IPP-Report

Max-Planck-Institut
für Plasmaphysik

Menglong Zhao

Towards a More Complete Description of Tokamak Edge Plasmas: SOLPS Simulations with Kinetic Electron Effects

IPP 2018-10
Mai 2018

Max-Planck-Institut für Plasmaphysik

Towards a More Complete Description of Tokamak Edge Plasmas: SOLPS Simulations with Kinetic Electron Effects

Menglong Zhao

Vollständiger Abdruck der von der
Fakultät für Physik
der Technischen Universität München zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
genehmigten Dissertation.

Vorsitzende: Prof. Dr. Katharina Krischer

Prüfende/-r der Dissertation:

1. Prof. Dr. Ulrich Stroth

2. Prof. Dr. Ulrich Gerland

Die Dissertation wurde am 21.02.2018 bei der Technischen Universität München
eingereicht und durch die

Fakultät für Physik am 12.04.2018 angenommen.

PhD Thesis in Physics at
Technische Universität München

Towards a More Complete Description of Tokamak Edge Plasmas: SOLPS Simulations with Kinetic Electron Effects

Menglong Zhao

January 26, 2018

First Tutor: Dr. A. V. Chankin

Second Tutor: Dr. D. P. Coster

Supervisor: Prof. Dr. U. Stroth

Contents

1 Introduction 1
1.1 Nuclear fusion 1
1.2 Confinement fusion 2
1.3 Tokamak, divertor and scrape-off layer 2
1.4 Scrape-off Layer plasma simulation 4
1.5 Thesis outline 6

2 Fluid description of the edge plasma 9
2.1 Fluid model 9
2.2 Braginskii model 11
2.3 SOLPS code package 14
2.3.1 B2.5 equations 14
2.4 1D SOLPS adaptation 18

3 Kinetic effects of parallel electron transport 21
3.1 Limitations of fluid models and kinetic factors 21
3.1.1 Heat conduction and thermal force coefficients 21
3.1.2 Sheath potential drop and electron sheath heat transmission coefficient 23
3.2 The KIPP code and kinetic electrons 27
3.2.1 Introduction to KIPP 27
3.2.2 Investigation of kinetic electrons using KIPP and a simple ion model 34

4 Iterative coupling of KIPP with SOLPS 45
4.1 Iterative coupling algorithm 45
4.1.1 Justification for the iterative algorithm 45
4.1.2 Analytical analysis of the iterative coupling scheme . . . 45
4.2 1D geometry for the iterative coupling 54
4.2.1 The simulation geometry 54
4.2.2 Grid cells generated in KIPP 55
4.2.3 Grid cells generated in SOLPS 55
4.3 Testing the iterative coupling scheme 56
4.3.1 Coupling setup 57
4.3.2 Convergence study 59

4.3.3 Deuterium density scan study 61
4.3.4 Limitations of the iterative coupling algorithm and a possible solution 71
4.3.5 Summary on the coupling scheme 75
4.4 Kinetic effects in the coupling runs 77
4.4.1 Pure D cases 77
4.4.2 D+C case 80
4.5 Summary 81

5 Comparison of the coupling results with SOLPS results 85
5.1 Density scan cases: pure deuterium 86
5.2 D+C case 89
5.3 Summary 91

6 Summary and conclusions 93

A Numerical details of SOLPS 99
A.1 Numerical implementation of B2.5 equations 99
A.2 1D adaptation 103
A.3 Adaptation of boundary conditions in SOLPS to the coupling scheme 104
A.3.1 The default boundary condition 104
A.3.2 The adapted boundary condition 104

B Details of the iterative coupling 107
B.1 The structure of SOLPS and KIPP 107
B.1.1 The structure of SOLPS 107
B.1.2 The structure of KIPP 110
B.2 Structure of the coupling algorithm 111
B.3 Numerical implementation of the coupling algorithm 114
B.3.1 Geometry 114
B.3.2 Definition of variables at cell centers and faces 114
B.3.3 First SOLPS loop with default coefficients 115
B.3.4 Transferring plasma profiles from the SOLPS part to the KIPP part 116
B.3.5 Effective heat transport coefficient c_e^{eff} calculated in the KIPP part 117
B.3.6 Effective thermal force coefficient $k_{\|}$ calculated in the KIPP part 121
B.3.7 Effective electron sheath heat transmission coefficient and sheath potential drop 122
B.4 Added control parameters and switches 126
B.4.1 in *runpar.dat* 126
B.4.2 in *b2mn.dat* 129

Chapter 1

Introduction

1.1 Nuclear fusion

Global energy consumption is increasing mostly due to population growth and rising living standards in developing countries [1]. Coal, fossil oil and natural gas, the energy resources human beings are currently relying on, are limited and not renewable. The demand for new sources of energy has become increasingly urgent. Fusion [2], as the fundamental energy source in the known universe, is regarded as a potential candidate for the ultimate source of energy on Earth. Fusion is the process in which two light nuclei (e.g. deuterium and tritium) fuse to form a heavier one (helium):

$$^2_1D + ^3_1T \longrightarrow ^4_2He(3.5\mathrm{Mev}) + ^1_0n(14.1\mathrm{Mev}) \tag{1.1}$$

During the fusion reaction a huge amount of energy 17.6Mev is released in the form of kinetic energy stored in helium (3.5Mev) and neutron (14.1Mev) due to the mass defect. The main reason that deuterium and tritium have been chosen as the fuel for thermonuclear reactions is that D-T reaction is the easiest fusion reaction. Although tritium does not exist naturally on Earth, it can be produced by combining fusion neutrons with abundant lithium metal. The fact that tritium has a very short radioactive half-life, and deuterium, which exists in the sea water, is practically unlimited, makes fusion energy particularly appealing [3]. Additionally, the fusion reactor is intrinsically safe because the Lawson criterion [4] which has to be fulfilled in order to make fusion reaction possible, is difficult to achieve. The fusion reaction can only happen when two nuclei come to each other close enough ($< 10^{-15}$m), however, they are positively charged, repelling each other. So they have to overcome the Coulomb potential. Due to quantum mechanical tunneling, the nuclei kinetic energy can be less than that required to overcome the Coulomb barrier. The fuel temperature should be at least $\sim$ 10keV. In this high temperature the fuel are already in "plasma" state. For fusion reaction ignition with the deuterium and tritium fuel, it requires [5]:

$$nT\tau_{\mathrm{E}} > 3 \times 10^{21}\mathrm{m}^{-3}\mathrm{keVs} \tag{1.2}$$

where n and T are the fuel plasma density and temperature. τ_{E} is energy confinement time. Achieving this criterion requires highly precise control system. If an accident happened, the fusion reaction would be terminated automatically [3].

1.2 Confinement fusion

There are two major branches of fusion energy research: Magnetic Confinement Fusion (MCF) and Inertial Confinement Fusion (ICF). They both aim at satisfying the Lawson criterion but with two extremely different approaches.

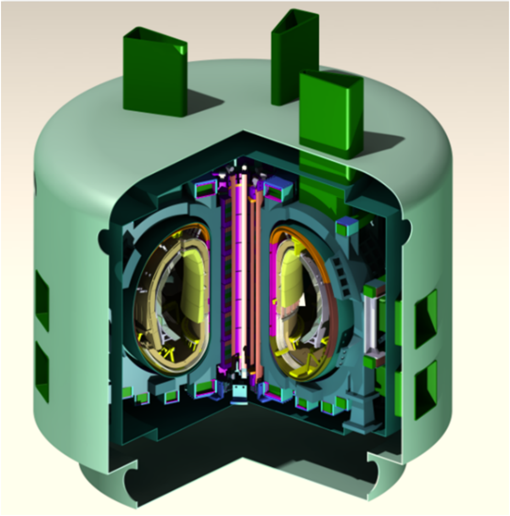


Figure 1.1: A tokamak device.

Magnetic Confinement Fusion (MCF) The fuel plasma inside a vacuum vessel is confined by external strong magnetic fields and is heated to $\sim 5-10\mathrm{keV}$ with ohmic and auxiliary heating methods. Compared to the air on Earth, plasma density in MCF is quite low $\sim 10^{20}\mathrm{m}^{-3}$. It aims at a self-sustaining steady state where the temperature is maintained by the produced high energy helium particles heating and the produced high energy neutrons are thermalized in the blanket to produce energy. Part of the thermalized neutrons are used for tritium breeding.

Inertial Confinement Fusion (ICF) The fuel target (typically a pellet containing deuterium and tritium) is heated and compressed by the powerful laser into extremely high density and temperature plasmas where the ignition condition can be reached. Compared to MCF, the confinement time in ICF is tiny ($\sim 10^{-9}\mathrm{s}$), but plasma density can be extremely high ($\geq 10^{32}\mathrm{m}^{-3}$).

1.3 Tokamak, divertor and scrape-off layer

The tokamak [5, 7–9], as one kind of MCF device, among various types of fusion devices, is the most promising one to realize fusion power in the near future. The tokamak (shown in Fig. 1.1) is a plasma confinement device with strong external toroidal magnets. The fuel plasma is confined in the center of the vessel by strong toroidal magnetic fields generated by toroidally arranged magnets. However, the pure toroidal field is not sufficient to confine the plasma

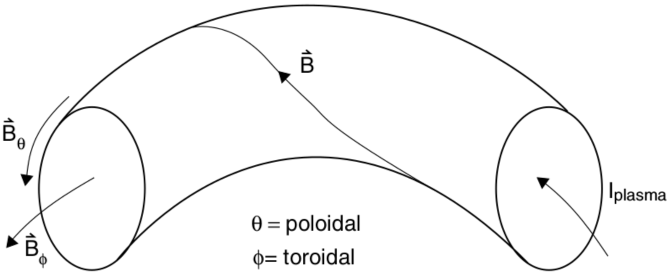


Figure 1.2: Magnetic field $\vec{B}$ is helical around the torus. $\vec{B}_\Phi$ is toroidal magnetic field. $\vec{B}_\theta$ is poloidal magnetic field. This figure is taken from [6].

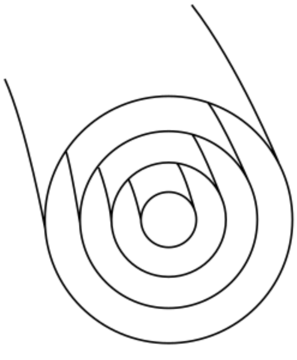


Figure 1.3: Poloidal cross section of Fig. 1.2. A set of nested magnetic flux surfaces are formed. This figure is taken from [6].

due to outward drifts. The toroidal plasma current I_p driven externally is also necessary to provide a poloidal magnetic field so that plasma pressure gradient force is balanced by magnetic forces. As shown in Fig. 1.2, the resulting total magnetic field B_t is helical around the torus. It has two components: the toroidal magnetic field B_Φ generated by toroidal magnets and the poloidal magnetic field B_θ generated by the plasma current I_p. Each magnetic field line lies on one of a nested set of toroidal flux surfaces [6], as shown in Fig. 1.3. Typically in a tokamak, the poloidal field is an order of magnitude lower than the toroidal field ($B_\theta \ll B_\Phi$).

The poloidal cross section of the plasma region is shown in Fig. 1.4, where the main fuel plasma is heated in the center to $5 \sim 10$keV. The so-called divertor configuration [5] is often adopted in large present day tokamaks such as ASDEX-Upgrade [10], JET [11], KSTAR [12], EAST [13], DIII-D [14] and future devices e.g. JT60-SA [15] and ITER [16]. An X-point ($B_\theta = 0$) is formed by cancelling the local poloidal magnetic field generated by the plasma current (I_p) with a current (I_D) flowing in poloidal coils near the divertor:

$$I_D = \frac{x}{b} I_p \tag{1.3}$$

where x and b are the distances from the divertor coils and the center of the plasma current to the X-point, respectively. By this, the magnetic field becomes open outside of the last closed flux surface (LCFS). The region outside of LCFS is called the Scrape-off Layer (SOL). The plasma transport along the magnetic field is much faster than that in the perpendicular direction. When the plasma moves out of the core into the SOL through LCFS, it is diverted along open magnetic field lines to the targets which are normally placed away from the confinement region [5]. The interaction of the plasma with the solid material causes neutral recycling and impurity sputtering [17]. The recycled fuel neutrals, as well as sputtered impurity neutrals, are ionized mostly in the divertor region instead of entering the confinement region and thereafter contaminating the plasma there. The radiation due to ionization and other atomic processes cools down the divertor plasma, forcing a substantial temperature gradient from the mid-plane to the target in the SOL, which allows the interaction of the divertor plasma with the target to occur at a low temperature, significantly reducing the impurity sputtering yields. Tungsten will be probably chosen as the first wall material in future fusion devices [18]. Avoiding impurity sputtering becomes increasingly important since only a small amount of tungsten in the core region can cool down the fuel plasma and terminate the fusion reaction [19]. Additionally, power exhaust is one of the critical issues in the future fusion devices [20]. In ITER, approximately 80% of the heating power due to the fusion reaction should be exhausted in the divertor region, hence the divertor should be well designed to have the capability of handling such a large power exhaust.

1.4 Scrape-off Layer plasma simulation

An accurate prediction of the heat flux deposited on divertor targets is required for the divertor design. For present tokamak edge plasma simulation codes such as SOLPS [21], the 2D multi-fluid Braginskii model [22] is implemented. It is based on moments of the Vlasov-Fokker Planck equation [2, 5, 8] (described in detail in section 2.2). The first three moments with unknown variables: ion and electron densities, velocities and temperatures, are commonly used in a fluid model with closure equations to close the hierarchical structure of moment equations. The closure equations are the relations of the higher order terms (the undetermined terms: ion and electron heat flux densities, thermal force coefficient) and the variables to be solved for. In the Braginskii model, the closure of moment equations is achieved by solving Vlasov-Fokker-Planck equation based on the assumption that the distribution function deviates only slightly from the Maxwellian due to the presence of gradients of density and temperature. However, with respect to the electron parallel transport in the SOL, this assumption is easily violated since the electron distribution function in particular near the target is, in most

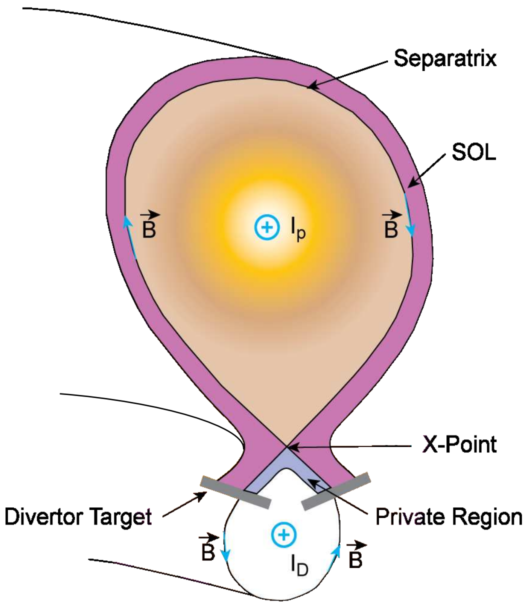


Figure 1.4: Cross section of the plasma and divertor targets.

cases, far away from the Maxwellian, being highly asymmetric due to the presence of super-thermal electrons which are much less collisional and carry the bulk of the heat flux [23–26]. Hence they contribute to the extended tails of down-streaming electrons near the target. The sheath potential drop, acting as an energy filter, reflects low energy electrons, thus cutting off the upstreaming electron distribution function [6]. In addition, parallel electron transport is non-local due to long mean free paths of super-thermal electrons [26–28].

Earlier studies attempted to simplify the kinetic equation in the super-thermal limit [28–31] by only considering the collisions of super-thermal electrons which are responsible for the heat transport and then to obtain a parallel heat transport equation with non-local terms which can be easily solved numerically. For other analytical studies, self-similar solutions [27, 32, 33] for the kinetic equation with a simplified collision operator are searched in order to obtain an analytical expression for parallel conductive heat flux density which converges to Braginskii formula in the collisional limit. These analytical formulas clarified the dependence of parallel conductive heat flux density on other macroscopic parameters despite many limitations posed on their applications by the prerequisite assumptions.

Another approach of including non-local effects is to close the hierarchical structure in a higher moment (the 4th moment) [34–42]. A higher order of a macroscopic parameter $\langle v^4 \rangle$ is introduced, but it cannot be determined self-consistently. This parameter $\langle v^4 \rangle$ is then closed by being related to lower order parameters by assuming a bi-Maxwellian distribution function [43].

Some other non-local expressions for the heat flux density are also proposed in various kinetic simulations [44–47]. They take into account the non-local nature of the conductive heat flux density by incorporating heat flux densities at all positions into one i.e. calculating the heat flux density at one certain location based on the whole parameter profiles, not just local values and their derivatives.

The studies mentioned above are all within the framework of a fluid model. However, from kinetic simulations [48–53] it was found that the electron distribution function near the target deviates significantly from the Maxwellian due to super-thermal electrons coming from upstream. Therefore kinetic simulations are necessary to elucidate the physics that the fluid model is not able to cover. The full kinetic equation is often reduced to a gyro-kinetic one by averaging the gyro-motion.

Two main approaches are adopted to solve the kinetic equation: particle-in-cell (PIC) [54] and continuum [55] methods.

Particle in cell (PIC) method In the PIC approach, the initial distribution function in the kinetic equation is recovered by a number of "macroparticles" statistically, each of which consists of many real particles. Then the evolution of the distribution function is represented by evolving the positions and velocities of the "macroparticles". This approach is numerically easier to implement in a code but it requires a large number of "macroparticles" to reduce the statistical noise. Some codes have been utilizing this method to investigate kinetic effects in edge plasmas, e.g. PARASOL [50, 56–64], BIT [65–73], W1 and W2 [74–76].

Continuum method In the continuum approach utilized in other codes: FPET [51], ALLA [49], COGENT [77–79], the kinetic equation is solved in a discretized phase space. The macroscopic parameters are easily calculated by taking moments of the distribution function. However, the simulation accuracy is limited by the resolution of velocity space grids that should be discretized in such a way so as to cover the wide range of temperatures along field lines. Implementation of the simplified BGK collision operator [80] which is numerically and physically simpler than the Fokker-Planck one [81], allows one to couple kinetic effects related to the long mean free path particles [82–84] and nonlocal boundary conditions with a fluid model [85]. Nevertheless, the simplified approximation of a full collision operator is not applicable to arbitrary profiles of density and temperature because of its low accuracy and the lack of conservativeness [83]. The existing kinetic codes are either imperfect in including all the physical processes that already exist in a fluid code or extremely time-consuming. Therefore the Kinetic Code for Plasma Periphery (KIPP) [86–89] was developed to investigate kinetic effects of parallel plasma transport in a systematic way with an aim of coupling it with SOLPS. It is supposed to be fast at a cost of sacrificing a few features but without losing accuracy of collision terms. On the other hand, SOLPS is a highly sophisticated code with self-consistent recycling and physical and chemical sputtering as well as atomic physics [90] which are the most sophisticated and time-consuming parts in the numerical implementation of a kinetic code. In this thesis, the sophisticated model of SOLPS and kinetic effects of electron parallel transport offered by KIPP are combined in the KIPP-SOLPS coupling algorithm.

