to construct a more accurate approximation for mixed conduction and convection. Such a scheme was proposed by Patankar [60]. Then it was generalized in [61] for a non-staggered grid case with a constant source within the cell. The idea of the method is simple and at the same time very prominent. First let's assume that within a region of one cell the diffusion coefficient, the velocity and source remains constant. After that we can solve a boundary value problem with a Dirichlet boundary conditions on one cell for such an equation:

$$\frac{\partial}{\partial x^3} \left(F\phi - \Gamma \frac{\partial \phi}{\partial x^3} \right) = S \tag{4.14}$$

where

$$F = \sqrt{g}V^3 \tag{4.15}$$
$$\Gamma = \sqrt{g}D^{33} \tag{4.16}$$
$$S = \sqrt{g}Q \tag{4.17}$$

Here the notation from [61] was used. For the specific assumptions we can replace the partial with full derivatives, so Eq. (4.14) is (omitting the indices)

$$\frac{d}{dx}\left(F\phi - \Gamma\frac{d\phi}{dx}\right) = S \tag{4.18}$$

Eq. (4.18) can be solved exactly if Γ and F are taken to be constant and if S is a function of x only or a linear function of ϕ. Here we consider only the case with a constant S. To solve the equation (4.18) first we directly integrate it. After integration we have:

$$F\Phi - \Gamma\frac{d\Phi}{dx} = Sx + c \tag{4.19}$$

This differential equation belongs to a class of first-order linear ordinary differential equations:

$$y'(x) + p(x)\,y(x) = r(x) \tag{4.20}$$

The general solution of such kind of equation can be written in the form

$$y = \exp(-a(x))\left(\int r(x)\exp(a(x))\,dx + k\right) \tag{4.21}$$

where $a(x) = \int p(x)\,dx$. We rewrite Equation (4.19) in the form of Equation (4.20)

$$\frac{d\Phi}{dx} - \frac{F}{\Gamma}\Phi = -\frac{1}{\Gamma}(Sx + c) \tag{4.22}$$

and assign $p(x)$ as

$$p(x) = -\frac{F}{\Gamma}$$

$r(x)$ as

$$r(x) = -\frac{1}{\Gamma}(Sx + c)$$

and $a(x)$ as

$$a(x) = -\frac{F}{\Gamma}x.$$

With these definition the resulting solution is written as

$$\Phi = \exp\left(\frac{F}{\Gamma}x\right)\left(\int -\frac{1}{\Gamma}(Sx + c)\exp\left(-\frac{F}{\Gamma}x\right)dx + k\right) \tag{4.23}$$

In the expression above the integral splits into two pieces which can be taken separately:

$$\begin{aligned} -\frac{1}{\Gamma}S\int x\exp\left(-\frac{Fx}{\Gamma}\right)dx &= \frac{S}{F}x\exp\left(-\frac{F}{\Gamma}x\right) + \frac{S\Gamma}{F^2}\exp\left(-\frac{F}{\Gamma}x\right) \\ -\int\frac{1}{\Gamma}C\exp\left(-\frac{Fx}{\Gamma}\right)dx &= \frac{C}{F}\exp\left(-\frac{Fx}{\Gamma}\right) \end{aligned} \tag{4.24}$$

Substituting the resulting expressions from (4.24) into Eq. (4.23) we get

$$\begin{aligned}\Phi &= \exp\left(\frac{F}{\Gamma}x\right)\left(\int -\frac{1}{\Gamma}\left(Sx+c\right)\exp\left(-\frac{F}{\Gamma}x\right)dx + k\right)\\ &= \frac{S}{F}x + \frac{S\Gamma}{F^2} + \frac{C}{F} + k\exp\left(\frac{Fx}{\Gamma}\right)\end{aligned} \tag{4.25}$$

To determine the constants we have to use the boundary conditions which in our case are as follows:

$$\begin{aligned}x = 0 &: \Phi = \Phi_0\\ x = L &: \Phi = \Phi_L\end{aligned} \tag{4.26}$$

We obtain coefficients:

$$\begin{aligned}k &= \frac{\Phi_L - \Phi_0 - \dfrac{SL}{F}}{\exp\left(\dfrac{SL}{F}\right) - 1}\\ C &= \Phi_0 F - \frac{S\Gamma}{F} - F\frac{\Phi_L - \Phi_0 - \dfrac{SL}{F}}{\exp\left(\dfrac{SL}{F}\right) - 1}\end{aligned} \tag{4.27}$$

After the substitution of the coefficients from (4.27) into the (4.25) we finally get the solution:

$$\Phi = \Phi_0 + \frac{S}{F}x + \frac{\Phi_L - \Phi_0 - (SL/F)}{\exp\left(FL/\Gamma\right) - 1}\left[\exp\left(\frac{Fx}{\Gamma}\right) - 1\right] \tag{4.28}$$

Then, the flux J^x is

$$J^x = F\Phi - \Gamma\frac{d\Phi}{dx} = F\Phi_0 + Sx - F\frac{\Phi_L - \Phi_0 - SL/F}{\exp\left(FL/\Gamma\right) - 1} - \frac{S\Gamma}{F} \tag{4.29}$$

With this definition, Equation (4.18) can be written in the form

$$\frac{dJ^x}{dx} = S \tag{4.30}$$

Now we can integrate Eq. (4.30) withing the cell e.g. from the interface point w to the interface point e. However here we should take the flux J^x and thus the source S, the convection F and the conduction Γ constant between grid points (e.g. P,W and P,E). After integration we have:

$$J^x_e - J^x_w = S_e\delta x^-_e + S_w\delta x^+_w \tag{4.31}$$

The solution (4.28) is continuous only between the grid points and thus we can obtain fluxes on the interface points basing on the correspondent solution for the nearby grid points. So for J^x_e we take δx_e as L, Φ_P as Φ_0, Φ_E as Φ_L. Also quantities F, Γ, S are taken as F_e, Γ_e, S_e. And coordinate x will be the distance from the first point i.e. δx^-_e. For the J^x_e flux we take correspondent values $\delta x_w, \Phi_W, \Phi_P, F_w, \Gamma_w, S_w, \delta x^-_w$ as $L, \Phi_0, \Phi_L, F, \Gamma, S, x$. In this case, the value Φ_P is taken as Φ_L, whereas for the J^x_e it was taken as Φ_0. So the fluxes look now as follows:

$$\begin{aligned}J^x_e &= F_e\Phi_P + S_e\delta x^-_e - F_e\frac{\Phi_E - \Phi_P - S_e\delta x_e/F_e}{\exp\left(F_e\delta x_e/\Gamma_e\right) - 1} - \frac{S_e\Gamma_e}{F_e}\\ J^x_w &= F\Phi_W + S_w\delta x^+_w - F_w\frac{\phi_P - \phi_W - S_w\delta x_w/F_w}{\exp\left(F_w\delta x_w/\Gamma_w\right) - 1} - \frac{S_w\Gamma_w}{F_w}\end{aligned} \tag{4.32}$$

Further, expressions for J_e^x and J_w^x are substituted into Eq. (4.31) and coefficients are combined such, that we end up with the expression:

$$C\Phi_W + B\Phi_P + A\Phi_E = R. \tag{4.33}$$

To do this we also introduce a so called Peclèt number, which is the ratio between convection and conduction.

$$P_w = \frac{Fwe\delta x_w}{\Gamma_w}, \tag{4.34}$$

and P_e as

$$P_e = \frac{F_e\delta x_e}{\Gamma_e}. \tag{4.35}$$

Also taking into account, that in our setup we have $\delta x_w^- = \delta x_w^+ = 0.5\delta x_w$ and $\delta x_e^- = \delta x_e^+ = 0.5\delta x_e$, we find A, B, C as

$$\begin{aligned} A &= -\frac{\Gamma_e}{\delta x_e}\left(\frac{P_e}{\exp P_e - 1}\right) \\ B &= F_e + \frac{F_e}{\exp P_e - 1} + \frac{F_w}{\exp P_w - 1} \\ C &= -\frac{\Gamma_w}{\delta x_w}\left(P_w + \frac{P_w}{\exp P_w - 1}\right) \end{aligned} \tag{4.36}$$

We denote as $A(P)$ the function:

$$A(P) = \frac{P}{\exp P - 1} \tag{4.37}$$

and rewrite the expression for C coefficient as

$$C = -\frac{\Gamma_w}{\delta x_w}\left(\frac{-P_w}{\exp(-P_w) - 1}\right).$$

Finally, we write the coefficients as

$$\begin{aligned} A &= -\frac{\Gamma_e}{\delta x_e}A(P_e) \\ B &= F_e - F_w - C - A \\ C &= -\frac{\Gamma_w}{\delta x_w}A(-P_w) \end{aligned} \tag{4.38}$$

Now, we come to the discussion of the source term and coefficient R. After some algebraic transformation we can obtain R:

$$R = S_e\delta x_e\left(\frac{1}{P_e} - \frac{1}{\exp P_e - 1}\right) + S_w\delta x_w\left(1 - \frac{1}{P_w} + \frac{1}{\exp P_w - 1}\right) \tag{4.39}$$

Another function can be introduced

$$W(P) = \frac{1}{P} - \frac{1}{\exp P - 1}. \tag{4.40}$$

which is connected with the previously introduced function A as

$$W(P) = \frac{1 - A(P)}{P}. \tag{4.41}$$

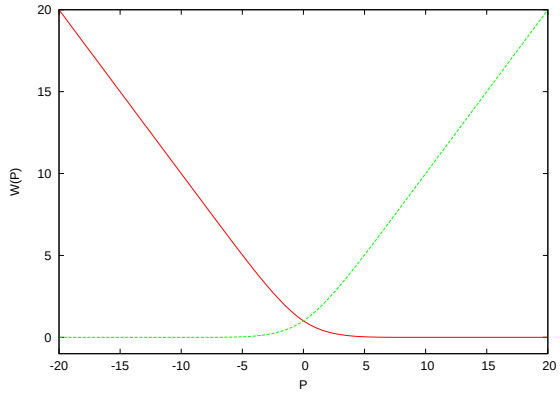

Figure 4.6: Red curve presents $A(p)$ and green $A(-P)$

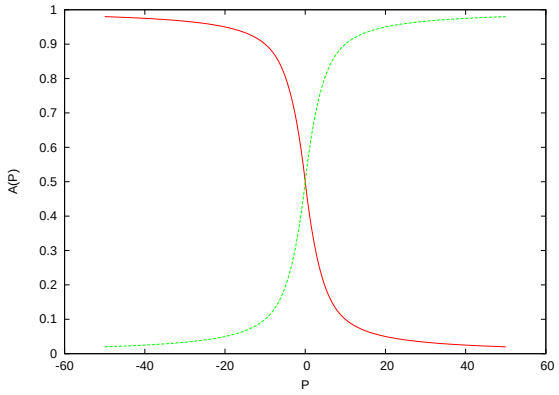

Figure 4.7: Red curve presents $W(p)$ and green $1 - W(P)$

With such a definition the coefficient R looks like:

$$R = S_e \delta x_e W(P_e) + S_w \delta x_w \left(1 - W(P_w)\right) \tag{4.42}$$

The functions A and W are illustrated in Figs. 4.6 and 4.7. For large negative values of the Peclèt number the function $A(P)$ can be approximated with the value of P itself and for $P \to +\infty$ it approaches 0. The opposite behaviour has the function $A(-P)$.
The $W(P)$ and $1 - W(P)$ act as step functions. The relative impact of the sources vary from 1 to 0 depending on the sign and the strength of the Peclèt number such that for large negative values the front source have the greatest impact and for large positive values the one behind. For zero Peclèt number the velocity contribution from both sources are equal ($\frac{1}{2}$ of the total). In most cases, the upwind scheme is enough for obtaining a correct solution, on the other hand computational costs of the generalized upwind scheme can be quite big, so that one can be satisfied with a simpler solution. Therefore, a simple upwind scheme is used and only the correction of the source for the pressure gradient term in the momentum equation is done with the algorithm described here. The implementation of such a weighting of the neighboring sources allows to avoid an unstable behaviour of the equation known in literature [61].

4.2.2 Discretization of the non-parallel terms

The non-parallel terms are treated in a completely different way. Here, we do not express terms as a divergence of the flux. Instead we use a chain rule to obtain "pure" derivatives over our major quantity and then calculate these derivatives with the help of the "free point" method. Taking the definition of the $L_{\angle}$ from Eq. (4.6) and using Eq. (4.4) we can write down:

$$L_{\angle} \circ f = \frac{1}{\sqrt{g}} \frac{\partial}{\partial x^i} \sqrt{g} \left(V^i f - D^{ij} \frac{\partial f}{\partial x^j} \right) \tag{4.43}$$

and after applying the chain rule we have:

$$L_{\angle} \circ f = \frac{1}{\sqrt{g}} \left[\frac{\partial}{\partial x^i} \left(\sqrt{g} V^i f \right) - \sqrt{g} D^{ij} \frac{\partial^2}{\partial x^i \partial x^j} f - f \frac{\partial^2}{\partial x^i \partial x^j} \left(\sqrt{g} D^{ij} \right) \right] \tag{4.44}$$

To discretize Eq. (4.44) we have to find out the discrete form of the first order derivatives $\dfrac{\partial}{\partial x^i}$ and the second order derivatives $\dfrac{\partial^2}{\partial x^i \partial x^j}$. Having an information about poloidal neighbours for

each point we want in general to calculate their weighting coefficients. The coefficients have to give us a derivative value while multiplying the function values in the neighbours and reference point on them:

$$f' = \sum_{i=0}^{N} a_i f_i, \tag{4.45}$$

where i is the index of neighbour and the reference value, N is the neighbours, a_i is the coefficient for i'th neighbour. A reference point is labeled as $i = 0$. In the same way we want to discretize all first and second derivatives. One of the possibilities to construct such a set of coefficients is to use the least squares method. We can estimate (using a first order derivatives) a function value in some neighbour point i using an expression:

$$\tilde{f}_i \approx f_0 + f'_x(x_i - x_0) + f'_y(y_i - y_0) \tag{4.46}$$

where,
f_0 : *the function value in the reference point*
$\tilde{f}_i$: *the estimated function value in the neighbour point i*
f'_x, f'_y : *partial derivatives* $\partial/\partial x$, $\partial/\partial y$
x_i, y_i : x *and* y *coordinates of a poloidal neighbour i*
x_0, y_0 : x *and* y *coordinates of the reference point*
The better we find partial derivatives f'_x, f'_y the closer the difference between the value of the function f in the point i and its estimation based on the reference point. As we have not only one neighbour we can construct the same estimations for all the neighbour points. We want to fit partial derivatives in the reference point in such a way, that they will approximate in a best way all the values of the functions in all the neighbours. To do this we say, that f'_x and f'_y have to minimize the following functional:

$$A = \sum_{i=1}^{N} (f_i - f_0 - f'_x \Delta x_i - f'_y \Delta y_i)^2 \omega_i^2 \tag{4.47}$$

where N stands for number of neighbours and ω_i as some additional weighting factors. In our realization we take these weighting factors reverse proportional to the distance from the reference point to the neighbour one. The reason for that is that we expect the bigger impact from the near points, then from far. To find the minimum for (4.47) derivatives should satisfy:

$$\begin{aligned} \frac{\partial A}{\partial f'_x} &= 0 \\ \frac{\partial A}{\partial f'_y} &= 0 \end{aligned} \tag{4.48}$$

and taking the partial derivatives of the functional A in (4.47) we get (in the following we omit the boundaries for the sum):

$$\begin{aligned} &\sum_i (f_i - f_0 - f'_x \Delta x_i - f'_y \Delta y_i) \omega_i^2 \Delta x_i \\ &\sum_i (f_i - f_0 - f'_x \Delta x_i - f'_y \Delta y_i) \omega_i^2 \Delta y_i \end{aligned} \tag{4.49}$$

The equation (4.49) can be solved over the derivatives f'_x, f'_y. The solution will be represented exactly as needed for determining coefficients like in (4.45). Coefficients will depend only on the differences Δx, Δy and weighting factors ω.

The same procedure can be done for determining second order derivatives $\frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial x \partial y}$. To do this, we construct first an approximation of the function f in some neighbour point i based on the reference point 0 as we did before, but keeping now the second order terms:

$$\tilde{f}_i \approx f_0 + f'_x(x_i - x_0) + f'_y(y_i - y_0) + \frac{1}{2}\left(f''_{xx}(x_i - x_0)^2 + 2f''_{xy}(x_i - x_0)(y_i - y_0) + f''_{yy}(y_i - y_0)^2\right) \tag{4.50}$$

Then, taking the definition above we construct another functional, that have to be minimized:

$$B = \sum_i \left[\left(f_i - f_0 - f'_x\Delta x_i - f'_y\Delta y_i\right) - \frac{1}{2}\left(f''_{xx}\Delta x_i^2 + 2f''_{xy}\Delta x_i\Delta y_i + f''_{yy}\Delta y_i^2\right)^2\right]^2 \omega_i^4 \tag{4.51}$$

To minimize Eq. (4.51) second derivatives should satisfy a set of differential equations:

$$\begin{aligned} \frac{\partial B}{\partial f''_{xx}} &= 0 \\ \frac{\partial B}{\partial f''_{xy}} &= 0 \\ \frac{\partial B}{\partial f''_{yy}} &= 0 \end{aligned} \tag{4.52}$$

which leads to the linear set of equations:

$$\begin{aligned} \sum_i \Delta x_i^2 \left[\varepsilon_i - \frac{1}{2}\left(f''_{xx}\Delta x_i^2 + 2f''_{xy}\Delta x_i\Delta x_y + f''_{yy}\Delta y_i^2\right)\right]\omega_i^4 \\ \sum_i \Delta x_i\Delta y_i \left[\varepsilon_i - \frac{1}{2}\left(f''_{xx}\Delta x_i^2 + 2f''_{xy}\Delta x_i\Delta x_y + f''_{yy}\Delta y_i^2\right)\right]\omega_i^4 \\ \sum_i \Delta y_i^2 \left[\varepsilon_i - \frac{1}{2}\left(f''_{xx}\Delta x_i^2 + 2f''_{xy}\Delta x_i\Delta x_y + f''_{yy}\Delta y_i^2\right)\right]\omega_i^4 \end{aligned} \tag{4.53}$$

where $\varepsilon_i = f_i - f_0 - f'_x\Delta x_i - f'_y\Delta y_i$. The first derivatives in the definition of ε are obtained with the help of the procedure already discussed. As we use here the same number of points as for the first derivative it is clear, that the resulting error of the procedure for obtaining the second order derivatives will be squared with respect to the one of the procedure for the first order derivatives.

4.2.3 Conduction-convection solver

In the present version of the main code "Findif" a single solver for solving the conduction-convection equation in a general form was developed. The code solves sequentially a number of such equations. Due to such a design it is easier for extensions and management. For example, withing the configuration file one can set which equations to solve, switch different sources on and off and manipulate all transport coefficients. For adding new equations (e.g. impurities) one has to define all physical quantities without modifying the main solver routines. Instead of using a fully implicit scheme (resulting in large sparse matrices, see Fig. 4.8) a so-called "semi-explicit" scheme is used. This means that for the parallel direction all terms are treated implicitly, whereas the rest of the terms are taken explicitly. This allows to reduce the amount of memory needed for the calculations and also to use a simple tridiagonal and cyclic tridiagonal solvers which are extremely fast. After using this scheme the amount of memory needed for all the coefficients passed to the solver reduces by 5-7 times (see Fig. 4.9). The time needed for

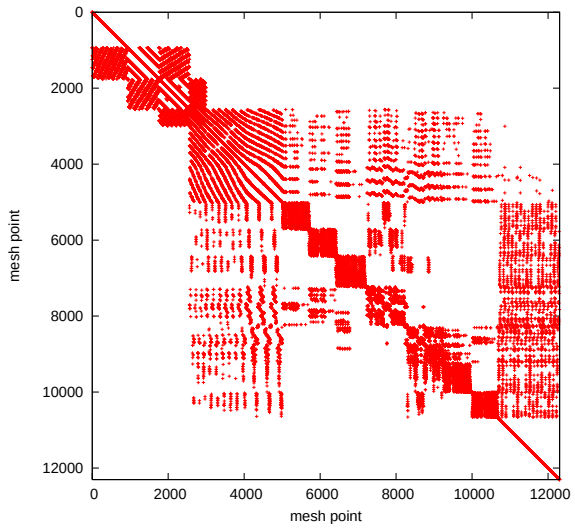

Figure 4.8: *The matrix used for solving the conduction-convection problem with a fully implicit scheme.*

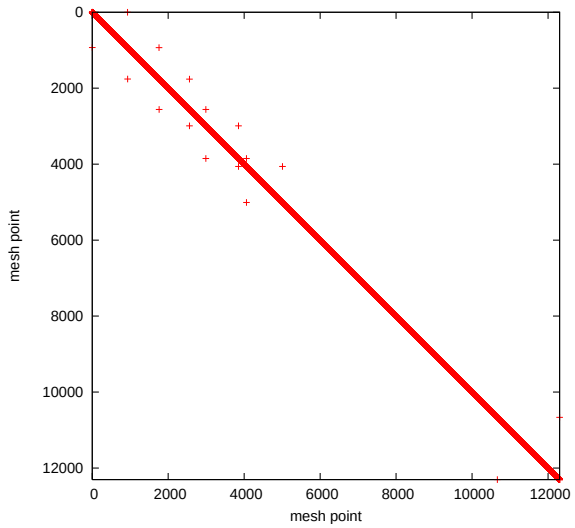

Figure 4.9: *The matrix used for solving the conduction-convection problem with a "semi-explicit" scheme.*

individual time step calculation decreases by 50 times. At the same time the procedure remains robust enough due to the fact that the dominant process of the parallel transport is calculated implicitly. It is also important to stress, that all the diagonal coefficients in all the non-parallel terms ($L_{\angle}$ operator) are taken implicitly. This is done to make the scheme stable.

With this, we complete the description of the algorithm of the finite difference conduction-convection solver. The detailed description of the "Findif" algorithm can be found in Algs. 4, 5 and 6. The "life cycle" of the frame work as a whole is illustrated in Fig. 4.10.

Algorithm 4 Main code. Return solution

Require: Config file, Mesh, Neighbours, Metric
  Read Config
  $ListOfCoefficients = FreePointMethod(Neighbours, Mesh)$ {Find the matrix of coefficients for further derivative calculation}
  $Iter \Leftarrow 0$
  Set $TimeStep$ from Config
  Init $CurrentSolution, OldSolution$ from Config
  while $Iter < MaxIter$ **do**
    if Is $TimeStep$ adaptive? **then**
      $TimeStep = RecalcuteTimeStep()$
    end if
    $Solution \Leftarrow ExecuteTimeStep()$
    $OldSolution \Leftarrow CurrentSolution$
    $CurrentSolution \Leftarrow Solution$
    $Res = GetResidual(CurrentSolution, OldSolution)$
    if Accuracy in Res achieved? **then**
      return $CurrentSolution$
    end if
  end while
  return $CurrentSolution$

Algorithm 5 $ExecuteTimeStep()$ procedure. Return solution on this timestep

Require: Config file, Mesh, Neighbours, Metric
  Init $FirstResidual$
  Init $OldIterSolution, CurrentIterSolution$
  $InnerIter \Leftarrow 0$
  while $Iter < MaxInnerIter$ **do**
    for all $Equation \in ListOfEquations$ **do**
      $Solve(Equation)$
    end for
    $OldIterSolution = CurrentIterSolution$
    $CurrentIterSolution = Solution$
    $Res = GetIterResidual(OldIterSolution, CurrentIterSolution)$
    if $Res/FirstResidual < eps$ **then**
      return $CurrentIterSolution$
    end if
  end while
  return $CurrentIterSolution$

Algorithm 6 *Solve*(*Equation*) procedure. Return solution for one equation

Require: Config file
  for all $line \in FieldLines$ **do**
    for all $p \in line$ **do**
      Execute calculations for selected configuration: 1D or 3D
      Calculate $\frac{d}{dx^3}\left(V^3 f - D^{33}\frac{df}{dx^3}\right)$ term implicitly with Upwind scheme
      Calculate all the other derivative terms explicitly
      Calculate explicit and implicit Source terms
    end for
    Apply *BoundaryConditions*, if needed
    Solve *Equation* in *line* with PROGONKA method
  end for
  return *Solution*

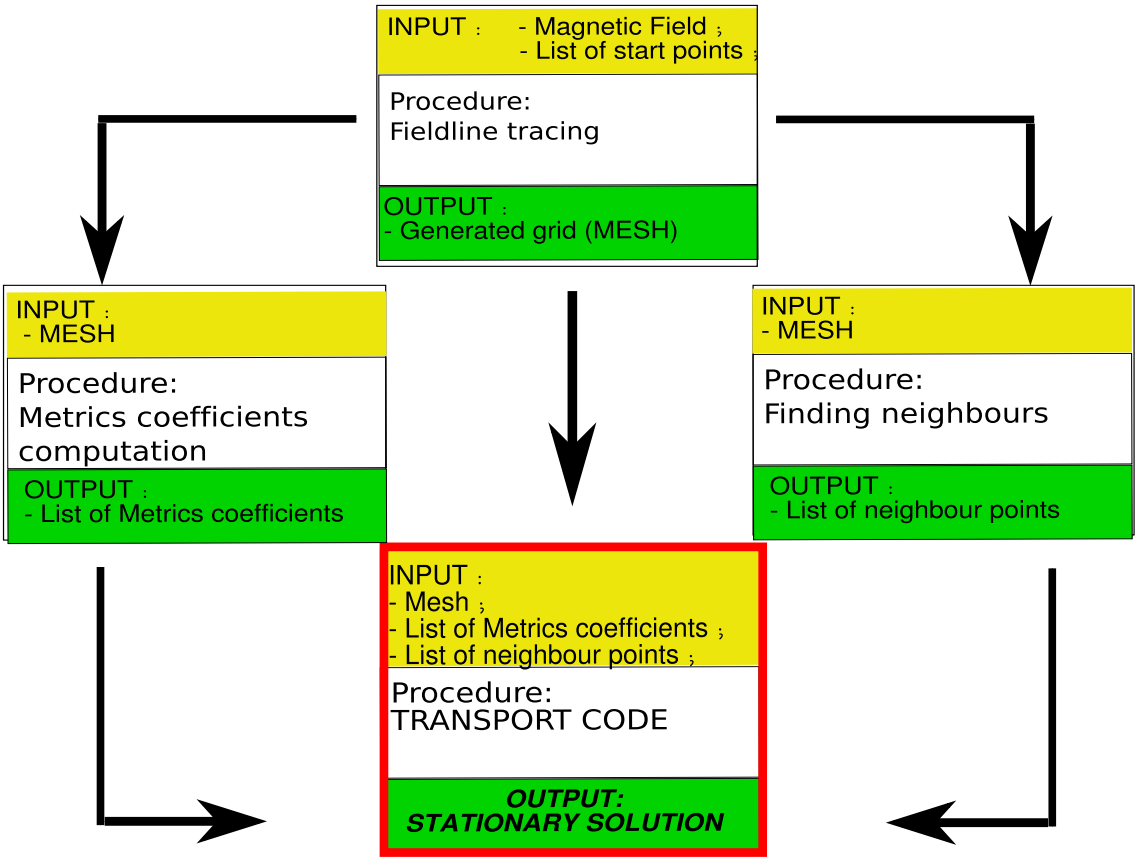

Figure 4.10: *The life cycle schematics of the FINDIF project. Describes the most important steps for obtaining a temperature, density and parallel velocity stationary solutions for some device.Consists of three preliminary steps and the main code.*

4.3 Validation of the conduction-convection solver

Every code needs validation with analytical solutions. As we have a conduction-convection problem it's worse to check separately two limiting cases: the purely conductive and the purely convective solution. For the purely conductive case, we can take the diffusion equation, e.g. for the electron temperature:

$$\frac{d}{dx}\kappa_{e||}\frac{dT_e}{dx} = 0 \tag{4.54}$$

In the equation above $\kappa_{e||}$ is the classical electron heat conduction coefficient, which depends on temperature as $\kappa \propto T_e^{\frac{5}{2}}$. With additional Dirichlet boundary conditions at the ends of the domain, the equation can be easily solved analytically. If we set $T_e(x = 0) = T_{e0}$ and $T_e(x = L) = T_{eL}$ then the solution of Eq. (4.54) is:

$$T_e = \left(\frac{x}{L}\left(T_{eL}^{\frac{7}{2}} - T_{e0}^{\frac{7}{2}}\right) + T_{e0}^{\frac{7}{2}}\right)^{\frac{2}{7}} \tag{4.55}$$

The result of the analytical and numerical solutions gives the same values up to the numerical noise in this case.

Another kind of limiting case is the purely convective case. For this kind of validity test we have used the simplest system of convective fluid equations, that can be solved analytically: the stationary coupled one for n, V in 1D. To obtain the solution we omit the viscosity term in the velocity equation and take the source for density as constant (although it is possible to obtain solutions without this constraint). As boundary conditions we suppose the velocity values given on both ends. We suppose the absolute values of these velocities to be equal, whereas the sign is chosen opposite. So, the system of equations we solve is:

$$\begin{cases} \dfrac{d\left(nV\right)}{dx} = S \\ \dfrac{d\left(nmV^2 + nT\right)}{dx} = 0 \end{cases} \tag{4.56}$$

After integration we have:

$$\begin{cases} nV = Sx + C_1 \\ nmV^2 + nT = C_2 \end{cases} \tag{4.57}$$

where C_1 and C_2 are constants determined by the boundary conditions. We express n from the first equation of Eqs. (4.57) and substitute it to the second one, the resulting quadratic equation with respect to V is:

$$V^2 m\left(Sx + C_1\right) - VC_2 + T\left(Sx + C_1\right) = 0. \tag{4.58}$$

The solution of Eq. (4.58) is:

$$V_{1,2} = \frac{C_2 \pm \sqrt{C_2^2 - 4mT\left(Sx + C_1\right)^2}}{2m\left(Sx + C_1\right)}. \tag{4.59}$$

Let $V = V_0$ for $x = 0$ and $V = -V_0$ for $x = L$. Using Eqs. (4.57) we obtain the expressions for coefficients C_1 and C_2:

$$C_1 = -\frac{SL}{2}, \tag{4.60}$$

$$C_2 = -\frac{SL\left(V_0^2 m + T\right)}{2V_0}. \tag{4.61}$$

Substitute the expressions (4.60), (4.61) to (4.59)

$$V_{+,-} = \frac{-\frac{SL}{2}\frac{V_0^2 m + T}{V_0} \pm \sqrt{\frac{(SL)^2}{4}\frac{\left(V_0^2 m + T\right)^2}{V_0^2} - 4mT\left(Sx - \frac{SL}{2}\right)^2}}{2m\left(Sx - \frac{SL}{2}\right)}, \tag{4.62}$$

This solution is defined for $\forall x \in [0, L]$ with $x \neq \frac{L}{2}$ with $V_0 \neq 0$. After applying the boundary condition $x = 0$ to Eq. (4.62) we found the following constraints on the solution V:

$$V = \begin{cases} V_+ & \text{if } mV_0^2 \geq T, \\ V_- & \text{if } mV_0^2 \leq T \end{cases} \tag{4.63}$$

For x, that lies in the neighbourhood of $\frac{L}{2}$, we obtain the following solution, getting rid of the indeterminate form $\frac{0}{0}$:

$$V = \frac{2TV_0}{SL\left(V_0^2 m + T\right)}\left(\frac{SL}{2} - Sx\right). \tag{4.64}$$

Figures 4.11 and 4.12 show the comparison between the analytical and numerical solutions, which agree quite well with some small differences. In Fig. 4.12 one can see a small difference between the two solutions near the walls (the first and the last point) caused by the very steep gradients. This relatively small difference in the velocity values leads to the visible discrepancy in the density profile, as one can see in Fig. 4.11, which disappears if one uses the analytical velocity profile.

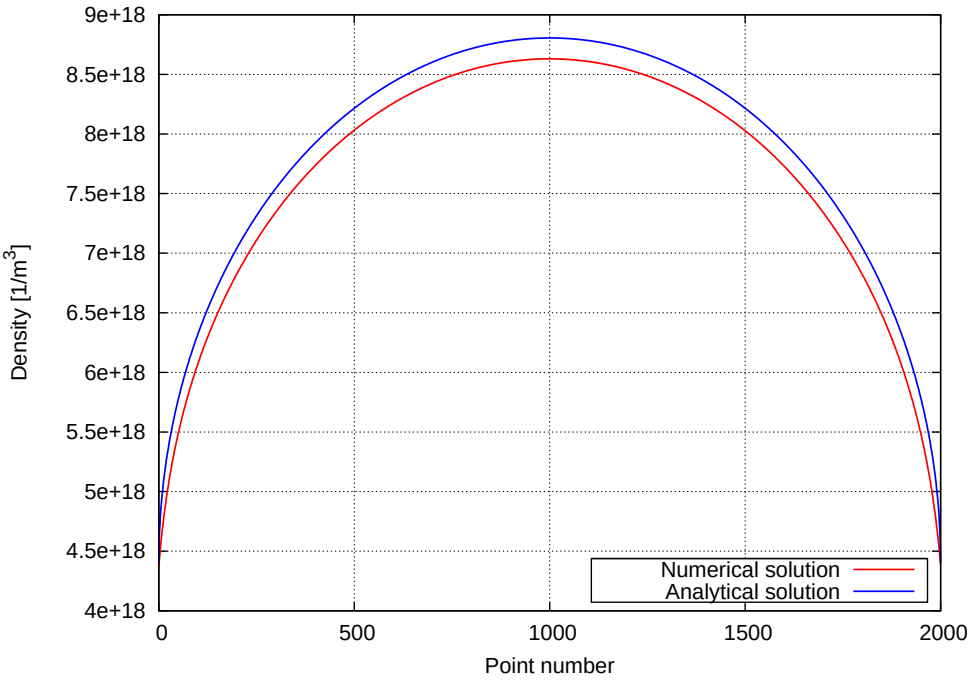

Figure 4.11: *Comparison of the density obtained from the analytical and numerical solution.*

The benchmark with a 1D code solving coupled pair of the density and velocity equations was done. The code named "FV1D" was provided by A. Runov. FV1D is a fully explicit code using

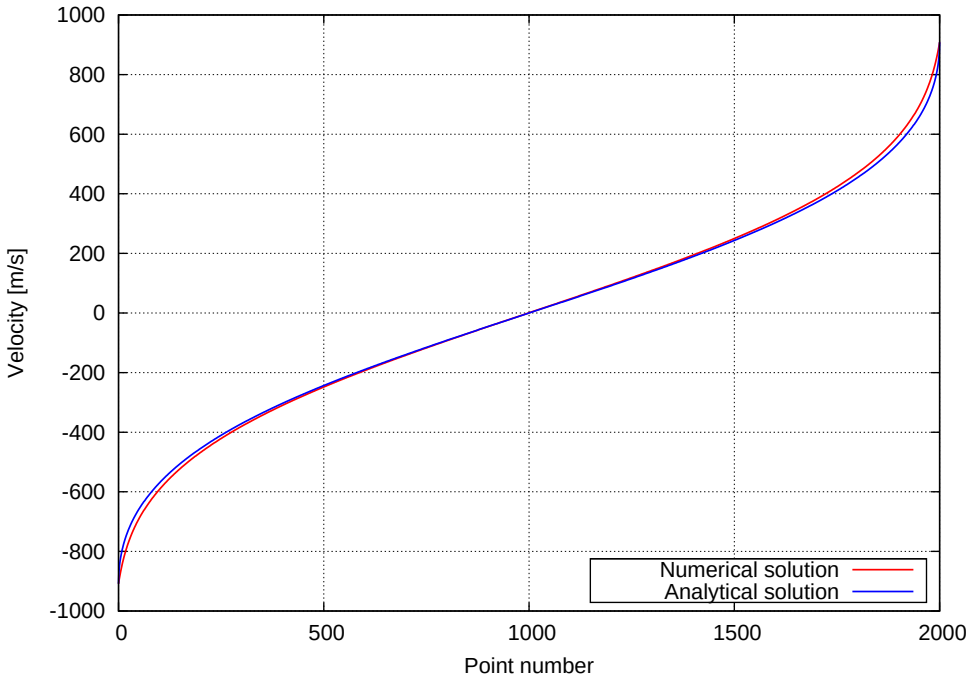

Figure 4.12: *Comparison of the velocity obtained from the analytical and numerical solution.*

a staggered grid (i.e. velocities are set on the cell interfaces). A fourth order Runge-Kutta ODE integrator is used for solving the time evolution. For the spatial part an upwind scheme is used. The benchmark was done for the case with the source of the plasma placed on the left side and target plate placed on the right. Four quantities, namely electron temperature, ion temperature, density and velocity were calculated in the Findif code. After obtaining temperatures from the Findif code (shown in Fig. 4.13) the values were used as a background temperature in the FV1D code. The results of comparisons are shown in Fig. 4.14 for density and Fig. 4.15 for velocity and show good agreement.

In this chapter, the code algorithm including pre-processing steps was discussed and the successful benchmark with analytical and numerical models was presented. Now, the tool is ready to be used for physics studies.

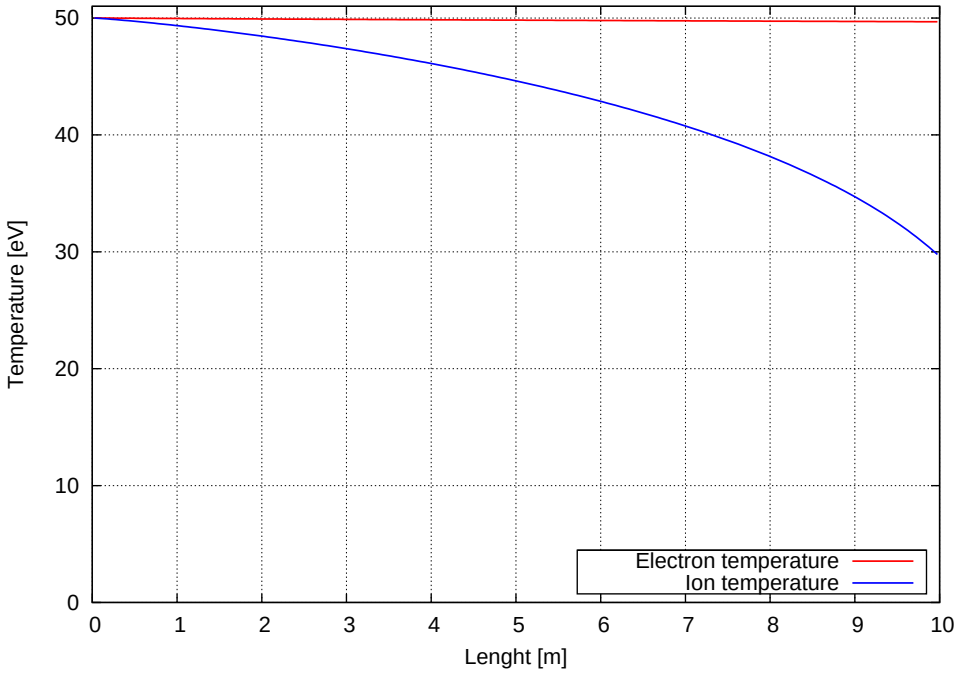

Figure 4.13: *Temperature profiles used for the benchmark with a 1D code.*

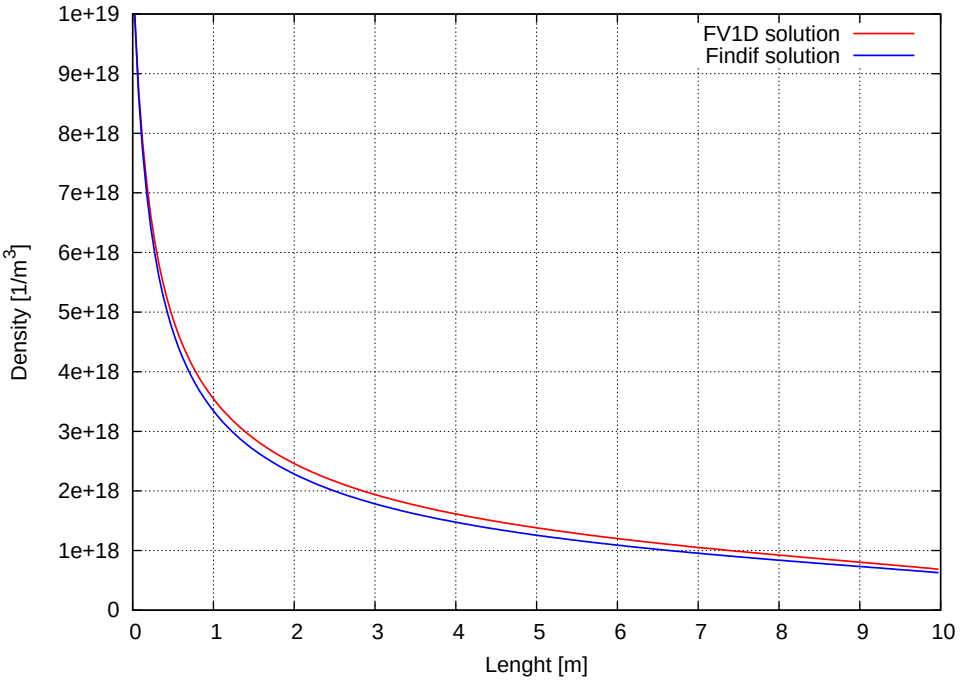

Figure 4.14: *Benchmark with a 1D code. Density.*

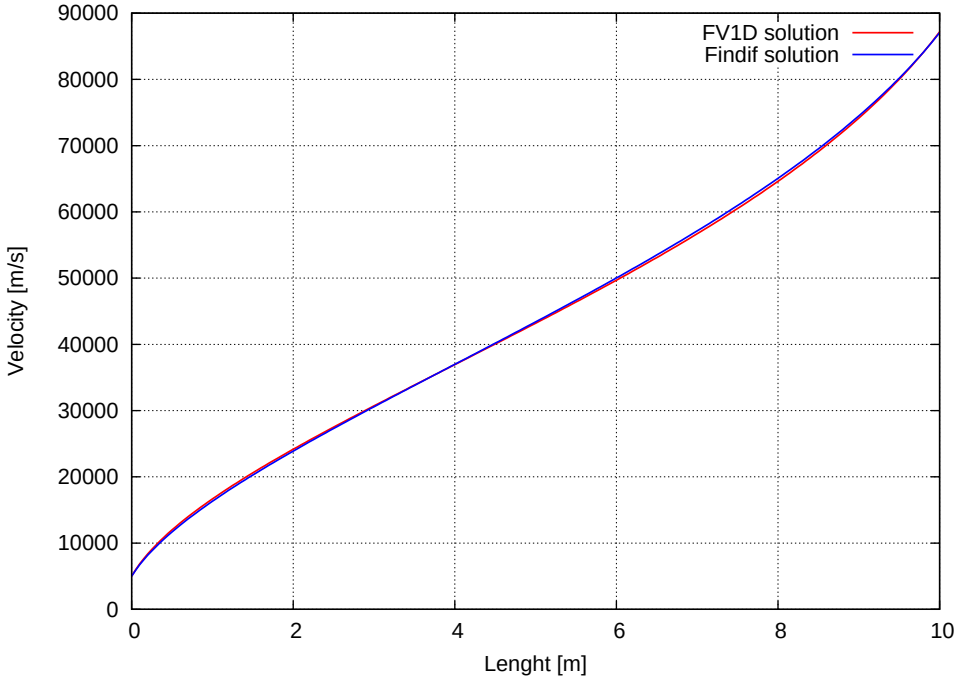

Figure 4.15: *Benchmark with a 1D code. Velocity.*

Chapter 5

Results

The aim of this work is to investigate the influence of the ergodicity and 3D configuration of the field on the heat transport. The study of conductive heat transport is the first step towards the analysis of the heat flux pattern, because for temperatures above about 50 eV the parallel heat conduction is usually dominant and convection terms or neutral sources can be neglected (at least for electrons). We present calculations for the W7-X stellarator and TEXTOR-DED.

5.1 Heat conduction

The electron and ion temperature equations include the coupling term, that describes energy exchange between electrons and ions.

$$\begin{aligned}
&\frac{3}{2}n_i\frac{\partial T_e}{\partial t}- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\kappa_e^A g^{33}+(\kappa_e^0-\kappa_e^A)(h^3)^2\right]\frac{\partial T_e}{\partial x^3}- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\kappa_e^A\left(g^{31}\frac{\partial T_e}{\partial x^1}+g^{32}\frac{\partial T_e}{\partial x^2}\right)\right]- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left[\kappa_e^A\left(g^{11}\frac{\partial T_e}{\partial x^1}+g^{12}\frac{\partial T_e}{\partial x^2}+g^{13}\frac{\partial T_e}{\partial x^3}\right)\right]- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left[\kappa_e^A\left(g^{21}\frac{\partial T_e}{\partial x^1}+g^{22}\frac{\partial T_e}{\partial x^2}+g^{23}\frac{\partial T_e}{\partial x^3}\right)\right] \\
&=\frac{3}{2}T_e\frac{\partial n_i}{\partial t}+k(T_i-T_e),
\end{aligned} \tag{5.1}$$

for electron temperature, and

$$\begin{aligned}
&\frac{3}{2}n_i\frac{\partial T_i}{\partial t}- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\kappa_i^A g^{33}+(\kappa_i^0-\kappa_i^A)(h^3)^2\right]\frac{\partial T_i}{\partial x^3}- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\kappa_i^A\left(g^{31}\frac{\partial T_i}{\partial x^1}+g^{32}\frac{\partial T_i}{\partial x^2}\right)\right]- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left[\kappa_i^A\left(g^{11}\frac{\partial T_i}{\partial x^1}+g^{12}\frac{\partial T_i}{\partial x^2}+g^{13}\frac{\partial T_i}{\partial x^3}\right)\right]- \\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left[\kappa_i^A\left(g^{21}\frac{\partial T_i}{\partial x^1}+g^{22}\frac{\partial T_i}{\partial x^2}+g^{23}\frac{\partial T_i}{\partial x^3}\right)\right] \\
&=\frac{3}{2}T_i\frac{\partial n_i}{\partial t}-k(T_i-T_e),
\end{aligned} \tag{5.2}$$

for ion temperature. Using the concept of a generalized conduction-convection equation, which was described in Chap. 4 (see Eqs. (4.4) and (4.5)) with zero velocities, we can rewrite both

equation:

$$A_t \frac{\partial T_\alpha}{\partial t} - \frac{1}{\sqrt{g}} \frac{\partial}{\partial x^i} \sqrt{g} \left(D_\alpha^{ij} \frac{\partial T_\alpha}{\partial x^j} \right) + \nu_\alpha T_\alpha = Q_\alpha \tag{5.3}$$

Identifying the terms we have:

$$\begin{aligned} D_\alpha^{ij} &= D_{\alpha\perp} g^{ij} + (D_{\alpha||} - D_{\alpha\perp}) h^i h^j \\ D_{\alpha\perp} &= \chi_{\alpha\perp} n \\ D_{\alpha||} &= \chi_{\alpha||} n \\ \nu_\alpha &= \frac{3 m_e n}{m_i} \frac{1}{\tau_{ei}} \\ Q^i &= \frac{3 m_e n}{m_i} \frac{1}{\tau_{ei}} T_e \\ Q^e &= \frac{3 m_e n}{m_i} \frac{1}{\tau_{ei}} T_i \end{aligned} \tag{5.4}$$

Here, the index α refers to the appropriate species either electrons or ions, $h^i = |\nabla x^i|$, n is the plasma density, m_e, m_i are the electron and ion masses, $\chi_{\alpha\perp}, \chi_{\alpha||}$ are the perpendicular and parallel thermal conductivities, respectively and τ_{ie} is the electron-ion collision time. We have also the contribution from the heat loss rate ν and energy sources Q_α from the coupling term.

5.1.1 W7-X geometry

For W7-X we analyze first a limiter configuration for an equilibrium configuration without ergodic effects and without target plates: here, the limiting structure is assumed to follow exactly the shape of an outer flux surface. Calculations were done for the electron temperature only. We use Dirichlet boundary conditions on the innermost and the outermost flux surfaces. On the innermost the fixed temperature value is taken $300eV$ and on the outermost $4.6eV$. The anomalous perpendicular diffusion coefficient is taken as $1m^2/s$. The density was taken constant on the whole domain and equal $10^{19}\ 1/m^3$.
This case is used to benchmark three codes with very different numerical approaches for the problem of anisotropic electron heat conduction in a 3D geometry including islands. Fig. 5.1 shows the resulting electron temperature distribution in the W7-X bean-like toroidal cut. The results for the electron temperature profile along the line shown in Fig. 5.1 with black color (a black line cut the profile on the level $z = 0$) are compared for BoRiS, E3D and Findif in Fig. 5.2.
A quite good agreement between these different numerical approaches is seen, especially because all of the codes have to interpolate quite different to obtain the 1-D cut results. As discussed in chapter 2 BoRiS is a finite volume code in global flux coordinates, E3D and Findif make use of LCMS, for which E3D is solving the transport equations with Monte Carlo methods, whereas Findif is using finite differences. All codes show the strong modification of the radial profiles due to the islands, where the existence of closed field lines flattens the radial profile.

W7-X is clearly not operated in a limiter mode, but has 3D divertor structures (as seen in Figs. 5.3 and 5.4).

These target plates cut through the islands and lead to large losses to the target plates resulting in much lower temperatures at the target plates and a different solution than for the limiter case (see Figs. 5.5 and 5.6 and compare with Fig. 5.1). For the discussed case we solve the set of equations Eqs. (5.1), (5.2) with a Dirichlet boundary conditions on the inner and outer boundaries and sheath conditions on the target plates. The boundary values for

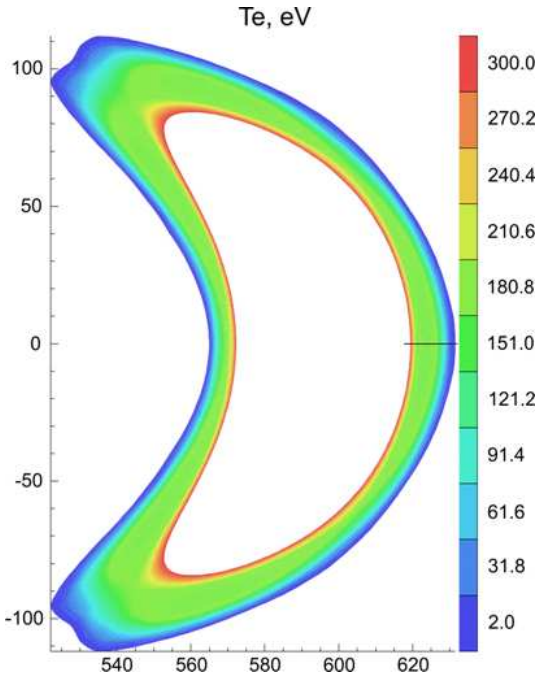

Figure 5.1: *Solution of the pure conduction problem for electron temperature. W7-X bean-like shape.*

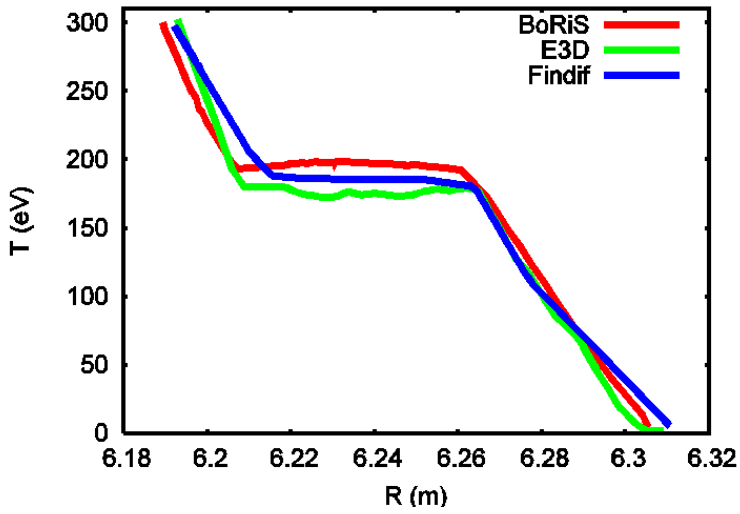

Figure 5.2: *Benchmark of three codes. Electron temperature along the line $z = 0$ on the W7-X bean-like cut is shown.*

both temperatures are taken equal: $150eV$ for inner and $5eV$ for outer boundaries respectively. Anomalous diffusion coefficient is taken as 2 m^2/s.

The heat flux pattern on the target plates reflects for electrons the spatial distribution of the connection length of the field lines hitting the plate (see Figs. 5.7,5.8, 5.9). This is also visible on the plot of the electron heat flux density at the target plate as a function of field line length (Fig. 5.10). The maximum values are clearly correlated with the longest field line lengths. Due to the radial diffusion, these large heat fluxes also fill up neighboured field lines with shorter lengths, therefore one gets a rather full triangle in the plot. In contrast to the electrons, ions have a rather hot-spot-like pattern with localized zones of high ion temperatures. These spots are

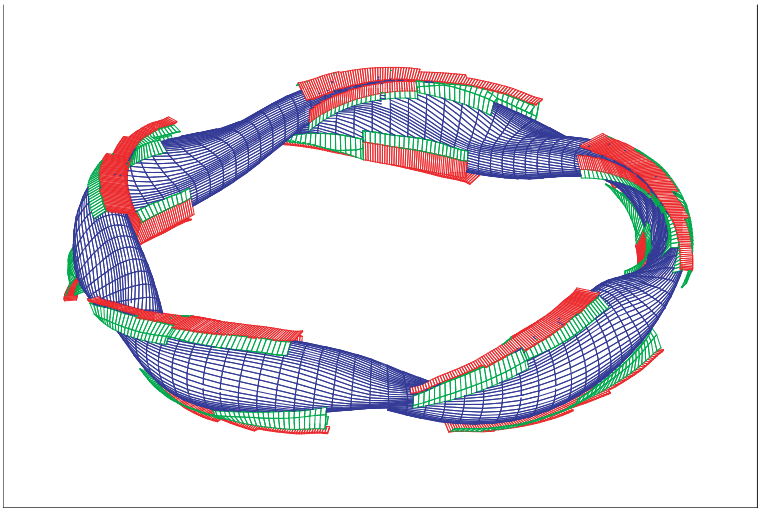

Figure 5.3: *Divertor structure on the W7-X geometry.*

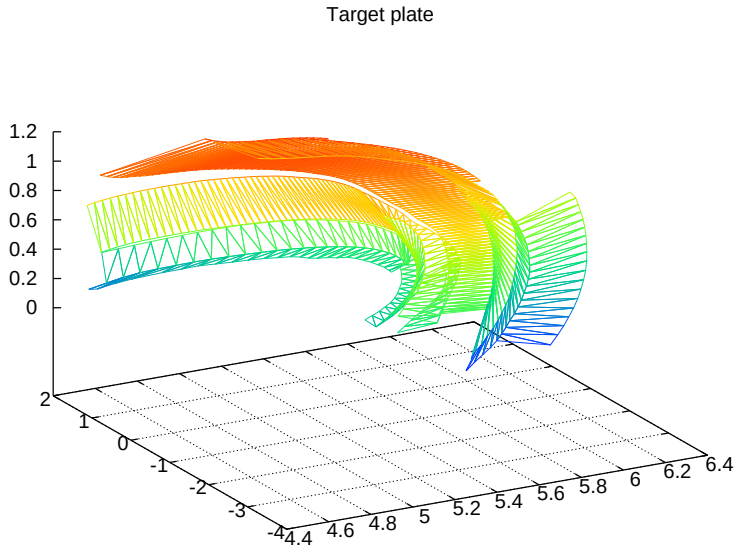

Figure 5.4: *One part of the divertor structure (W7-X), used for the 3D view of the solution later on.*

created by the coupling terms of electrons to ions, which are heating the ions at those locations where the density and electron temperature are simultaneously relatively high. Therefore, this is a convolution of the electron temperature distribution with the (background) electron density. In this run, we used already for the ion density the results of the following chapter, where in addition to the energy equations also the parallel momentum and continuity equation is solved. This density distribution has also large differences in its spatial distribution and creates then this unusual pattern together with the connection lengths. In the distribution of the ion heat fluxes, the localized patterns are already much more broadened, because the maximum heat fluxes do not have to be connected with the largest ion temperatures, because the boundary heat fluxes are also determined by the particle fluxes (the kinetic boundary condition for a sheath discussed in chapter 2 see Eq. (2.5)). These particle fluxes are again used from the more

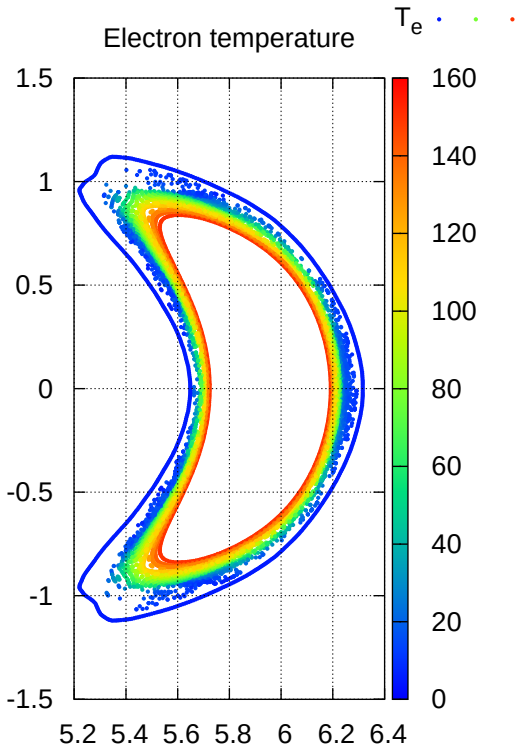

Figure 5.5: *Electron temperature on the "bean-like" toroidal cut.*

complex simulation in the next chapter, but only for the boundary conditions, whereas in the rest of the domain all convective terms are set to zero. This procedure allows (see next section) a clear separation of conductive and convective effects for the heat flux pattern in this W7-X case. The ion heat fluxes as a function of connection length show due to these effects discussed before a more sparse pattern and are less filled up to the maxima compared to the electrons. If one compares the radial profiles of electrons and ions (as was done before for the limiter case), we recognize clearly the regions where within the islands the intersection with the target plates act as strong sinks and reduce strongly the temperatures there. Also, the radial decay of the electron profile is faster than the ion profile, which is expected, because the parallel heat conduction of electrons is much larger than of ions (due to the mass difference). Therefore, the parallel loss is stronger for electrons and their profile decays faster.