1.5 Thesis outline

The second chapter introduces the Braginskii model and the SOLPS code package as well as its adaptation to a 1D geometry which is used by the present version of KIPP. The third chapter describes the limitations of the fluid model and basic features of KIPP. A family of self-similar KIPP cases, varying only by upstream plasma collisionality [6], with the parabolic ion velocity profile reaching ion sound speed at the target are run. The results showed that in a small region near the target at a distance of the order of electron mean free path (MFP) calculated based on plasma parameters at the target, the fluid model is not applicable due to non-local effects [48]. In the fourth chapter, an iterative algorithm of coupling KIPP with SOLPS is fully investigated. A special treatment of the region near the target for low upstream collisionalities is required. Additionally, in the fifth chapter, the coupling results are compared with the results obtained by using only SOLPS to identify electron kinetic effects. Appendix A describes the numerical implementation of the Braginskii model in the SOLPS code and the adaptation to the 1D geometry, while the numerical details of the coupling algorithm are presented in Appendix B.

Chapter 2

Fluid description of the edge plasma

The Vlasov-Fokker-Planck equation [2,81] is the fundamental kinetic equation to describe a plasma:

$$\frac{\partial f_\alpha}{\partial t} + \vec{v} \cdot \nabla_{\vec{r}} f_\alpha + \frac{q_\alpha}{m_\alpha} \left(\vec{E} + \vec{v} \times \vec{B} \right) \cdot \nabla_{\vec{v}} f_\alpha = \left(\frac{\partial f_\alpha}{\partial t} \right)_c \tag{2.1}$$

with $f_\alpha(\vec{r}, \vec{v}, t)$ being the distribution function of species α with the mass m_α and the electric charge q_α, at position $\vec{r}$, with velocity $\vec{v}$ at time t. $\vec{E}\left(\vec{r}, t\right)$ and $\vec{B}\left(\vec{r}, t\right)$ are the macroscopic electric and magnetic fields, respectively, at position $\vec{r}$ and time t. The right hand side is the Fokker-Planck collision term, which describes long-range Coulomb interactions considered in section 3.2.1.

The numerical solution of the above equation requires large computational resources. A fluid model is often used to speed up calculations in the tokamak edge plasma.

2.1 Fluid model

A fluid model can be derived by taking moments of the Vlasov-Fokker-Planck equation (Eq. (2.1)) [2,5,22]. Moment equations can be derived by multiplying Eq. (2.1) by an arbitrary function $\phi\left(\vec{v}\right)$ and integrating over the velocity space.

Particle conservation equation When $\phi\left(\vec{v}\right) = 1$, the moment equation becomes the particle conservation equation:

$$\frac{\partial n_\alpha}{\partial t} + \nabla \cdot \left(n_\alpha \vec{u}_\alpha \right) = 0 \tag{2.2}$$

where density and fluid velocity (the average velocity of all α particles):

$$n_\alpha = \int f_\alpha d\vec{v} \tag{2.3}$$

$$\vec{u}_\alpha = \frac{1}{n_\alpha}\int f_\alpha \vec{v} d\vec{v} \tag{2.4}$$

and the right hand side is zero because the Fokker-Planck collision term conserves particles:

$$\int \left(\frac{\partial f_\alpha}{\partial t}\right)_c d\vec{v} = 0 \tag{2.5}$$

Momentum conservation equation When $\phi\left(\vec{v}\right) = m_\alpha \vec{v}$, the moment equation becomes the momentum conservation equation:

$$m_\alpha \frac{\partial n_\alpha \vec{u}_\alpha}{\partial t} + \nabla \cdot \left(m_\alpha n_\alpha \vec{u}_\alpha \vec{u}_\alpha\right) = -\nabla \cdot \hat{P}_\alpha + n_\alpha q_\alpha \left(\vec{E} + \vec{u}_\alpha \times \vec{B}\right) + \vec{R}_\alpha \tag{2.6}$$

where $\hat{P}_\alpha$ is the pressure tensor and $\vec{R}_\alpha$ is the friction force exerted on species α by collisions with other species. They are defined by:

$$\hat{P}_\alpha = \int m_\alpha \vec{v'}\vec{v'} f_\alpha d\vec{v} \tag{2.7}$$

$$\vec{R}_\alpha = \int m_\alpha \vec{v'} \left(\frac{\partial f_\alpha}{\partial t}\right)_c d\vec{v} \tag{2.8}$$

with the random velocity $\vec{v'}$ defined as:

$$\vec{v'} = \vec{v} - \vec{u}_\alpha \tag{2.9}$$

The pressure tensor $\hat{P}_\alpha$ can be split into two parts: the scalar pressure p_α which only exists in the diagonal elements and the viscosity $\hat{\Pi}_\alpha$ that arises due to the deviation of the distribution function from spherical symmetry:

$$\hat{P}_\alpha = p_\alpha \hat{I} + \hat{\Pi}_\alpha \tag{2.10}$$

Pressure, as well as temperature, are defined as:

$$p_\alpha = n_\alpha T_\alpha \tag{2.11}$$

$$T_\alpha = \frac{1}{n_\alpha}\int \frac{1}{3} m_\alpha \vec{v'}^2 f_\alpha d\vec{v} \tag{2.12}$$

The viscosity tensor is defined as:

$$\hat{\Pi}_\alpha = \int m_\alpha \left(\vec{v'}\vec{v'} - \frac{1}{3}\vec{v'}^2 \hat{I}\right) f_\alpha d\vec{v} \tag{2.13}$$

Energy conservation equation When $\phi\left(\vec{v}\right) = \frac{1}{2}m_\alpha \vec{v}^2$, the moment equation becomes the energy conservation equation:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\frac{1}{2}n_\alpha m_\alpha \vec{u}_\alpha^2 + \frac{3}{2}n_\alpha T_\alpha\right) &+\nabla\cdot\Bigg(\frac{1}{2}n_\alpha m_\alpha \vec{u}_\alpha^2 \vec{u}_\alpha + \\ &\qquad \frac{5}{2}n_\alpha T_\alpha \vec{u}_\alpha + \hat{\Pi}_\alpha\cdot\vec{u}_\alpha + \vec{q}_\alpha^{\text{cond}}\Bigg) \\ &= q_\alpha n_\alpha \vec{E}\cdot\vec{u}_\alpha + \vec{R}_\alpha\cdot\vec{u}_\alpha + Q_\alpha\end{aligned} \tag{2.14}$$

where conductive heat flux density:

$$\vec{q}_\alpha^{\text{cond}} = \int \frac{1}{2}m_\alpha \vec{v'}^2 \vec{v'} f_\alpha d\vec{v} \tag{2.15}$$

and heat exchange between species α and other species due to collisions:

$$Q_\alpha = \int \frac{1}{2}m_\alpha \vec{v'}^2 \left(\frac{\partial f_\alpha}{\partial t}\right)_c d\vec{v} \tag{2.16}$$

Eqs. (2.2), (2.6) and (2.14) are basic equations of a fluid model with the unknown variables: density, velocity and temperature, defined by Eqs. (2.3), (2.4) and (2.12), to be solved for. As one can see, density is the zeroth moment of the distribution function, with fluid velocity being the first moment and temperature being the second moment. A fluid model is attempting to reduce the complexity of a kinetic model and give an approximate information of macroscopic plasma parameters. However, one should note that a higher moment variable due to the second term of Eq. (2.1) is always present and undetermined in every moment equation. In principle, the information given by Eq. (2.1) can be recovered by the infinite number of moment equations. In practice, some closure equations after the energy conservation equation, relating the undetermined variables to density, velocity and temperature, are given based on reasonable assumptions.

2.2 Braginskii model

The basic fluid equations are rewritten here:

$$\frac{\partial n_\alpha}{\partial t} + \nabla\cdot\left(n_\alpha \vec{u}_\alpha\right) = 0 \tag{2.17}$$

$$m_\alpha \frac{\partial n_\alpha \vec{u}_\alpha}{\partial t} + \nabla\cdot\left(m_\alpha n_\alpha \vec{u}_\alpha \vec{u}_\alpha\right) = -\nabla p_\alpha - \nabla\cdot\boldsymbol{\hat{\Pi}_\alpha} + n_\alpha q_\alpha\left(\vec{E} + \vec{u}_\alpha\times\vec{B}\right) + \boldsymbol{\vec{R}_\alpha} \tag{2.18}$$

$$\begin{aligned}\frac{\partial}{\partial t}\left(\frac{1}{2}n_\alpha m_\alpha \vec{u}_\alpha^2 + \frac{3}{2}n_\alpha T_\alpha\right) &+\nabla\cdot\Bigg(\frac{1}{2}n_\alpha m_\alpha \vec{u}_\alpha^2 \vec{u}_\alpha + \\ &\qquad \frac{5}{2}n_\alpha T_\alpha \vec{u}_\alpha + \hat{\Pi}_\alpha\cdot\vec{u}_\alpha + \boldsymbol{\vec{q}_\alpha^{\text{cond}}}\Bigg) \\ &= q_\alpha n_\alpha \vec{E}\cdot\vec{u}_\alpha + \vec{R}_\alpha\cdot\vec{u}_\alpha + \boldsymbol{Q_\alpha}\end{aligned} \tag{2.19}$$

It is convenient to eliminate the kinetic energy [22] from Eq. (2.19), especially for numerical implementation. The so-called internal energy conservation equation can then be derived:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\frac{3}{2}n_\alpha T_\alpha\right) &+\nabla\cdot\left(\frac{3}{2}n_\alpha T_\alpha \vec{u}_\alpha + \boldsymbol{\vec{q}_\alpha^{\text{cond}}}\right)\\ &= -\left(\hat{\Pi}_\alpha\cdot\nabla\right)\cdot\vec{u}_\alpha - n_\alpha T_\alpha \nabla\cdot\vec{u}_\alpha + \boldsymbol{Q_\alpha}\end{aligned} \tag{2.20}$$

There are three unknown variables, n_α, $\vec{u}_\alpha$ and T_α to be solved for. However, there are still undetermined terms in red, which are also the moments of the distribution function. Determining the closure terms requires the solution of the kinetic equation. Nevertheless, this process can be simplified with some reasonable assumptions. The most famous derivation of the closure terms is from Braginskii [22], which is based on splitting f_α into a Maxwellian and a small correction:

$$f_\alpha = f_\alpha^M + f_\alpha^1 \tag{2.21}$$

where

$$f_\alpha^M = \frac{n_\alpha}{\left(2\pi T_\alpha/m_\alpha\right)^{3/2}}\exp\left(-\frac{m_\alpha\left(\vec{v}-\vec{u_\alpha}\right)}{2T_\alpha}\right) \tag{2.22}$$

is a Maxwellian distribution function and $|f_\alpha^1| \ll f_\alpha^M$. If the distribution is Maxwellian with constant density, velocity and temperature everywhere, all the closure terms would be zero and the thermal equilibrium would be already achieved. In typical situations, source terms are present, leading to finite density and temperature gradients. In the collisional limit, the distribution function can be described as Eq. (2.21), of which the non-Maxwellian part f_α^1 is the response to gradients. If only one species of ions is considered with charge Z_i, i.e. $\alpha = i, e$, and a strong external magnetic field $\vec{B}$ is present, the Braginskii model gives the friction term:

$$\vec{R}_e = -\vec{R}_i = \vec{R}_u + \vec{R}_T \tag{2.23}$$

where friction and thermal forces:

$$\vec{R}_u = en_e\left(\frac{\vec{j}_\parallel}{\sigma_\parallel} + \frac{\vec{j}_\perp}{\sigma_\perp}\right) \tag{2.24}$$

$$\vec{R}_T = -k_\parallel n_e \nabla_\parallel T_e - \frac{3en_e^2}{2\sigma_\perp B^2}\vec{B}\times\nabla T_e \tag{2.25}$$

The subscript '$\|$' denotes the projection of the vector along the magnetic field, '$\perp$' the projection perpendicular to the magnetic field. The thermal force coefficient:

$$k_\parallel = 0.71 \tag{2.26}$$

for $Z_i = 1$. The current density is defined as:

$$\vec{j} = e\left(Z_i n_i u_i - n_e u_e\right) \tag{2.27}$$

and the electrical conductivities (for $Z_i = 1$) are:

$$\sigma_{\parallel} = 1.96\frac{e^2 n_e \tau_e}{m_e} \tag{2.28}$$

$$\sigma_{\perp} = \frac{e^2 n_e \tau_e}{m_e} \tag{2.29}$$

with defining ion and electron collision times [5]: τ_i and τ_e, as

$$\tau_i = 12\pi^{3/2}\frac{\varepsilon_0^2 m_i^{1/2} T_i^{3/2}}{n_i Z_i^4 e^4 \ln\Lambda} \tag{2.30}$$

$$\tau_e = 3\left(2\pi\right)^{3/2}\frac{\varepsilon_0^2 m_e^{1/2} T_e^{3/2}}{n_i Z_i^2 e^4 \ln\Lambda} \tag{2.31}$$

where ε_0 is vacuum permittivity and $\ln\Lambda$ is Coulomb logarithm, of order 10 in tokamak plasmas [5]. One component of the viscosity tensor is shown here (more details can be found in [22]):

$$\Pi_\alpha^{\parallel,\parallel} = -\eta_{\alpha\parallel}\left(2\nabla_{\parallel}\cdot\vec{u}_\alpha - \frac{2}{3}\nabla\cdot\vec{u}_\alpha\right) \tag{2.32}$$

where viscosity coefficients (for $Z_i = 1$) are:

$$\eta_{i\parallel} = 0.96 n_i T_i \tau_i \tag{2.33}$$

$$\eta_{e\parallel} = 0.73 n_e T_e \tau_e \tag{2.34}$$

Conductive heat fluxes are:

$$\vec{q}_i^{\,\text{cond}} = -c_i\frac{n_i T_i \tau_i}{m_i}\nabla_{\parallel} T_i + \vec{q}_{i\perp}^{\,\text{cond}} \tag{2.35}$$

$$\vec{q}_e^{\,\text{cond}} = -c_e\frac{n_e T_e \tau_e}{m_e}\nabla_{\parallel} T_e + \vec{q}_{e\perp}^{\,\text{cond}} \tag{2.36}$$

and the heat exchange terms are:

$$Q_i = Q_\Delta = \frac{3 m_e n_e}{m_i \tau_e}\left(T_e - T_i\right) \tag{2.37}$$

$$Q_e = -\vec{R}_e\cdot\left(\vec{u}_e - \vec{u}_i\right) - Q_\Delta \tag{2.38}$$

with heat conduction coefficients:

$$c_i = 3.9 \tag{2.39}$$

$$c_e = 3.16 \tag{2.40}$$

for $Z_i = 1$.

Now we have unknown variables n_e, n_i, $\vec{u}_e$, $\vec{u}_i$, T_e, T_i determined by Eqs. (2.17), (2.18), (2.19) with the closure terms replaced by Eqs. (2.23), (2.32), (2.36), (2.35), (2.38) and (2.37) (replacing α with e, i in all equations).

2.3 SOLPS code package

The analytical solution of the Braginkii equations is not possible, especially in the SOL where complicated magnetic geometry, often with an X point, is present, and various atomic processes (e.g. neutral recycling, impurity sputtering, ionization and recombination) occur. The numerical solution is thus necessary to gain better insight into the SOL physics. SOLPS [21], an abbreviation for Scrape-off Layer Plasma Simulator, is a code package developed for the tokamak plasma edge physics community. It includes 5 parts [91](the version of SOLPS5.0):

DG generates the geometry of the tokamak device to be simulated.

Carre generates the simulation grid in the geometry from DG, based on the tokamak equilibrium produced by an equilibrium reconstruction code (e.g. EFIT).

B2.5 solves the Braginskii model in the grid generated by Carre with treating neutrals as one kind of fluid.

Eirene solves the kinetic model for neutrals based on Monte-Carlo method for the given plasma background from B2.5 and calculates the associated particle, momentum and energy sources for B2.5.

B2plot graphically presents simulation results.

B2.5 and Eirene are the main parts of this code package. Since neutrals have no charge and are not bounded by the magnetic field, they have comparatively long mean free paths and can either be treated as a fluid (by B2.5) or kinetically (by Eirene). However, for the following discussions and coupling tests, neutrals will be treated as a fluid since the 1D geometry is used in this work and the fluid treatment is computationally significantly faster. Additionally, the neutrals are not directly involved in the coupling algorithm which describes interactions between the electron kinetic and fluid energy conservation equations.

2.3.1 B2.5 equations

SOLPS is a 2D fluid code, assuming that a tokamak device is toroidally symmetric, with poloidal direction from the inner target to the outer one denoted by x and radial direction from the inner region to the vessel wall denoted by y, as shown in Fig. 2.1. The magnetic field has toroidal (B_z) and poloidal (B_x) components in a tokamak device. The transport in the poloidal direction x in SOLPS arises due to the transport in the parallel ($\|$) and perpendicular ($\perp$) directions as shown in Fig. 2.2 ("$\perp$" was called diamagnetic direction in [21]). One should note that $\perp$ here is different from the one discussed before. Here and in later discussions the subscript $\perp$ denotes the perpendicular direction which is perpendicular to both the magnetic field and the radial direction y

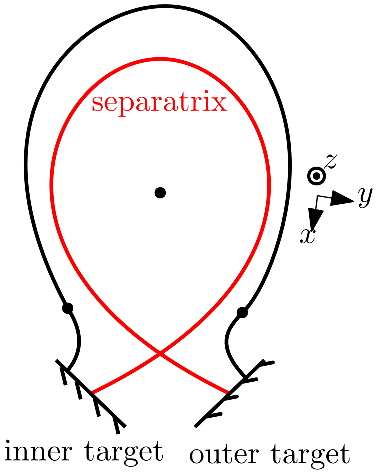


Figure 2.1: Poloidal cross section of a tokamak with lower single null. x is along the poloidal direction from the inner target pointing to the outer target, y is along the radial direction from the core pointing to the wall and z is along the toroidal direction. All variables are assumed to be constant along z in SOLPS.

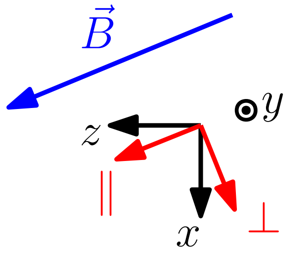


Figure 2.2: The magnetic field $\vec{B}$ in a tokamak has two components: toroidal field (TF) B_z along the toroidal direction z and poloidal field (PF) B_x along the poloidal direction x. The parallel direction $\parallel$ is along the magnetic field while the perpendicular direction $\perp$ is perpendicular to both the parallel ($\parallel$) and radial (y) directions. Fluxes in the poloidal direction (x) are actually sums of projections of parallel and perpendicular fluxes on the poloidal direction (x).

(detailed discussions can be found in section 2.4).

We would like to remind again that, in later discussions, x and y indicate the poloidal and radial directions, respectively, while $\parallel$ and $\perp$ indicate the direction parallel to the magnetic field and the perpendicular direction which is perpendicular to both the parallel ($\parallel$) and radial (y) directions, respectively. And in later discussions, 'perpendicular' transport means the transport only in the perpendicular direction ($\perp$).

B2.5 equations

The equations to be solved in SOLPS (more details can be found in [21]) are

$$\frac{\partial n_i}{\partial t} + \nabla_x \Gamma_{ix} + \nabla_y \Gamma_{iy} = S_{n_i} \tag{2.41}$$

$$n_e = \sum_i Z_i n_i \tag{2.42}$$

$$\frac{\partial}{\partial t}\left(m_i n_i u_{i\parallel}\right) + \nabla \cdot \left(m_i n_i u_{i\parallel} \vec{u}_i\right) = -\nabla_\parallel \left(p_i + p_e\right) - \nabla \cdot \vec{\Pi}_i^{\parallel} + S_{i\parallel} \tag{2.43}$$

$$\nabla_x j_x + \nabla_y j_y = 0 \tag{2.44}$$

$$\begin{aligned} &\frac{\partial}{\partial t}\left(\frac{3}{2} n_i T_i\right) + \nabla_x \left(F_{ix} T_i + q_{ix}^{\text{cond}}\right) + \nabla_y \left(F_{iy} T_i + q_{iy}^{\text{cond}}\right) - Q_\Delta \\ &\qquad = S_{Ei} - n_i T_i \nabla \cdot \vec{u}_i - \left(\hat{\Pi}_i \cdot \nabla\right) \cdot \vec{u}_i \end{aligned} \tag{2.45}$$

$$\frac{\partial}{\partial t}\left(\frac{3}{2} n_e T_e\right) + \nabla_x \left(F_{ex} T_e + q_{ex}^{\text{cond}}\right) + \nabla_y \left(F_{ey} T_e + q_{ey}^{\text{cond}}\right) + Q_\Delta$$

$$= S_{Ee} - n_e T_e \nabla \cdot \vec{u}_e + \frac{1}{en_e} \vec{j} \cdot \vec{R}_e \tag{2.46}$$

Particle conservation equations Eq. (2.41) is the ion particle conservation equation. Γ_{ix} and Γ_{iy} are ion particle flux densities in poloidal and radial directions, respectively. As discussed above, Γ_{ix} is equal to the sum of projections of the ion particle flux density both in parallel and perpendicular directions on the x direction:

$$\Gamma_{ix} = b_x \Gamma_{i\|} + b_z \Gamma_{i\perp} \tag{2.47}$$

where the parallel ion particle flux density:

$$\Gamma_{i\|} = n_i u_{i\|} \tag{2.48}$$

and

$$b_x = \frac{B_x}{\sqrt{B_x^2 + B_z^2}} \tag{2.49}$$

$$b_z = \frac{B_z}{\sqrt{B_x^2 + B_z^2}} \tag{2.50}$$

The perpendicular ion particle flux density will be discussed in the following paragraph. Electron density is determined by the quasi-neutrality Eq. (2.42) instead of solving the electron particle conservation equation.

Momentum conservation equations As one may note, only the parallel ion momentum conservation equation is solved in SOLPS. The perpendicular one is not present because the transport in the directions ($\perp$ and y) perpendicular to $\vec{B}$ in a tokamak device is dominated by drifts and anomalous transport due to turbulence, not yet well elucidated by theory. Instead, ion particle flux densities in these directions are given by:

$$\Gamma_{i\perp} = \Gamma_{i\perp}^{\mathrm{d}} + \Gamma_{i\perp}^{\mathrm{a}} \tag{2.51}$$

$$\Gamma_{iy} = \Gamma_{iy}^{\mathrm{d}} + \Gamma_{iy}^{\mathrm{a}} \tag{2.52}$$

where ion drift particle flux densities:

$$\Gamma_{i\perp}^{\mathrm{d}} = \Gamma_{i\perp}^{\mathrm{ExB}} + \Gamma_{i\perp}^{\mathrm{dia}} \tag{2.53}$$

$$\Gamma_{iy}^{\mathrm{d}} = \Gamma_{iy}^{\mathrm{ExB}} + \Gamma_{iy}^{\mathrm{dia}} \tag{2.54}$$

and ion anomalous particle flux densities:

$$\Gamma_{i\perp}^{\mathrm{a}} = D^n \nabla_\perp n_i \tag{2.55}$$

$$\Gamma_{iy}^{\mathrm{a}} = D^n \nabla_y n_i \tag{2.56}$$

with D^n being the anomalous diffusion coefficient, which is a free parameter specified by the user in SOLPS. The drift terms are mainly E×B and diamagnetic drifts. The divergence of the electric current is zero (Eq. (2.44)). The poloidal current has two components:

$$j_x = b_x j_\parallel + b_z j_\perp \tag{2.57}$$

The parallel current $j_\parallel$ is determined by the Ohm's law (derived from the electron momentum equation neglecting inertia terms):

$$j_\parallel = \sigma_\parallel \left(-\nabla_\parallel \Phi + \frac{1}{en_e} \nabla_\parallel p_e + \frac{k_\parallel}{e} \nabla_\parallel T_e \right) \tag{2.58}$$

Similar to the specification of the perpendicular ion particle flux, the radial and perpendicular components are determined by the non-ambipolar fluxes mainly from the diamagnetic drifts and other effects including inertia, viscosity and ion-neutral friction (the full formula can be found in [21]):

$$j_\perp = j_\perp^{\text{dia}} + j_\perp^{\text{other}} \tag{2.59}$$

$$j_y = j_y^{\text{dia}} + j_y^{\text{other}} \tag{2.60}$$

Hence electric potential can be determined by solving the charge conservation equation and thus electric currents and electron parallel velocity can be obtained as a result.

Internal energy conservation equations Eq. (2.46) is the internal energy conservation equation for electrons implemented in SOLPS. The poloidal electron heat flux density includes the convective piece:

$$q_{ex}^{\text{conv}} = F_{ex} T_e \tag{2.61}$$

where F_{ex} is poloidal electron particle flux density multiplied by a coefficient:

$$F_{ex} = b_x F_{e\parallel} + b_z F_{e\perp} \tag{2.62}$$

$$F_{e\parallel} = \frac{3}{2} \Gamma_{e\parallel} \tag{2.63}$$

$$F_{e\perp} = \frac{3}{2} \Gamma_{e\perp}^{\text{ExB}} + \frac{5}{2} \Gamma_{e\perp}^{\text{dia}} + \frac{5}{2} \Gamma_{e\perp}^{\text{a}} \tag{2.64}$$

with Γ being particle flux density, and the conductive piece:

$$q_{ex}^{\text{cond}} = b_x q_{e\parallel}^{\text{cond}} + b_z q_{e\perp}^{\text{cond}} \tag{2.65}$$

$$q_{e\parallel}^{\text{cond}} = -c_e \frac{n_e T_e \tau_e}{m_e} \nabla_\parallel T_e \tag{2.66}$$

$$q_{e\perp}^{\text{cond}} = -\chi_{e\perp} n_e \nabla_\perp T_e \tag{2.67}$$

where $\chi_{e\perp}$ is the anomalous thermal conductivity, similar to the diffusion coefficient, which is a free parameter specified by the user. τ_e is the electron collision time. The radial convective and conductive electron heat flux densities have the same forms as the perpendicular ones (Eqs. (2.64) and (2.67)):

$$q_{ey}^{\mathrm{conv}} = F_{ey} T_e \tag{2.68}$$

$$q_{ey}^{\mathrm{cond}} = -\chi_{e\perp} n_e \nabla_y T_e \tag{2.69}$$

where

$$F_{ey} = \frac{3}{2}\Gamma_{ey}^{\mathrm{ExB}} + \frac{5}{2}\Gamma_{ey}^{\mathrm{dia}} + \frac{5}{2}\Gamma_{ey}^{\mathrm{a}} \tag{2.70}$$

The ion heat flux densities have the same forms as the electron heat flux densities. The ion temperature T_i in SOLPS is assumed to be the same for all ion species and neutrals.

Now we have the unknown variables n_i, n_e, $u_{i\parallel}$, $j_\parallel$, T_i, T_e and Φ determined by the Eqs. (2.41), (2.42), (2.43), (2.58), (2.45), (2.46) and (2.44). The numerical implementation of these equations is described in Appendix A.

2.4 1D SOLPS adaptation

SOLPS was created as a 2D code, with poloidal and radial directions based on the assumption that all parameters in the toroidal direction are constant. For a number of applications, a 1D version of SOLPS was created [92, 93]. The kinetic code KIPP (described in detail in section 3.2.1) is mainly aimed at coupling parallel kinetic transport with SOLPS, therefore it is the 1D SOLPS version without currents and drifts, with variables only varying in the poloidal direction, that is used for testing the coupling algorithm (described in Chapter 4) since parallel kinetic effects have no direct interaction with the radial transport. This section describes how the 1D geometry is set up in SOLPS.

Fig. 2.3 shows the 1D grid cells generated along the poloidal coordinate (x) in SOLPS. Only one cell is created in the radial direction. The radial transport (e.g. Γ_{iy}, Γ_{ey}, q_{iy} or q_{ey}) is disabled by switching off drifts and forcing radial gradients to 0 ($\nabla_y = 0$). Hence radial particle and heat flux densities in this version of SOLPS through the south and north faces of cell x are

$$\Gamma_y(x) = 0 \tag{2.71}$$

$$q_y(x) = 0 \tag{2.72}$$

Without the radial transport it becomes essentially a 1D problem in the poloidal direction, but, as discussed above, the poloidal transport is the sum of projections of parallel transport along the magnetic field line and the transport in the perpendicular direction in the magnetic flux surface on the poloidal direction (see Fig. 2.4):

$$\Gamma_x(x) = b_x \Gamma_\parallel(x) + b_z \Gamma_\perp(x) \tag{2.73}$$

$$q_x(x) = b_x q_\parallel(x) + b_z q_\perp(x) \tag{2.74}$$

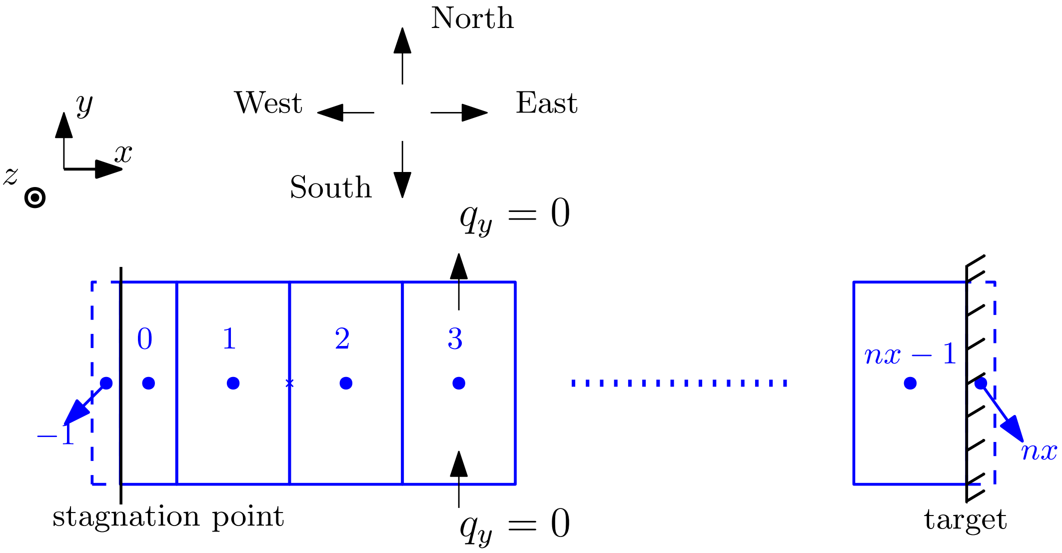


Figure 2.3: 1D grid for SOLPS is generated along the poloidal direction (x). Radial particle and heat flux densities (e.g. heat flux densities through north and south cell faces $q_y = 0$ for all species) are switched off (details can be found in the text). The number of cells is $nx + 2$, with two guard cells: cell -1 and nx, attached to the west and east boundaries.

We also attempted to achieve a real 1D geometry by artificially forcing $\Gamma_\perp = 0$ and $q_\perp = 0$ for all species in Eqs. (2.73) and (2.74), with only parallel transport left. However, neutrals are not bounded by the magnetic field, so,

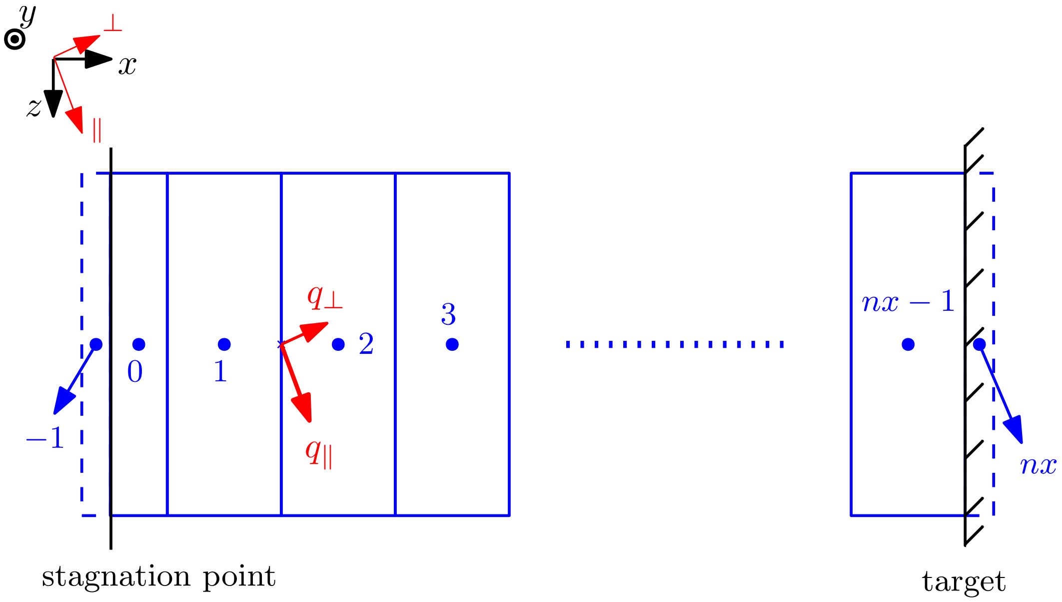


Figure 2.4: 1D grid for SOLPS (the top view of the grid in Fig. 2.3). 1D transport (e.g. q_x) along the poloidal direction (x) at the left face of a certain cell (e.g. cell 2) is the sum of projections of the transport in parallel and perpendicular directions.

the transport of neutrals from the target where they are recycled to upstream regions is mainly attributed to the perpendicular term (the second term in Eq. (2.73)) with the magnetic field $B_z \gg B_x$. Prohibiting neutrals flowing across $\vec{B}$ makes them accumulate in the last two or three cells adjacent to the target, finally leading to crashes in SOLPS. A steady state cannot be achieved. Hence we chose to keep the terms with "$\perp$" in this work. Since the anomalous diffusion coefficient D^n and thermal conductivity $\chi_{e\perp}$, $\chi_{i\perp}$ are much smaller than the parallel ones for plasmas, the perpendicular terms are not important. They are regarded as source terms (see discussions in chapter 4).

Hence the adaptation of the B2.5 equations to the 1D geometry results in:

$$\frac{\partial n_i}{\partial t} + \nabla_x \Gamma_{ix} = S_{n_i} \tag{2.75}$$

$$n_e = \sum_i Z_i n_i \tag{2.76}$$

$$\nabla_x j_x = 0 \tag{2.77}$$

$$\frac{\partial}{\partial t}\left(m_i n_i u_{i\parallel}\right) + \nabla \cdot \left(m_i n_i u_{i\parallel} \vec{u}_i\right) = -\nabla_\parallel \left(p_i + p_e\right) - \nabla \cdot \vec{\Pi}_i^{\parallel} + S_{i\parallel} \tag{2.78}$$

$$\frac{\partial}{\partial t}\left(\frac{3}{2} n_i T_i\right) + \nabla_x \left(F_{ix} T_i + q_{ix}^{\mathrm{cond}}\right) - Q_\Delta = S_{Ei} - n_i T_i \nabla_\parallel u_{i\parallel} - \left(\hat{\Pi}_i \cdot \nabla\right) \cdot \vec{u}_i \tag{2.79}$$

$$\frac{\partial}{\partial t}\left(\frac{3}{2} n_e T_e\right) + \nabla_x \left(F_{ex} T_e + q_{ex}^{\mathrm{cond}}\right) + Q_\Delta = S_{Ee} - n_e T_e \nabla_\parallel u_{e\parallel} + \frac{1}{e n_e} j_x R_{ex} \tag{2.80}$$

where R_{ex} is the projection of the thermal force on the poloidal direction. All other variables have the same forms as discussed above.

Chapter 3

Kinetic effects of parallel electron transport

3.1 Limitations of fluid models and kinetic factors

This section describes four electron-related "kinetic factors" (this name was used in [52, 53]): electron heat conduction coefficient c_e, thermal force coefficient $k_\|$, sheath potential drop $\Delta\phi$ and electron sheath heat transmission coefficient γ_e. Their specifications are necessary for numerical solutions of a fluid model since they cannot be determined self-consistently within the framework of the fluid model. The following subsections will present the classical approaches of specifying the four kinetic factors [6, 22] and thereafter the limitations.