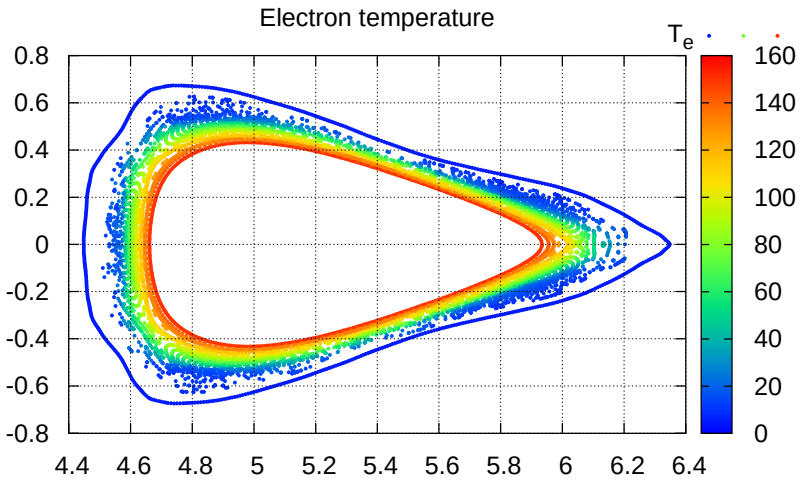

Figure 5.6: *Electron temperature on the "triangle-like" toroidal cut.*

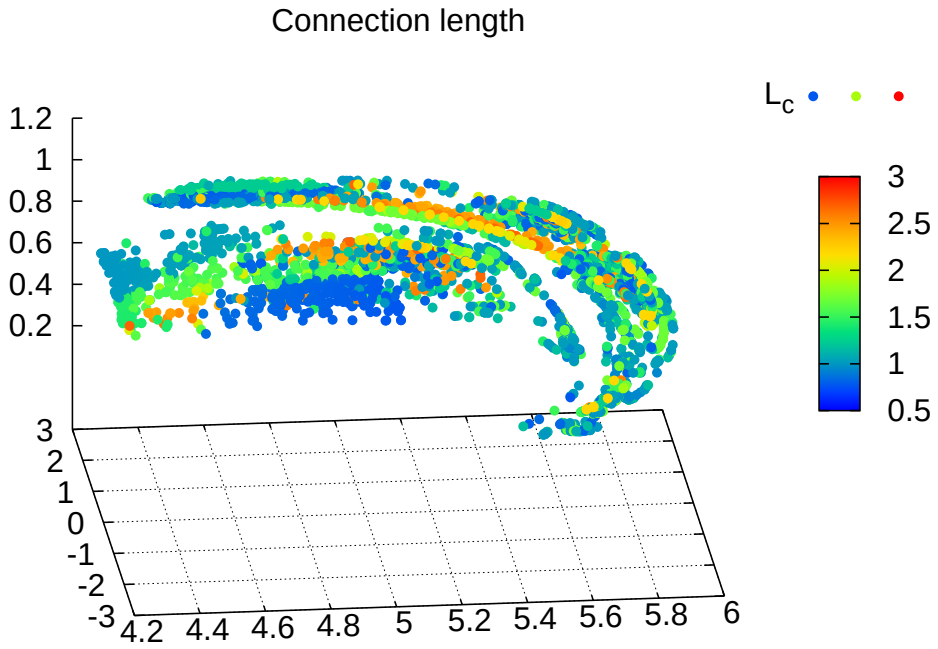

Figure 5.7: *Connection length on the target plates.*

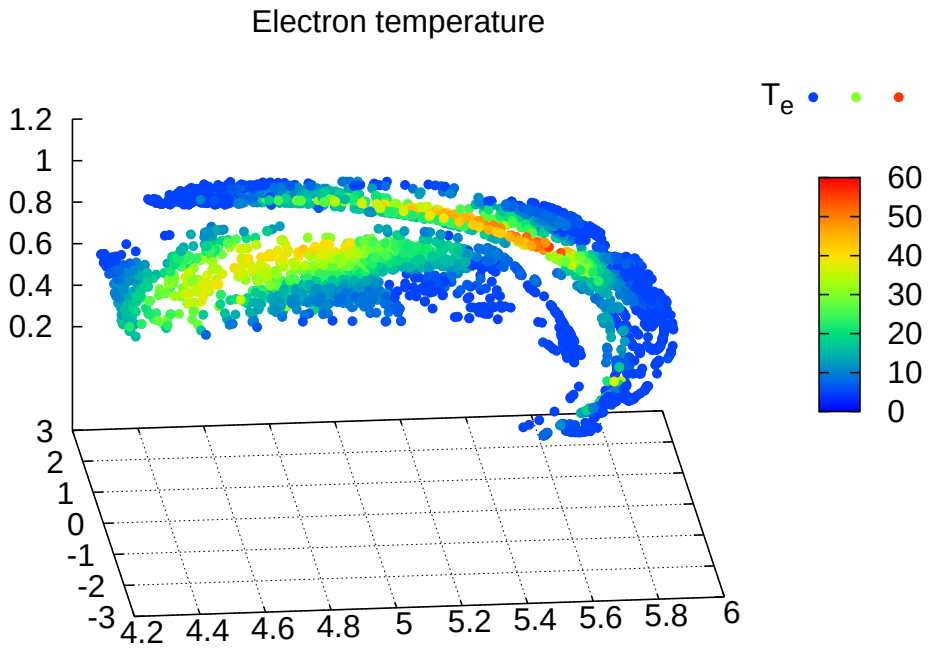

Figure 5.8: *Electron temperature on the target plate.*

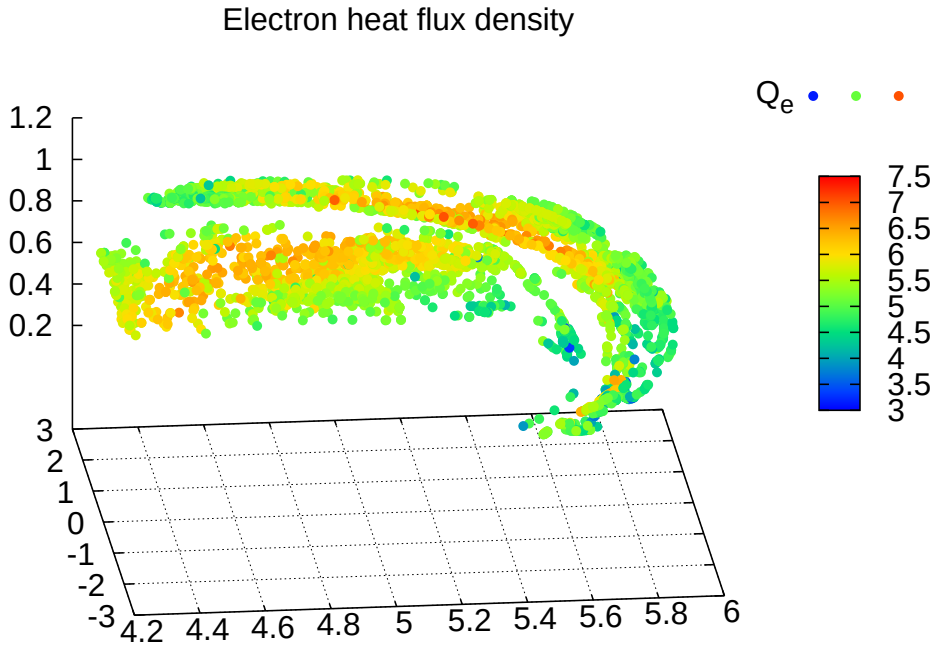

Figure 5.9: *Electron heat flux on the target plate.*

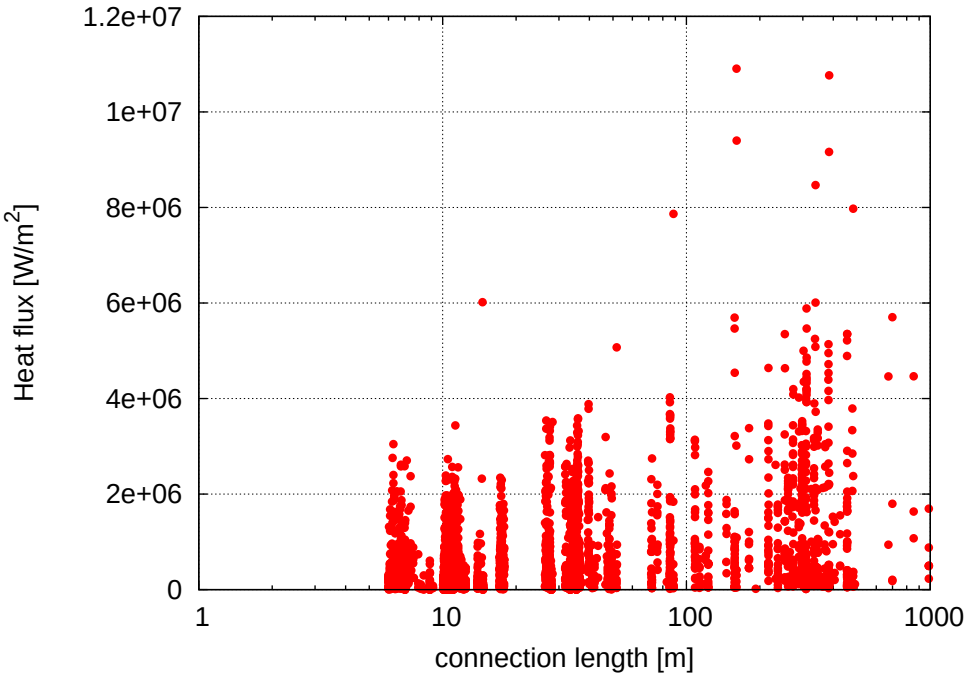

Figure 5.10: *Electron heat flux on the target plate with respect to the connection length.*

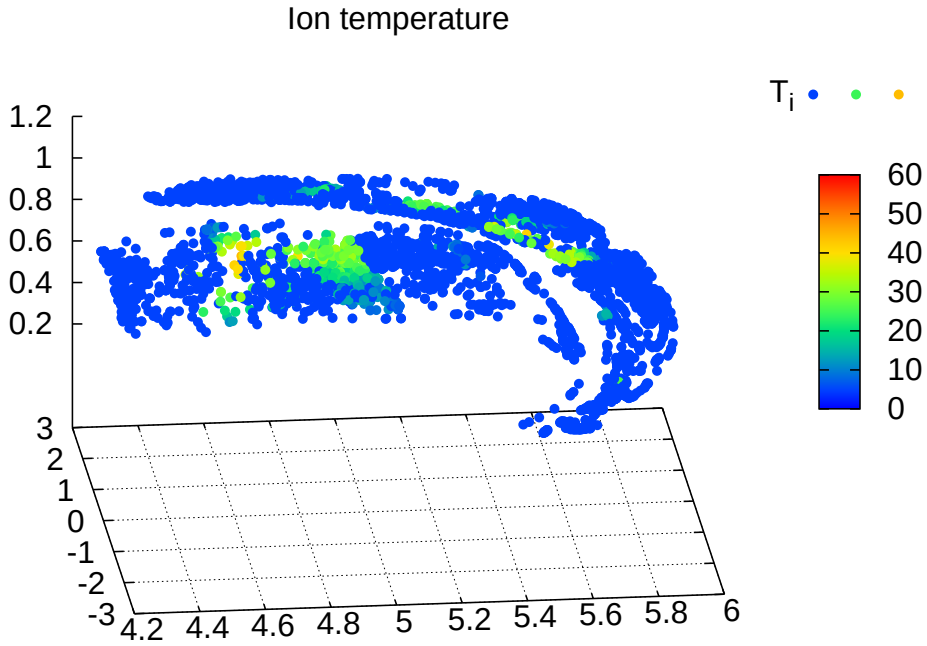

Figure 5.11: *Ion temperature on the target plate.*

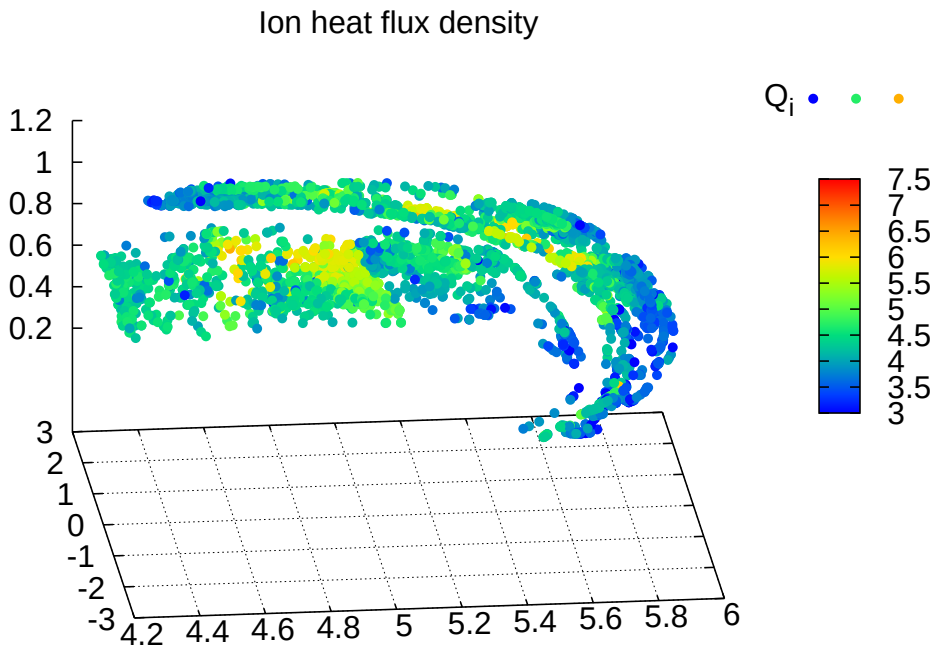

Figure 5.12: *Ion heat flux on the target plate.*

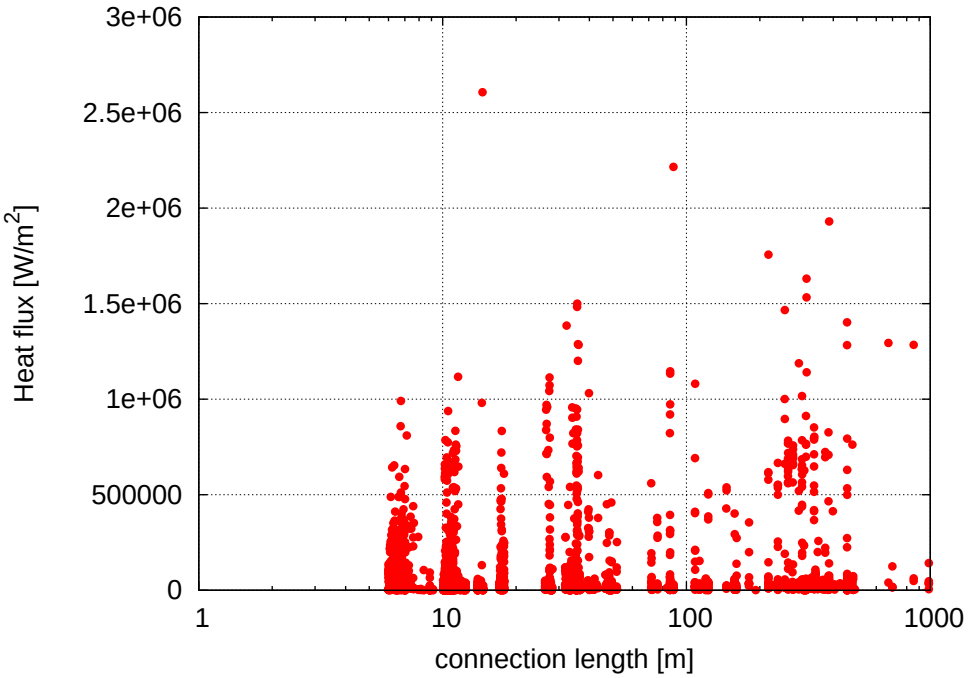

Figure 5.13: *Ion heat flux on the target plate with respect to the connection length.*

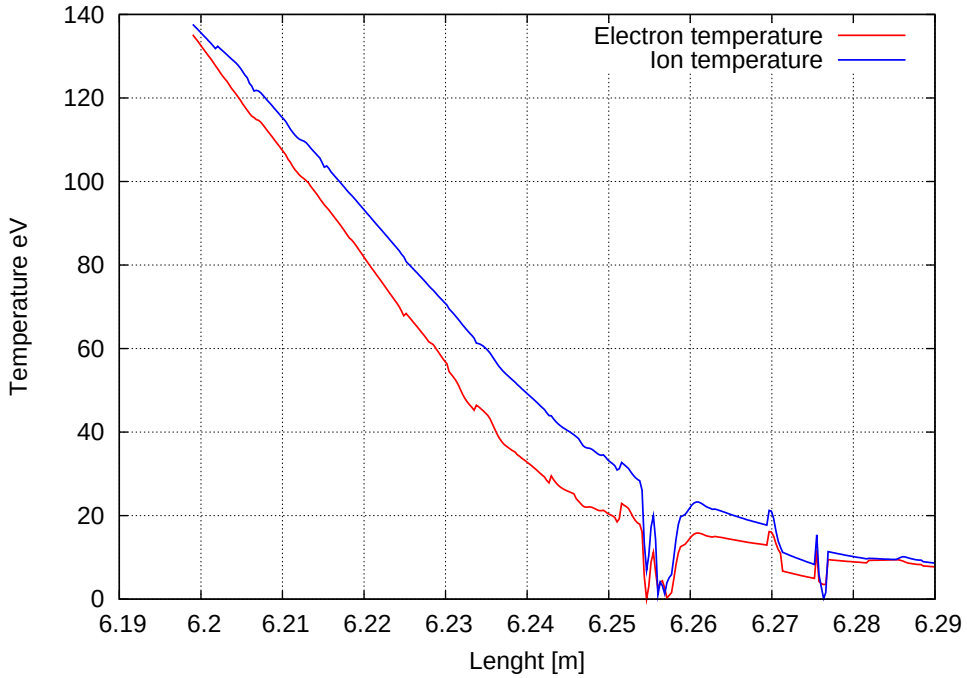

Figure 5.14: *Electron and ion temperature radial profile.*

5.1.2 TEXTOR-DED geometry

TEXTOR-DED

The Dynamic Ergodic Divertor in TEXTOR [62] is an excellent tool to study the basic properties of ergodic transport. The main component of the DED is a set of magnetic perturbation coils, which imposes the magnetic perturbation in the plasma edge region. The set consists of 16 individual coils (four quadruples) plus two compensation coils located at the high-field side. The base mode of the magnetic perturbation can be varied by the connections of the power supplies to the coils between $(m, n) = (12, 4)$, $(6, 2)$ and $(3, 1)$, where (m, n) refers to the poloidal and toroidal mode number. The spectrum of the magnetic perturbation of $(12, 4)$ and $(6, 2)$ base mode is optimized to influence rational flux surfaces in the plasma edge, whereas the magnetic perturbation in the $(3, 1)$ configuration reaches the plasma core and investigations here focus primarily on studies of the physics of the tearing modes [63, 64]. This work is constrained to the $(12, 4)$ and $(6, 2)$ base modes. The side bands of magnetic perturbation create island chains, which at certain level of the DED current overlap creating perturbed volume.

For the proper analysis of such a complicated topology, one inevitably needs 3D transport codes, which could describe such a variety of magnetic field lines. Here, the influence of the field structure on the heat transport patterns for different operation modes $(12, 4)$ and $(6, 2)$ will be discussed.

Construction of the grid

In order to solve the plasma transport in the 3D boundary of TEXTOR-DED a numerical mesh has to be generated as described in detail in Chap. 4.

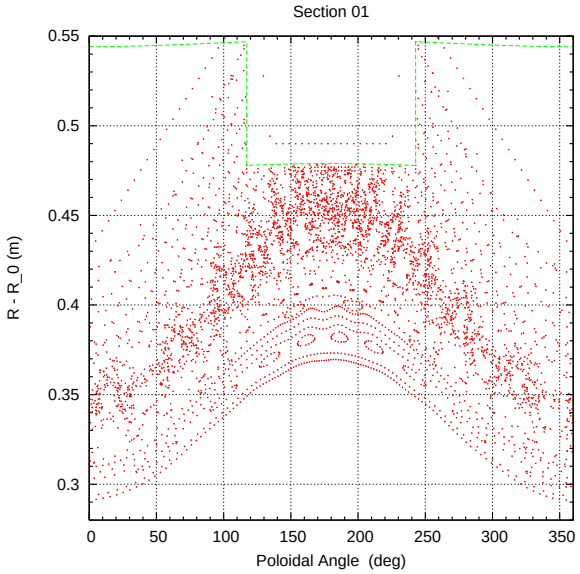

Figure 5.15: TEXTOR-DED mesh (Poincaré section) for the toroidal angle $\Phi = 0^o$ (mode 12/4). Here the poloidal radius R is shifted by R_0 (=1.752 m)

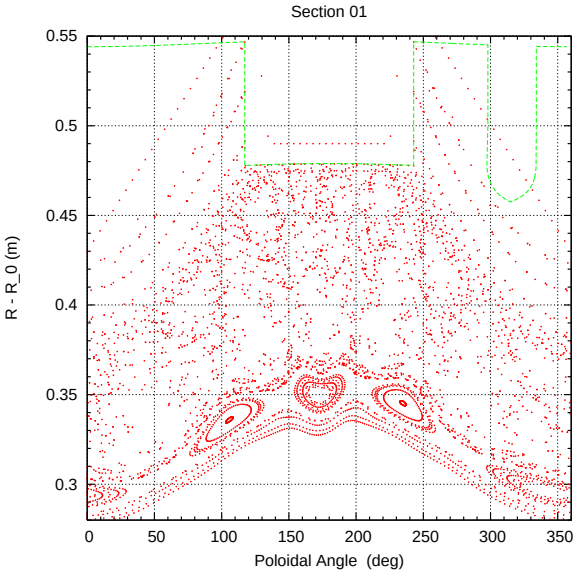

Figure 5.16: TEXTOR-DED mesh (Poincaré section) for the toroidal angle $\Phi = 0^o$ (mode 6/2). Here the poloidal radius R is shifted by R_0 (=1.752 m)

In the present work we constructed meshes for two modes 12/4 and 6/2. For the 12/4 mode the current in the DED coils $I_{DED} = 11$ kA and the toroidal magnetic field $B_T = 1.9$ T and for the 6/2 mode $I_{DED} = 7.5$ kA and $B_T = 2$ T. We start the line at one cut (say $\phi = 0^o$) and we trace the field line around the torus. We generate a mesh point whenever a field line intersects

a cut. By following many field lines around the torus, for many toroidal turns ($\sim$ 100), we generate many points on each cut.

In our finite difference code we make use of an optimized mesh. It consists of two classes of magnetic field lines:

- "*closed*" field lines, where the distance between starting and ending points is smaller than ion gyroradius (2 mm in this case).
- "*open*" field lines, which appear in the outermost plasma edge, they start and end on the DED surface or on the wall structures. These field lines are much shorter than the "closed" ones. The requirement for them is such that they intersect at least three Poincaré sections.

A full 3D mesh consists of a series of Poincaré plots spaced at regular toroidal intervals. In Figs. 5.15 and 5.16 we show the sections at toroidal angle $\Phi = 0^o$ for the mode 12/4 and 6/2 of the TEXTOR-DED tokamak. The mesh for the 12/4 mode is constructed from 629 lines and contains 108776 points whereas 842 lines is used for the 6/2 mode mesh and it contains 119047 points. It should be noted that in the grid we have also points which do not belong to the considered 24 cuts. The reason is that the open lines which are followed from the points on the cuts does not necessarily ends at the Poincaré plots and thus they can hit the wall structures in the areas between the cuts.

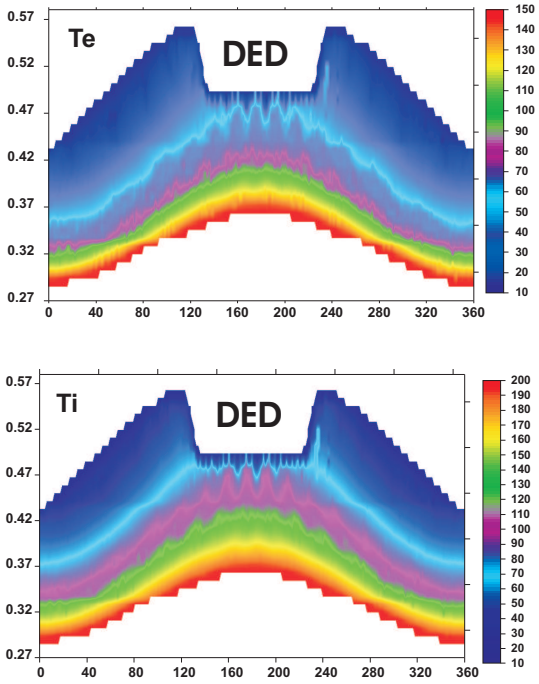

Figure 5.17: Distribution of plasma temperatures T_e and T_i (in eV) at the Poincarè sections ($\phi = 0$) for $n_e = 10^{18}\ m^3$ (mode 12/4). The vertical axis shows the plasma radius in respect to the magnetic (in m) and the poloidal angle is shown on the horizontal axis.

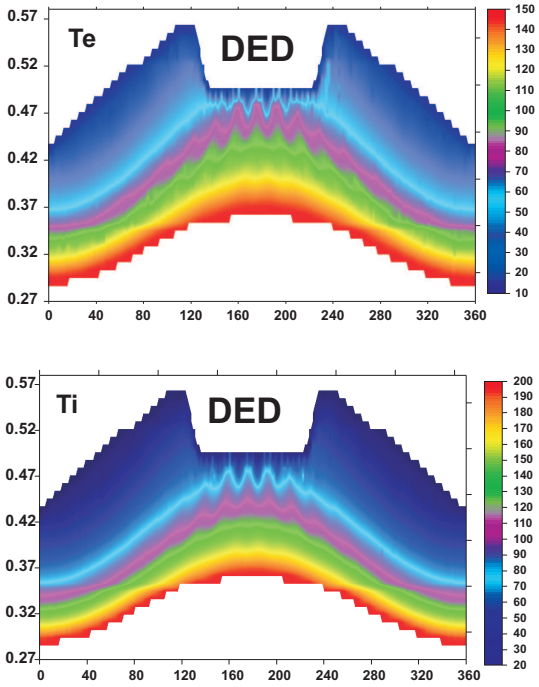

Figure 5.18: Distribution of plasma temperatures T_e and T_i (in eV) at the Poincarè sections ($\phi = 0$) for $n_e = 5 \times 10^{19}\ m^3$ (mode 12/4). The vertical axis shows the plasma radius in respect to the magnetic (in m) and the poloidal angle is shown on the horizontal axis.

Results

The finite difference code FINDIF has been used to investigate the energy transport in the complex 3D TEXTOR-DED tokamak geometry, where the plasma transport is closely related

to the structure of the magnetic field lines.

Structures of different origin can be recognized in the meshes in Figs. 5.15 and 5.16. In the

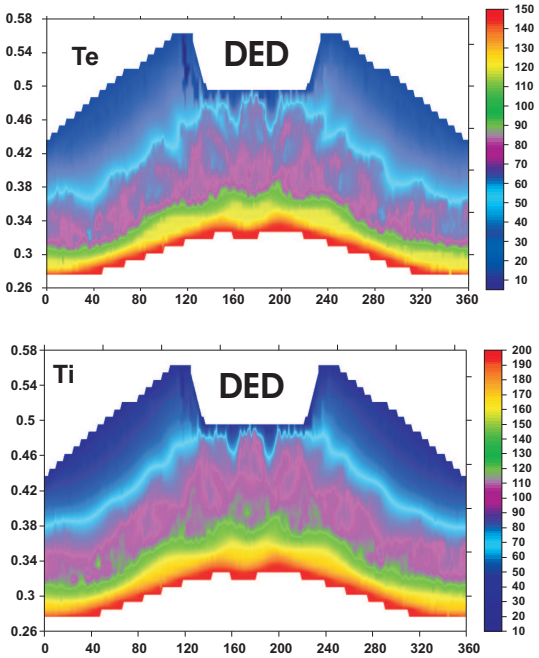

Figure 5.19: Distribution of plasma temperatures T_e and T_i (in eV) at the Poincarè sections ($\phi = 0$) for $n_e = 10^{18}\ m^3$ (mode 6/2). The vertical axis shows the plasma radius in respect to the magnetic (in m) and the poloidal angle is shown on the horizontal axis.

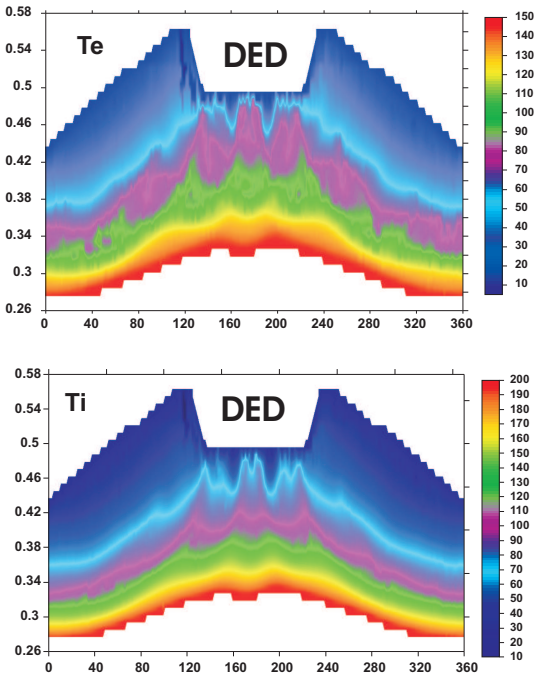

Figure 5.20: Distribution of plasma temperatures T_e and T_i (in eV) at the Poincarè sections ($\phi = 0$) for $n_e = 5 \times 10^{19}\ m^3$ (mode 6/2). The vertical axis shows the plasma radius in respect to the magnetic (in m) and the poloidal angle is shown on the horizontal axis.

region close to the core plasma, the close field lines form flux surfaces as well as island chains. Next, we have regions with closed ergodic lines which are followed by wide area of open field lines. We have solved the transport equations for T_e and T_i for different values of the plasma density ($n_e = 0.1, 1, 5 \times 10^{19}$ m^{-3}) in order to assess the effect of n_e on the energy transport. For the radial transport we have assumed constant conductivity coefficients: $\chi_{e\perp} = \chi_{i\perp} = 5$ m^2/sec being of the order of Bohm diffusion. Similar values have been used in the 2D simulation of TEXTOR edge plasmas with the TECXY code [65]. Constant plasma temperatures have been specified ($T_e^{core} = 150\,eV$, $T_i^{core} = 200\,eV$) at the core boundary for all considered plasma densities.

At the divertor target plates, sheath conditions [66] have been considered, whereas the constant temperatures at the wall structures have been assumed: ($T_e^{wall} = 15\,eV$, $T_i^{wall} = 20\,eV$).

In Figs. 5.17-5.20 the spatial distribution of the electron and ion temperatures at one of the Poincaré sections ($\phi = 0$) for different densities ($n_e = 10^{18}$ m^{-3}, 5×10^{19} m^{-3}) and for both modes are shown. In all four graphs the abscissa represents the poloidal angle (in degrees) and the ordinate vessel minor radius (in meters). Electron and ion temperatures are expressed in eV and energy fluxes in $W/m2$ (see colorbars on the right hand side of the graphs). It is apparent, that the magnetic topology is closest reflected in the solution of the electron temperature field at $n_e = 10^{18}$ m^{-3}, which indicates that the parallel transport is dominant. If the density increases, in particular for T_i, the heat conductivity is reduced forcing development of temperature gradients along field lines and in addition, the role of the radial transport increases leading to the stronger mixing of temperatures from different regions, and consequently the mesh structures are less pronounced. However, the hot spot structure of the temperature distribution close to the target

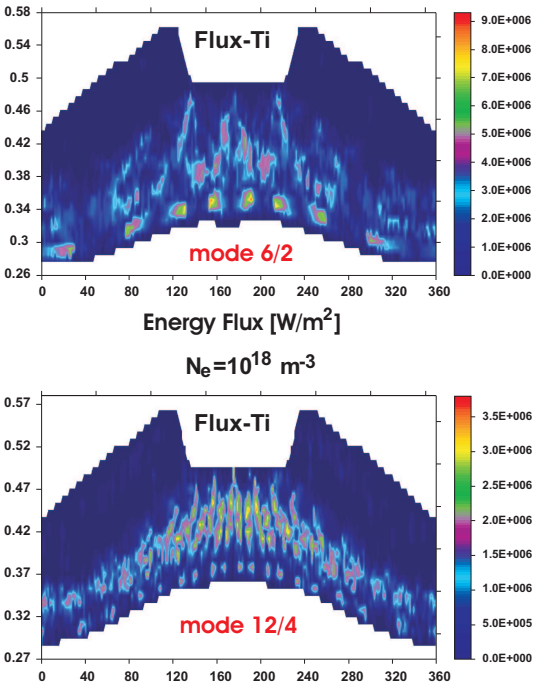

Figure 5.21: Ion energy fluxes for $n_e = 10^{18}$ m^{-3} and both modes.