3.1.1 Heat conduction and thermal force coefficients

The closure equations in the previous section are only valid under the assumption that the collisionality is high enough so that the electron conductive heat flux density and thermal force scale linearly with the local temperature gradient:

$$\boldsymbol{q_{e\|}^{\mathrm{cond}}} = -\boldsymbol{c_e n \tau_e \frac{T_e}{m_e} \nabla_\| T_e} \tag{3.1}$$

$$\boldsymbol{R_{T_\|}} = -\boldsymbol{k_\| n \nabla_\| T_e} \tag{3.2}$$

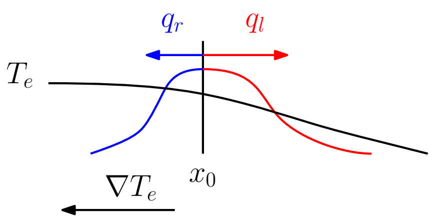


Figure 3.1: Schematic of the heat flux density at a particular location x_0 from the kinetic point of view.

Below we explain the physical origin of these formulas using kinetic arguments.

Electron heat conduction coefficient c_e Considering a certain location x_0 in the plasma (shown in Fig. 3.1) where an almost negligible electron temperature gradient, pointing to the left, is present, the net heat flux density at x_0 is the difference between heat fluxes of electrons moving from the left to right and those moving in the opposite direction:

$$q = q_l - q_r \tag{3.3}$$

Let's assume that the electron density and temperature at x_0 are n_0 and T_0. In the collisional limit with a tiny temperature gradient, the distribution of the electrons streaming to the right f_l^M and the other one f_r^M can be approximated as half-Maxwellian distributions with slightly different T_l and T_r, respectively. Hence the heat flux density coming from the left and right are:

$$q_l = \frac{1}{2} m_e \int_0^{\infty} f_l^M v^2 v_{\parallel} d\vec{v} = 2T_l\Gamma_l \tag{3.4}$$

$$q_r = \frac{1}{2} m_e \int_{-\infty}^{0} f_r^M v^2 v_{\parallel} d\vec{v} = 2T_r\Gamma_r \tag{3.5}$$

where Γ_l and Γ_r are half-Maxwellian random particle flux densities and $\Gamma_l \approx \Gamma_r$. Electrons streaming to the right are characterized by temperature T_l, which is the temperature at the location approximately one electron mean free path λ_0 to the left of x_0. And it is the same case for electrons streaming to the left. We assume:

$$\lambda_0 \ll L_T \tag{3.6}$$

T_l and T_r can be related to T_0 by first order Taylor expansion:

$$T_l \approx T_0 - \lambda_0 \nabla_{\parallel} T \tag{3.7}$$

$$T_r \approx T_0 + \lambda_0 \nabla_{\parallel} T \tag{3.8}$$

Substitution of Eqs. (3.7), (3.8), (3.4) and (3.5) into Eq. (3.3) leads to:

$$q = -c\frac{n_0 T_0 \tau_0}{m_e} \nabla_{\parallel} T \tag{3.9}$$

where c is a constant of the order of unity. The net electron heat flux density q, directed from the left to right, corresponds to the conductive part in a fluid concept.

Thermal force coefficient $k_{\parallel}$ Similarly, the thermal friction force can also be described by a simple physical picture [6, 22] (see Fig. 3.2). Let's assume ions are at rest at x_0. Electrons moving from the left to right and those streaming in the opposite direction are colliding with ions. The thermal force exerted on electrons R_T at x_0 consists of two forces opposite to electron moving directions: R_l and R_r,

$$R_T = R_r - R_l \tag{3.10}$$

which can be approximated as:

$$R_l \approx m_e \Gamma_l \nu_l \propto m_e n_0 v_{\text{th}} \nu_l \tag{3.11}$$

$$R_r \approx m_e \Gamma_r \nu_r \propto m_e n_0 v_{\text{th}} \nu_r \tag{3.12}$$

where v_{th} is the electron thermal velocity at x_0, $v_{\text{th}} = \sqrt{T_0/m_e}$. ν_l and ν_r are the collision frequencies with ions for the electrons streaming from the left to right and those moving in the opposite direction, respectively. They have the following scaling with temperature:

$$\nu_l \propto T_l^{3/2} \tag{3.13}$$

$$\nu_r \propto T_r^{3/2} \tag{3.14}$$

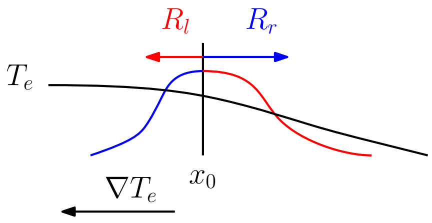


Figure 3.2: Schematic of the thermal force at a particular location x_0 from the kinetic point of view.

Based on the assumption from Eq. (3.6), ν_l and ν_r can be related to the collision frequency ν_0 at x_0 by first order Taylor expansion:

$$\nu_l \approx \nu_0 - \lambda_0 \frac{\partial \nu}{\partial T} \nabla_{\parallel} T \tag{3.15}$$

$$\nu_r \approx \nu_0 + \lambda_0 \frac{\partial \nu}{\partial T} \nabla_{\parallel} T \tag{3.16}$$

Substitution of Eqs. (3.15), (3.16), (3.11) and (3.12) into Eq. (3.10) leads to:

$$R_T \propto m_e n_0 v_{\text{th}} \left(\nu_r - \nu_l\right) \propto n_0 \nabla_{\parallel} T \tag{3.17}$$

Limitations The prerequisite ($\lambda_0 \ll L_T$) for plasmas to be in the collisional limit is often not valid in edge plasmas. From previous 1D kinetic simulations [23–26, 88], the conductive heat flux was found to be carried mostly by Heat Carrying Electrons (HCE) even in the collisional limit as shown in Fig. 3.3. According to [88], the maximum energy flux density is achieved at $v_{\parallel} = 2.82 v_{\text{th}}$, $v_{\perp} = 1.98 v_{\text{th}}$, corresponding to $5.95 T_e$ of the kinetic energy. The mean free path of HCE is $\lambda_{\text{HCE}} = v_{\parallel} \times \tau_{\text{HCE}} \approx 25 \lambda_{\text{th}}$ [88]. HCE are experiencing much fewer collisions than thermal electrons. The collisional limit condition (Eq. (3.6)) is often violated for HCE even when thermal electrons are collisional.

3.1.2 Sheath potential drop and electron sheath heat transmission coefficient

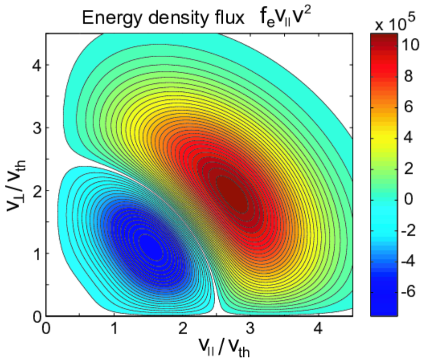


Figure 3.3: The contour plot of the conductive heat flux (taken from the reference [88]). $v_{||}$ and $v_\perp$ are velocity parallel to the magnetic field line and gyro-motion velocity perpendicular to the magnetic field line in the velocity space. $v_{\rm th} = \sqrt{T_e/m_e}$.

A thin layer called "Debye sheath" [94] is present in the region where the plasma interacts with the target. The sheath is so thin that it is usually collisionless and fully kinetic. Hence it cannot be described by the fluid model, as a result, the boundary conditions for the fluid equations (section 2.4) have to be specified separately. Here only the boundary kinetic factors for equations related to electrons are discussed: $\Delta\phi$ for the charge conservation equation (Eq. (2.77)) and γ_e for the electron energy conservation equation (Eq. (2.80)).

Sheath potential drop $\Delta\phi$ The boundary condition for the charge conservation equation (Eq. (2.77)) cannot be determined self-consistently [6, 94–98] in a fluid code since, as discussed above, the Debye sheath is fully kinetic and not resolved, e.g. for the above mentioned geometry in SOLPS (Fig. 2.3) the east boundary is technically the sheath edge instead of the target. The determination of the sheath potential drop $\Delta\phi$ depends on the local electron distribution function, which is unknown at the boundary shown in Fig. 3.4. In order to specify a proper boundary condition for Eq. (2.77), the potential drop through the sheath $\Delta\phi$ has to be given. One can assume the sheath to be thin and collisionless, but the plasmas are often collisional enough for elec-

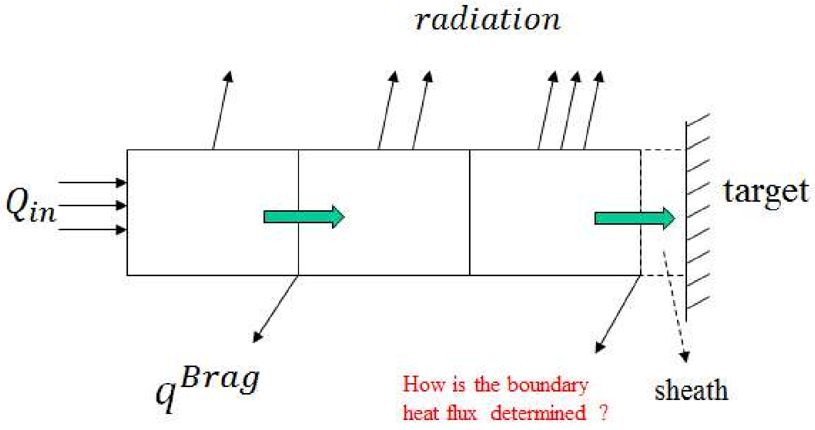


Figure 3.4: Sheath potential drop and heat transport through the boundary can not be determined self-consistently in a fluid code since the Debye sheath is present between the simulation boundary and the target, and the sheath physics is beyond the capability of a fluid code.

trons to have the Maxwellian distribution function at the sheath edge (sheath entrance). Within the sheath, there are no particle sinks, and electrons satisfy Boltzmann relation with the Maxwellian distribution function with constant T_e in an electrostatic field [6]:

$$n_e(x) = n_t \mathrm{exp}\left(\frac{e(\phi(x) - \phi_t)}{T_e}\right) \tag{3.18}$$

where n_t and ϕ_t are the electron density and potential at the sheath edge. Then the electron density at the solid wall (the target) [6]:

$$n_w = n_t \mathrm{exp}\left(\frac{e\Delta\phi}{T_e}\right) \tag{3.19}$$

where $\Delta\phi$ is the sheath potential drop, i.e. the potential difference between the wall and the sheath edge, $\Delta\phi = \phi_w - \phi_t$. The electron flux density reaching the wall equals to the one-way random particle flux density for a Maxwellian [6]:

$$\Gamma_{ew} = \frac{1}{4} n_w \left(\frac{8T_e}{\pi m_e}\right)^{\frac{1}{2}} = \frac{1}{4} n_t \left(\frac{8T_e}{\pi m_e}\right)^{\frac{1}{2}} \mathrm{exp}\left(\frac{e\Delta\phi}{T_e}\right) \tag{3.20}$$

where $\Delta\phi$ is the potential drop across the sheath. Without particle sinks in the sheath, the electron particle flux density through the boundary $\Gamma_{et} = \Gamma_{ew}$. Without currents, ambipolar flow at the boundary is achieved, giving [6]:

$$\begin{aligned} \Gamma_{it} &= \Gamma_{et} && (3.21) \\ \Rightarrow \quad n_t C_s &= \frac{1}{4} n_t \left(\frac{8T_e}{\pi m_e}\right)^{\frac{1}{2}} \mathrm{exp}\left(\frac{e\Delta\phi}{T_e}\right) && (3.22) \\ \Rightarrow \quad \frac{e\Delta\phi}{T_e} &= \frac{1}{2} ln \left(2\pi \frac{m_e}{m_i}\left(1 + \frac{T_i}{T_e}\right)\right) && (3.23) \end{aligned}$$

For hydrogen plasmas with $T_e \approx T_i$,

$$\frac{e\Delta\phi}{T_e} \approx -3 \tag{3.24}$$

which is often taken as the boundary condition for the potential equation in fluid codes [6].

Electron sheath heat transmission coefficient γ_e The boundary condition for the electron energy conservation equation cannot be self-consistently determined either. As mentioned above, a Maxwellian distribution is assumed to obtain the sheath potential drop. This assumption, which is, however, not exactly true even for very collisional plasmas, leads to incorrect boundary heat flux. In practice, the distribution function of electrons at the solid wall (the target) cannot be a perfect Maxwellian [6]. In the collisional limit, a cut-off Maxwellian distribution function [6] can be assumed at the boundary (sheath

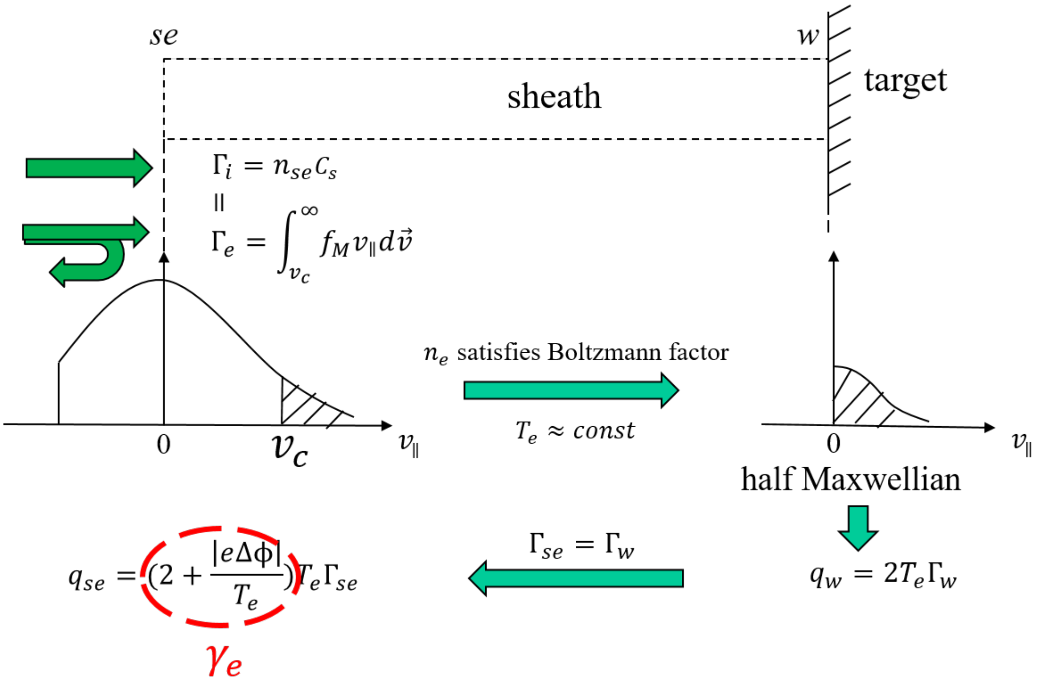


Figure 3.5: In the collisional limit, a cut-off Maxwellian distribution function is assumed at the simulation boundary (se). The cut-off parts (denoted in shadow) are the electrons which have enough energy to overcome the potential drop through the sheath and then be deposited on the wall (target). Right on the target, there are no electrons coming back upstream due to the absorbtion. So, a half-Maxwellian distribution function is assumed for electrons.

edge), as shown in Fig. 3.5, with the cut-off high energy tail being absorbed by the wall and bulk electrons with parallel electron velocity lower than the cut-off velocity v_c being reflected back into the simulation domain due to the potential drop within the sheath. Right at the target the deposited electrons, corresponding to the cut-off part at the sheath edge, have a half-Maxwellian distribution due to the absorption, meaning that there are no upstreaming electrons at the target. Since there are no particle sources in the sheath, $\Gamma_w = \Gamma_{se}$. The deposited electron parallel heat flux density on the target, based on a half-Maxwellian distribution, is [6]:

$$q_w = 2T_w\Gamma_w \tag{3.25}$$

Similar to the assumption above, $T_e \approx const$ is assumed throughout the whole sheath, meaning $T_{et} \approx T_w$. Every electron that is deposited on the target has gone through the sheath, losing the energy $|e\Delta\phi|$ which needs to be added to the electrons right at the sheath edge. Therefore the electron heat flux density through the boundary is [6]:

$$\begin{aligned} q_{se} &= \gamma_e T_{et}\Gamma_{se} && (3.26) \\ \gamma_e &= 2 + \frac{|e\Delta\phi|}{T_{et}} && (3.27) \end{aligned}$$

Now we have prescribed the boundary conditions for the charge and energy conservation equations [6]:

$$\Delta\phi \approx -\frac{3T_{et}}{e} \tag{3.28}$$

$$\gamma_e = 2 + \frac{|e\Delta\phi|}{T_{et}} \approx 5 \tag{3.29}$$

Limitations Electrons at the sheath edge can hardly be Maxwellian or cut-off Maxwellian, especially in medium upstream collisionalities [6, 43, 99], since the electrons that overcome the sheath potential drop and contribute to the boundary heat flux may come from far upstream [99], being characterized by significantly higher temperature than the local one [96, 98, 100–107] due to the temperature variation along a flux tube in the edge. Hence the prerequisite for deriving Eqs. (3.28) and (3.29) is not always valid in the edge. From previous kinetic simulations [52, 53], the non-local effect can significantly influence the values of γ_e and $\Delta\phi$.

3.2 The KIPP code and kinetic electrons

3.2.1 Introduction to KIPP

The Vlasov-Fokker-Planck equation is being solved in KIPP [86, 87]. At the present stage, the code is focusing on the electron parallel transport:

$$\frac{\partial \tilde{f}_e}{\partial \tilde{t}} + \tilde{v}_\parallel \nabla_\parallel \tilde{f}_e - \tilde{E}_\parallel \frac{\partial \tilde{f}_e}{\partial \tilde{v}_\parallel} = \left(\frac{\partial \tilde{f}}{\partial \tilde{t}}\right)_{\text{coll.}} + \tilde{S}_{\text{E}} + \tilde{S}_{\text{p}} \tag{3.30}$$

where the tilde sign "$\sim$" denotes that all parameters appearing in this code are dimensionless, normalized by reference parameters: density n_0, temperature T_0, velocity v_0 and collision logarithm Λ_0, with $T_0 = m_e v_0^2$, normally taken at the stagnation point. This sign will be omitted in the following discussion in this section. f_e is a 3D distribution function with two dimensions in velocity space: parallel and gyro-averaged perpendicular velocity, and one dimension in physical space along the magnetic field. S_{E} and S_{p} are electron energy and particle sources respectively.

Eq. (3.30) is solved by using an operator splitting scheme [86], with parallel free-streaming for 1/2 time step followed by Coulomb collision and the electric field force over one time step, followed again by the other 1/2 time step free-streaming [86, 108, 109].