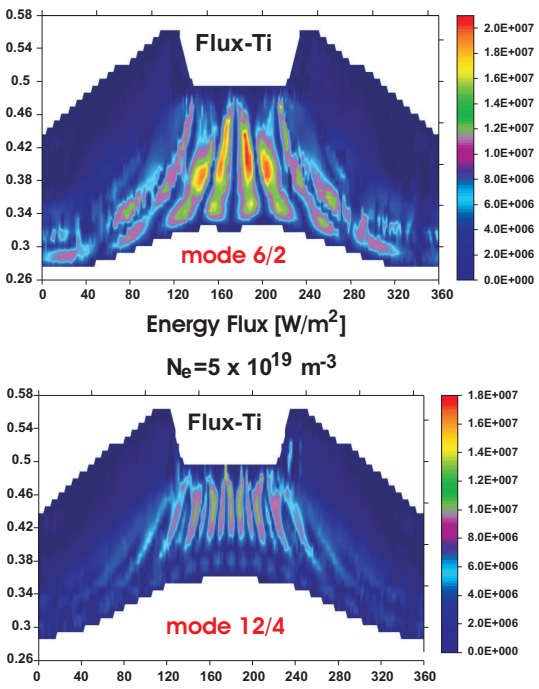

Figure 5.22: Ion energy fluxes for $n_e = 5 \times 10^{19}$ m^{-3} nd both modes.

is always preserved. It can be well seen from Figs. 5.21- 5.22 where the distribution of the total ion heat flux ($|\mathbf{q}| \equiv \mathbf{n_i^2 g_{mn}} \chi_{\mathbf{i}}^{\mathbf{mj}} \frac{\partial \mathbf{T_i}}{\partial \mathbf{x_j}} \chi_{\mathbf{i}}^{\mathbf{nk}} \frac{\partial \mathbf{T_i}}{\partial \mathbf{x_k}}$) at the Poincaré section $\phi = 0$ is shown. It appears that the maximum of the energy flow is concentrated in very well localized areas elongated radially from the core to the target. It should be noted that in accordance with the previous modeling attempts [67, 68] the heat preferable goes through the edges of an island.

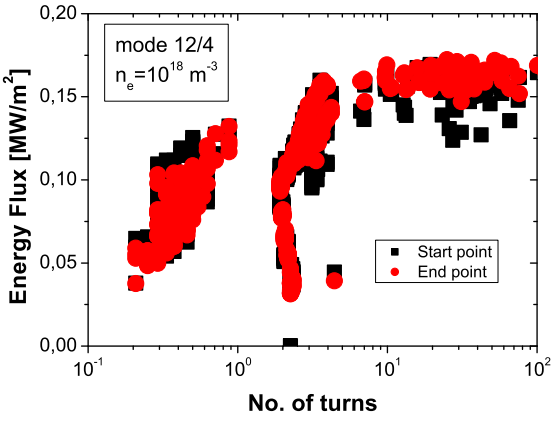

Figure 5.23: Total electron energy fluxes at target plates (start and end points of field lines) versus field line length (number of toroidal turns) for $n_e = 10^{18}$ m^{-3} (mode 12/4)

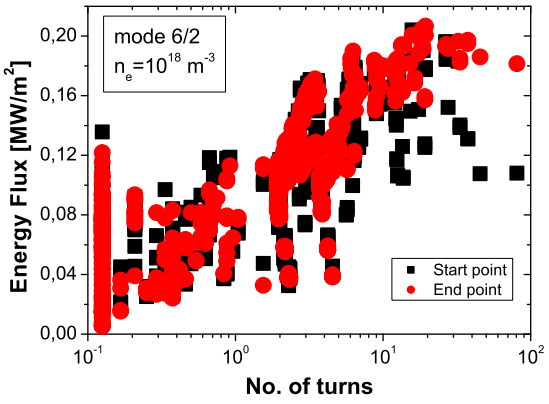

Figure 5.24: Total electron energy fluxes at target plates (start and end points of field lines) versus field line length (number of toroidal turns) for $n_e = 10^{18}$ m^{-3} (mode 6/2)

In order to consider the question which field lines, short or long, contribute stronger to the heat load to the target, we have analyzed the total parallel energy fluxes at the start and end points of open field lines as a function of the field line length and the results are plotted in Figs. 5.23-5.24, for both modes and plasma density $n_e = 10^{18}$ m^{-3}. It can be seen that heat is

preferably transported to the target by field lines with longer connection length. In case of the mode 6/2, there is a tendency that the heat delivered to the target increases with the field line length. However, in the case of the mode 12/4, such conclusion is only true for magnetic field lines, which have relatively short connection length, less than about connection length shorter than 40 toroidal turns (roughly 10 poloidal turns). For longer field lines the heat transmitted to the target does not depend on the field line length. As it has been discussed in [36, 37]

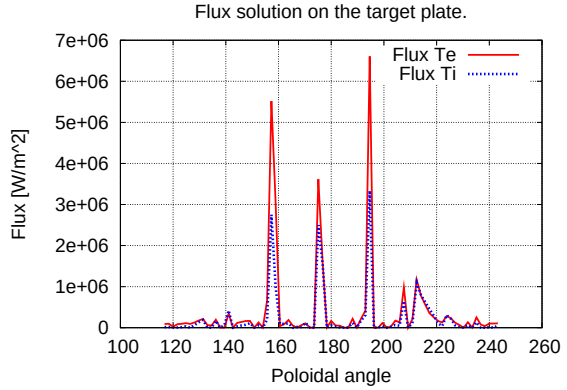

Figure 5.25: Poloidal profile of electron and ion energy fluxes at the divertor targets for $n_e = 10^{18}\ \ m^{-3}$ and 12/4 mode.

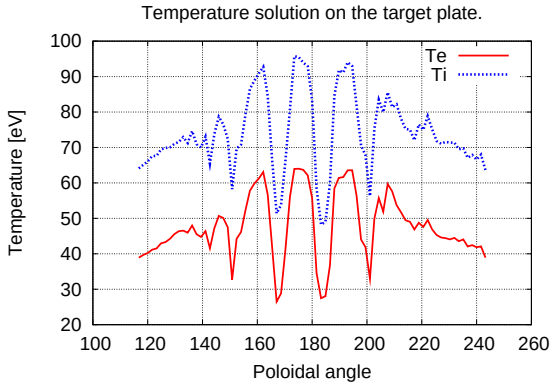

Figure 5.26: Poloidal profile of electron and ion temperature at the divertor targets for $n_e = 10^{18}\ \ m^{-3}$ and 12/4 mode.

the heat flux depends on the radial penetration of magnetic field lines, these which have longer connection length penetrate deeper within the Kolmogorov length. Most likely magnetic field lines with ($L_c \approx 40$ toroidal turns) reach (within the Kolmogorov length) the innermost part of the perturbed volume. We note that the difference in the heat flux between long and short field lines is not large (factor $\sim 4-5$) whereas the field line length changes by three orders. Since the number of short field lines is much larger than long field lines (~ 5 times) therefore short field lines might contribute to the total heat load as strongly as the long ones. We note that the picture is qualitatively the same for other densities. This result is not confirmed by experimental findings (see e.g. [37]), where the heat flux brought to the target by field lines with long connection length is reduced at higher densities. This would suggest that other terms, like convective transport might be important. It should be noted also, that since the plasma temperature at the core interface is the same for all densities, the changes to the heat fluxes due to larger densities could be override by the changes to the input energy fluxes.

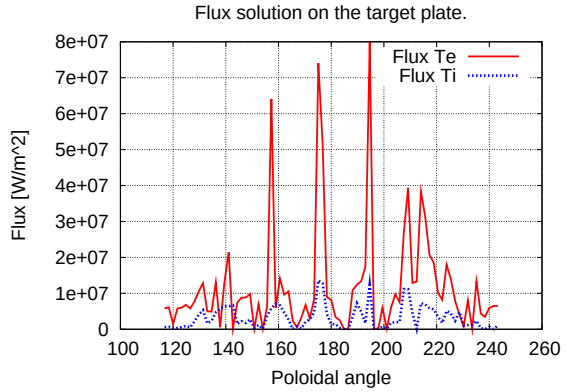

Figure 5.27: Poloidal profile of electron and ion energy fluxes at the divertor targets for $n_e = 5 \times 10^{19}$ m^{-3} and 12/4 mode.

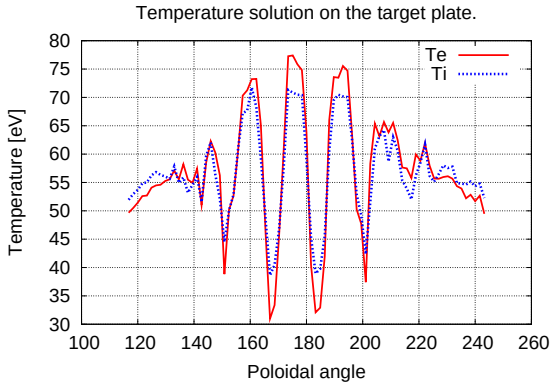

Figure 5.28: Poloidal profile of electron and ion temperature at the divertor targets for $n_e = 5 \times 10^{19}$ m^{-3} and 12/4 mode.

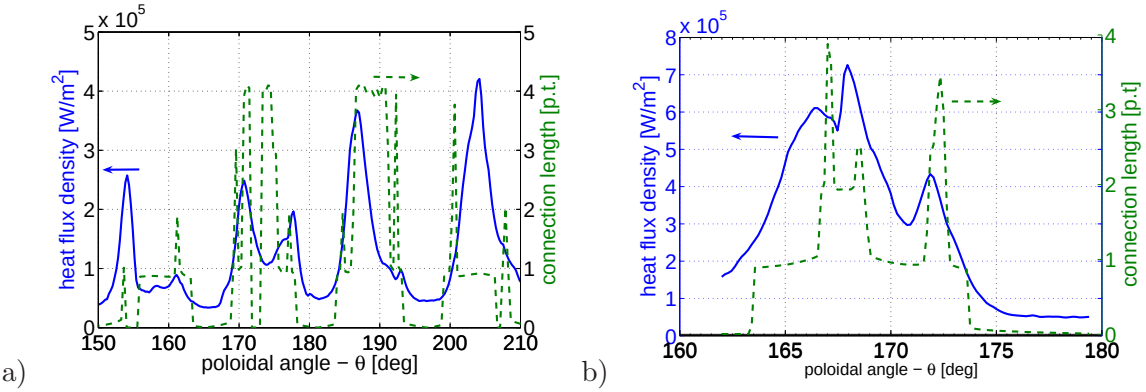

Figure 5.29: The measured heat flux density (blue curves, left ordinates) and calculated connection length (green curves, right ordinates) profiles as a function of the poloidal angle for a) 12/4 mode (TEXTOR #95592) and b) 6/2 mode (TEXTOR #1000975)

In Figs.5.25-5.28 poloidal profiles of total energy fluxes and electron and ion temperatures at the target plates for the toroidal position $\phi = 180^o$ are shown for the lowest and highest density and for the mode 12/4. It can be seen that for low plasma density the electron and ion energy fluxes are comparable but the temperatures are different. However for the high plasma density the electron energy flux is dominant but the plasma temperatures are very similar due to strong energy exchange between ions and electrons. From the energy flux poloidal distribution the hot spot nature of the heat load to the target plates is well visible. The heat flux is strongly poloidally nonuniform with large isolated peaks close to divertor center. Correspondingly the temperature is poloidally modulated with maxima at he same positions as the heat peaks.

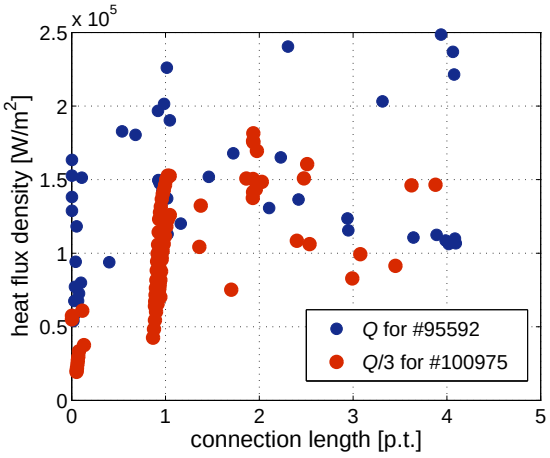

Figure 5.30: The measured heat flux density deposited on the surface of the DED as a function of the field line connection length. The data is evaluated from Fig. 5.29. Blue points show the dependence for the 12/4 mode (TEXTOR #95592, $150 < \theta < 165$), the red ones for the 6/2 base mode (TEXTOR #1000975) divided by 3.

The immanent feature of the heat flux deposition pattern in TEXTOR-DED experiment is that it follows the topology of magnetic field - it has the same helicity and the maximum of the energy deposited corresponds to the field lines with longest connection length. Two examples of the experimental data are shown in Fig. 5.29. Here the heat flux density (the blue curves) and the corresponding connection length profiles (as calculated by ATLAS [29]) are presented

as a function of poloidal angle for the 12/4 mode (Fig. 5.29a) and 6/2 (Fig. 5.29b). For the latter mode the number of stripes is reduced by factor of 2 due to lower value of the toroidal mode. Unfortunately the surface of the DED is not ideally aligned, i.e. some of the tile edges are protruding over the divertor surface, shadowing some parts of the heat deposition patterns. However, we have tried to choose least affected part of the divertor surface to evaluate profiles on. The heat flux density is calculated from the temperature evolution as measured by an infrared camera (the thermographic setup and method of heat flux evaluation has been discussed in [27]).

It is clear that the heat flux density is strongly related to the magnetic field line connection length. This is also shown in Fig.5.30. Here the measured heat flux density is plotted against the connection length (in poloidal turns) of the field lines intersecting the DED surface at the point of the heat deposition. For both modes deposited energy increases with increasing connection length (L_c)up to $2-3$ poloidal turns, what is roughly 7-10 toroidal turns. This is in fair agreement with calculations for the 12/4 mode as presented in Fig. 5.23. For the 6/2 base mode the experimental behavior is not satisfactory reproduced by results of simulations. However, one should note that the heat profile is affected by the tile edge at $\theta \approx 166^\circ$ (see Fig. 5.29b) and it is reduced by effects of shadowing [29] at $\theta \approx 173^\circ$. Also the resolution of the measurement is much lower than for the modeled profiles. In addition, since for the short field lines and high temperatures the convection might be important, the discrepancy between calculated and experimental results could be associated to the lack of the convective terms in our model. The importance of that effect will be discussed in next section.

5.2 Extended model

For an improved description of the heat flux pattern the plasma transport equations have to be extended (see chapter 3). Here, we discuss results for W7-X of this extended model and compare it with the purely conductive case.

5.2.1 1D solution for W7-X

As a first step, we study the characteristics of the 1-D solution along the field lines for one long field line. The geometry information was included through the metric coefficients calculated for the line. The number of cuts taken for this kind of calculations is 500 per one period. Such a great amount of cuts is needed due to the strong variations of the metric coefficients and Jacobian along the field line. The total number of points along that line is 4208 with approximately 8.4 toroidal turns. To analyze the structure in the Jacobian we show both the graph for the whole domain and the graph for one turn. The same kind of pictures are done for density and velocity solution. It can be clearly seen from the pictures, that variations of the density and parallel velocity values coincide with variations of the Jacobian.On the other hand temperature solution was not so strongly modified with respect to the metric coefficient variations due to the much larger parallel heat conduction smoothing the profile.

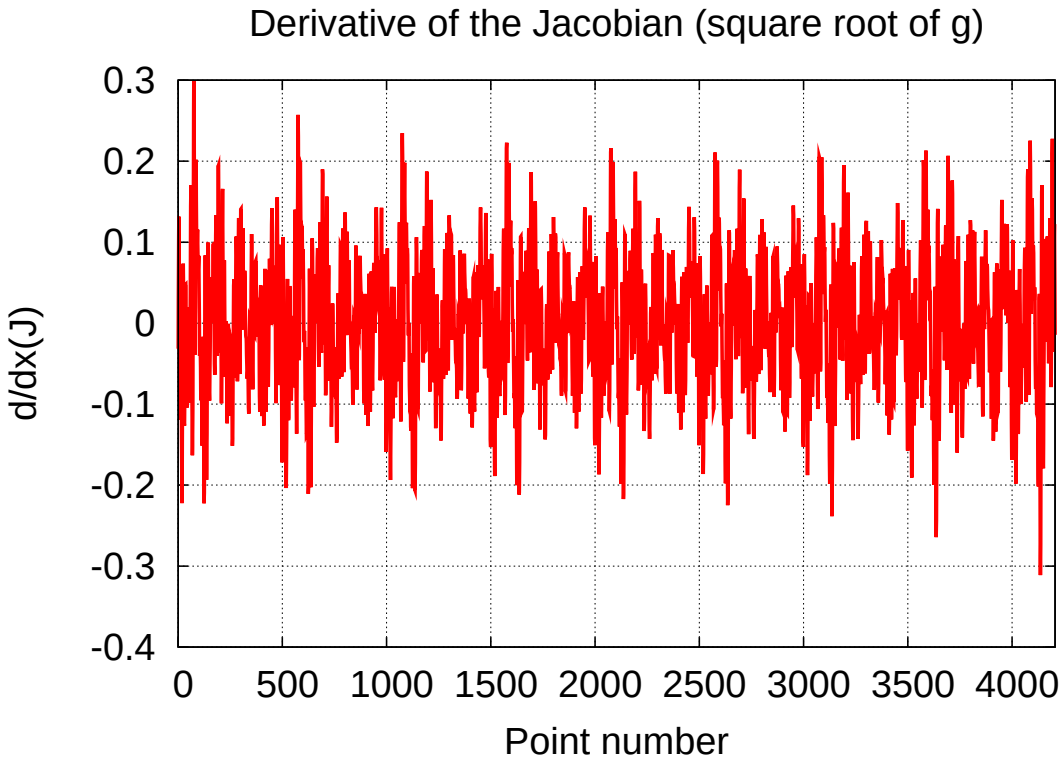

Figure 5.31: *Derivative of the Jacobian taken with a central difference scheme in one local magnetic coordinate system. The full line is shown.*

It is also important to stress here, that due to such strong variations of the metric coefficients we are forced to take such a large number of cuts as needed for resolving properly these variations. Thus, the number of cuts can not be less then 500 per period. Combining this with proper resolution in radial direction we will end up for the total 3D mesh with an approximate number of about 5 million points or even more. Another kind of complexity appears due to the fact,

that the appropriate time step for calculations is strongly connected to the distance between points and thus should be decreased as far as the distance between cuts are increased. Both demands lead to too large numerical resources for a sequential code and therefore the code needs to be parallelized. The parallelization of the code will be possible using domain-decomposition. There are two possible splitting strategies for our setup. The first is a division of the grid with respect to field lines and the second a division with respect to toroidal cuts. The first allows to make the whole chain of calculations (up to obtaining the solution on the current time step) independently on each processor, but for the construction of the radial terms information about all the quantities from the neighbours is needed. This will need further exchange of information between processors. The second strategy allows one easily to calculate contributions from the radial terms, but at the same time all coefficients from each line have to be gathered on a single processor to have the possibility to obtain the solution for the parallel terms with a tridiagonal solver. The most promising setup would be a mixing of both cases. As a result, each processor can be used for two kind of calculations: at a first step the right hand side coefficients from the radial transport for one or several toroidal cuts are calculated. As the second step, after all processors have finished the first step and after exchanging data, it can be used for the calculation of the resulting values along the field line.

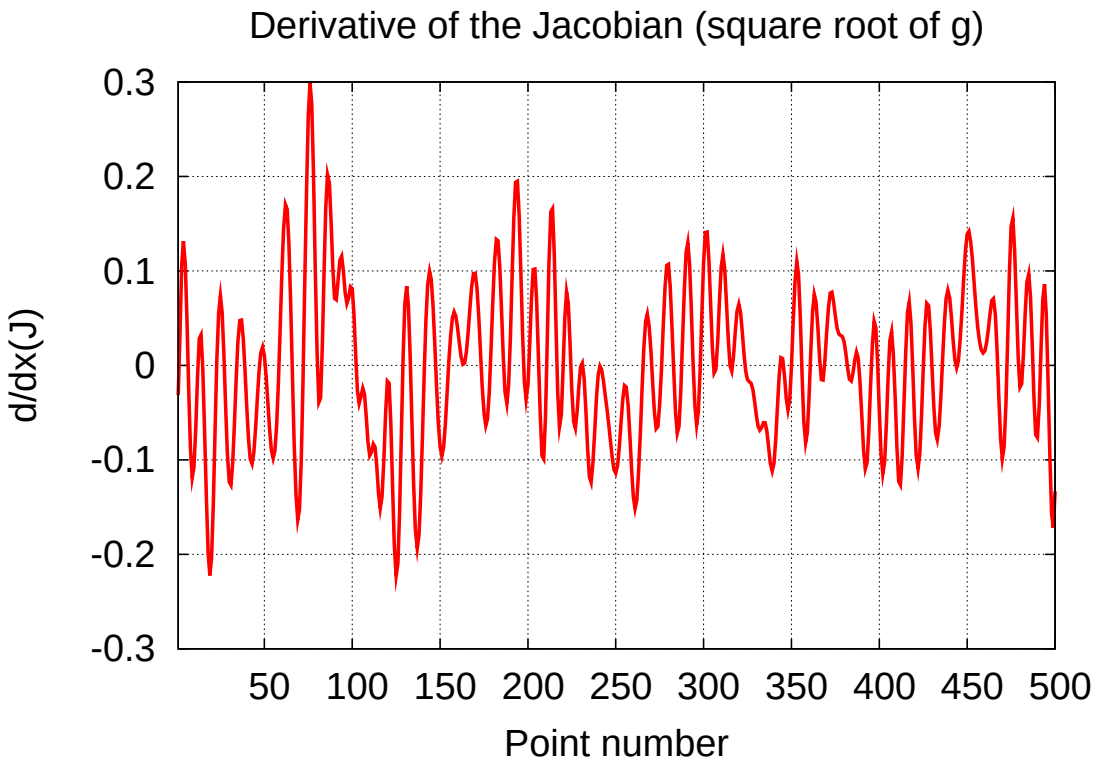

Figure 5.32: *Derivative of the Jacobian taken with a central difference scheme in one local magnetic coordinate system. One turn is shown.*

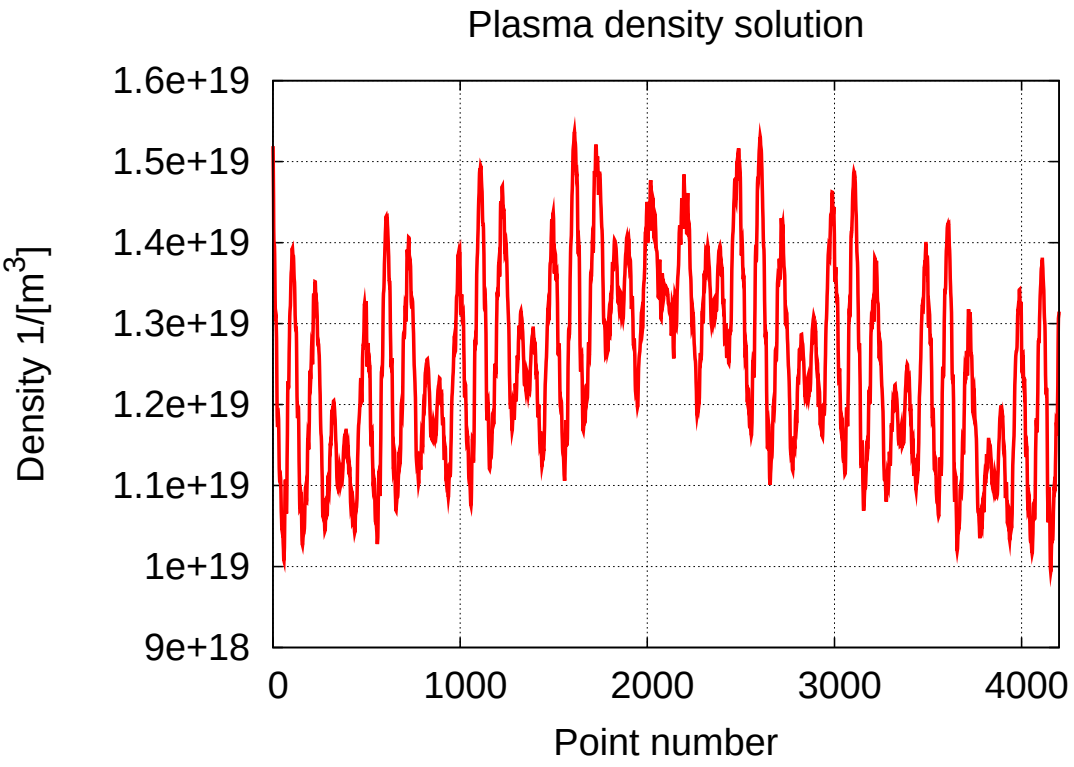

Figure 5.33: *Density solution along the full field line.*

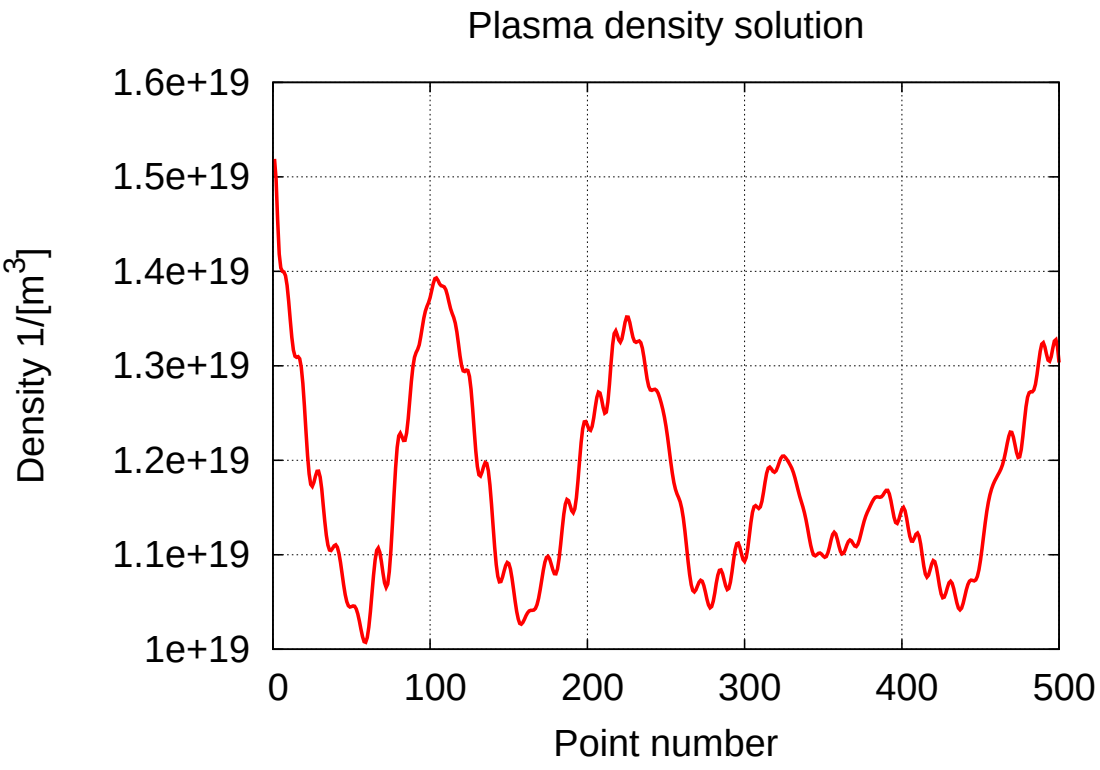

Figure 5.34: *Density solution along the field line for one turn.*

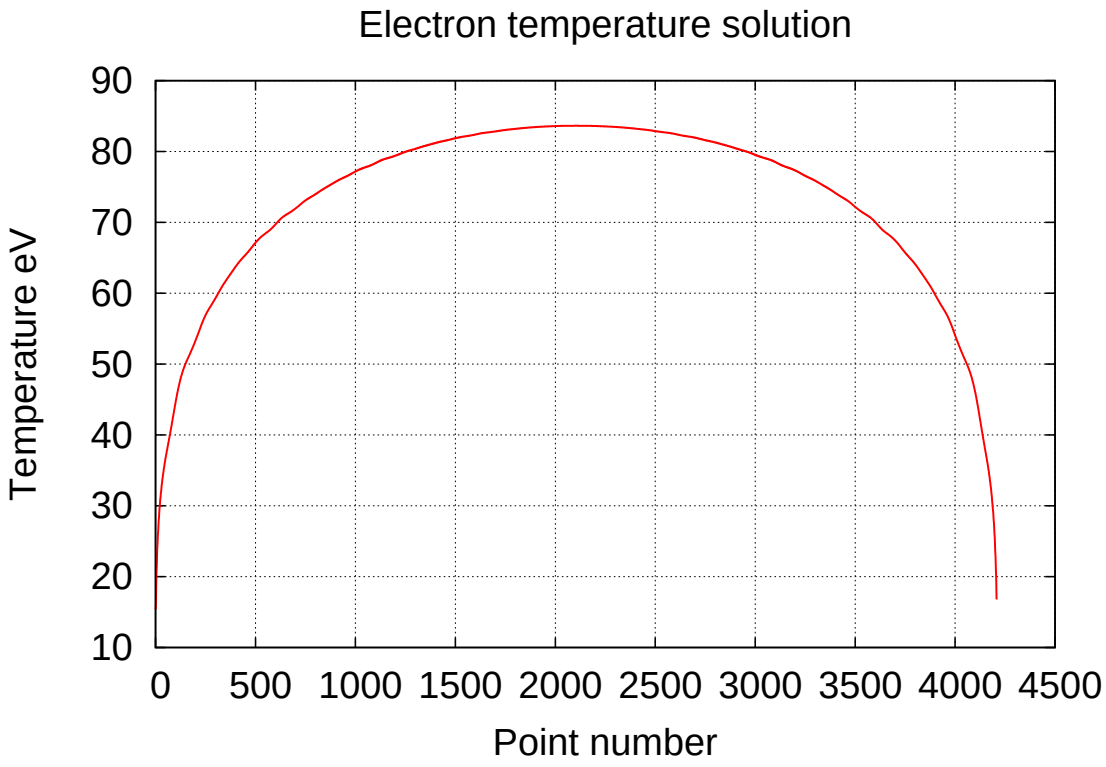

Figure 5.35: *Electron temperature solution along the full field line.*

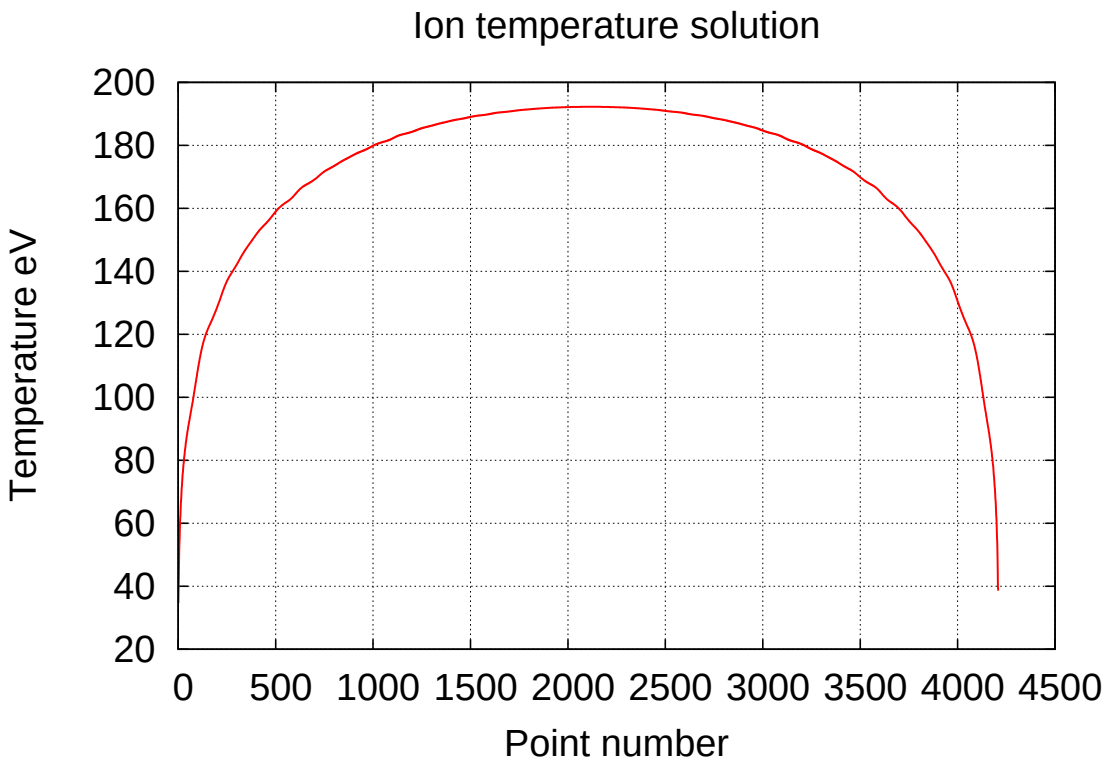

Figure 5.36: *Ion temperature solution along the full field line.*

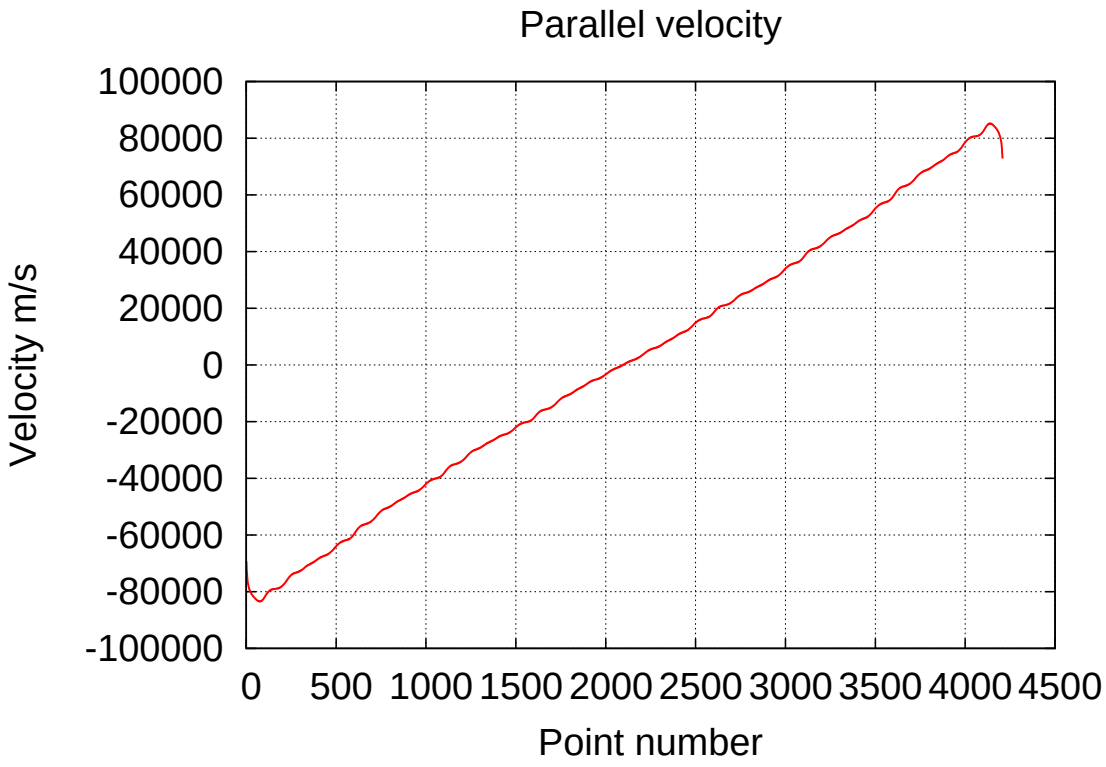

Figure 5.37: *Parallel velocity solution along the full field line.*

5.2.2 3D W7-X case

Based on the results of the previous section, we would have to use a rather large grid to resolve the metrics requesting a parallelized code. However, within the practical limits of this thesis, we decided to use a different ansatz. The additional existence of radial transport will smooth-out most of the fine structure of the 1D solution driven by the metric coefficients. Therefore, we created a smoothed metrics of the W7-X case which allows to reduce the number of toroidal cuts to 50. This procedure enabled the use of the sequential code without the need for parallelizing and guaranteed the successful test of the physics extensions in a reasonable run time. The temperature boundary conditions for the considered case are taken the same as for the pure conductive case described before, i.e. $150eV$ for inner and $5eV$ for outer boundaries, respectively, anomalous conductivity coefficients are taken 2 m^2/s for both T_e and T_i. In addition we have a particle diffusion coefficient 5 m^2/s and perpendicular anomalous viscosity 0.5 m^2/s. For parallel plasma velocity we set 0 fixed value on both inner and outer boundaries. At the target plate we use Bohm criteria with $V_{||}$ set to sound speed. For the plasma density we use again Dirichlet boundary conditions for inner and outer boundaries with values 10^{20} and $5.0 \cdot 10^{18}$ $1/m^3$ respectively. The boundary conditions at the target plates for densities represents the outflow of the particles. The density values are completely determined by the previous value along the field line (pure convective transport). Therefore, no boundary conditions are needed.

Comparing the results of the extended model with the results of the pure conductive problem allows us to identify the importance of the convective effects for the power load at the target plates. The contour plots of the electron and ion temperatures on the bean-like and triangle toroidal cut do not show any clear difference between extended and purely conductive model, therefore we do not show them here. The effect of the target plates show up clearly in the contours of the parallel velocity, where the acceleration towards the target plates due to the Bohm condition is obvious. Analyzing the smaller flow velocities the islands show a rather symmetric splitting of the flows towards the different targets. Plasma density contour plots on the cuts look very similar to electron and ion temperatures. Differences appear only in the radial profiles.

Analyzing the radial profiles of electron and ion temperatures in Fig. 5.45, we observe the same expected trend as in the pure conductive case: electron temperatures drop steeper than

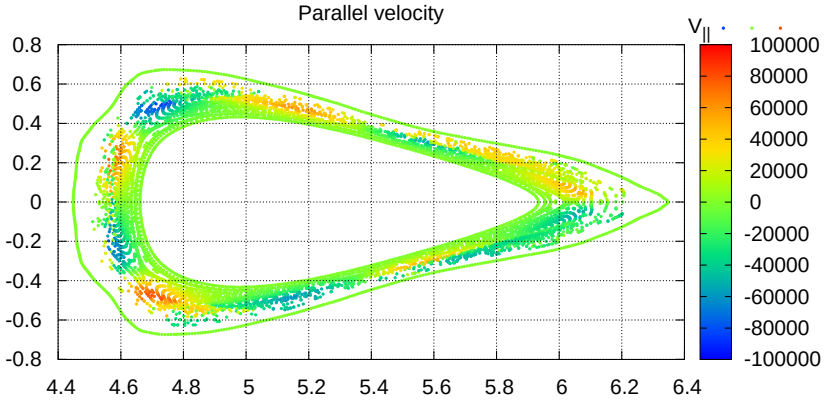

Figure 5.38: *Parallel velocity solution on the "triangle-like" toroidal cut.*

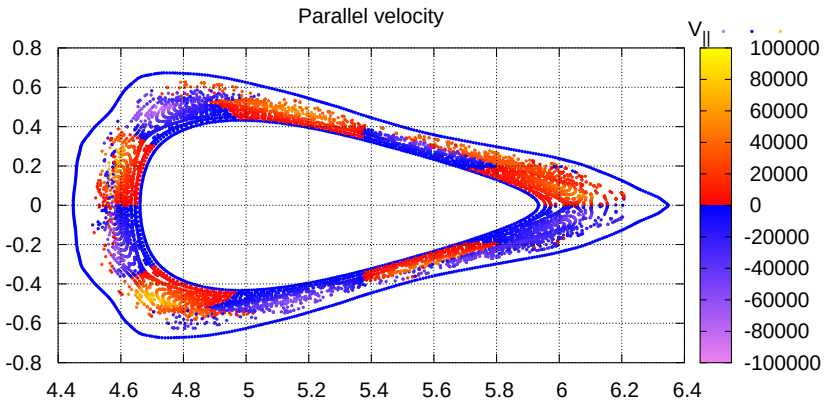

Figure 5.39: *Parallel velocity solution (flows separation) on the "triangle-like" toroidal cut.*

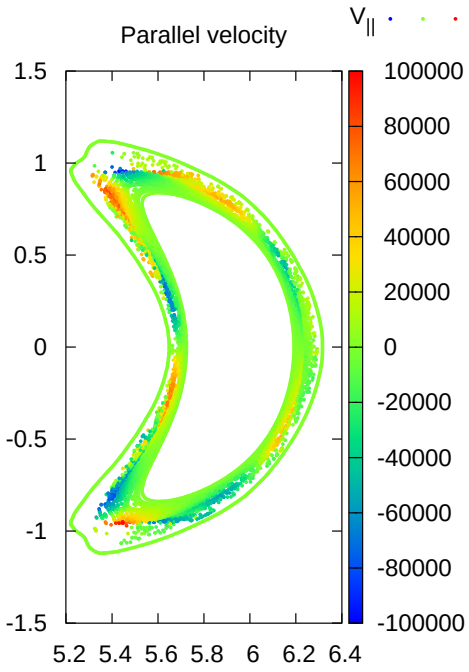

Figure 5.40: *Parallel velocity solution on the "bean-like" toroidal cut.*

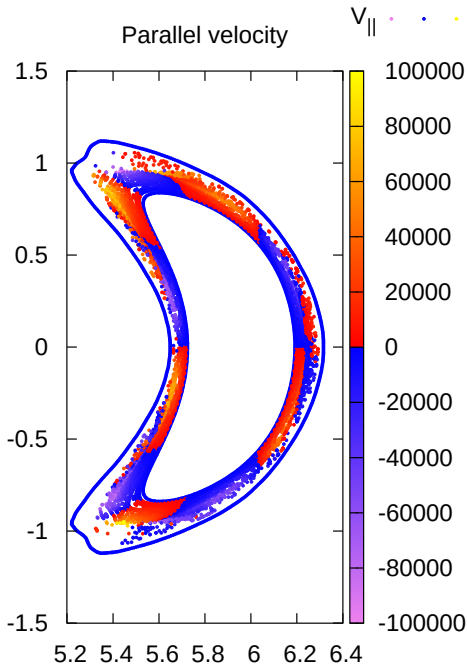

Figure 5.41: *Parallel velocity solution (flows separation) on the "bean-like" toroidal cut.*

ion temperatures. The profile of the ion density is similar than the ion temperature, because both have similar parallel losses. The ion density is characterized in parallel direction by the parallel momentum equation, which has the parallel viscosity in it. The parallel viscosity now

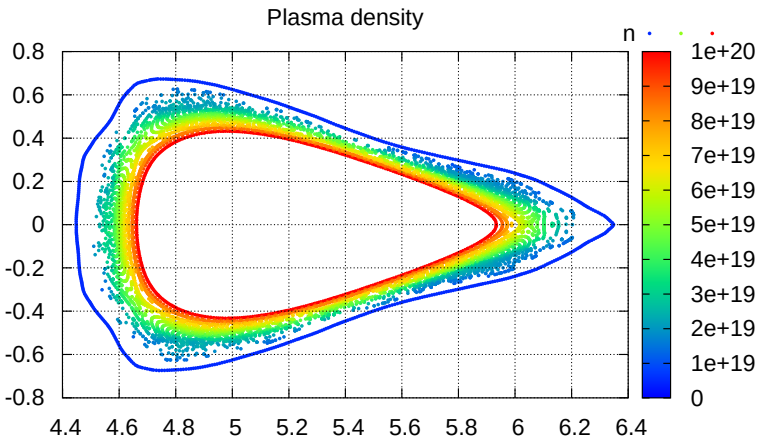

Figure 5.42: *Density solution on the "triangle-like" toroidal cut.*

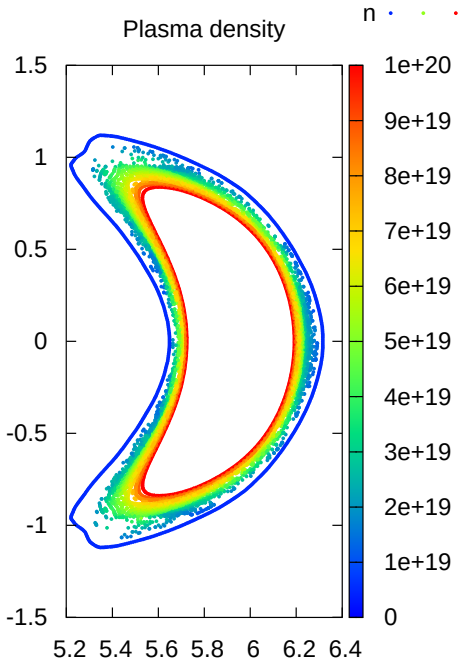

Figure 5.43: *Density solution on the "bean-like" toroidal cut.*

scales as the parallel ion heat conduction and therefore this similarity is expected. However, we see a broader radial profile of the density compared with the ion temperature and this is because we used a larger anomalous transport coefficient for particle transport than for the ion

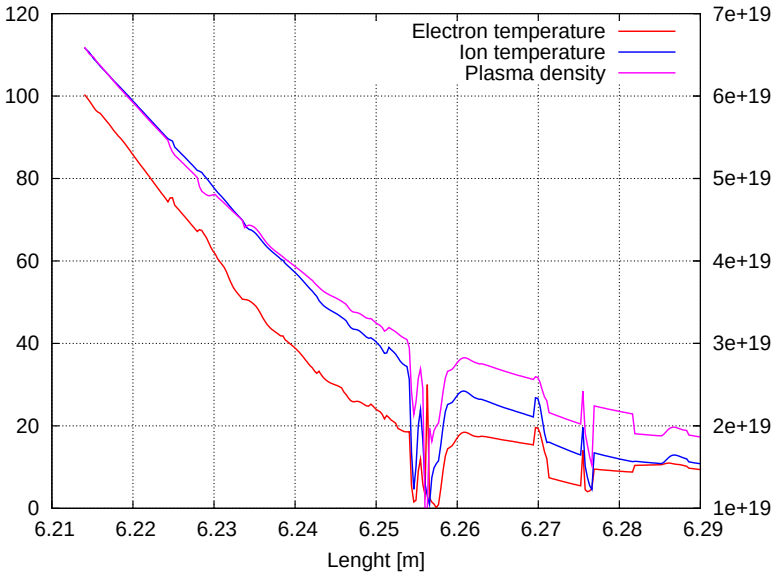

Figure 5.44: *Radial profiles of electron temperature, ion temperature and density.*

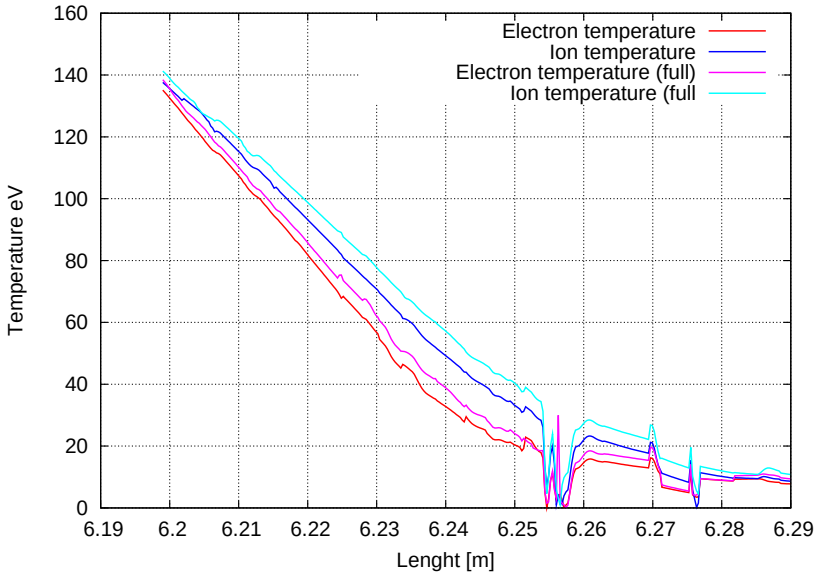

Figure 5.45: *Radial ion and electron temperatures profiles for two cases, i.e. only conduction and conduction-convection.*

transport.

The comparison of the temperature profiles between the full and the purely conductive case (Fig. 5.45) shows very similar profiles. Both temperatures just experience some constant shift due to the convection, where the effect is stronger for ions than for electrons. This is again due to the dominance of the parallel heat conduction term for electrons and therefore less pronounced convective effects. The same is obvious looking at the target plate patterns. Electrons do not change much, but just get larger values due to the additional convective energy. Therefore, also the heat flux density distribution is mostly determined by the field line lengths. In contrast, the ion heat flux density now shows a quite different pattern than for the conductive case: it is strongly determined by the convective heat flux and is more homogeneous than in the

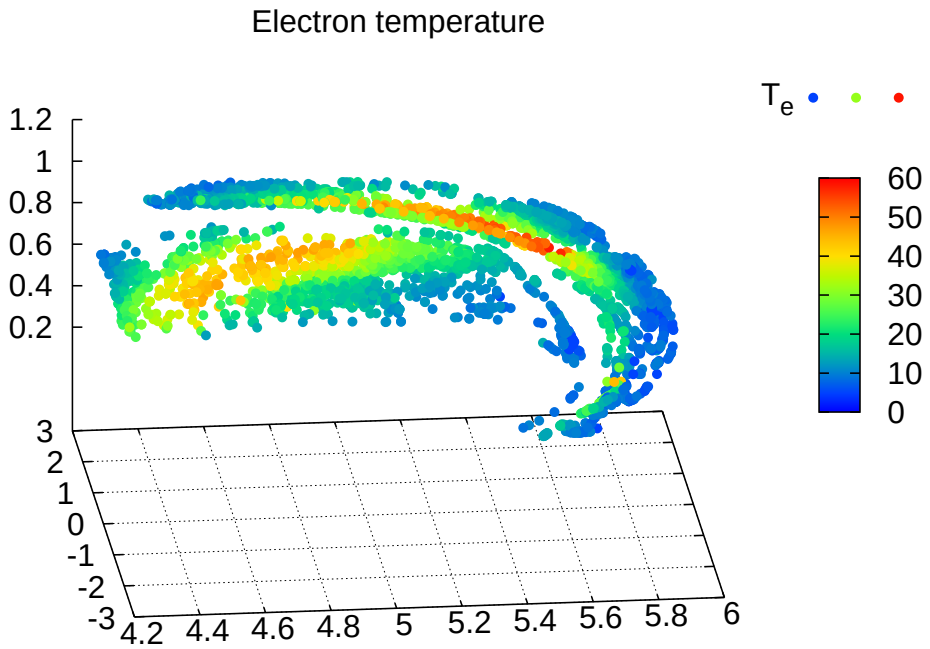

Figure 5.46: *Electron temperature on the target plate.*

conductive case. Also, it is broader than the electron heat flux distribution. The filling effect and broadening also gets visible in the plot of ion heat flux density versus field line length, where now the figure looks much similar to the electron heat flux and more triangle-like with homogeneous distribution of the points.

The coincidence of the maxima of electron heat flux, ion heat flux and particle flux densities in the target patterns also show the importance of the convective heat fluxes and the need for including them into the analysis of heat flux distributions. Also, in contrast to the pure conductive case ions contribute as much to the heat flux densities as electrons.

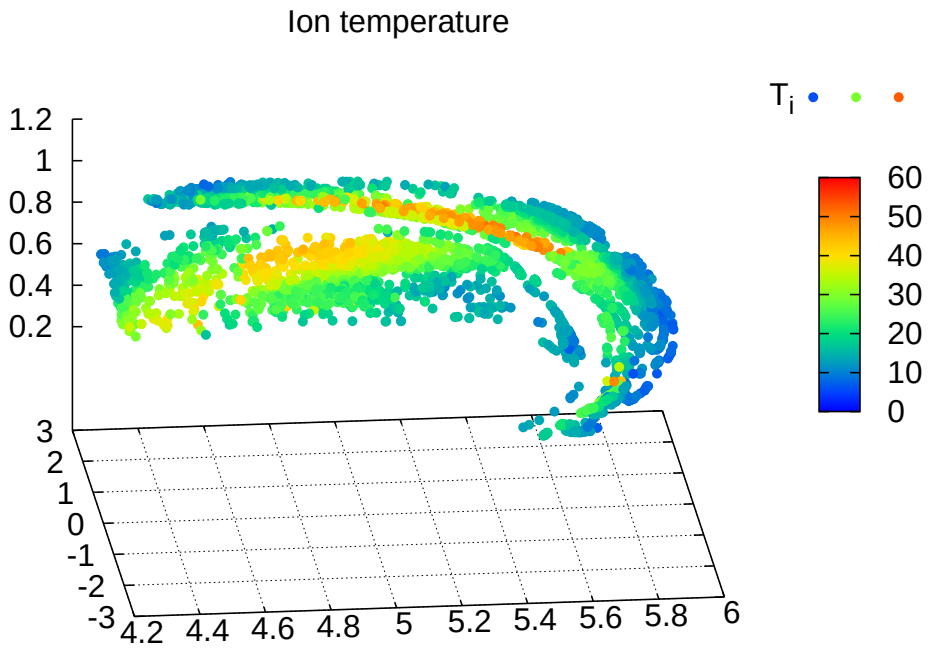

Figure 5.47: *Ion temperature on the target plate.*

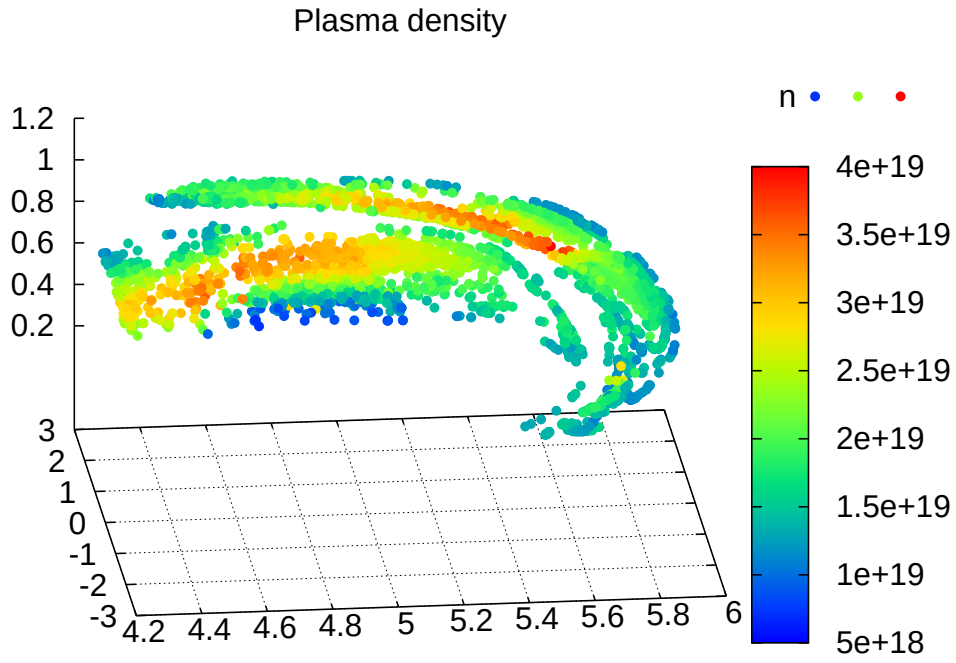

Figure 5.48: *Density on the target plate.*

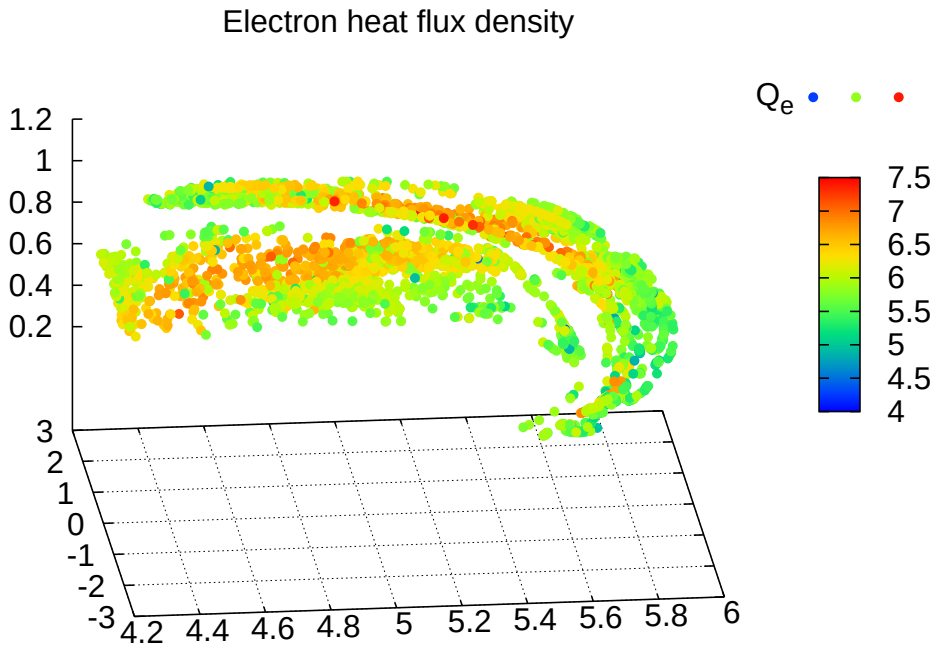

Figure 5.49: *Electron heat flux on the target plate.*

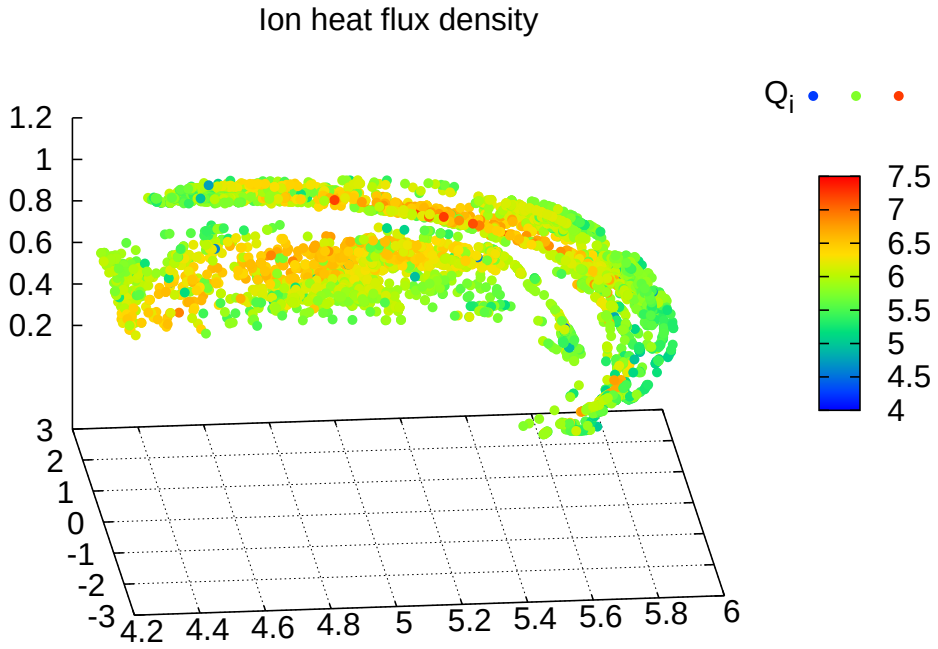

Figure 5.50: *Ion heat flux on the target plate.*

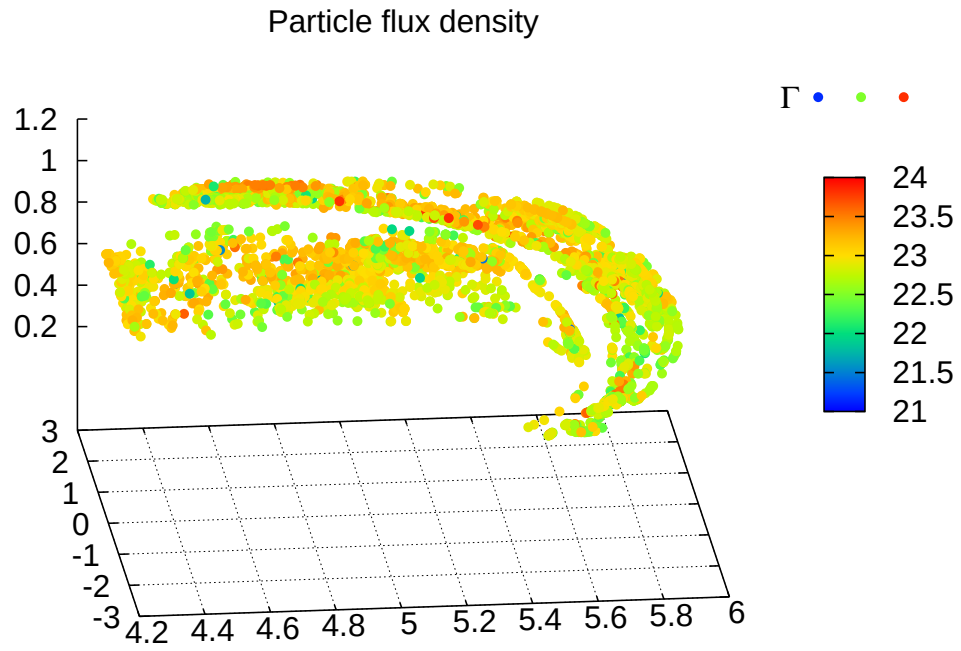

Figure 5.51: *Particle flux on the target plate.*

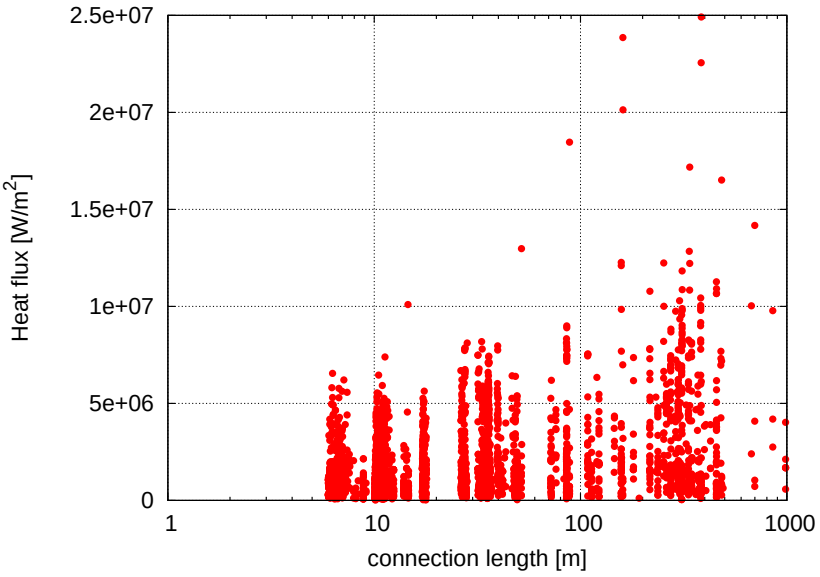

Figure 5.52: *Electron heat flux on the target plate as a function of the connection length.*