The numerical implementation of the terms: collision ($\left(\frac{\partial f}{\partial t}\right)_{\text{coll.}}$), sources ($S_{\text{E}}$ and S_{p}), free-streaming ($v_\parallel \nabla_\parallel f_e$) and electric field ($E_\parallel \frac{\partial f_e}{\partial v_\parallel}$), and the boundary condition will be discussed below.

Collision term

$\left(\frac{\partial f}{\partial t}\right)_{\text{coll.}}$ is the collision term [87]:

$$\left(\frac{\partial f}{\partial t}\right)_{\text{coll.}} = -\sum_{\beta=e,i}\left(\frac{\partial \Gamma^{\beta}_{v_\parallel}}{\partial v_\parallel} + \frac{\partial \Gamma^{\beta}_{v_\perp}}{v_\perp \partial v_\perp}\right) \tag{3.31}$$

where

$$\Gamma^{\beta}_{v_\parallel} = -D^{\beta}_{v_\parallel} f_e - D^{\beta}_{v_\parallel v_\parallel}\frac{\partial f_e}{\partial v_\parallel} - D^{\beta}_{v_\parallel v_\perp}\frac{\partial f_e}{\partial v_\perp} \tag{3.32}$$

$$\Gamma^{\beta}_{v_\perp} = -D^{\beta}_{v_\perp} f_e - D^{\beta}_{v_\perp v_\parallel}\frac{\partial f_e}{\partial v_\parallel} - D^{\beta}_{v_\perp v_\perp}\frac{\partial f_e}{\partial v_\perp} \tag{3.33}$$

$\Gamma^{\beta}_{v_\parallel}$ and $\Gamma^{\beta}_{v_\perp}$ are flux densities in the velocity space. $D^{\beta}_{v_\parallel}$ and $D^{\beta}_{v_\perp}$ are dynamic friction coefficients, $D^{\beta}_{v_\parallel v_\parallel}$ and $D^{\beta}_{v_\perp v_\perp}$ are diffusion coefficients, $D^{\beta}_{v_\parallel v_\perp}$ and $D^{\beta}_{v_\perp v_\parallel}$ are pitch-angle scattering coefficients. The superscript β can be "e": electron, or "i": ion. The coefficients in Eqs. (3.32) and (3.33) are functions of mass, charge and the distribution function of species "β", e.g. for the pitch-angle scattering coefficient:

$$D^{\beta}_{v_\parallel v_\perp} = D^{\beta}_{v_\parallel v_\perp}(Z_\beta, m_\beta, f_\beta) \tag{3.34}$$

The details of the form of these coefficients can be found in [87,110,111]. The numerical details of the implementation of the collision term can be found in [111] and references therein.

Source terms

Formulas for source terms S_E and S_p depend on a specific problem being studied. In order to be compatible with sources in a fluid model where particle and energy sources are separately applied to particle and energy conservation equations (e.g. Eqs. (2.75) and (2.80)), the energy and particle sources are numerically separated in KIPP, meaning that the particle source term only adds particles into cells without introducing energy and vice versa.

By default, the energy source scheme $F_E(f)$ represents the process of uniform electron power input or subtraction which converts one Maxwellian into another without changing the number of particles and momentum. This is equivalent to increasing or decreasing the temperature in a fluid concept.

The electron particle source scheme $F_S(f)$ modifies the electron density by scaling up f_e evenly in velocity space with subsequent power removal with the uniform energy source scheme $F_E(f)$ to compensate for the energy content change during this process:

$$f_1(\vec{v}) = \alpha f_0(\vec{v}) \tag{3.35}$$

$$f_2(\vec{v}) = F_E(f_1(\vec{v})) \tag{3.36}$$

where

$$\alpha = \frac{n_0 + \Delta n}{n_0} \tag{3.37}$$

The density is increased by Δn in Eq. (3.35), while extra power $\frac{3}{2}\Delta n T_e$ is introduced. The following energy source scheme $F_E(f_1(\vec{v}))$ is thus applied to remove such power in order to make sure that the energy content is conserved.

Other forms of energy and particle source schemes will be used for specific problems in the following section.

Free-streaming term

The free-streaming term is the so-called *free-streaming advection*, implemented numerically based on the *Reconstruct-Evolve-Average Algorithm* (REA) [86] in the concept of Finite Volume Method [55]:

$$v_\parallel \nabla_\parallel f_e = -\frac{F^n_{i-\frac{1}{2}} - F^n_{i+\frac{1}{2}}}{\Delta s_i} \tag{3.38}$$

where $F^n_{i-\frac{1}{2}}$ and $F^n_{i+\frac{1}{2}}$ are the discretized numerical fluxes of $v_\parallel f_e$ at the left and right faces of cell i at time t_n, respectively. Then the determination of the free-streaming term is dependent on specifications of the numerical flux $F^n_{i-\frac{1}{2}}$ at cell faces at any time t_n. The specification of $F^n_{i-\frac{1}{2}}$ is based on the process of reconstructing the distribution function on account of the cell average distribution function f^n_i [55] (details are discussed below), which is the average value of the distribution function at time t_n in cell i.

In order to achieve second order accuracy, non-zero slopes are used to reconstruct the distribution function based on *Fromm's method* in every cell, e.g. the distribution function in cell i at time t_n is reconstructed as increasing or decreasing from the left face to right with a constant slope and with the average value f^n_i being at the cell center:

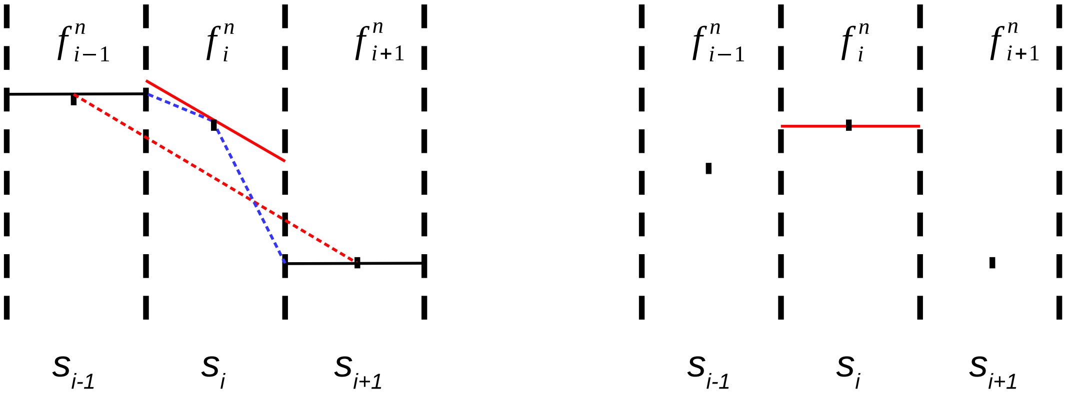
$$\sigma^n_i = \frac{f^n_{i+1} - f^n_{i-1}}{s_{i+1} - s_{i-1}} \tag{3.39}$$

with σ^n_i being the slope at cell i and time t_n, s_i being the center of cell i. Additionally, the *total variation diminishing* (TVD) condition [55] has to be satisfied to achieve non-oscillatory solutions:

$$TV(f^{n+1}) \leq TV(f^n) \tag{3.40}$$

where $TV(f)$ is a function called *total variation*, defined as:

$$TV(f^n) = \sum_i \left| f^n_i - f^n_{i-1} \right| \tag{3.41}$$



(a) σ_i^n is specified as the gentlest one of the three slopes: the solid red slope and the other two dashed blue slopes.

(b) $\sigma_i^n = 0$ in the case of $ab \leq 0$.

Figure 3.6: Linear reconstruction of f_i^n with non-zero slope based on the Fromm's method with the MC limiter.

Fromm's method only satisfies the TVD condition when the solution is smooth enough. Therefore the *monotonized central-difference limiter* (MC limiter) is applied to the *Fromm's method*. Assuming that the spatial grids are uniform, the reconstruction becomes:

$$\sigma_i^n = \text{minmod}\left(\frac{f_{i+1} - f_{i-1}}{2\Delta s}, 2\frac{f_i - f_{i-1}}{\Delta s}, 2\frac{f_{i+1} - f_i}{\Delta s}\right) \tag{3.42}$$

where minmod is a function defined by:

$$\text{minmod}\,(a, b) = \begin{cases} a & \text{if } |a| < |b| \text{ and } ab > 0 \\ b & \text{if } |b| < |a| \text{ and } ab > 0 \\ 0 & \text{if } ab \leq 0 \end{cases} \tag{3.43}$$

$$\text{minmod}\,(a, b, c) = \text{minmod}\,(\text{minmod}\,(a, b)\,, c) \tag{3.44}$$

The reconstruction based on the *Fromm's method* with the MC limiter is illustrated in Fig. 3.6. Once the linear reconstruction is done at time t_n, the numerical flux through each cell face can be calculated:

$$F^n_{i-\frac{1}{2}} = v_\parallel \left[f^n_{i-1} + \frac{1}{2}\left(\Delta s_{i-1} - v_\parallel \Delta t\right)\sigma^n_{i-1}\right] \qquad \text{if } v_\parallel \geq 0 \tag{3.45}$$

$$F^n_{i-\frac{1}{2}} = v_\parallel \left[f^n_i - \frac{1}{2}\left(\Delta s_i + v_\parallel \Delta t\right)\sigma^n_i\right] \qquad \text{if } v_\parallel < 0 \tag{3.46}$$

Due to the fact that the variation of the edge plasma density and temperature can be large, non-uniform spatial grids are used with much finer grids in the divertor region where atomic physics processes are present leading to shorter temperature and density scale lengths. Additionally, the velocity space is also non-uniform with high resolution for the low velocity and much coarser resolution for very high velocity grids. Generally, the distribution functions at very high velocity cells tend to be zero. Under some extreme conditions, with non-uniform grids, free-streaming based on the reconstruction using *Fromm's method* with the MC limiter may introduce negative values in those coarser velocity cells, which may propagate to other cells with advancing in time and finally ruin the simulation. An example is shown in Fig. 3.7. Let's assume there are three adjacent cells, numbered $i-1$, i, $i+1$, with corresponding average distribution functions at time t_n: f^n_{i-1}, f^n_i, f^n_{i+1} in each cell, respectively. The widths of the cells are assumed to be $\Delta s_{i-1} = \Delta s_i = 2\Delta s_{i+1} = 2$. When one does the reconstruction in cell i, the slope calculated by the MC limiter is $-\frac{4}{3}$ (shown as red dashed line in Fig. 3.7). This reconstruction introduces negative values near the right face of cell i, hence the free-streaming solver may give $f^{n+1}_{i+1} < 0$ if $v_{\|} > 0$ and $v_{\|}\Delta t < \frac{1}{2}$.

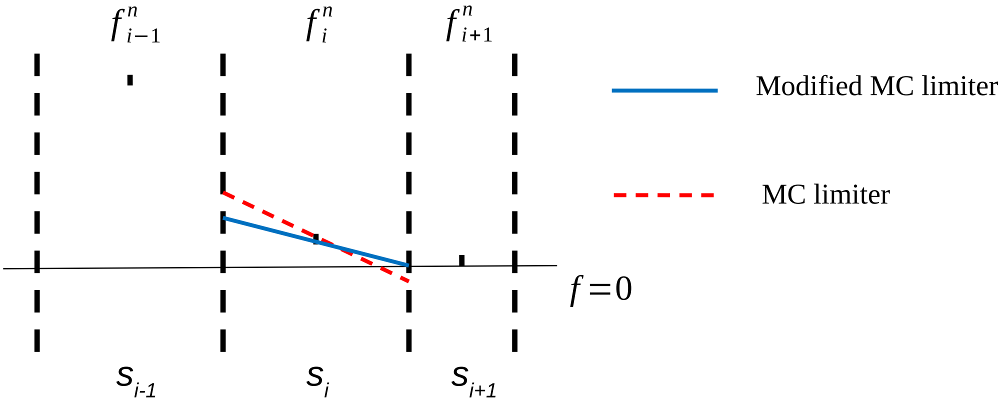


Figure 3.7: The slope at time t_n in cell i is calculated based on the MC limiter (red dashed line) and the modified MC limiter (blue solid line). The example shown in this figure has the following assumptions: the widths of the cells are $\Delta s_{i-1} = \Delta s_i = 2\Delta s_{i+1} = 2$ with the average values of the distribution functions in the three cells being $f^n_{i-1} = 5$, $f^n_i = 1$, $f^n_{i+1} = 0$.

We made a modification (shown as the blue line in Fig. 3.7) for avoiding negative values in such extreme situations without any impacts on the solution when it is smooth by limiting the drop of the right end of the blue solid line to zero. Then the reconstruction of the slope in cell i at time t_n becomes:

$$\sigma^n_i = \text{minmod}\left(\sigma, sign\,(\sigma) \cdot \frac{2|f^n_i|}{\Delta s_i}\right) \tag{3.47}$$

$$\sigma = \text{minmod}\left(\frac{f^n_{i+1} - f^n_{i-1}}{s_{i+1} - s_{i-1}}, 2\frac{f^n_i - f^n_{i-1}}{s_i - s_{i-1}}, 2\frac{f_{i+1} - f_i}{s_{i+1} - s_i}\right) \tag{3.48}$$

$$sign\,(\sigma) = \begin{cases} \sigma/|\sigma| & \text{if } \sigma \neq 0 \\ 0 & \text{if } \sigma = 0 \end{cases} \tag{3.49}$$

Additionally, it is known that the evolving and averaging processes cannot possibly increase the *total variation* [55]. The modified MC limiter does not introduce extra instabilities since it tends to decrease the *total variation* in the reconstruction process.

The modification on the MC limiter has no noticeable influence on the solutions in the simulations described in this work.

Electric field term

Plasma neutrality is maintained by adjusting $E_{\|}$ to compensate for parallel momentum generation in each spatial cell [86]. This is only a numerical procedure without requiring solutions of any equations, which gives the electric field contribution due to electron pressure gradient. The electric field contribution due to the thermal force is calculated separately, and related to the momentum generation in a cell due to Coulomb collisions. It is blended into the implicit scheme for the collision operator [86]. Ambipolarity of parallel plasma transport means that the total electric field is the sum of these two contributions.

Boundary conditions

Boundary for velocity space KIPP is currently developed as a 3D (1D2V) code. As described above, all parameters are normalized by the reference parameters: n_0, T_0, Λ_0, v_0. The velocity grids are created based on the reference parameter v_0. The highest velocity resolved in all simulation cases is $v_{max} = 7v_0$. Distribution functions outside of this range are treated as zero. The justification for this choice was given in [86].

Boundary for spatial space: the stagnation point KIPP is currently designed for 1D space along the magnetic field line with the stagnation point and the target at the left and right ends, respectively. As mentioned above, *Fromm's method* with modified MC limiter is used for the free-streaming solver. It can be clearly seen that the information from two adjacent cells to both left and right sides of cell i is necessary for the calculation of the free-streaming term. However, *Fromm's method* is not applicable at the left or right boundaries since no information outside of the boundaries is available. There are $smax+1$ cells created for the 1D space and they are numbered: $0, 1, \cdots, smax$. Discretization of the free-streaming term at boundary cells 0 and $smax+1$ are $-\frac{F^n_{-\frac{1}{2}}-F^n_{\frac{1}{2}}}{\Delta s_0}$ and $-\frac{F^n_{smax-\frac{1}{2}}-F^n_{smax+\frac{1}{2}}}{\Delta s_{smax}}$, respectively. Determination of the numerical fluxes $F^n_{-\frac{1}{2}}$ and $F^n_{smax+\frac{1}{2}}$ require boundary conditions. The left boundary is the center of cell 0, assumed as the stagnation point where the reconstruction is based on the 1st order upwind scheme:

$$\sigma_0^n = 0 \tag{3.50}$$

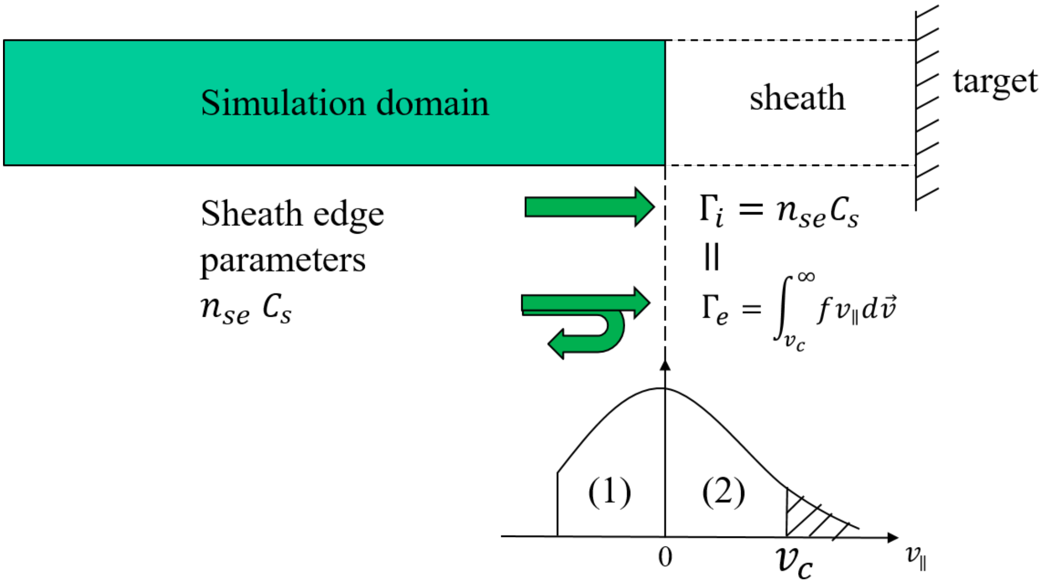


Figure 3.8: The distribution of the downstreaming electrons is determined by upstream parameters. The upstreaming electrons (marked as (1)) are specified as the electrons (marked as (2)) reflected back into the simulation domain.

and the numerical flux entering the boundary is assumed as:

$$F^n_{-\frac{1}{2}}\left(v_\parallel, v_\perp\right) = -F^n_{\frac{1}{2}}\left(-v_\parallel, v_\perp\right) \tag{3.51}$$

making the stagnation point a reflective boundary.