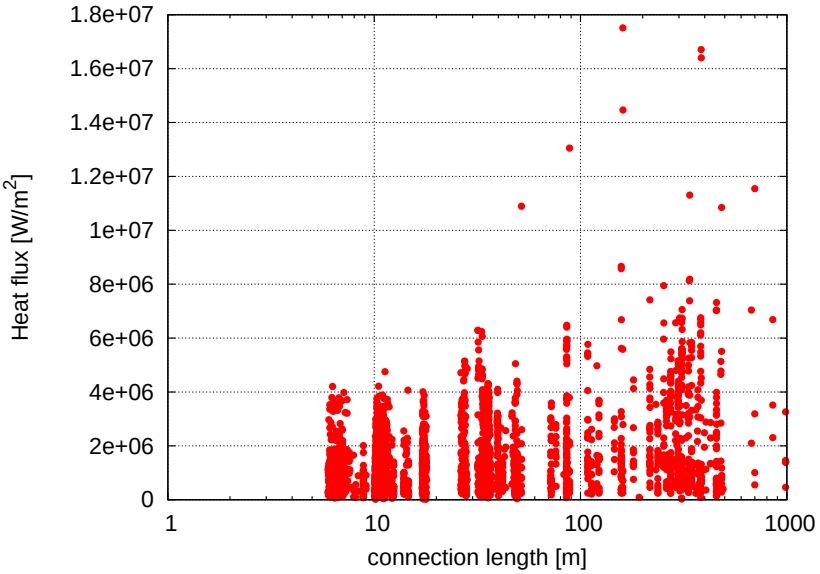

Figure 5.53: *Ion heat flux on the target plate as a function of the connection length.*

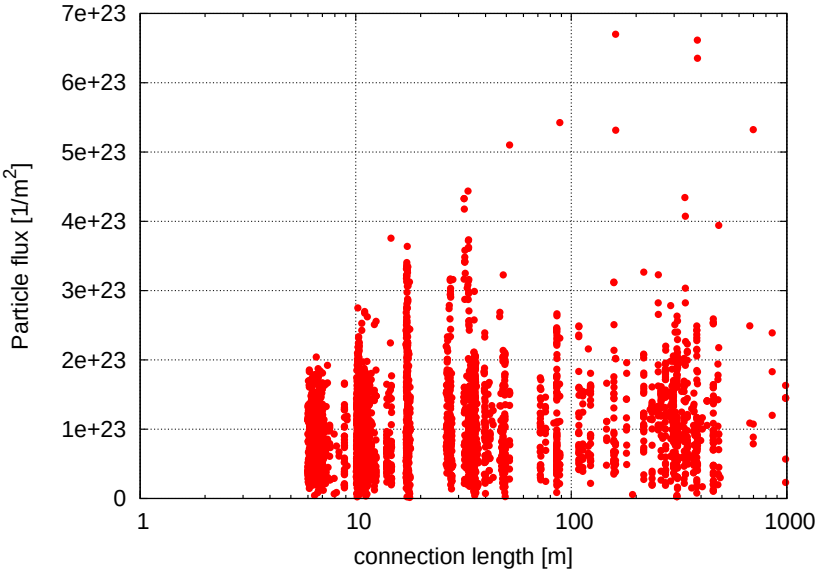

Figure 5.54: *Particle flux on the target plate as a function of the connection length.*

Chapter 6

Effect of neutrals

In the divertor region, in addition to the plasma transport processes the interaction with neutrals is quite important, because they act as sink and sources for particles, energy and momentum. Here, we describe neutral fluid model coupled to the plasma fluid equations to study the effects of neutrals.

6.1 Neutral fluid model

Neutral recycling and further plasma-neutral interaction plays a great role in the SOL physics. The most accurate description of the problem is given by Monte Carlo methods, although a fluid model (e.g. described in [69], [70], [71]) usually makes reasonably good approximation in the case of the neutral mean free path, λ_N, is small compared to the characteristic spatial scale length.
The presented neutral fluid model consists of the particle and momentum balance equations. In addition to that we suppose, that all heavy species (ions and neutrals) have the same temperature, which gives us corrections for the ion temperature equation. The particle neutral fluid equation looks like:

$$\frac{\partial n_N}{\partial t} + \vec{\nabla} \cdot \left(n_N \vec{V_N} \right) = -S_i + S_r, \tag{6.1}$$

where n_N is the neutral density, $\vec{V_N}$ is the neutral flow velocity, and $S_{N,i}, S_{N,r}$ stands for ionization and recombination sources, respectively. For the neutral parallel momentum equation the full Navier-Stokes equation is used:

$$\begin{aligned} &\frac{\partial}{\partial t} \left(m n_N V_{||N} \right) + \vec{\nabla} \cdot \left(m n_i \vec{V}_N V_{||N} - \hat{\eta}_N \vec{\nabla} V_{||N} \right) = \\ &- \nabla_{||} \rho_N + m n_i n_N K_{cx} \left(V_{||i} - V_{||N} \right) + m S_r V_{||i} - m S_i V_{||N}, \end{aligned} \tag{6.2}$$

where $V_{||i}$ is the ion parallel flow velocity, $V_{||N}$ is the neutral parallel flow velocity, m is the atom mass, n_i is the background plasma density, ρ_N is the neutral pressure, η_N is the neutral viscosity, and K_{cx} is the change-exchange rate. The perpendicular neutral velocity is calculated in the diffusive approximation:

$$V_{\perp N} = -D_N \left(\nabla_{||} n_N / n_N + \nabla_\perp T_i / T_i \right), \tag{6.3}$$

In order to receive a realistic solution we have to use a flux limiting procedure that reduces the diffusive perpendicular particle fluxes to some reasonable values, known from the kinetic modeling. So we replace $j_{N\perp}$ with

$$\frac{j_{N\perp}}{\left(1 + \alpha \left(\frac{j_{N\perp}}{n_N V_{T_i}/4} \right)^\gamma \right)^{1/\gamma}}, \tag{6.4}$$

where typical parameter values for α and γ are 1 and 2, respectively.

The above-mentioned flux limiting procedure is applied to the perpendicular neutral velocity.

6.2 Transport equations including neutrals

In this section we formulate the "Findif" set of equations including neutrals. The plasma density equation has only one difference to Eq. 3.34, that it includes now ionization and recombination sources in the right hand side. Ionization is a source of plasma, whereas recombination is a sink (the plasma transforms to the neutrals).
For the neutral density equation we express the perpendicular velocity in an explicit form and apply the flux limiting procedure to all its components. Hence, we write down the neutral density equation in a following form:

$$\begin{aligned}&\frac{\partial n_N}{\partial t}+\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[n_N(h^3V_{N||}+V^3_{N\perp})\right]+\\&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left(n_NV^1_{N\perp}\right)+\\&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left(n_NV^2_{N\perp}\right)=-S_i+S_r;\end{aligned}\tag{6.5}$$

We can express ionization source S_i and recombination sink S_r as:

$$\begin{aligned}S_i&=K_in_Nn_i,\\S_r&=K_rn_in_i;\end{aligned}\tag{6.6}$$

The perpendicular neutral velocity $V_{N\perp}$ is taken as:

$$\begin{aligned}V^3_{N\perp}=&-D_{N\perp}\left((g^{33}-(h^3)^2)\frac{\partial n_N}{\partial x^3}+g^{31}\frac{\partial n_N}{\partial x^1}+g^{32}\frac{\partial n_N}{\partial x^2}\right)/n_N-\\&D_{N\perp}\left((g^{33}-(h^3)^2)\frac{\partial T_h}{\partial x^3}+g^{31}\frac{\partial T_h}{\partial x^1}+g^{32}\frac{\partial T_h}{\partial x^2}\right)/T_h\\V^3_{N\perp}=&-D_{N\perp}\left(g^{21}\frac{\partial n_N}{\partial x^1}+g^{22}\frac{\partial n_N}{\partial x^2}+g^{23}\frac{\partial n_N}{\partial x^3}\right)/n_N-\\&D_{N\perp}\left(g^{21}\frac{\partial T_h}{\partial x^1}+g^{21}\frac{\partial T_h}{\partial x^2}+g^{23}\frac{\partial T_h}{\partial x^3}\right)/T_h\\V^3_{N\perp}=&-D_{N\perp}\left(g^{11}\frac{\partial n_N}{\partial x^1}+g^{12}\frac{\partial n_N}{\partial x^2}+g^{13}\frac{\partial n_N}{\partial x^3}\right)/n_N-\\&D_{N\perp}\left(g^{11}\frac{\partial T_h}{\partial x^1}+g^{11}\frac{\partial T_h}{\partial x^2}+g^{13}\frac{\partial T_h}{\partial x^3}\right)/T_h\end{aligned}\tag{6.7}$$

For plasma parallel momentum balance equation we also have additional sources describing the interaction with the neutrals:

$$S_{V_{||i}}=mn_in_nK_{cx}(V_{N||}-V_{i||})-mS_rV_{i||}+mS_iV_{N||}.\tag{6.8}$$

From the expression above it is clear, that there are three ways of changing plasma velocity: the ionization acts as a source of the momentum for the plasma (the same as in the case of the plasma density), the recombination on the opposite side is the sink of the momentum and the charge exchange term forwards the momentum from the plasma to neutrals and vice-versa.

The neutral velocity looks as follows:

$$
\begin{aligned}
& mn_N \frac{\partial}{\partial t}\left(V_{N||}\right) + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[mn_N V_{N||}(h^3 V_{N||} + V^3_{N\perp})\right] - \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\eta_N\left(g^{31}\frac{\partial V_{N||}}{\partial x^1} + g^{32}\frac{\partial V_{N||}}{\partial x^2} + g^{33}\frac{\partial V_{N||}}{\partial x^3}\right)\right] + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left[mn_N V_{N||}V^1_{N\perp} - \eta_N\left(g^{11}\frac{\partial V_{N||}}{\partial x^1} + g^{12}\frac{\partial V_{N||}}{\partial x^2} + g^{13}\frac{\partial V_{N||}}{\partial x^3}\right)\right] + \\
& \frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left[mn_N V_{N||}V^2_{N\perp} - \eta_N\left(g^{21}\frac{\partial V_{N||}}{\partial x^1} + g^{22}\frac{\partial V_{N||}}{\partial x^2} + g^{23}\frac{\partial V_{N||}}{\partial x^3}\right)\right] \\
& = -g^{33}\frac{\partial p_N}{\partial x^3} - mV_{N||}\frac{\partial n_N}{\partial t} + \\
& mn_N n_N K_{cx}(V_{i||} - V_{N||}) + mS_r V_{i||} - mS_i V_{N||};
\end{aligned}
\tag{6.9}
$$

Here we see the same sources as was added to the plasma velocity equation but with the opposite sign.

In the electron temperature equation an additional source term S_{E_e} was added which represents the energy loss due to hydrogenic line radiation and the energy gain from the three body recombination process. It is very important to include the last source because it keeps electron temperature high enough and does not allow to drop infinitely due to the radiation mechanism. According to our model all the heavy species in the plasma (namely, the ions and neutrals) have the same temperature. Hence, we summarize the ion temperature and the neutral temperature

equations in a common heavy particle temperature equation:

$$
\begin{aligned}
&\frac{3}{2}(n_i+n_N)\frac{\partial}{\partial t}\left(T_h\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\frac{5}{2}(n_i(h^3V_{i||}+V_{i\perp}^3)+n_N(h^3V_{N||}+V_{N\perp}^3))T_h-\left[\kappa_h^A g^{33}+(\kappa_h^0-\kappa_h^A)(h^3)^2\right]\frac{\partial T_h}{\partial x^3}\right]-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left[\kappa_h^A\left(g^{31}\frac{\partial T_h}{\partial x^1}+g^{32}\frac{\partial T_h}{\partial x^2}\right)\right]+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left[\frac{5}{2}\left(n_iV_{i\perp}^1+n_NV_{N\perp}^1\right)T_h-\kappa_h^A\left(g^{11}\frac{\partial T_h}{\partial x^1}+g^{12}\frac{\partial T_h}{\partial x^2}+g^{13}\frac{\partial T_h}{\partial x^3}\right)\right]+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left[\frac{5}{2}\left(n_iV_{i\perp}^2+n_NV_{N\perp}^2\right)T_h-\kappa_h^A\left(g^{21}\frac{\partial T_h}{\partial x^1}+g^{22}\frac{\partial T_h}{\partial x^2}+g^{23}\frac{\partial T_h}{\partial x^3}\right)\right]\\
&=\frac{3}{2}T_h\frac{\partial(n_i+n_N)}{\partial t}-\frac{\partial}{\partial t}\left(\frac{1}{2}m_in_iV_{i||}^2+\frac{1}{2}m_in_NV_{N||}^2\right)-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}\left(\frac{1}{2}m_in_iV_{i\perp}^1(V_{i||})^2+\frac{1}{2}m_in_NV_{N\perp}^1(V_{N||})^2\right)-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}\left(\frac{1}{2}m_in_iV_{i\perp}^2(V_{i||})^2+\frac{1}{2}m_in_NV_{N\perp}^2(V_{N||})^2\right)-\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\left(\frac{1}{2}m_in_i\left(V_{i\perp}^3+h^3V_{i||}\right)(V_{i||})^2+\frac{1}{2}m_in_i\left(V_{N\perp}^3+h^3V_{N||}\right)(V_{N||})^2\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^1}\sqrt{g}g^{11}\left(\frac{1}{2}\eta_i^A\frac{\partial(V_{i||})^2}{\partial x^1}+\frac{1}{2}\eta_N\frac{\partial(V_{N||})^2}{\partial x^1}\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^2}\sqrt{g}g^{22}\left(\frac{1}{2}\eta_i^A\frac{\partial(V_{i||})^2}{\partial x^2}+\frac{1}{2}\eta_N\frac{\partial(V_{N||})^2}{\partial x^2}\right)+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}g^{33}\frac{1}{2}\eta_N\frac{\partial(V_{N||})^2}{\partial x^3}+\\
&\frac{1}{\sqrt{g}}\frac{\partial}{\partial x^3}\sqrt{g}\frac{1}{2}\left[\eta_i^Ag^{33}+(\eta_i^0-\eta_i^A)(h^3)^2\right]\frac{\partial(V_{i||})^2}{\partial x^3}-\\
&(V_{i||}h^3+V_{i\perp}^3)g^{33}\frac{\partial p_e}{\partial x^3}+k(T_e-T_h)+S_{E_h};
\end{aligned}
\tag{6.10}
$$

6.3 Implementation details

There are two main processes which define the diffusive behaviour of the neutrals - charge exchange and neutral-neutral interaction with cross-sections σ_{cx} and σ_{nn} respectively. The mean free path of a neutral H atom λ before exchanging charge with H^+ ion of velocity v_i is

$$\lambda_{cx}=\frac{v_0}{n_i<\sigma_{cx}v_i>}, \tag{6.11}$$

where $v_0=v_i=\sqrt{2kT_0/m_0}$. Such a process can be described as diffusive with a diffusion coefficient $D_{cx}=v_0/(n_i\sigma_{cx})$. By analogy we can describe the process of neutral-neutral collisions: $D_{NN}=v_0/(n_N\sigma_{NN})$. To construct the diffusion coefficient taking into account both processes we take the harmonic average of D_{cx} and D_{NN}. The viscosity coefficient for both parallel and radial cases is taken as $\eta=m_Nn_ND_N$, where D_N is the above introduced diffusion coefficient. For the practical calculations we take $\sigma_{cx}=2.0\cdot10^{-18}$ and $\sigma_{nn}=2.0\cdot10^{-19}$.

Reaction rates for ionization, charge exchange and recombination are taken as:

$$\begin{aligned} K_i &= \frac{3 \cdot 10^{-14} \left(T_e/10.\right)^2}{\left(3 + \left(T_e/10.\right)^2 + \dfrac{0.1}{\left(T_e/10.\right)^5}\right)} \\ K_{cx} &= \sigma_{cx} v_0 \\ K_r &= 4.684 \cdot 10^{-19} \frac{1}{T_e^{\frac{3}{2}} \left(\frac{1}{T_e} + 0.0434\right)} \end{aligned} \tag{6.12}$$

All described reaction rates with respect to the temperature are shown in Fig. 6.1.

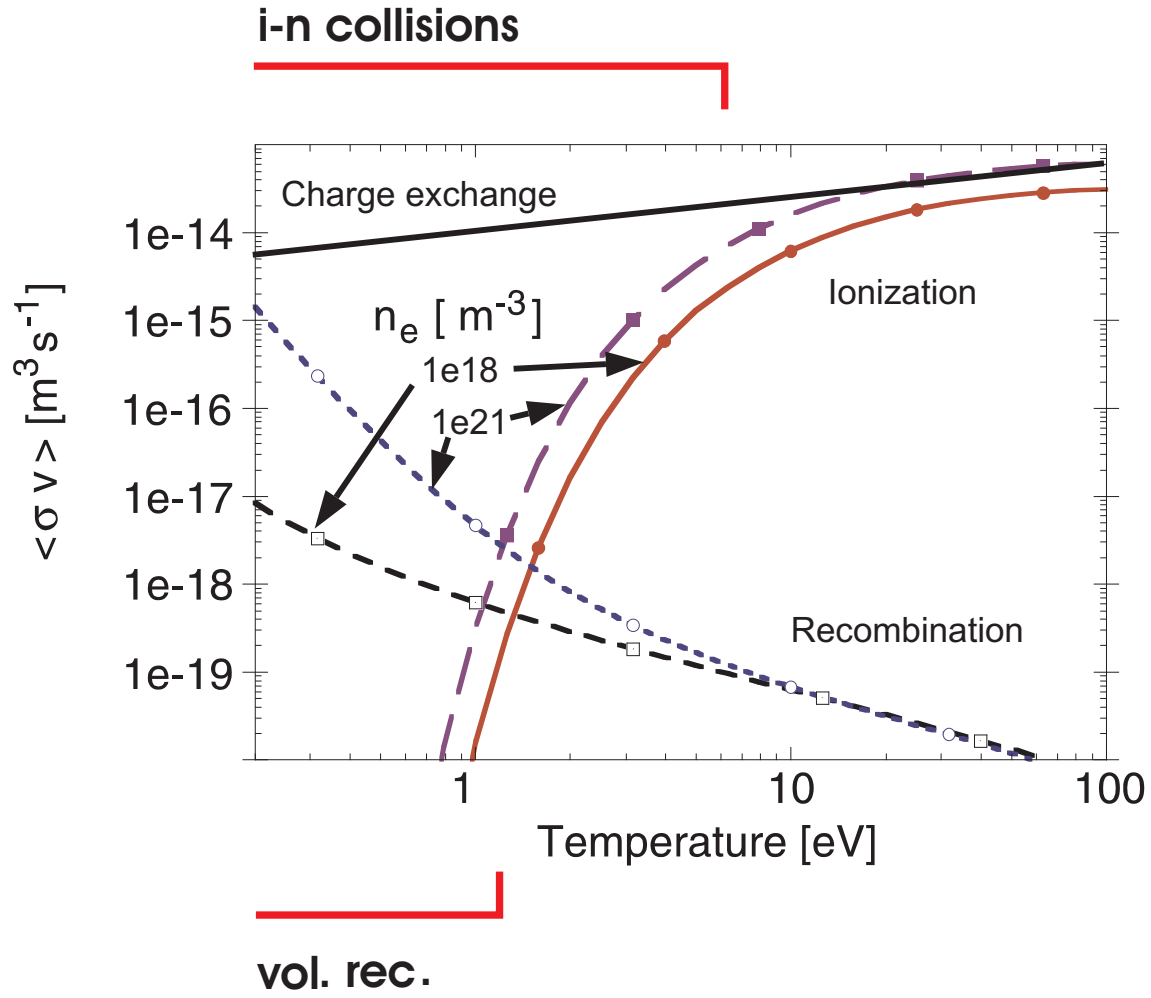

Figure 6.1: *Reaction rates for charge-exchange, ionization and recombination for hydrogen (in $m^3 s^{-1}$) versus temperature (in eV).*

There are two kind of energy sources included in the code: hydrogenic line radiation and three body recombination. The hydrogenic line radiation source is taken as $S_{E_e,h} = -25.0\, eV \cdot K_i$. The three body recombination describe collision of three particles - two electrons and one ion. As a result of such a collision appears a neutral and the second electron gain 75% of recombination energy (13.6 eV for hydrogen), the difference is radiated. Thus we have $S_{E_e,r} = 0.75 \cdot 13.6\, eV \cdot K_r$.

An important part of the neutral model are the boundary conditions. Determining recycling boundary conditions for the neutral fluid model, one faces the problem transferring kinetic boundary conditions of reflected particles into a fluid picture. The recycling boundary condition

means, that a part of plasma particle flux Γ_i impinging on the target plates returns as a neutral flux Γ_N directed towards the plasma: $\Gamma_{N\perp} = \gamma\Gamma_{i\perp}$. This relation does not allow to determine the neutral density and neutral velocity separately and, thus, an additional condition is needed. It is impossible to determine a boundary value of either neutral density or neutral velocity based on the knowledge of the macroscopic parameters as these values itself are integral quantities self-consistently defined by the process of plasma-neutral interaction. In practice, the neutral velocity is taken as a fraction of the sound speed from comparison with full kinetic Monte Carlo simulations taking the spatial average over the first cell. In combination with the above relation it allows to determine then the neutral density.

6.4 1D solution

In this section, we present results of the solution of the plasma transport problem including neutral interaction in 1D by the neutral fluid model described before. The 1D setup is not applicable to real devices, but it allows to obtain results for interpretation and verification of the code. The total field line length was 1m.
For the 1D cases presented here, we used a plasma source boundary condition on the left and target plate boundary conditions on the right. For incoming plasma we used constant values. On the right side, we apply sheath boundary conditions for plasma and recycling conditions for neutrals described before. We present here two cases with and without neutrals and for two different temperatures at the left boundary of 50 and 10 eV. The plasma density at the left boundary is prescribed to 10^{19} $1/m^3$ and velocity to 5000 m/s for all cases. The neutral recycling coefficient γ is kept as 0.85. The penetration depth of the neutrals can be estimated with Eq. 6.11. In our case for a plasma density at the target plate of about 10^{19} we get $\lambda_{cx} \approx 5$ cm.

Figs. 6.3, 6.2 show the drop of the heavy particles temperature with respect to the case without neutrals. This drop is caused by the redistribution of energy on the kinetic part included by adding neutrals. The electron energy also drops, but only slightly compared with the heavy particles temperature. The reason of such a drop is the coupling term which brings electron temperature towards the heavy particle temperature. Radiation losses for electrons are still too small due to the low neutral density to appear in the profile. For plasma density, the rise of the profile near the target plate due to recycling can be observed in Fig. 6.4. For the lower temperature we see, that due to the ionization of the neutrals a large recycling neutral flux converts to the plasma flux as seen in Fig. 6.5. It fills the whole domain such, that the density inside the domain can be bigger than the boundary density.
In the case of 50 eV the coupling of the momentum equations between ions and neutrals is still quite small and only small differences for the parallel velocity of ions with and without neutrals appear. For the 10 eV case the charge-exchange coupling gets quite strong and the neutrals slow down the plasma flow considerably being on the other hand accelerated themselves.
All these results support the correct implementation of the neutral fluid model into the code. A full run of W7-X including neutrals and ions experiences unfortunately a resolution problem. The metrics requires a very fine resolution as also the neutral dynamics does due to the fast profile decay on the length scale of the neutral mean free path. Therefore, only a parallel code will be able to treat such a case, which is beyond the scope of this work.

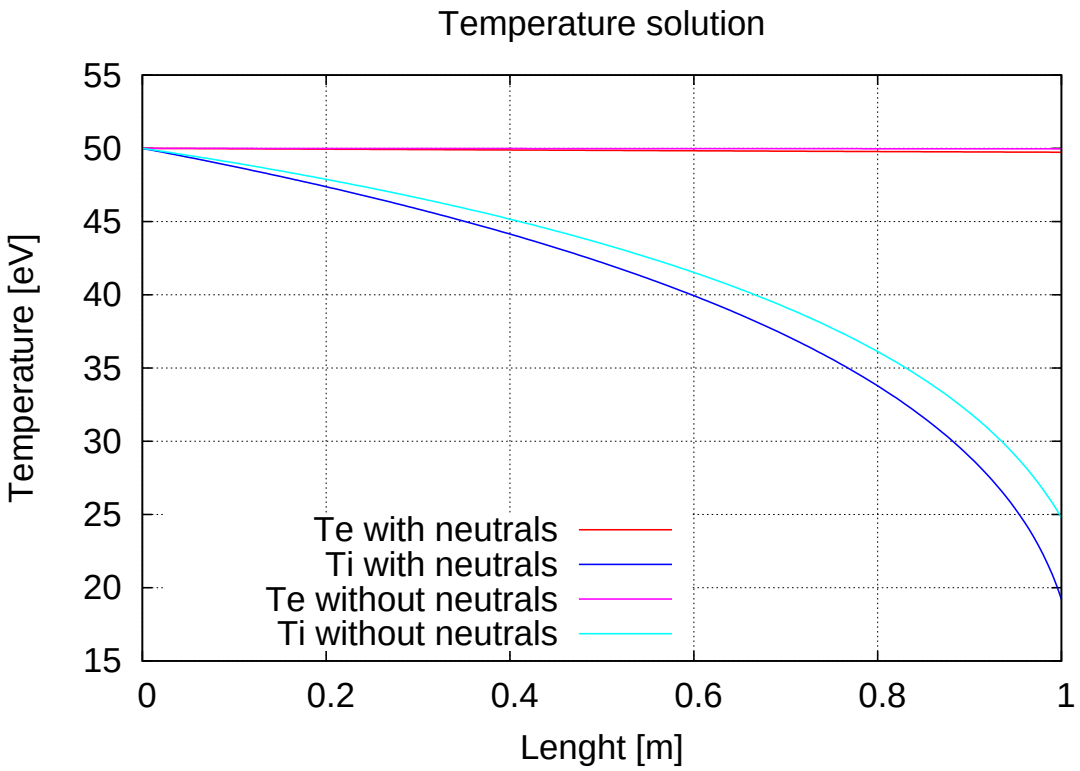

Figure 6.2: Temperature solution for a fixed temperature at the left boundary of $50eV$.

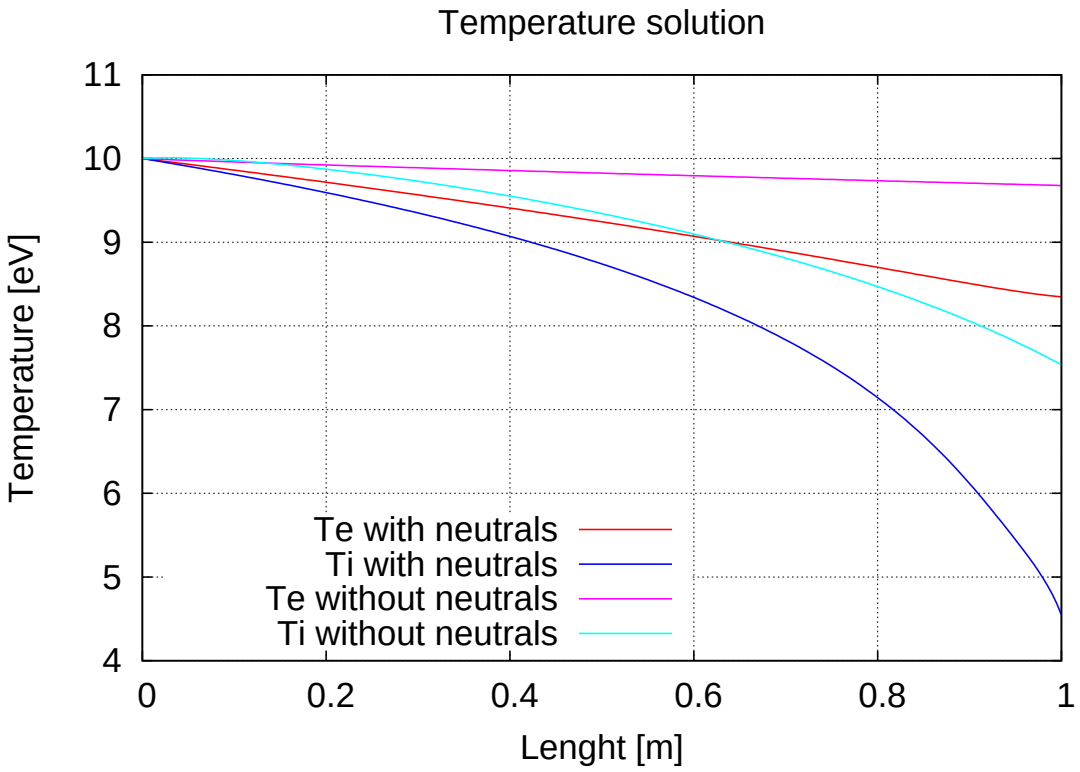

Figure 6.3: Temperature solution for a fixed temperature at the left boundary of $10eV$.