Boundary for spatial space: the target On the other hand, the right boundary is the right face of cell *smax*, assumed as the target (technically sheath edge). Concerning the specification of the boundary condition for the target, one would naturally think of dealing with sheath physics and obtaining a proper distribution function at the sheath edge. However, KIPP is designed to be coupled with SOLPS. For numerical simplicity, Poisson equation and hence the Debye sheath is not dealt with in KIPP since solving the Debye sheath requires a very small time step $\Delta t \sim \tau_g$ (the gyro-motion time) and a small spatial cell size $ds \sim L_D$ (the width of the Debye sheath). Instead, the logical sheath boundary condition is implemented [112], of which the main idea is to give a reasonable electron distribution function at the sheath edge without having to solve the sheath region of inherently small space and time scales. The results compare reasonably well with those of analytical analysis and those from simulation with solving the sheath region, since it captures the main physics of the sheath illustrated in Fig. 3.8. The logical sheath boundary condition is utilized to specify only the numerical flux entering the boundary ($v_\parallel < 0$) since the downstreaming flux ($v_\parallel > 0$) out of the final boundary is determined self-consistently by the upstream conditions:

$$F^n_{smax+\frac{1}{2}}\left(v_\parallel, v_\perp\right) = v_\parallel \left[f^n_{smax} + \frac{1}{2}\left(\Delta s_{smax} - v_\parallel \Delta t\right)\sigma^n_{smax}\right] \quad \text{if } v_\parallel > 0 \tag{3.52}$$

The reconstruction of the boundary cell σ^n_{smax} is based on *Lax-Wendorf's method*:

$$\sigma^n_{smax} = \frac{f^n_{smax} - f^n_{smax-1}}{s_{smax} - s_{smax-1}} \tag{3.53}$$

Based on the logical sheath condition, the numerical flux entering the boundary is assumed as:

$$F^n_{smax+\frac{1}{2}}\left(v_\parallel, v_\perp\right) = \begin{cases} -F^n_{smax+\frac{1}{2}}\left(-v_\parallel, v_\perp\right) & \text{if } -v_c < v_\parallel < 0 \\ 0 & \text{if } v_\parallel \leq -v_c \end{cases} \tag{3.54}$$

where the critical velocity v_c ($v_c > 0$) is determined by the assumption of ambipolar flow through the final boundary:

$$\begin{aligned} & \Gamma_{it} &= \Gamma_{et} \qquad & (3.55) \\ \longrightarrow \qquad & n_{se}C_s &= \int_{v_c}^{\infty} F^n_{smax+\frac{1}{2}}\left(v_\parallel, v_\perp\right) d\vec{v} \qquad & (3.56) \end{aligned}$$

where n_{se} and C_s are boundary density and ion acoustic speed. The electrons with parallel velocity larger than the critical velocity (denoted by the shadow in Fig. 3.8) are absorbed by the target while the left part of the downstreaming electrons (marked by (2)) is reflected at some positions within the sheath and have the same velocity in opposite direction when they come back to the sheath edge, becoming upstreaming electrons (marked by (1)) with part (1) symmetric to part (2). Therefore if the downstreaming electrons at the boundary determined by upstream profiles are denoted as $f_t^{\mathrm{down}}(v_\parallel, v_\perp)$ for $v_\parallel > v_c$, the logical boundary condition is equivalent to the assumption that the distribution function at the simulation boundary is prescribed as:

$$f_t^{\mathrm{kipp}}(v_\parallel, v_\perp) = \begin{cases} f_t^{\mathrm{down}}(v_\parallel, v_\perp) & \text{if } v_\parallel > 0 \\ f_t^{\mathrm{down}}(-v_\parallel, v_\perp) & \text{if } -v_c \leq v_\parallel \leq 0 \\ 0 & \text{if } v_\parallel < -v_c \end{cases} \tag{3.57}$$

3.2.2 Investigation of kinetic electrons using KIPP and a simple ion model

The KIPP code has been extensively tested and demonstrated high precision [86–89]. Before directly coupling KIPP with SOLPS, kinetic electrons described by Eq. (3.30) with a simple ion model were used to evaluate the degree of the deviation of kinetic results from the fluid assumptions in order to develop an adequate coupling scheme.

Simulation model

A series of cases based on a family of self-similar background ion profiles were run. The only variable parameter was upstream dimensionless collisionality

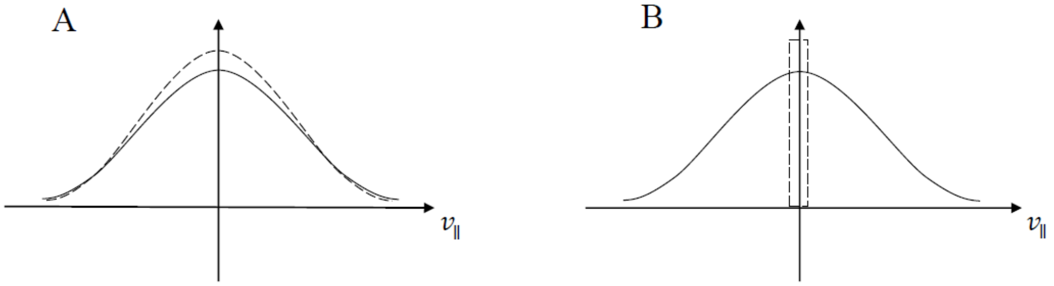


Figure 3.9: Schematics of two numerical schemes of introducing electron particle sources. Solid lines are distribution functions before the particle sources, while the dashed lines are the distribution function after the particle source (A) was introduced and the particle source (B) itself, respectively. These two particle sources only change density with conserving energy and momentum. Case A is the uniform particle source scheme $F_S(f)$ (see section 3.2.1); Case B is an extremely cold particle source scheme, adding electrons only in the velocity cell: $v_\parallel = 0$, $v_\perp = 0$.

(varying from 13 to 130), $\nu^* \equiv L/\lambda_0$. As was pointed out earlier, quasi-neutrality in KIPP is maintained by adjusting parallel electric field $E_\parallel$ (see section 3.2.1). To characterize the effect of collisionality on kinetic factors, for all the scan cases ion profiles are assumed to satisfy a simple SOL 1D model [6] with some additional specifications:

1. Total pressure conservation: $2n_{i0}T_{i0} = 2n_{i(s)}T_i(s) + n_i(s)m_i v_i^2(s)$.
2. Parabolic profile of flow speed u_i along B: $u_i(s) = \frac{s^2}{L^2}C_s$, with the ion sound speed calculated at the target $C_s = \sqrt{\frac{T_e+T_i}{m_i}}$.
3. Equal ion and electron temperatures: $T_i = T_e$.
4. Two numerical schemes for particle sources are implemented for each collisionality (see Fig. 3.9) as described later in detail.

n_{i0}, T_{i0} are upstream dimensionless ion density and temperature at the stagnation point, respectively, assumed to be 1. s is the distance away from the stagnation point. L is the system length and λ_0 is the thermal electron mean free path calculated using parameters at the stagnation point. The assumption of equal ion and electron temperatures is used for simplification of the model and generalization of its results. The variation of the free parameter ν^* is achieved by changing the dimensionless system length.

Geometry

A 1D grid in real space is used. Cells are non-uniformly distributed along the magnetic field with much finer cells near the target. The two ends of the simulation domain have two different boundary conditions, with the left one regarded as a reflective boundary and the right one treated as an energy filter for electrons, absorbing those with parallel velocity $v_\parallel \geq v_c$ while reflecting those with $v_\parallel < v_c$ back into the simulation domain with the same absolute velocity but opposite direction (see section 3.2.1). Ion profiles are adjusted at each time step according to the equation: $v_i(s) = \frac{s^2}{L^2} C_s$. For electrons, in addition to energy sources, particle sources are also necessary to compensate for the divergence of parallel particle flux $nv_\parallel$ and the target sink due to the lack of self-consistent recycling source.

Collisionality scan cases

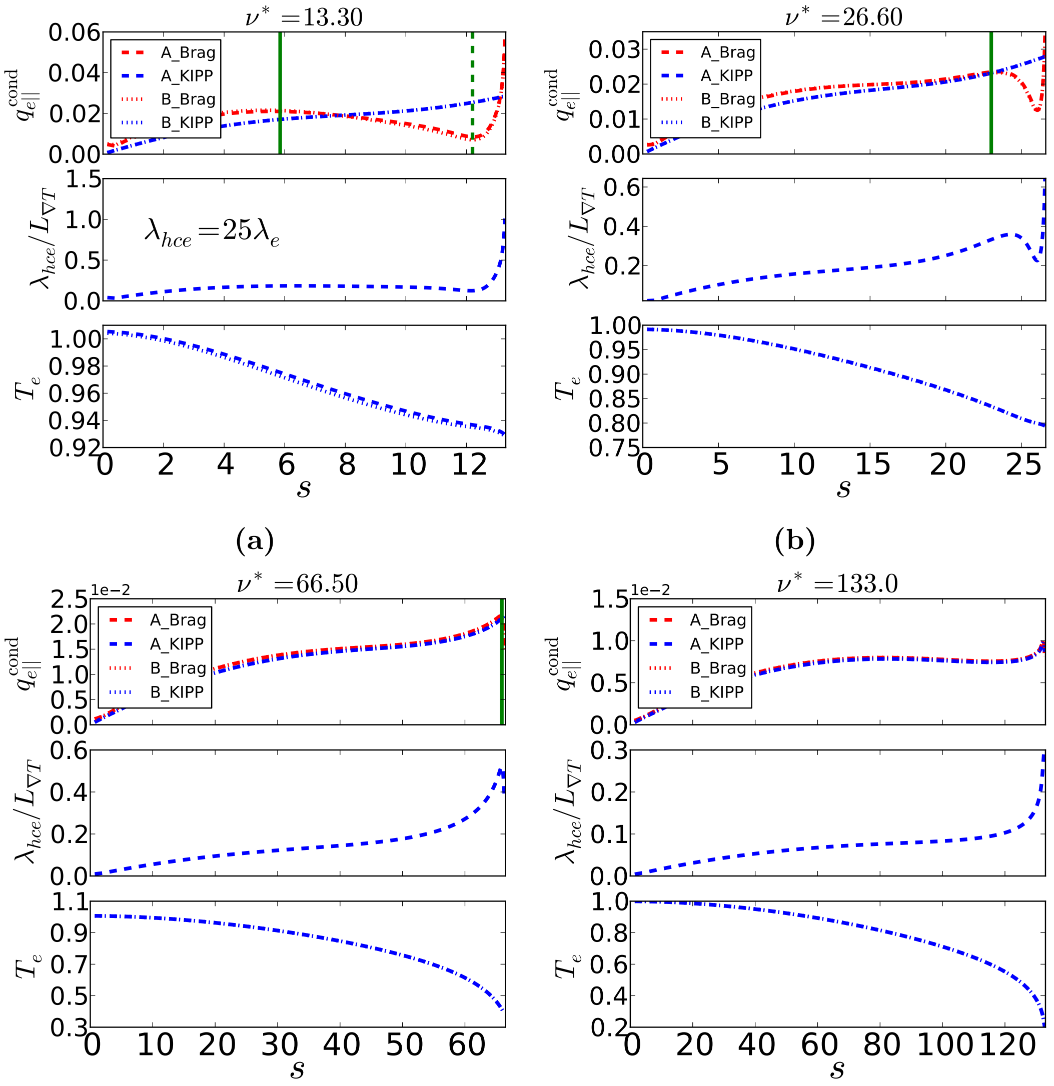
As mentioned in section 3.2.1, particle and energy sources are numerically separated. The energy source is described by the scheme $F_E(f)$ (section 3.2.1). Two numerical schemes are implemented for modelling the electron particle source as shown in Fig. 3.9. Case A is a "uniform" particle source achieved by scaling up f_e evenly in the velocity space with the subsequent energy correction, described by the scheme $F_S(f)$ from Eqs. (3.35) and (3.36); Case B has an "extremely cold" particle source, by adding all electrons into $f\,(0, 0, s)$. The two cases are the same from the fluid model's point of view, but totally different kinetically.

Cases with various upstream collisionality: $\nu^* = 13.30$, $\nu^* = 26.60$, $\nu^* = 66.50$, $\nu^* = 133.0$, are run. The steady state macroscopic parameter profiles of electron temperature (T_e) and conductive heat flux density ($q_{e\parallel}$), local collisionality ($\lambda_{hce}/L_{\nabla T}$, the ratio of HCE mean free path to electron temperature scale length), are shown in Fig. 3.10. It can be clearly seen that the steady state profiles from the two particle source models are the same for all collisionalities since heat carrying electrons (discussed later in this section) are not affected by the source model and hence the relation between conductive heat flux density and temperature gradients remain unchanged i.e. the heat conduction coefficient is not affected by the choice of the particle source model.

For the high collisionality case $\nu^* = 133$, the conductive heat flux density calculated in KIPP has already converged to the Braginskii conductive heat flux density as shown in Fig. 3.10d. In contrast, for comparatively low collisionalities $\nu^* = 13.30$ or 26.60 (Figs. 3.10a and 3.10b), although local collisionality ($\lambda_e/L_{\nabla T}$ used in [26] where λ_e is local thermal electron mean free path and $L_{\nabla T}$ is electron temperature scale length) remains always quite high:

$$\lambda_{hce}/L_{\nabla T} \approx 25\lambda_{hce}/L_{\nabla T} < 1 \tag{3.58}$$

through most of the SOL, it shows flux limiting upstream and flux enhancement downstream [43]. It is clear from these figures that the occurrence of the flux limiting and flux enhancement is unrelated to the local collisionality.



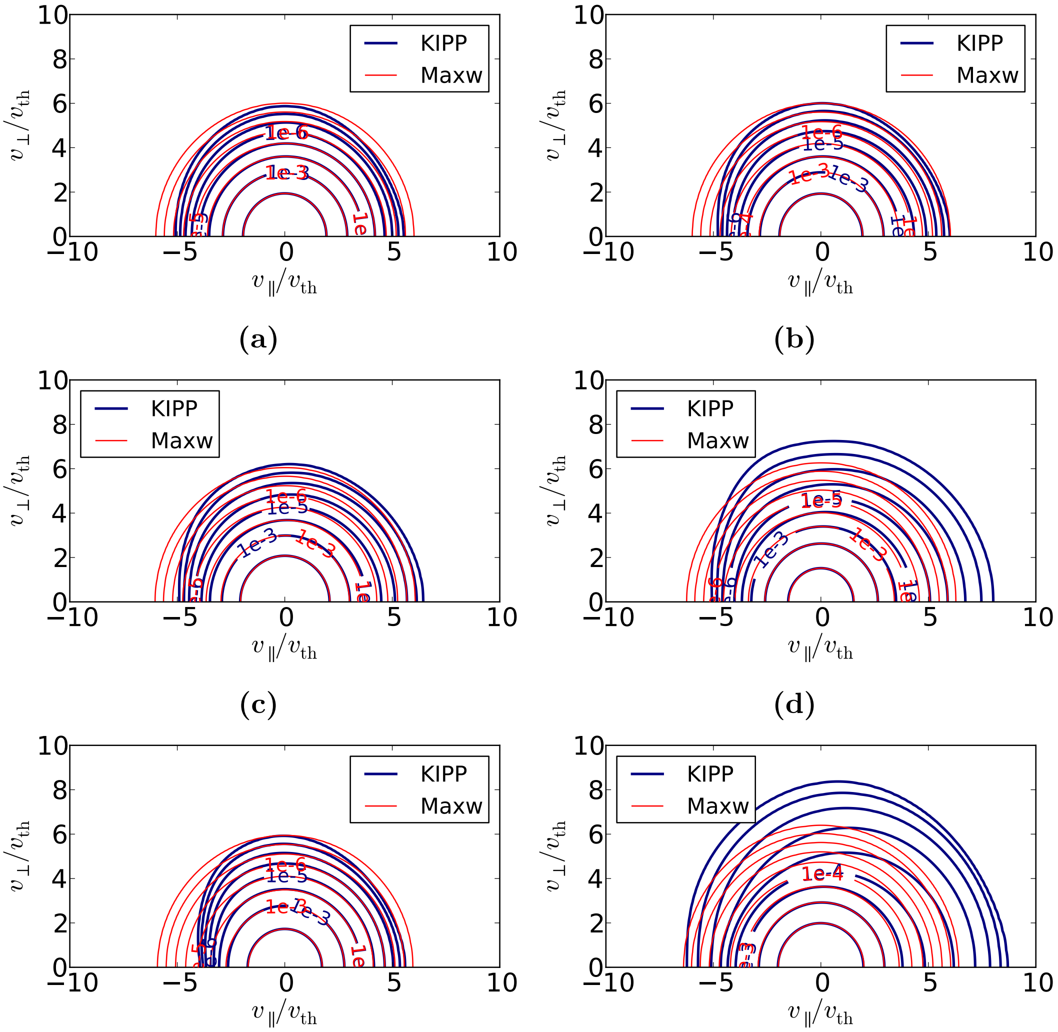
(a) **(b)**

(c) **(d)**

Figure 3.10: Cases with various upstream collisionalities are shown: (a) $\nu^* = 13.30$; (b) $\nu^* = 26.60$; (c) $\nu^* = 66.50$; (d) $\nu^* = 133.0$. All parameters are dimensionless. 'A' and 'B' denote cases with the uniform particle source (corresponding to Case A in Fig. 3.9) and with the extremely cold particle source (corresponding to Case B in Fig. 3.9), respectively. For the $q_{e\|}$ profile, 'KIPP' denotes the conductive electron heat flux density calculated in KIPP while 'Brag' denotes the one calculated with the Braginskii formula based on the electron temperature profile obtained from KIPP. $\lambda_{\mathrm{hce}}/L_{\nabla T}$ is the ratio of heat carrying electrons mean free path to the temperature scale length. The positions $s \approx 5.85$, $s \approx 12.20$ (dashed) in (a), $s \approx 23.00$ in (b), $s \approx 65.96$ in (c) are marked by the green vertical lines. f_e at these positions are plotted in Fig. 3.11.

Flux enhancement downstream

In the region near the target shown in Figs. 3.10a, 3.10b and 3.10c, from locations where the Braginskii conductive heat flux density starts to decrease with s (marked by solid vertical green lines) to the target, kinetic results for the



(a) (b)

(c) (d)

(e) (f)

Figure 3.11: 2D contour plots of f_e at positions: (a) $s \approx 5.85$ for $\nu^* = 13.30$, (b) $s \approx 23.00$ for $\nu^* = 26.60$, (c) $s \approx 37.20$ for $\nu^* = 39.90$, (d) $s \approx 65.96$ for $\nu^* = 66.50$. (e) is at $s \approx 12.20$ for collisionality scan case $\nu^* = 13.30$. (f) is at the same position for the case with enhanced radiation. In all figures, red curves indicate local Maxwellian distributions and blue curves indicate distribution functions from KIPP.

conductive heat flux density deviate substantially from the Braginskii theory. This deviation is probably due to the electron temperature profile flattening in the corresponding region, or due to non-local transport [26–28].

Temperature profile flattening The width of this region, between the solid green vertical line (the first one in Fig. 3.11a) and the right end, is reduced with the increase of the upstream collisionality ν^*, however, it stays unchanged in units of the mean free path at the target ($\sim 5\lambda_t$ with λ_t being thermal electron mean free path at the target). It indicates that the temperature profile flattening and significant discrepancies between KIPP and Braginskii heat conduction coefficients, as well as thermal force coefficients, in this region are somehow related to the target cut-off effect that acts as a sink for high energy electrons thus modifying the target distribution function f_t asymmetrically.

f_e at the positions where Braginskii conductive heat flux density starts to decrease with s are plotted in Figs. 3.11a, 3.11b, 3.11c and 3.11d for cases with $\nu^* = 13.30$, $\nu^* = 26.60$, $\nu^* = 39.90$ (note that this case is not shown in Fig. 3.10), and $\nu^* = 66.50$, respectively. Fig. 3.11e shows the distribution function inside the region at $s \approx 12.20$ for the case with $\nu^* = 13.30$. The comparison of f_e and the local Maxwellian distribution shows that the cut-off distribution of the upstreaming electrons starts to appear for all cases, and there are obvious extended high energy tails of the downstreaming electrons for medium collisionality cases. It means that the Braginskii conductive heat flux formula is not applicable in the region approximately 5 times the mean free path away from the target.

Non-local transport In comparatively low collisionality cases, shown in Fig. 3.11d and 3.11f, a significant expansion of contours for the downstreaming electrons compared to the Maxwellian f_M occur (super-high energy tails) due to non-local transport. This, however, is only the case for extremely high energies, far above the heat carrying electrons (HCE) energies. Such electrons are in absolute minority and they don't contribute much to the parallel heat flux. Up to the HCE energies, f_e are quite close to the Maxwellians, with the $f_e - f_M$ difference hardly visible on the figures, which also applies to Figs. 3.11a, 3.11b, 3.11c, 3.11d. This indicates that the effective HCE collisionality in these cases is medium, allowing only for rather moderate heat flux limiting and enhancement.

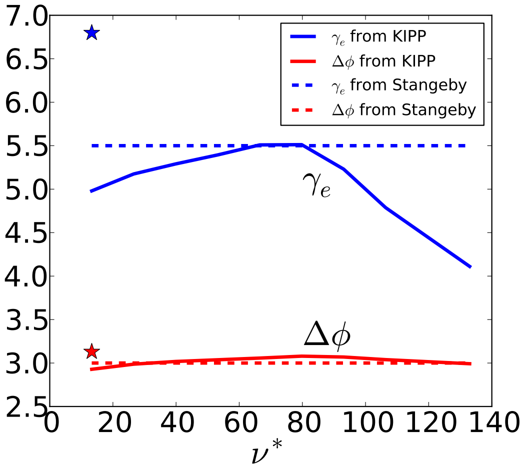


Figure 3.12: Sheath heat transmission coefficient (γ_e) and potential drop ($\Delta\phi$) against collisionality both for the collisionality scan cases (solid lines) and the case with enhanced radiation (stars). Dashed lines are the theoretical values from the reference [6].

The heat transmission coefficient is quite sensitive to the upstream collisionality, while the sheath potential drop agrees well with the classical value used for a fluid model, as shown in Fig. 3.12. The case with enhanced radiation (stars in Fig. 3.12) is discussed below.

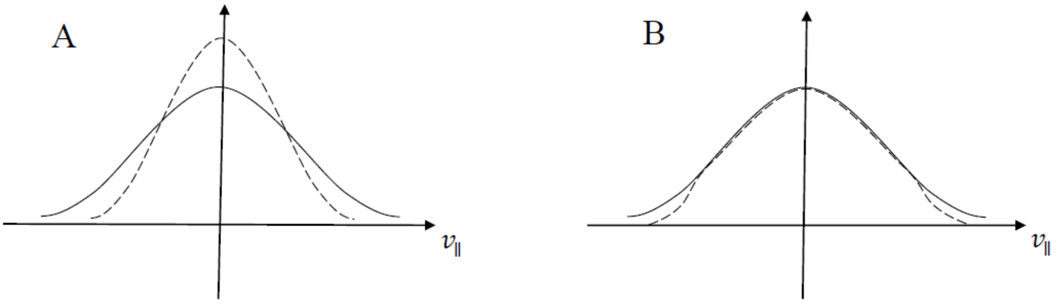


Figure 3.13: Two numerical schemes of modelling radiation power. Solid and dashed curves indicate distribution functions before and after the power removal. Case A is the case with applying uniform energy source ($\Delta E < 0$); Case B is mimicking electron impact ionization power sink with subtracting power only from electrons with kinetic energy above a specified threshold ($E_{th} = 7T_t$ in this case).

Case with enhanced radiation

From collisionality scan cases above, one can see that a physically interesting case with $\nu^* = 13.30$, corresponding to ITER outside mid-plane separatrix parameters ($\nu^* \sim 12$): electron density $n_u = 4 \times 10^{19}\text{m}^{-3}$, electron temperature $T_u = 150\text{eV}$, parallel length from mid-plane to the outer target $L_\parallel \approx 65\text{m}$, does not result in a large temperature drop without a strong energy sink downstream. Therefore an artificial radiation power sink is introduced to radiate 83% of the total input power, spatially distributed according to the $(s/L)^3$ dependence along the simulation domain to force the T_e drop by factor 10. Similar to the particle source, two different numerical schemes for modelling radiation power are implemented, of which case A is using a fluid type radiation model with uniform energy source scheme $F_E(f)$ (Fig. 3.13A) and case B is using the model in which the energy sink is achieved by subtracting power only from the high energy tail mimicking very approximately the electron impact ionization processes at low T_e. The threshold energy is set at about 7 times the target temperature: $E_{th} \approx 7T_t$. The radiation power is then evenly subtracted from electrons with total kinetic energy above that threshold ($0.5m_e(v_\parallel^2 + v_\perp^2) > 7T_t$, Fig. 3.13B), afterwards, these electrons are placed into ($v_\parallel = 0, v_\perp = 0$):

$$f_1(v_\parallel, v_\perp) = \begin{cases} \alpha f_0(v_\parallel, v_\perp) & E(v_\parallel, v_\perp) > E_{th} \\ f_0(v_\parallel, v_\perp) & E(v_\parallel, v_\perp) \leq E_{th} \end{cases} \tag{3.59}$$

$$f_1(0,0) = f_0(0,0) + \int_{E(v_\parallel, v_\perp) > E_{th}} (1-\alpha) f_0(v_\parallel, v_\perp) d\vec{v} \tag{3.60}$$

The case with $\nu^* = 13.30$ is then rerun with the artificial radiation sink. Steady state profiles of electron density, temperature and local collisionality with two different radiation power schemes are shown in Fig. 3.14. Let's first discuss profiles with the uniform energy source (Case A). By comparing the KIPP and Braginskii conductive heat flux densities in Fig. 3.14, one can see a moderate flux limiting upstream and enhancement downstream by factor 1.5. Compared to the distribution function (shown in Fig. 3.11e) at $s \approx 12.20$ in the case with $\nu* \approx 13.30$ without artificial radiation for the collisionality scan study, the same position at $s \approx 12.20$ (indicated by the vertical green line in Fig. 3.14) is chosen to show the steady state distribution function (3.11f) where the distinct high energy tail of downstreaming electrons is seen. However, no cut-off distribution (due to the Debye sheath effect) is seen in this case, since, with the radiation sink, target T_e drops substantially, resulting in decreasing λ_t and moving this position out of the $\sim 5\lambda_t$ region. With the presence of the super-high energy tail, the sheath heat transmission coefficient increases by $\sim 40\%$, compared to the corresponding scan case, as shown in Fig. 3.12.

Comparing the two power input schemes

The comparison of Cases A and B (Fig. 3.14) demonstrates again that no difference between the cases appears in the steady state macroscopic parameter profiles for the two models of radiation power sink, which seems counter-intuitive, since at cells near the target, in the steady state, the power is mainly balanced between free-streaming heat flux coming into the cell and radiation power sink taken from the cell. Two different radiation power sinks would be

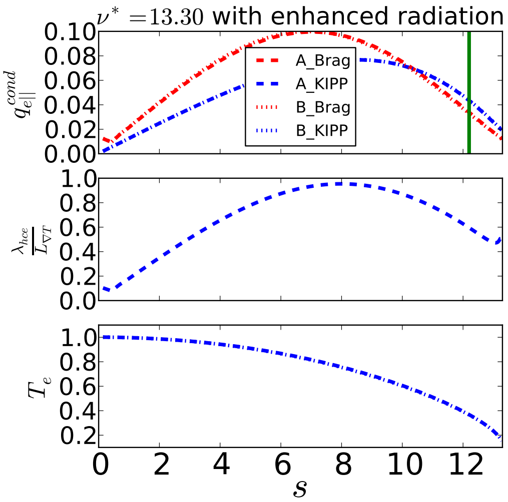


Figure 3.14: Profiles of conductive heat flux density, local collisionality and electron temperature for the case with enhanced radiation. Case A: uniform radiation model (corresponding to Fig. 3.13A); Case B: subtracting radiation energy from electron high energy tail (corresponding to Fig. 3.13B). The green vertical line marks the position $s \approx 12.20$, to plot the distribution function in a steady state (shown in Fig. 3.11f).

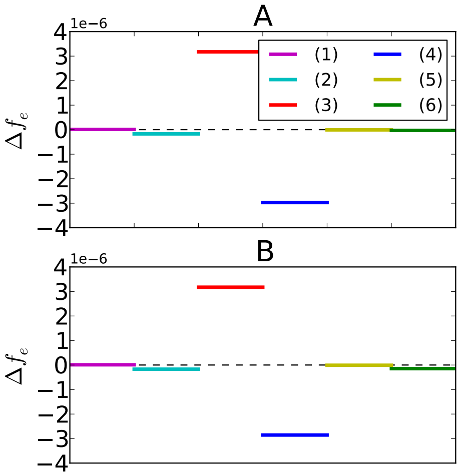
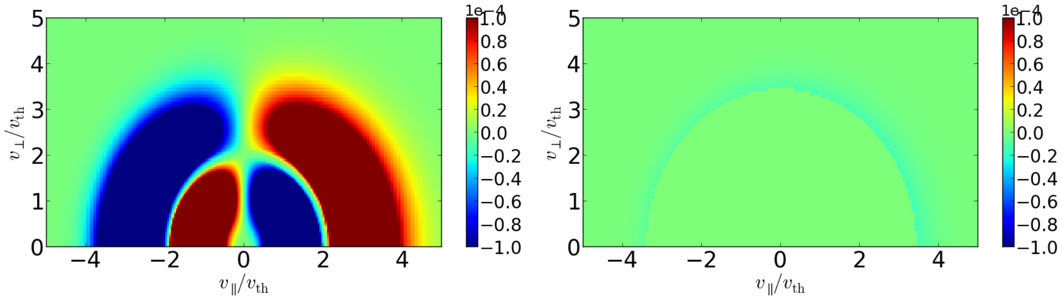


Figure 3.15: Changes of f_e (Δf_e) for one time step at the velocity cell $v_\| \approx 3.54T_e$, $v_\perp \approx 3.54T_e$ for the spatial cell adjacent to the target, due to terms with: (1) power input, (2) electric field, (3) free-streaming, (4) collisions, (5) particle source and (6) radiation power sink. A) shows the power balance among various terms with the uniform energy source scheme (Case A in Fig. 3.13A). B) shows the power balance among various terms with removing power only from high energy electrons (Case B in Fig. 3.13B).



(a) Δf_e change due to free-streaming. **(b)** Δf_e change due to radiation.

Figure 3.16: In a steady state, with advancing one time step, distributions of Δf_e in velocity space at the cell adjacent to the target are shown for Case A (Fig. 3.13A) with the uniform energy source scheme due to the free-streaming term (A) and the radiation term (B).

expected to result in two different steady state profiles. The reasons for this are discussed below.

At the last spatial cell adjacent to the target, the change of f_e for one time step (Δf_e) in velocity space due to the free-streaming and radiation loss for Case B are shown in Figs. 3.16a and 3.16b, respectively. The Δf_e due to the radiation loss is rather uniform compared to that due to the free-streaming term despite the threshold being 7 times the target temperature. One velocity cell with $v_\parallel \approx 3.54T_e, v_\perp \approx 3.54T_e$ (T_e is the local electron temperature) is selected to show the Δf_e balance between different terms. As one can see in Fig. 3.15, for both cases with two different radiation schemes, the Δf_e balance in this velocity cell is mainly between the free-streaming and collision terms, and the distribution function change Δf_e due to the radiation is several orders of magnitude lower than the main balance terms. Therefore the difference between numerical schemes in the radiation term for Cases A and B does not result in any difference in the balance of heat carrying electrons (HCE) velocity cells and the high energy tails, thus leading to similar transport coefficients, resulting in similar macroscopic parameter profiles.

A somewhat surprising result that the two methods of energy input give close results for the heat transport can be additionally explained by two factors. First, across most of the spatial grid, not very close to the target, energy even in the Case B is being removed mostly from thermal electrons, as local T_e is much higher than that near the target and 'high energy tail' removal is done actually from bulk (thermal) electrons. Closer to the target, the ∇T_e has already been formed by sources upstream. And near the target, the region where T_e is low enough so that the difference between the two methods of power removal could lead to two different distribution function, is too narrow to make an impact on ∇T_e, so the two T_e profile look almost identical on the figure.

Summary

Simulation results confirm that kinetic effects of electron parallel transport are primarily attributed to non-local transport of high energy electrons characterized by the extended high energy tail in the downstreaming direction. A significant deviation of the kinetic solution from the fluid one is seen in the region about $5\lambda_t$ from the target, meaning that this region might require special treatment in the coupling scheme.

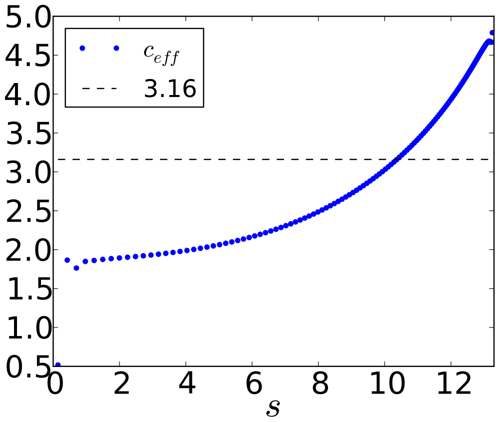


Figure 3.17: Profile of the effective heat conduction coefficient defined by Eq. (3.61).

In a steady state, $q_{e\|}$ and hence plasma parameter profiles are mainly determined by the spatial distribution of energy and particle sources. For the above simulation cases, they are rather insensitive to kinetic details of where (in which location of f_e in velocity space) sources are introduced.

The profile of an effective heat conduction coefficient c_{eff} from the case with enhanced radiation is shown in Fig. 3.17. c_{eff} is defined as the ratio of conductive heat flux calculated in KIPP to the Braginskii one, multiplied by 3.16:

$$c_{eff} = 3.16 \times \left(\frac{1}{2} m_e \int f_e v^2 v_\| d\vec{v} - \frac{5}{2} T_e \Gamma_{e\|} \right) / q_{e\|}^{\mathrm{Brag}} \tag{3.61}$$

where $\Gamma_{e\|}$ is electron particle flux density and $q_{e\|}^{\mathrm{Brag}}$ is the Braginskii electron conductive heat flux density calculated based on the KIPP electron temperature profile. Contrary to values of heat flux limiter being strongly and non-uniformly varied along the poloidal direction, this strong dependence on local collisionalities found in previous PIC kinetic simulations [52, 53], the effective heat conduction coefficient from upstream to downstream calculated in KIPP is varied smoothly. This encourages us to attempt to do coupling by replacing the local c_e in SOLPS with the effective c_{eff} calculated in KIPP.

Chapter 4

Iterative coupling of KIPP with SOLPS

As discussed in section 3.1, the fluid description of electrons is not always valid in the SOL, kinetic corrections are necessary for arbitrary collisionalities. Since the fluid model in SOLPS is already highly sophisticated with self-consistent recycling, physical and chemical sputtering, as well as atomic physics models, which are the most time-consuming parts of a kinetic code, KIPP was developed to account for kinetic effects of electron parallel transport in SOLPS, leaving the rest of the physical models intact. An algorithm of coupling KIPP with SOLPS will be investigated in this chapter.

4.1 Iterative coupling algorithm

4.1.1 Justification for the iterative algorithm

One difficulty with coupling a kinetic code to a fluid one is that they have very different time scales. For a plasma with constant $T = 50eV$, $n = 1.0 \times 10^{19}\mathrm{m}^{-3}$ along a 25m flux tube, the time scale of the fluid model used in a fluid code is characterized as the upstream ion transport time $\sim 10^{-4}$s, however, the time scale of a kinetic code without resolving the gyro-motion can be characterized as the electron collision time at the target $\sim 10^{-7}$s. The direct real-time coupling is obviously not possible [113]. One can think of an iterative coupling algorithm instead, which may offer a possible way of coupling a kinetic code to a fluid one. In the following sections, an iterative coupling algorithm will be investigated.