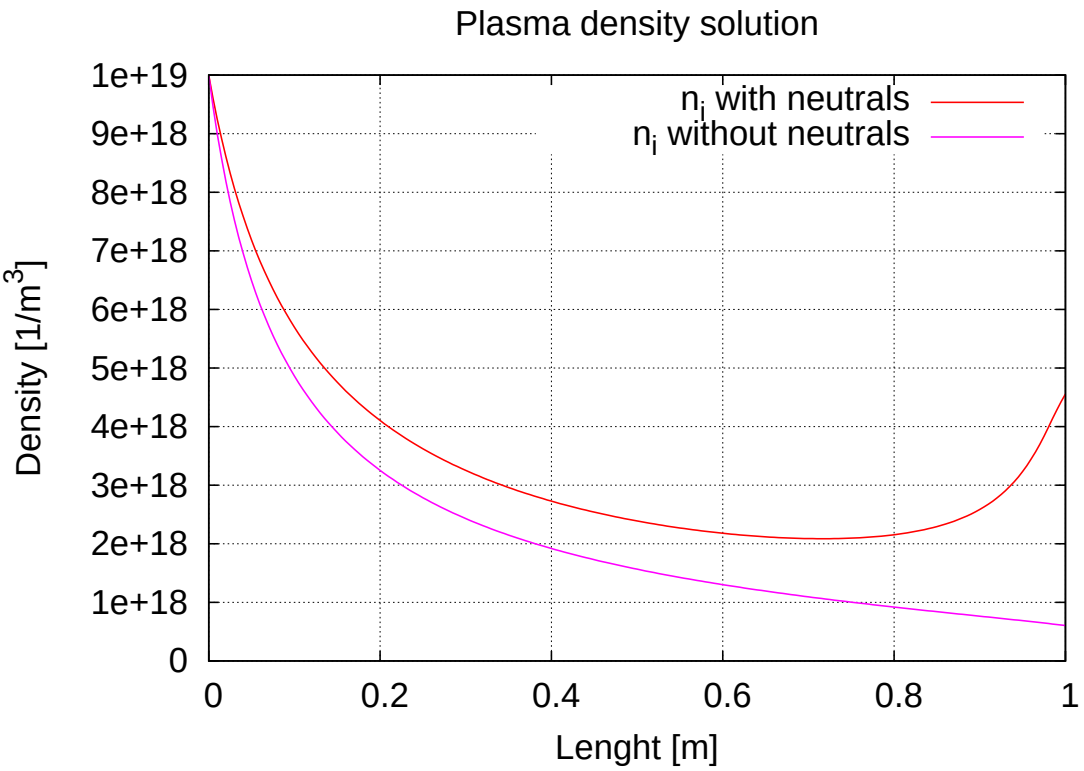

Figure 6.4: Plasma density solution for a fixed temperature at the left boundary of $50eV$.

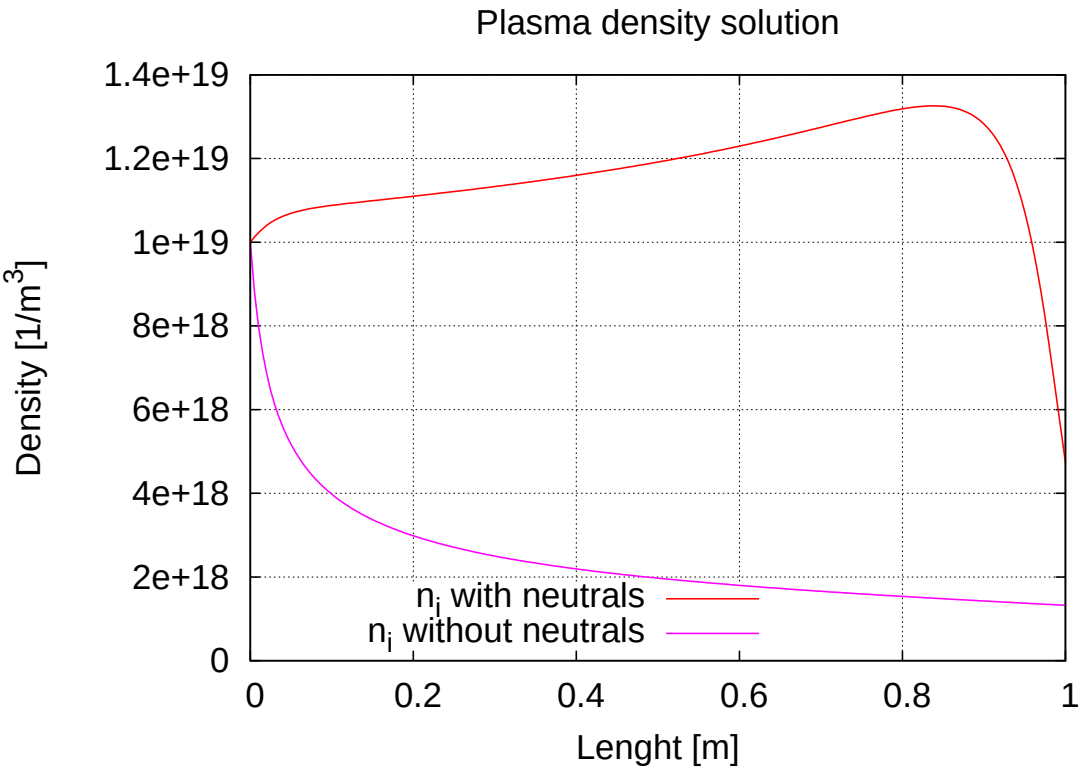

Figure 6.5: Plasma density solution for a fixed temperature at the left boundary of $10eV$.

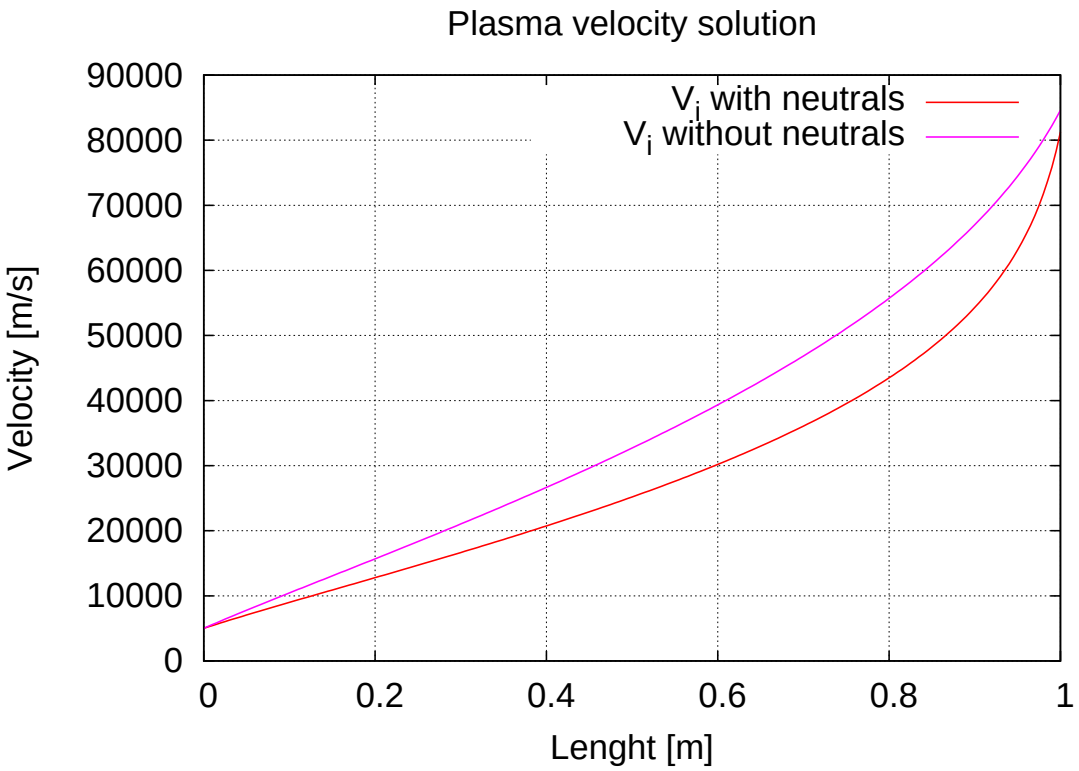

Figure 6.6: Plasma velocity solution for a fixed temperature at the left boundary of $50eV$.

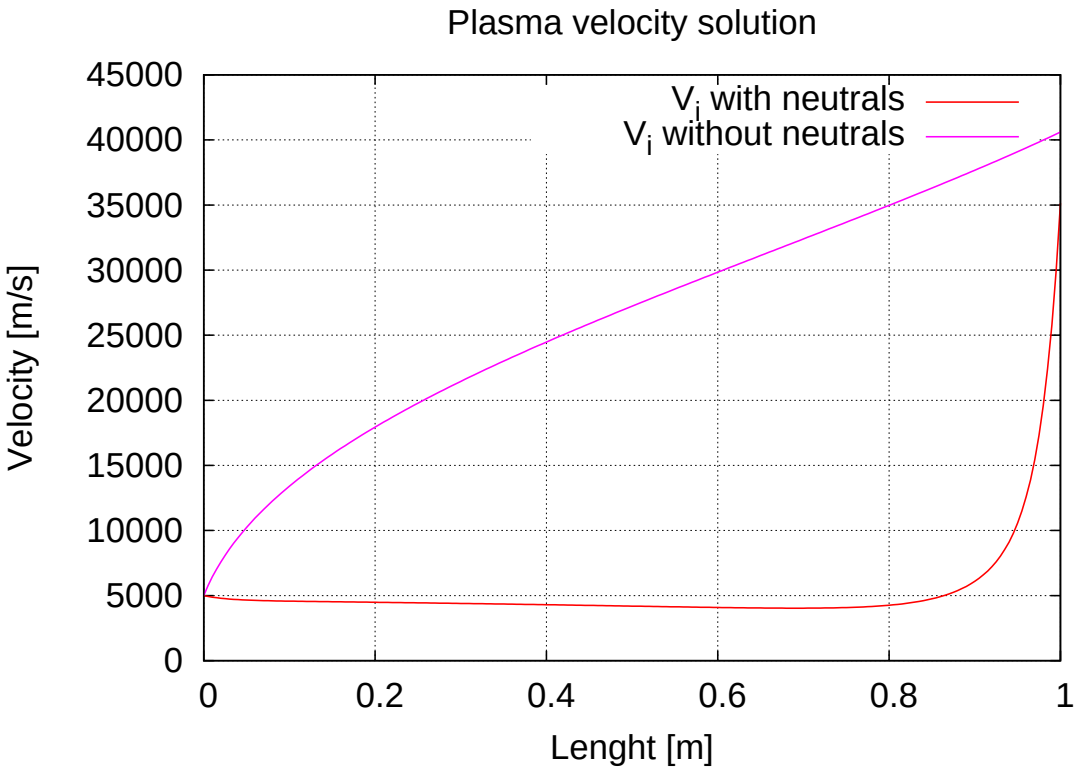

Figure 6.7: Plasma velocity solution for a fixed temperature at the left boundary of $10eV$.

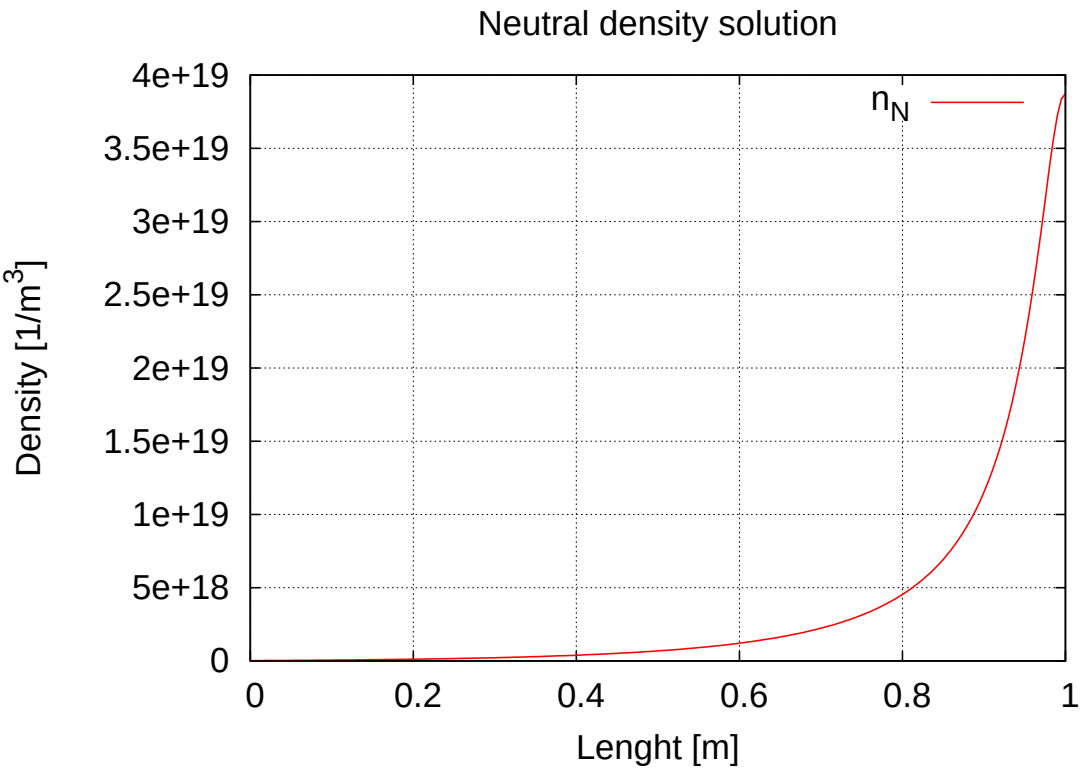

Figure 6.8: Neutral density solution for a fixed temperature at the left boundary of $50eV$.

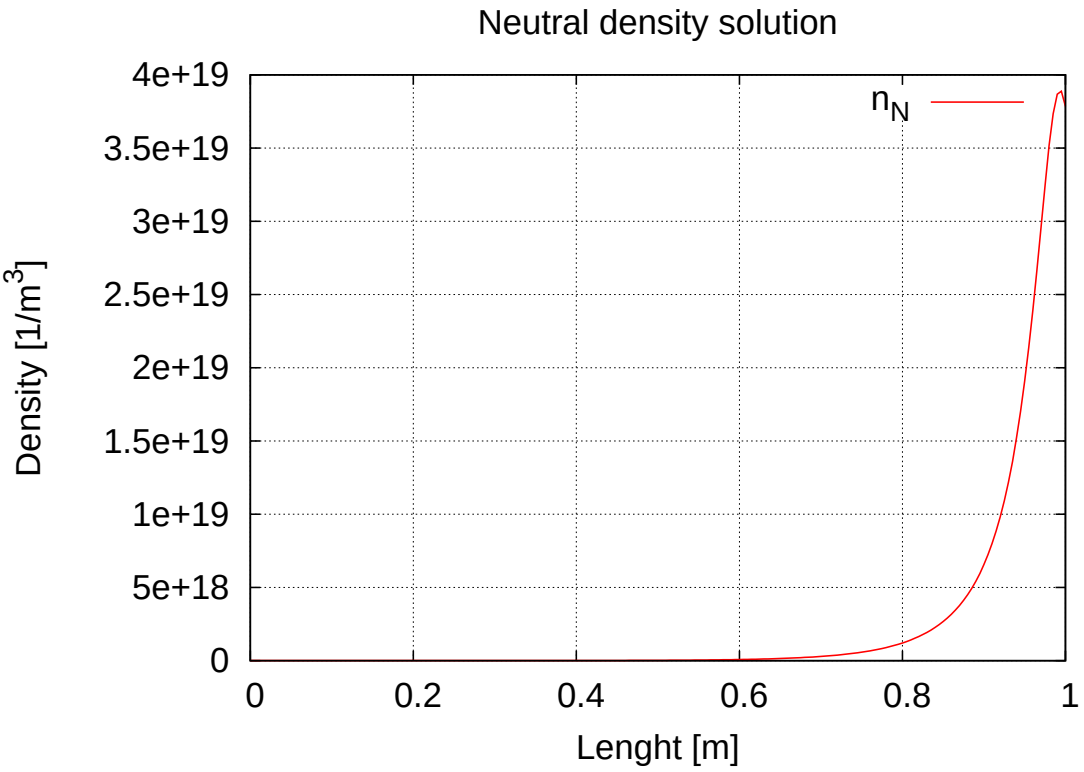

Figure 6.9: Neutral density solution for a fixed temperature at the left boundary of $10eV$.

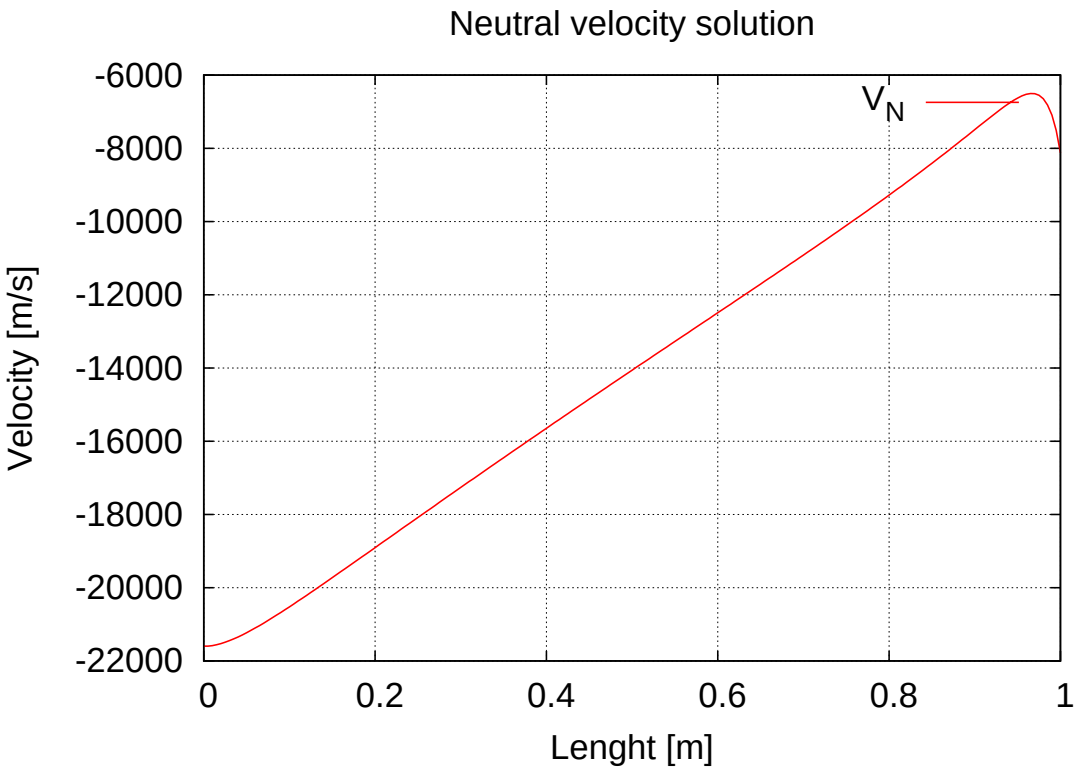

Figure 6.10: Neutral velocity solution for a fixed temperature at the left boundary of $50eV$.

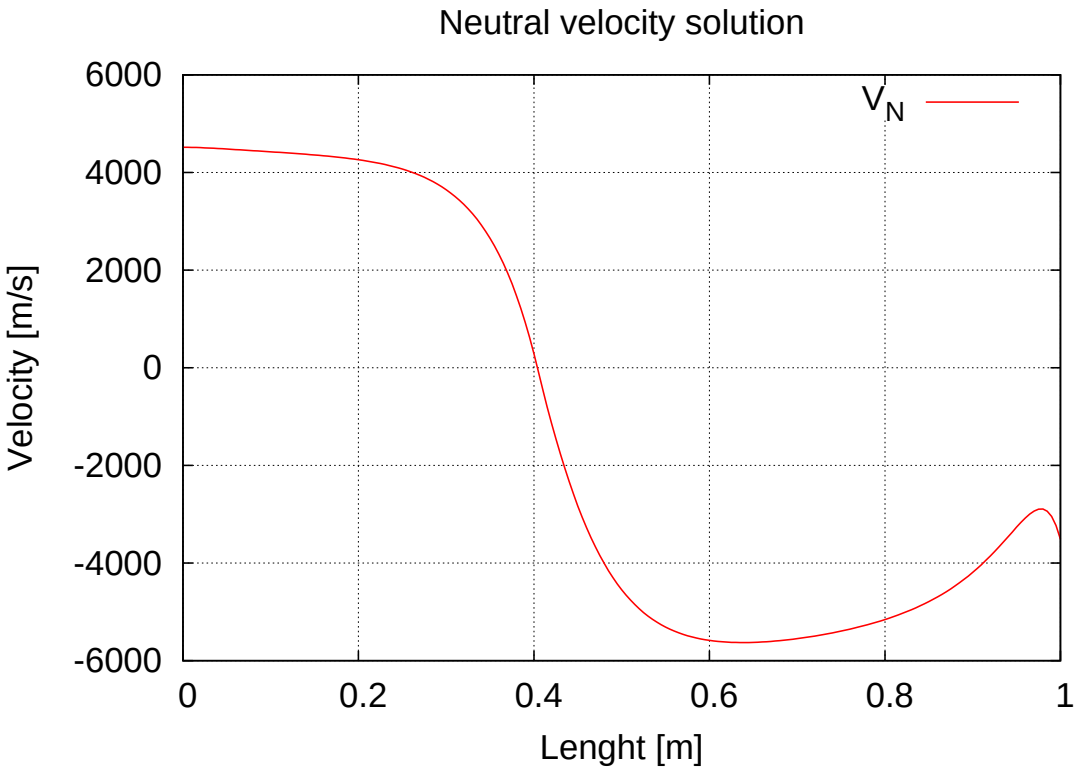

Figure 6.11: Neutral velocity solution for a fixed temperature at the left boundary of $10eV$.

Chapter 7

Conclusions

The present work was concerned with developing a finite difference code for modeling the physics of the plasma edge in fusion devices. It allows the treatment of a complex 3D geometry and an ergodic structure of field lines in the plasma edge, like in W7-X and TEXTOR-DED, using a finite difference code approach. The code described in the present work requires only a magnetic field configuration data file and specified boundary conditions in order to generate solutions of the plasma transport equations. The concept of local magnetic coordinates was used to handle the different geometries consistently. The pre-processing steps of the general algorithm are grid construction, finding poloidal neighbours and calculation of metric coefficients. For this, a general tracing procedure was developed in a sequential and a parallel version to solve the system of ordinary differential equations for the different local magnetic coordinate systems (LCMS) needed for the discretization stencil.

The neighbours are found with a shadowing algorithm. To solve the coupled set of transport equations in the LCMS representation a solver for conduction-convection equations using a "semi-explicit" generalized upwind scheme for non-staggered grids was developed. For the parallel direction all terms are treated implicitly, whereas the rest of the terms are calculated explicitly. This allows to reduce the amount of memory needed for the calculations and also to use simple tridiagonal and cyclic tridiagonal solvers which are extremely fast. At the same time the procedure remains robust enough due to the fact that the dominant process of the parallel transport is calculated implicitly. The non-parallel terms are not expressed as a divergence of the flux. Instead, all derivatives are derived with the help of the "free point" method. This conduction-convection solver was successfully validated with analytical solutions and benchmarked with two other codes.

The key physics question addressed in the present work is the problem of power loads at the divertor plates, including the influence of ergodicity and 3D configuration of the field on the heat transport. The study of conductive heat transport is the first step towards the analysis of the heat flux pattern, because for temperatures above about 50 eV the parallel heat conduction is usually dominant and convection terms or neutral sources can be neglected (at least for electrons). A W7-X divertor case was studied as a reference case for the purely conductive problem. The target plates cut through the islands and lead to large losses to the target plates resulting in much lower temperatures at the target plates and a different solution than for a limiter case. The heat flux pattern on the target plates reflects for electrons the spatial distribution of the connection length of the field lines hitting the plate. The maximum values are clearly correlated with the longest field line lengths. The radial decay of the electron profile is faster than the ion profile, because the parallel heat conduction of electrons is much larger than of ions (due to the mass difference) and therefore the parallel loss is stronger.

The energy transport in the complex 3D TEXTOR-DED tokamak geometry with large ergodic regions was studied, where the plasma transport is closely related to the structure of the magnetic field lines. The magnetic topology is closest reflected in the solution of the electron temperature field at low densities, which indicates that the parallel transport is dominant. If the density increases, the heat conductivity is reduced forcing development of temperature gradients along field lines, in addition, the role of the radial transport increases leading to the stronger mixing of temperatures from different regions, and consequently the mesh structures are less

pronounced.

However, the hot spot structure of the temperature distribution close to the target is always preserved. It appears that the maximum of the energy flow is concentrated in very well localized areas elongated radially from the core to the target, because the heat preferable goes through the edges of an island. heat is preferably transported to the target by field lines with longer connection length. In case of the mode 6/2, there is a tendency that the heat delivered to the target increases with the field line length. However, in the case of the mode 12/4, such conclusion is only true for magnetic field lines, which have relatively short connection length, less than about connection length shorter than 40 toroidal turns (roughly 10 poloidal turns). For longer field lines the heat transmitted to the target does not depend on the field line length for all densities. This result is not confirmed by experimental findings, where the heat flux brought to the target by field lines with long connection length is reduced at higher densities. This suggests that other terms, like convective transport might be important.

Therefore, the effect of convective terms are discussed for a W7-X divertor case and compared with the purely conductive case. This allows to identify the importance of the convective effects for the power load patterns at the target plates. Electrons do not change much, but just get larger values due to the additional convective energy. Their heat flux density distribution is mostly determined by the field line lengths as in the purely conductive case. In contrast, the ion heat flux density now shows a quite different pattern than for the conductive case: it is strongly determined by the convective heat flux and is more homogeneous than in the conductive case. Also, it is broader than the electron heat flux distribution. The coincidence of the maxima of electron heat flux, ion heat flux and particle flux densities in the target patterns also show the importance of the convective heat fluxes and the need for including them into the analysis of heat flux distributions. Also, in contrast to the pure conductive case, ions contribute as much to the heat flux densities as electrons.

A neutral fluid model was coupled to the plasma fluid equations to study the effects of neutrals. In 1D, the correct implementation of the neutral fluid model into the code was validated. However, a full run of W7-X including neutrals and ions was not possible within this work, because there is a severe resolution problem. The metrics requires a very fine resolution as also the neutral dynamics does due to the fast profile decay on the length scale of the neutral mean free path. Therefore, only a parallel code will be able to treat such a case, which was beyond the scope of this work.

In summary, a new finite difference code for general plasma edge transport problems was successfully developed and first applications analyzing the heat flux patterns of W7-X and TEXTOR-DED were presented. The full capability of this concept will require further numerical developments of the code according to concepts presented already in this thesis: a domain-decomposition based parallel version will have to be created to satisfy the resolution requirements of the metrics and neutral dynamics.

Appendix A

Implicit algorithm of the Free Point Method

The Free point method was proposed by Dyachenko for the simulation of the plasma focus dynamics. It is an unstructured grid method suitable for large deformation problems. An extended variant of this method has been developed by Jach and Stepniewski.

The idea of the method is based on the assumption, that with each mesh point a set of neighbouring points is associated. These neighbour points should be spread more or less homogeneously around the central point and they can be changed during the evolution of the system. On the basis of the information collected from the neighbours one can find the first and second derivatives.

Let us consider a point i and the set of its neighbours, defined by points $f_1, f_2, ...f_{m(i)}$ ($m(i)$ stands for the number of neighbours of point i). The first derivatives of the two dimensional function $f(x, y)$ are computed by the least squares method based on finding the minimum of the following functional:

$$A = \sum_k \left(f_k - f_i - f_x \triangle x_k - f_y \triangle y_k\right)^2 w_k^2, \tag{A.1}$$

where $k = 1, 2, ..., m(i)$. In the following, we use only sum over (k) without specifying the limits. The condition for the minimum of the functional A is:

$$\frac{\partial A}{\partial f_x} = 0; \frac{\partial A}{\partial f_y} = 0, \tag{A.2}$$

giving us two equations for the unknown first derivatives f_x, f_y of function f,i.e.

$$\sum_k \left(f_k - f_i - f_x \triangle x_k - f_y \triangle y_k\right) w_k^2 \triangle x_k = 0 \tag{A.3}$$

$$\sum_k \left(f_k - f_i - f_x \triangle x_k - f_y \triangle y_k\right) w_k^2 \triangle y_k = 0. \tag{A.4}$$

$$\tag{A.5}$$

The above system of two linear algebraic equations can be written as:

$$\begin{bmatrix} X2 & XY \\ XY & Y2 \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} FX \\ FY \end{bmatrix}, \tag{A.6}$$

where

$$X2 = \sum_k \triangle x_k^2 w_k^2;$$
$$Y2 = \sum_k \triangle y_k^2 w_k^2;$$
$$XY = \sum_k \triangle x_k \triangle y_k w_k^2;$$
$$FX = \sum_k (f_k - f_i) \triangle x_k w_k^2;$$
$$FY = \sum_k (f_k - f_i) \triangle y_k w_k^2.$$

By solving this system we find the first derivatives:

$$f_x = \frac{FX \cdot Y2 - FY \cdot XY}{DET}; f_y = \frac{FY \cdot X2 - FX \cdot XY}{DET}, \tag{A.7}$$

where $DET = X2 \cdot Y2 - (XY)^2$. The derivatives of function $f(x, y)$, f_x, f_y at mesh point i can be written as a linear combination of values of the function f in the central and neighbouring points.

$$f_x = \sum_k A_x^k \left(f_k - f_i\right) \tag{A.8}$$
$$f_y = \sum_k A_y^k \left(f_k - f_i\right),$$

where coefficients A_x^k, A_y^k are:

$$A_x^k = \frac{w_k^2}{DET} \left(Y2 \cdot \triangle x_k - XY \cdot \triangle y_k\right)$$
$$A_y^k = \frac{w_k^2}{DET} \left(X2 \cdot \triangle y_k - XY \cdot \triangle x_k\right).$$

Finally, we have

$$f_x = -f_i \sum_k A_x^k + \sum_k A_x^k \cdot f_k$$
$$f_y = -f_i \sum_k A_y^k + \sum_k A_y^k \cdot f_k.$$

From these expressions we can find the contribution from each neighbour to the derivatives f_x and f_y. The corresponding metric elements are:

$$[f_x] \Rightarrow [f_x]_{ii} = -\sum_k A_x^k$$
$$[f_x]_{ik} = A_x^k$$

$$[f_y] \Rightarrow [f_y]_{ii} = -\sum_k A_y^k$$
$$[f_y]_{ik} = A_y^k$$

The problem of finding the second derivatives can be split into two steps. First, the first derivatives are found as described above, and, then, in each point we define the functional being the Taylor expansion of the function $f(x, y)$ in the vicinity of point i:

$$B = \sum_k \left[(f_k - f_i - f_x \triangle x_k - f_y \triangle y_k) - \frac{1}{2} \left(f_{xx} \triangle x_k^2 + 2 f_{xy} \triangle x_k \triangle y_k + f_{yy} \triangle y_k^2 \right) \right]^2 w_k^4. \quad \text{(A.9)}$$

Let us introduce $\varepsilon_k = f_k - f_i - f_x \triangle x_k - f_y \triangle y_k$ being the difference between value f_k and its linear approximation (residual at point k). Now the functional B can be written as

$$B = \sum_k \left[\varepsilon_k - \frac{1}{2} \left(f_{xx} \triangle x_k^2 + 2 f_{xy} \triangle x_k \triangle y_k + f_{yy} \triangle y_k^2 \right) \right]^2 w_k^4, \quad \text{(A.10)}$$

and we can evaluate the residual ε_k:

$$\begin{aligned} \varepsilon_k &= f_k - f_i - \sum_j A_x^j \left(f_j - f_i \right) \triangle x_k - \sum_j A_y^j \left(f_j - f_i \right) \triangle y_k \\ &= f_k - f_i \cdot \left[1 - \sum_j \left(A_x^j \triangle x_k + A_y^j \triangle y_k \right) \right] - \sum_j f_j \left(A_x^j \triangle x_k + A_y^j \triangle y_k \right) \\ &= -f_i \cdot \left[1 - \sum_j \left(A_x^j \triangle x_k + A_y^j \triangle y_k \right) \right] - \sum_j f_j \left(A_x^j \triangle x_k + A_y^j \triangle y_k - \delta_{jk} \right), \end{aligned} \quad \text{(A.11)}$$

where δ_{jk} is the Kronecker symbol. Second derivatives are found from the conditions for the minimum of the functional B:

$$\begin{aligned} \frac{\partial B}{\partial f_{xx}} &= 0; \\ \frac{\partial B}{\partial f_{2xy}} &= 0; \\ \frac{\partial B}{\partial f_{yy}} &= 0. \end{aligned}$$

Thus,

$$\begin{aligned} \sum_k \triangle x_k^2 \left[\varepsilon_k - \frac{1}{2} \left(f_{xx} \triangle x_k^2 + 2 f_{xy} \triangle x_k \triangle y_k + f_{yy} \triangle y_k^2 \right) \right] w_k^4 &= 0 \\ \sum_k \triangle x_k \triangle y_k \left[\varepsilon_k - \frac{1}{2} \left(f_{xx} \triangle x_k^2 + 2 f_{xy} \triangle x_k \triangle y_k + f_{yy} \triangle y_k^2 \right) \right] w_k^4 &= 0 \\ \sum_k \triangle y_k^2 \left[\varepsilon_k - \frac{1}{2} \left(f_{xx} \triangle x_k^2 + 2 f_{xy} \triangle x_k \triangle y_k + f_{yy} \triangle y_k^2 \right) \right] w_k^4 &= 0 \end{aligned}$$

and the set of algebraic equations for the unknown derivatives f_{xx}, f_{xy}, f_{yy} finally looks as follows:

$$\begin{aligned} f_{xx} \sum_k \triangle x_k^4 w_k^4 + 2 f_{xy} \sum_k \triangle x_k^3 \triangle y_k w_k^4 + f_{yy} \sum_k \triangle x_k^2 \triangle y_k^2 w_k^4 &= 2 \sum_k \varepsilon_k \triangle x_k^2 w_k^4, \\ f_{xx} \sum_k \triangle x_k^3 \triangle y_k w_k^4 + 2 f_{xy} \sum_k \triangle x_k^2 \triangle y_k^2 w_k^4 + f_{yy} \sum_k \triangle x_k \triangle y_k^3 w_k^4 &= 2 \sum_k \varepsilon_k \triangle x_k \triangle y_k w_k^4, \\ f_{xx} \sum_k \triangle x_k^2 \triangle y_k^2 w_k^4 + 2 f_{xy} \sum_k \triangle x_k \triangle y_k^3 w_k^4 + f_{yy} \sum_k \triangle y_k^4 w_k^4 &= 2 \sum_k \varepsilon_k \triangle y_k^2 w_k^4. \end{aligned}$$

Let us introduce for simplification:

$$
\begin{aligned}
X4 &= \sum_k \triangle x_k^4 w_k^4 \\
Y4 &= \sum_k \triangle y_k^4 w_k^4 \\
X3Y &= \sum_k \triangle x_k^3 \triangle y_k w_k^4 \\
X2Y2 &= \sum_k \triangle x_k^2 \triangle y_k^2 w_k^4 \\
XY3 &= \sum_k \triangle x_k \triangle y_k^3 w_k^4 \\
FX2 &= 2\sum_k \varepsilon_k \triangle x_k^2 w_k^4 \\
FY2 &= 2\sum_k \varepsilon_k \triangle y_k^2 w_k^4 \\
FXY &= 2\sum_k \varepsilon_k \triangle x_k \triangle y_k w_k^4.
\end{aligned}
$$

and the set of equations can be rewritten as:

$$
\begin{bmatrix} X4 & X3Y & X2Y2 \\ X3Y & X2Y2 & XY3 \\ X2Y2 & XY3 & Y4 \end{bmatrix} \begin{bmatrix} f_{xx} \\ 2f_{xy} \\ f_{yy} \end{bmatrix} = \begin{bmatrix} FX2 \\ FXY \\ FY2 \end{bmatrix}. \tag{A.12}
$$

The solution of the three algebraic equations has the form:

$$
\begin{aligned}
f_{xx} &= FX2 \cdot A1 + FXY \cdot A2 + FY2 \cdot A3 \\
2f_{xy} &= FX2 \cdot B1 + FXY \cdot B2 + FY2 \cdot B3 \\
f_{yy} &= FX2 \cdot C1 + FXY \cdot C2 + FY2 \cdot C3,
\end{aligned}
$$

where

$$A1 = \frac{1}{DET3}\begin{vmatrix} X2Y2 & XY3 \\ XY3 & Y4 \end{vmatrix}$$
$$A2 = \frac{-1}{DET3}\begin{vmatrix} X3Y & X2Y2 \\ XY3 & Y4 \end{vmatrix}$$
$$A3 = \frac{1}{DET3}\begin{vmatrix} X3Y & X2Y2 \\ X2Y2 & XY3 \end{vmatrix}$$
$$B1 = \frac{1}{DET3}\begin{vmatrix} X3Y & XY3 \\ X2Y2 & Y4 \end{vmatrix}$$
$$B2 = \frac{-1}{DET3}\begin{vmatrix} X4 & X2Y2 \\ X2Y2 & Y4 \end{vmatrix}$$
$$B3 = \frac{1}{DET3}\begin{vmatrix} X4 & X2Y2 \\ X3Y & XY3 \end{vmatrix}$$
$$C1 = \frac{1}{DET3}\begin{vmatrix} X3Y & X2Y2 \\ X2Y2 & XY3 \end{vmatrix}$$
$$C2 = \frac{-1}{DET3}\begin{vmatrix} X4 & X3Y \\ X2Y2 & XY3 \end{vmatrix}$$
$$C3 = \frac{1}{DET3}\begin{vmatrix} X4 & X3Y \\ X3Y & X2Y2 \end{vmatrix}$$
$$DET3 = \begin{vmatrix} X4 & X3Y & X2Y2 \\ X3Y & X2Y2 & XY3 \\ X2Y2 & XY3 & Y4 \end{vmatrix}.$$

Now, introducing

$$FX2 = \sum_k \varepsilon_k \gamma_k^{xx},$$
$$FY2 = \sum_k \varepsilon_k \gamma_k^{yy}$$
$$FXY = \sum_k \varepsilon_k \gamma_k^{xy},$$

where

$$\gamma_k^{xx} = 2\left(\triangle x_k\right)^2 \cdot w_k^4,$$
$$\gamma_k^{yy} = 2\left(\triangle y_k\right)^2 \cdot w_k^4,$$
$$\gamma_k^{xy} = 2\triangle x_k \triangle y_k \cdot w_k^4,$$

we can write the solution in the following form

$$f_{xx} = \sum_k \varepsilon_k \left(A1 \cdot \gamma_k^{xx} + A2 \cdot \gamma_k^{xy} + A3 \cdot \gamma_k^{yy}\right),$$
$$f_{xy} = \sum_k \varepsilon_k \left(B1 \cdot \gamma_k^{xx} + B2 \cdot \gamma_k^{xy} + B3 \cdot \gamma_k^{yy}\right),$$
$$f_{yy} = \sum_k \varepsilon_k \left(C1 \cdot \gamma_k^{xx} + C2 \cdot \gamma_k^{xy} + C3 \cdot \gamma_k^{yy}\right),$$

or by denoting

$$\begin{aligned}\alpha_k^{xx} &= A1\cdot\gamma_k^{xx} + A2\cdot\gamma_k^{xy} + A3\cdot\gamma_k^{yy},\\ \alpha_k^{xy} &= B1\cdot\gamma_k^{xx} + B2\cdot\gamma_k^{xy} + B3\cdot\gamma_k^{yy},\\ \alpha_k^{yy} &= C1\cdot\gamma_k^{xx} + C2\cdot\gamma_k^{xy} + C3\cdot\gamma_k^{yy},\end{aligned}$$

the solution can be written as

$$\begin{aligned}f_{xx} &= \sum_k \varepsilon_k\alpha_k^{xx},\\ f_{xy} &= \sum_k \varepsilon_k\alpha_k^{xy},\\ f_{yy} &= \sum_k \varepsilon_k\alpha_k^{yy}.\end{aligned}$$

Let us first consider the term f_{xx}. Substituting here Eq. (A.11) for ε_k we can write:

$$\begin{aligned}f_{xx} &= \sum_k \varepsilon_k\alpha_k^{xx}\\ &= \sum_k \alpha_k^{xx}\cdot\left[-f_i\left(1-\triangle x_k\sum_j A_x^j-\triangle y_k\sum_j A_y^j\right)-\sum_j f_j\left(\triangle x_k A_x^j+\triangle y_k A_y^j-\delta_{jk}\right)\right]\\ &= -f_i\cdot\sum_k\alpha_k^{xx}+f_i\cdot\sum_k\alpha_k^{xx}\triangle x_k\left(\sum_j A_x^j\right)+f_i\cdot\sum_k\alpha_k^{xx}\triangle y_k\left(\sum_j A_y^j\right)\\ &- \left(\sum_k\alpha_k^{xx}\triangle x_k\right)\sum_j f_j A_x^j-\left(\sum_k\alpha_k^{xx}\triangle y_k\right)\sum_j f_j A_y^j+\sum_k\alpha_k^{xx}f_k.\end{aligned}$$

From the above expression we can find the contribution from each neighbour to the second derivative f_{xx}. The correspondent metric elements are:

$$\begin{aligned}[f_{xx}]\Rightarrow[f_{xx}]_{ii} &= -\sum_k\alpha_k^{xx}+\sum_k\alpha_k^{xx}\triangle x_k\left(\sum_j A_x^j\right)+\sum_k\alpha_k^{xx}\triangle y_k\left(\sum_j A_y^j\right),\\ [f_{xx}]\Rightarrow[f_{xx}]_{ik} &= \alpha_k^{xx}-A_x^k\cdot\left(\sum_j\alpha_j^{xx}\triangle x^j\right)-A_y^k\cdot\left(\sum_j\alpha_j^{xx}\triangle y^j\right).\end{aligned}$$

Expressions fro f_{xy}, f_{yy} can be easily found from the above equations by simply replacing α_k^{xx} by α_k^{xy} and α_k^{yy}, respectively.

Bibliography

[1] R. Schneider. *Plasma Edge Physics for Tokamaks*. IPP-Report 12/1 Max-Planck-Institut für Plasmaphysik, Greifswald, Germany, 2001.

[2] P.C. Stangeby. The Bohm–Chodura plasma sheath criterion. *Physics of Plasmas*, 2(3):702–706, 1995.

[3] P.C. Liewer. Measurements of microturbulence in tokamaks and comparisons with theories of turbulence and anomalous transport. *Nuclear Fusion*, 25(5):543–621, 1985.

[4] J.D. Lawson. Some criteria for a power producing thermonuclear reactor. In *Proceedings of the Physical Society*, volume B70, page 6, 1957.

[5] J.A. Wesson. *Tokamaks*. Clarendon Press, Oxford, 1987.

[6] K. Itoh and S.-I. Itoh. The role of electric field in confinement. *Plasma Physics and Controlled Fusion*, 38(1):1–49, 1996.

[7] K. Ida. Experimental studies of the physical mechanism determining the radial electrical field and its radial structure in a toroidal plasma. *Plasma Physics and Controlled Fusion*, 40(8):1429–1488, 1998.

[8] D. Reiter, G.H. Wolf, and H. Kever. Burn condition, helium particle confinement and exhaust efficiency. *Nuclear Fusion*, 30(10):2141–2155, 1990.

[9] D.E. Post and K. Lackner. Plasma models for impurity control experiments. In D.E. Post and R. Behrisch, editors, *Physics of Plasma Wall Interaction in Controlled Fusion (Proc. NATO Workshop ValMorin, Quebec, Canada)*, pages 627–693, New York, 1984. Plenum Press.

[10] D.P. Coster. Divertor design: issues raised by steady state and advanced tokamak operation, 2000. Invited paper for the 2000 Toki Conference.

[11] A. Field, C. García–Rosales, D. Naujoks, C.S. Pitcher, G. Lieder, G. Fussmann, S. Hirsch, R. Radtke, U. Wenzel, and R. Neu. Studies of divertor target plate erosion in the ASDEX Upgrade tokamak. *Journal of Nuclear Materials*, 220–222:553–557, 1995. 11th PSI Mito, Japan, 23.–27. May 1994.

[12] C.S. Pitcher and P.C. Stangeby. Experimental divertor physics. *Plasma Physics and Controlled Fusion*, 39(6):779–930, 1997.

[13] H. Renner, J. Boscary, V. Erckmann, H. Greuner, H. Grote, J. Sapper, E. Speth, F. Wesner, M. Wanner, and W7–X team. The capabilities of steady state operation at the stellarator W7–X with emphasis on divertor design. *Nuclear Fusion*, 40:1083–1093, 2000.

[14] G. Grieger, W. Lotz, P. Merkel, J. Nührenberg, J. Sapper, and E. Strumberger. Physics optimization of stellarators. *Physics of Fluids*, B4:2081–2091, 1992.

[15] R. Chodura. Plasma flow in the sheath and presheath of a scarpe–off layer. In D.E. Post and R. Behrisch, editors, *Physics of Plasma Wall Interaction in Controlled Fusion (Proc. NATO Workshop Val–Morin, Quebec, Canada)*, pages 99–134, New York, 1984. Plenum Press.

[16] R. Behrisch. Surface erosion by electrical arcs. In D.E. Post and R. Behrisch, editors, *Physics of Plasma Wall Interaction in Controlled Fusion (Proc. NATO Workshop Val-Morin, Quebec, Canada)*, New York, 1984. Plenum Press.

[17] R. Chodura. Plasma flow in the sheath and presheath of a scrape–off layer. volume 131 of *NATO Advanced Science Institute*, pages 99–134, New York, 1986. NATO Science Committee, Plenum Press.

[18] P.C. Stangeby and A.V. Chankin. The ion velocity (Bohm–Chodura) boundary condition at the entrance to the magnetic presheath in the presence of diamagnetic and $\vec{E} \times \vec{B}$ drifts in the scrape–off layer. *Physics of Plasmas*, 2(3):707–715, 1995.

[19] P.C. Stangeby. *The Plasma Boundary of Magnetic Fusion Devices*. IOP, Bristol, 2000.

[20] S.I. Braginskii. Transport Processes in a Plasma. *Rev. Plasma Phys.*, 1, 1965.

[21] S.S. Abdullaev, K.H. Finken, Jakubowski M.W., S.V. Kasilov, M. Kobayashi, D. Reiser, D. Reiter, A.M. Runov, and R. Wolf. Overview of magnetic structure induced by the textor-ded and the related transport. *Nuclear Fusion*, 43:299–313, 2003.

[22] A. at al. Runov. Extensions of the 3-dimensional plasma transport code e3d. *Contributions to Plasma Physics*, 44, 2004.

[23] A. B. Rechester and M. N. Rosenbluth. Electron heat transport in a tokamak with destroyed magnetic surfaces. *Physical Review Letters*, 40(1):38–41, Jan 1978.

[24] P. Ghendrih, A. Grosman, and H. Capes. Theoretical and experimental investigations of stochastic boundaries in tokamaks. *Plasma Physics and Controlled Fusion*, 38(10):1653–1724, 1996.

[25] A. V. Nedospasov and M. Z. Tokar. Conception of divertorless tokamak reactor with turbulent plasma blanket. *Journal of Nuclear Materials*, 93-94, 1980.

[26] A. Grosman, Ph. Ghendrih, B. Meslin, and D. Guilhem. Power and particle flux to the neutraliser plates of the tore supra ergodic divertor modules. *Journal of Nuclear Materials*, 241-243, 1997.

[27] M. W. Jakubowski, S. S. Abdullaev, K. H. Finken, and M. the TEXTOR team Lehnen. Heat deposition patterns on the target plates of the dynamic ergodic divertor. *Journal of Nuclear Materials*, 337-339, 2005.

[28] T. E. Evans, R. K. W. Roeder, J. A. Carter, and B. I. Rapoport. Homoclinic tangles, bifurcations and edge stochasticity in diverted tokamaks. *Contributions to Plasma Physics*, 44, 2004.

[29] M. W. Jakubowski, S. S. Abdullaev, K. H. Finken, and the TEXTOR Team. Modelling of the magnetic field structures and first measurements of heat fluxes for textor-ded operation. *Nuclear Fusion*, 2004.

[30] P. Ghendrih, A. Grosman, and H. Capes. Theoretical and experimental investigations of stochastic boundaries in tokamaks. *Plasma Phys. Control. Fusion*, 38, 1996.

[31] M. Becoulet, H. Capes, and P. et al. Ghendrih. Experiments and 3d non-linear modelling of heat transport in ergodic zone on tore supra. *Contributions to Plasma Physics*, 40, 2000.

[32] T. Eich, D. Reiser, and K. H. Finken. Two dimensional modelling approach to transport properties of the textor-ded laminar zone. *Nuclear Fusion*, 40, 2000.

[33] Schmitz O. et al. Characterization of transport in the stochastic edge layer of textor by analysis of the radial and poloidal distribution of electron density and temperature. *Journal of Nuclear Materials*, 363-365, 2007.

[34] Jakubowski M. W. et al. Change of the magnetic-field topology by an ergodic divertor and the effect on the plasma structure and transport. *Physical Review Letters*, 2006.

[35] Jakubowski M. W. et al. Observation of the heteroclinic tangles in the heat flux pattern of the ergodic divertor at textor. *Journal of Nuclear Materials*, 363-365, 2007.

[36] F. Nguyen, P. Ghendrih, and A. Grosman. Interaction of stochastic boundary layer with plasma facing components. *Nuclear Fusion*, 1997.

[37] Jakubowski M. W. et al. *Plasma Physics and Controlled Fusion*, 2007.

[38] A.H. Boozer. Plasma equilibrium with rational magnetic surfaces. *Phys. Fluids*, 24, 1981.

[39] A.H. Boozer. Evaluation of the structure of ergodic fields. *Phys. Fluids*, 26, 1983.

[40] W.D. at al. D'haeseleer. *Flux Coordinates and Magnetic Field Structure / A Guide to a Fundamental Tool of Plasma Theory*. Springer-Verlag Berlin and Heidelberg GmbH, 1990.

[41] M. Borchardt, J. Riemann, R. Schneider, and X. Bonnin. W7–X edge modelling with the 3D SOL fluid code BoRiS. *Journal of Nuclear Materials*, 290–293(3):546–550, 2001.

[42] Y. Feng, F. Sardei, and J. Kisslinger. 3D fluid modelling of the edge plasma by means of a Monte Carlo technique. *Journal of Nuclear Materials*, 266–269:812–818, 1999.

[43] R. Schneider, M. Borchardt, J. Riemann, A. Mutzke, and S. Weber. Concept and status of a 3D SOL fluid code. *Contributions to Plasma Physics*, 40(3–4):340–345, 2000. 7th PET Conference, Tajimi, Gifu, Japan, 4–6 October, 1999.

[44] A.M. Runov, S. Kasilov, J. Riemann, M. Borchardt, D. Reiter, and R. Schneider. Benchmark of the 3–dimensional plasma transport codes E3D and BoRiS. *Contributions to Plasma Physics*, 42(2–4):169–174, 2002.

[45] A.M. Runov, S. Kasilov, D. Reiter, N. McTaggart, X. Bonnin, and R. Schneider. Transport in complex magnetic geometries: 3D modelling of ergodic edge plasmas in fusion experiments. *Journal of Nuclear Materials*, 313(3):1292–1297, 2003.

[46] R. Schneider, D.P. Coster, B.J. Braams, P. Xantopoulos, V.A. Rozhansky, S.P. Voskoboynikov, L. Kovaltsova, and H. Bürbaumer. B2–SOLPS5.0: SOL transport code with drifts and currents. *Contributions to Plasma Physics*, 40(3–4):328–333, 2000. 7th PET Conference, Tajimi, Gifu, Japan, 4–6 October, 1999.

[47] J. Riemann, M. Borchardt, R. Schneider, X. Bonnin, A. Mutzke, and T.D. Rognlien. Hierarchy tests of edge transport models (BoRiS, UEDGE). *Journal of Nuclear Materials*, 313(3):1030–1035, 2003.

[48] T.D. Rognlien, J.L. Milovich, M.E. Rensink, and G.D. Porter. A fully implicit, time dependent 2–D fluid code for modeling tokamak edge plasmas. *Journal of Nuclear Materials*, 196–198:347–351, 1992.

[49] R. Zagorski, H. Gerhauser, and H.A. Claassen. Numerical simulation of the TEXTOR edge plasma including drifts and impurities. *Contributions to Plasma Physics*, 38(1–2):61–66, 1998.

[50] P. Xanthopoulos and F. Jenko. Clebsch-type coordinates for nonlinear gyrokinetics in generic toroidal configurations. *Physics of Plasmas*, 13, 2006.

[51] R. Schneider, X. Bonnin, K. Borrass, D.P. Coster, H. Kastelewicz, D. Reiter, Rozhansky V.A., and B.J. Braams. Plasma edge physics with b2-eirene. *Contributions to Plasma Physics*, 46, 2006.

[52] Martine Baelmans. *Code improvements and Applications of a Two-dimensional Edge Plasma Model for Toroidal Devices*. PhD thesis, 1993.

[53] *Numerical recipies in C++: The Art of Scientific Computing*. 2002.

[54] W. Dorland, F. Jenko, M. Kotschenreuther, and B. N. Rogers. Electron temperature gradient turbulence. *Phys. Rev. Lett.*, 85(26):5579–5582, Dec 2000.

[55] F. Jenko, F. Dorland, M. Kotschenreuther, and B. N. Rogers. Electron temperature gradient driven turbulence. *Physics of Plasmas*, 7, 2000.

[56] S. P. Hirshman, U. Schwenn, and J. Nhrenberg. Improved radial differencing for three-dimensional magnetohydrodynamic equilibrium calculations. *Journal of Computational Physics*, 87, 1990.

[57] N. McTaggart. *Finite Difference Code for 3d Plasma Edge Modelling*. PhD thesis, Ernst-Moritz-Arndt Universitaet, Greifswald, Germany, 2004.

[58] *Sur la sphre vide*. Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk, 1934.

[59] V.F. Dyachenko and V. S. Imshennik. Two-Dimensional Magnetohydrodynamic Model of the Plasma Focus in the Z-Pinch(in Russian). *Voprosy Teorii Plazmy*, 8, 1974.

[60] S.V. Patankar. *Numerical Heat Transfer and Fluid Flow*. McGraw-Hill, New York, 1980.

[61] G. D. Thiart. Finite difference scheme for the numerical solution of fluid flow and heat transfer problems on nonstaggered grids. *Numerical Heat Transfer, Part B: Fundamentals An International Journal of Computation and Methodology*, 23, 1993.

[62] Dynamic ergodic divertor. special issue devoted to the ded. *Fusion Engineering and Design*, 37, 1997.

[63] Wolf R. C. et al. Effect of the dynamic ergodic divertor in the textor tokamak on mhd stability, plasma rotation and transport. *Nuclear Fusion*, 45, 2005.

[64] Koslowski H. R. et al. Dependence of the threshold for perturbation field generated m/n = 2/1 tearing modes on the plasma fluid rotation. *Nuclear Fusion*, 46, 2006.

[65] Gerhauser H. et al. Numerical modelling of pump limiter biasing on textor-94 and tore supra. *Nuclear Fusion*, 42, 2002.

[66] P. C. Stangeby and G. M. McCracken. Plasma boundary phenomena in tokamaks. *Nuclear Fusion*, 1990.

[67] A. M. Runov, D. Reiter, S. V. Kasilov, M. F. Heyn, and W. Kernbichler. Monte carlo study of heat conductivity in stochastic boundaries: Application to the textor ergodic divertor. *Physics of Plasmas*, 8, 2001.

[68] Kobayashi M. et al. 3d numerical transport study of the edge ergodized plasma in textorded. *Nuclear Fusion*, 44, 2004.

[69] T.D. Rognlien, P.N. Brown, R.B. Campbell, T.B. Kaiser, D.A. Knoll, P.R. McHugh, G.D. Porter, M.E. Rensink, and G.R. Smith. 2–D fluid transport simulations of gaseous/radiative divertors. *Contributions to Plasma Physics*, 34(2–3):362–367, 1994. 4th Workshop on Plasma Edge Theory, October 1993, Varenna, Italy.

[70] F. Wising, D.A. Knoll, S.I. Krasheninnikov, T.D. Rognlien, and D.J. Sigmar. Simulation of detachment in ITER–geometry using the UEDGE code and a fluid neutral model. *Contributions to Plasma Physics*, 36(2–3):309–313, 1996.

[71] M.V. Umansky and T.D. et al. Rognlien. Modeling of localized neutral particle sources in 3D edge plasmas. *Journal of Nuclear Materials*, 2003.

Acknowledgments

First, I sincerely thank my advisor Dr. Ralf Schneider, for his continuous support, encouragement and inspiration. He is the best advisor that i have ever had. He taught me how to approach a research problem and express my ideas. Thank to him I have learned what type of the scientist I would like to become and what level I must aim for.

I also would like to thank Dr. Roman Zagorski for his hospitality during my visit to Poland and Garching and for patience and insistence during our discussions.

I am deeply indebted to Dr. Pavlos Xantopolous, who always inspired and supported me. Thank to him the idea of the unified TRACER code has become the reality.

I would like to say the words of appreciation to Dr. Alexei Runov for our discussions and support.

A special thank to Jutta Gauger for her invaluable help in making my life in Germany easier and for her endless patience while helping me.

Many thanks to the Stellarator Theory group and at Institute für Physik, Greifswald University for being friendly and collaborative.

Finally, I want to thank all my friends and family, and, especially, my wife, Tanya.

Hiermit erkläre ich, daß diese Arbeit bisher von mir weder an der Mathematisch-Naturwissenschaftlichen Fakultät der Ernst-Moritz-Arndt-Universität Greifswald noch einer anderen wissenschaftlichen Einrichtung zum Zwecke der Promotion eingereicht wurde.
Ferner erkläre ich, daß ich diese Arbeit selbständig verfaßt und keine anderen als die darin angegebenen Hilfsmittel benutzt habe.

Greifswald, im Juli 2008

(Oleksandr Kalentev)

List of Publications

1. "3D numerical simulations of energy transport in the stochastic boundary of TEXTOR-DED with a finite difference method", R. Zagórski, M.W. Jakubowski, O. Kalentev, O. Schmitz, R. Schneider and W. Stepniewski Nucl. Fusion 48 (2008).

2. "Validation and application of numerically generated coordinate systems in gyrokinetic simulations", P. Xanthopoulos, D. Mikkelsen, F. Jenko, W. Dorland and O. Kalentev, submitted to Physics of Plasmas, July 2008.

Oleksandr Kalentev

Date of birth:
11.06.1979 in Kharkov, Ukraine

Family status
Married , since 04.06.2005 to Tetyana Ivanovska.

Education:
Masters (Specialist): Karazin Kharkov National University, Ukraine, 1996 - 2002.

Positions:
TREDEX Co. – private company, Kharkov, Ukraine, programmer,2002-2004.
Program-Ace – private company, Kharkov, Ukraine, programmer, team leader,2004-2005.
Max–Planck–Institut für Plasmaphysik, Greifswald, Germany, Doktorand, 2005 - present.

Work description:

- Developing and maintaining a program for digital analysis of encephalograthic mesuarements.
- 3D games programming.
- Developing a finite difference 3D plasma transport code as part of the Ph.D work at Max–Planck–Institut für Plasmaphysik, Greifswald, Germany.

Scholarships:

- Member of the International Max Planck Research School on Bounded Plasmas (IMPRS) Greifswald, Germany, since August 2005.