4.1.2 Analytical analysis of the iterative coupling scheme

One iterative coupling scheme, which avoids the difficulty caused by different time scales, is proposed here. SOLPS passes converged profiles of macroscopic plasma parameters to KIPP, while KIPP passes effective kinetic factors back

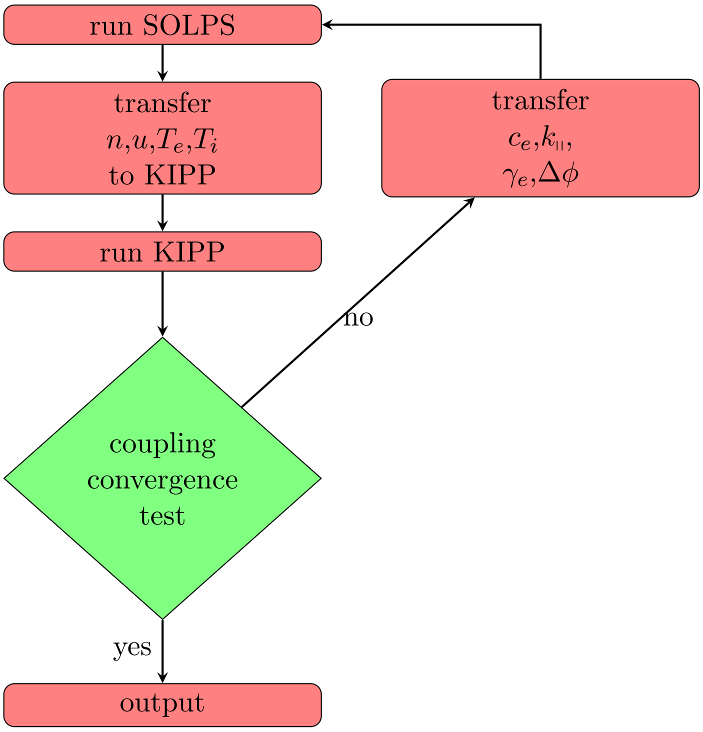


Figure 4.1: The schematic of the KIPP-SOLPS coupling algorithm.

to SOLPS. This process is repeated until coupling convergence is reached. The flowchart of this process is shown in Fig. 4.1. It consists of 4 main steps:

1. At the start of a coupled case, run SOLPS with default electron heat conduction coefficients ($c_e = 3.16$), thermal force coefficients ($k_{\parallel} = 0.71$), sheath potential drop (Eq. 3.28) and electron sheath heat transmission coefficient (Eq. 3.29). For a continuation of a coupled case, run SOLPS with the modified (effective) electron heat conduction coefficients (c_{eff}), thermal force coefficients (k_{eff}), sheath potential drop ($\Delta\phi_{eff}$) and electron sheath heat transmission coefficient (γ_{eff}). Calculations of the effective kinetic factors will be given below.

2. Transfer profiles of electron density, velocity and temperature, ion density and temperature, particle flux density from SOLPS to KIPP.

3. Maintaining the transferred profiles by automatic energy and particle sources, run KIPP and obtain new effective electron heat conduction coefficients (c_{eff}), thermal force coefficients (k_{eff}), sheath potential drop ($\Delta\phi_{eff}$) and electron sheath heat transmission coefficient (γ_{eff}).

4. Check if the coupling convergence has been reached: if yes, output all profiles and coefficients; if not, transfer kinetic factors back to SOLPS and continue with Step 1.

The iterative coupling is automatically carried out by some user-specified parameters. In this section, analytical formulas of the modifications of the kinetic factors by KIPP are described, while more numerical details of the coupling algorithm can be found in Appendix B. The above mentioned Step 3 is the most challenging since fluid equations in SOLPS and the kinetic equation in KIPP are not explicitly related to each other. The correspondence of terms involved to calculate the effective kinetic factors should be clarified. The following part of this section first compares equations solved in SOLPS and KIPP separately and then derives formulas for calculating effective kinetic factors in KIPP.

Comparison of electron balance equations between KIPP and SOLPS

The main goal of the iterative scheme is to incorporate kinetic effects of parallel electron transport calculated by KIPP into SOLPS. Equations solved in SOLPS and KIPP are compared below.

Equations solved for electrons in SOLPS Since KIPP is only tackling electron parallel transport, while, as discussed in section 2.4, 1d SOLPS transport is essentially the sum of poloidal projections of parallel and perpendicular transport, the particle and heat flux densities calculated based on f_e in KIPP here correspond to parallel parts in Eqs. (2.73) and (2.74). The electron equations (Eqs. (2.76), (2.58) and (2.80)) solved in the SOLPS 1D geometry are shown again here with moving the perpendicular terms to the right hand side and replacing ∇_x terms with $\nabla_\parallel$ and $\nabla_\perp$ terms (this will be discussed later in this paragraph):

$$\frac{\partial n_e}{\partial t} + \nabla_\parallel \left(\Gamma_{e\parallel}\right) = S_p - \nabla_\perp \Gamma_{e\perp} \tag{4.1}$$

$$e n_e E_\parallel = \boldsymbol{R_{T_\parallel}} - \nabla_\parallel p_e \tag{4.2}$$

$$\begin{aligned} & \frac{\partial}{\partial t}\left(\frac{3}{2} n_e T_e\right) + \nabla_\parallel \left(\frac{3}{2} n_e T_e u_{e\parallel} + \boldsymbol{q_{e\parallel}^{\mathrm{cond}}}\right) \\ = \; & -n_e T_e \nabla_\parallel u_{e\parallel} - Q_\Delta + S_{Ee} - \nabla_\perp q_{e\perp} \end{aligned} \tag{4.3}$$

with the particle and heat flux densities in the perpendicular direction:

$$\Gamma_{e\perp} = \Gamma_{e\perp}^{\mathrm{a}} \tag{4.4}$$

$$q_{e\perp} = \frac{5}{2} \Gamma_{e\perp} T_e - \chi_e^{\mathrm{E}} n_e \nabla_\perp T_e \tag{4.5}$$

where only anomalous transport is left, and the closure equations:

$$\boldsymbol{q_{e\parallel}^{\mathrm{cond}}} = \boldsymbol{-c_e n_e \tau_e \frac{T_e}{m_e} \nabla_\parallel T_e} \tag{4.6}$$

$$\boldsymbol{R_{T_\parallel}} = \boldsymbol{-k_\parallel n_e \nabla_\parallel T_e} \tag{4.7}$$

where $c_e = 3.16$, $k_\parallel = 0.71$ (default values for all species under the choice of "Braginskii" model in SOLPS). S_p and S_{Ee} are volumetric particle and energy sources, $E_{\|}$ is the electric field, and Eq. (4.2) gives the profile of electric potential. We remind that gradients along x can be expanded as follows:

$$\begin{aligned} \nabla_x V_x &= \nabla_x b_x V_{\|} + \nabla_x b_z V_{\perp} \\ &= \nabla_{\|} V_{\|} + \nabla_{\perp} V_{\perp} \end{aligned} \tag{4.8}$$

since gradients in the toroidal direction are assumed zero. $\vec{V}$ is a vector, which can be particle or heat fluxes (flux densities), with only the poloidal component present in the 1D geometry. Hence the terms $\nabla_{\|} V_{\|}$ and $\nabla_{\perp} V_{\perp}$ in Eqs. (4.1), (4.2) and (4.3) are from the transformations of the term $\nabla_x V_x$.

Equations solved for electrons in KIPP The main equation to be solved in KIPP is the Vlasov-Fokker-Planck equation for electron parallel transport shown again here (not dimensionless):

$$\frac{\partial f_e^{\mathrm{k}}}{\partial t} + v_{\|} \nabla_{\|} f_e^{\mathrm{k}} - \frac{e E_{\|}}{m_e} \frac{\partial f_e^{\mathrm{k}}}{\partial v_{\|}} = \left(\frac{\partial f^{\mathrm{k}}}{\partial t} \right)_{\mathrm{coll.}} + S_{\mathrm{energy}} + S_{\mathrm{particle}} \tag{4.9}$$

where $f_e^{\mathrm{k}}(v_{\|}, v_{|}, s)$ is the 3D electron distribution function: 2D in velocity space and 1D in real space along the magnetic field. KIPP can be easily adapted to evolving f_e^{k} while maintaining fixed density and temperature profiles by specifying automatic energy and particle sources. The first three moment equations following from Eq. (4.9) in KIPP are:

$$\frac{\partial n_e^{\mathrm{k}}}{\partial t} + \nabla_{\|} \left(\Gamma_{e\|}^{\mathrm{k}} \right) = S_p^{\mathrm{k}} \tag{4.10}$$

$$e n_e^{\mathrm{k}} E_{\|}^{\mathrm{k}} = \boldsymbol{R}_{\boldsymbol{T}_{\|}}^{\mathbf{k}} - \nabla_{\|} p_e^{\mathrm{k}} \tag{4.11}$$

$$\frac{\partial}{\partial t} \left(\frac{3}{2} n_e^{\mathrm{k}} T_e^{\mathrm{k}} \right) + \nabla_{\|} \boldsymbol{q}_{\boldsymbol{e}\|}^{\mathbf{k}} = -e n_e^{\mathrm{k}} u_{e\|}^{\mathrm{k}} E_{\|}^{\mathrm{k}} + Q_c^{\mathrm{k}} + S_{Ee}^{\mathrm{k}} \tag{4.12}$$

where all variables with the superscript k are defined in the same way as those in section 2.1 but calculated based on the distribution function in KIPP, f_e^{k}. The thermal force and electron parallel heat flux density are shown here:

$$\boldsymbol{R}_{\boldsymbol{T}_{\|}}^{\mathbf{k}} = \int m_e v' \left(\frac{\partial f^{\mathrm{k}}}{\partial t} \right)_{\mathrm{coll.}} d\vec{v} \tag{4.13}$$

$$\boldsymbol{q}_{\boldsymbol{e}\|}^{\mathbf{k}} = \frac{1}{2} m_e \int f_e^{\mathrm{k}} v^2 v_{\|} d\vec{v} \tag{4.14}$$

Modifications of kinetic factors in KIPP

As pointed out above in the Step 3 of the coupling scheme, profiles of electron density, velocity and temperature etc. are transferred from SOLPS to KIPP and maintained there:

$$n_e^{\mathrm{k}} = n_e \tag{4.15}$$

$$u_{e\parallel}^{\mathrm{k}} = u_{e\parallel} \tag{4.16}$$

$$T_e^{\mathrm{k}} = T_e \tag{4.17}$$

$$\Gamma_{e\parallel}^{\mathrm{k}} = \Gamma_{e\parallel} \tag{4.18}$$

by adjusting S_p^{k} and S_{Ee}^{k}.

Effective thermal force coefficient Eqs. (4.10) and (4.11) are consistent with Eqs. (4.1) and (4.2) which are solved in SOLPS. The profile of effective thermal force coefficient can be obtained at the end of each KIPP loop (the quasi-steady state is supposed to be achieved, details can be found in section 4.3.1). This coefficient is defined as:

$$\begin{aligned} \boldsymbol{k_{eff}} &= R_{T_\parallel}^{k} / \left(-n_e \nabla_\parallel T_e\right) \\ &= \int m_e v' \left(\frac{\partial f^{\mathrm{k}}}{\partial t}\right)_{\text{coll.}} d\vec{v} / \left(-n_e \nabla_\parallel T_e\right) \end{aligned} \tag{4.19}$$

In a coupling steady state (which means that SOLPS runs, with the modified kinetic factors that are calculated in KIPP based on the profiles from the previous SOLPS run, to reach convergence and the profiles out of it are the same as those from the previous run) where the time-dependent terms can be dropped, the electron perpendicular particle flux in SOLPS is automatically included in the particle source in KIPP:

$$S_p^{\mathrm{k}} = S_p - \nabla_\perp \Gamma_{e\perp} \tag{4.20}$$

Effective heat conduction coefficient The kinetic factor, heat conduction coefficient c_e, exists in the electron energy conservation equation, however, a difficult point needs to be clarified when calculating the effective one, since the internal electron energy equation (Eq. (4.3)) is solved in SOLPS. In order to be compared with Eq. (4.3), subtracting Eq. (4.11) multiplied by $u_{e\parallel}^{\mathrm{k}}$ from Eq. (4.12) leads to:

$$\frac{\partial}{\partial t}\left(\frac{3}{2} n_e^{\mathrm{k}} T_e^{\mathrm{k}}\right) + \nabla_\parallel \left(\boldsymbol{q_{e\parallel}^{\mathrm{k}}} - n_e^{\mathrm{k}} T_e^{\mathrm{k}} u_{e\parallel}^{\mathrm{k}}\right) = -n_e^{\mathrm{k}} T_e^{\mathrm{k}} \nabla_\parallel u_{e\parallel}^{\mathrm{k}} + Q_c^{\mathrm{k}} - u_{e\parallel}^{\mathrm{k}} R_{T_\parallel}^{\mathrm{k}} + S_{Ee}^{\mathrm{k}} \tag{4.21}$$

By comparing Eq. (4.3) with Eq. (4.21), the profile of the effective heat conduction coefficient can be obtained in KIPP as:

$$\begin{aligned} \boldsymbol{c_{eff}} &= \left(q_{e\parallel}^{\mathrm{k}} - n_e^{\mathrm{k}} T_e^{\mathrm{k}} u_{e\parallel}^{\mathrm{k}} - \frac{3}{2} n_e T_e u_{e\parallel}\right) / \left(-n_e \tau_e \frac{T_e}{m_e} \nabla_\parallel T_e\right) \\ &= \left(\frac{1}{2} m_e \int f_e^{\mathrm{k}} v^2 v_\parallel d\vec{v} - \frac{5}{2} n_e T_e u_{e\parallel}\right) / \left(-n_e \tau_e \frac{T_e}{m_e} \nabla_\parallel T_e\right) \end{aligned} \tag{4.22}$$

And, in the coupling steady state, the perpendicular heat flux is automatically included in the energy source term:

$$S_{E_e}^{\mathrm{k}} = S_{E_e} - \nabla_{\perp} q_{e\perp} \tag{4.23}$$

since the energy source due to the collision term is:

$$\begin{aligned} Q_c^{\mathrm{k}} &= u_{e\|}^{\mathrm{k}} R_{T_\|}^{\mathrm{k}} + \frac{3 m_e n_e^{\mathrm{k}}}{m_i \tau_e^{\mathrm{k}}} \left(T_i^{\mathrm{k}} - T_e^{\mathrm{k}} \right) \\ &= u_{e\|} R_{T_\|} - Q_{\Delta} \end{aligned} \tag{4.24}$$

Effective sheath potential drop Now we deal with the implementation of kinetic effects into the boundary conditions. The two boundary coefficients: the sheath potential drop and electron heat transmission coefficients in SOLPS, are obtained by assuming a distribution function at the boundary (sheath edge). As pointed out in section 3.2.1, KIPP can give a self-consistent distribution function (Eq. (3.57)) at the boundary without solving Poisson's equation for the Debye sheath. And a critical velocity v_c determined by Eq. (3.56) can thus be achieved, which corresponds to a floating potential drop through the sheath which gives ambipolar flux. Hence an effective potential drop can be obtained:

$$\boldsymbol{\Delta\phi_{eff}} = \frac{m_e v_c^2}{2e} \tag{4.25}$$

The parallel heat flux density through the boundary (or at the target), calculated in KIPP, is:

$$q_{t\|}^{\mathrm{k}} = \frac{1}{2} m_e \int_{v_c}^{\infty} f_t^{\mathrm{k}} v^2 v_{\|} d\vec{v} \tag{4.26}$$

In later discussions, "boundary" and "target" have the same meaning: the right end of the simulation boundary. The left end is the stagnation point (more details about the geometry can be found in section 4.2).

Effective electron sheath heat transmission coefficient We would like to remind that the default boundary condition for the internal electron energy conservation equation (Eq. (4.3)) in SOLPS is applied as an energy sink at the guard cell nx shown in Fig. 4.2. It is not straightforward to directly calculate the effective electron sheath heat transmission coefficient γ_{eff} in KIPP. In the following discussions of this subsection we first analyse the energy balance at cells adjacent to the target and then realize the adaptation of the SOLPS boundary condition. Based on the adapted boundary condition, we then derive the formula for calculating the effective electron sheath heat transmission coefficient in KIPP.

We first analyse the energy balance of cell nx (the guard cell) in a SOLPS run. The heat fluxes on the left and right faces of the guard cell are balanced,

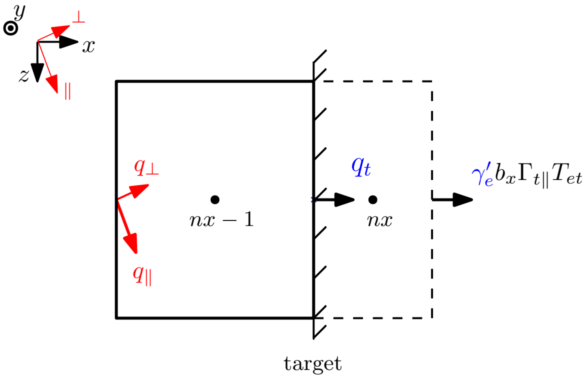


Figure 4.2: In the default SOLPS run, the boundary electron heat flux is determined by the user-specified constant c_1 or γ_e' ($\gamma_e' = c_1 + |e\Delta\phi|/T_{et}$). So $q_t = \gamma_e' b_x \Gamma_{t\parallel} T_{et}$.

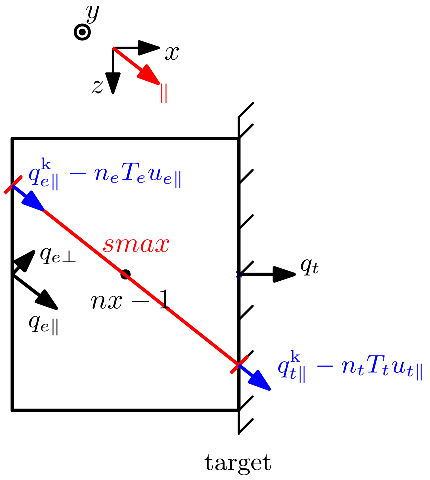


Figure 4.3: Electron heat flux densities through faces of the cell $nx-1$ in SOLPS (in black) and of the corresponding cell *smax* in KIPP (in blue). In a coupling run, q_t is supposed to be given based on the self-consistent boundary heat flux density $q_{t\parallel}^{\mathrm{k}}$.

implying that the poloidal heat flux density through the boundary is specified, as shown in Fig. 4.2, as:

$$\begin{aligned} q_t &= \gamma_e' b_x \Gamma_{t\parallel} T_{et} \\ &= \left(c_1 + \frac{|e\Delta\phi|}{T_{et}} \right) b_x \Gamma_{t\parallel} T_{et} \end{aligned} \tag{4.27}$$

where $\Gamma_{t\parallel}$ and T_{et} are electron particle flux density and electron temperature at the boundary, c_1 is a user-specified constant. γ_e' is a coefficient, defined as:

$$\gamma_e' \equiv c_1 + \frac{|e\Delta\phi|}{T_{et}} \tag{4.28}$$

It is not the electron sheath heat transmission coefficient. The relation between the coefficient γ_e' specified for the SOLPS default boundary condition and the electron sheath heat transmission coefficient will be discussed below and summarized in Appendix B.3.7.

We then analyse the energy balance of cell $nx-1$ in the steady state of a SOLPS run. One should note again that the electron heat flux density in the poloidal direction is defined according to Eq. (2.74) at cell faces for internal cells, but defined as q_t at the boundary. Based on Eq. (4.3), in the steady state, at cell $nx-1$, as shown in Fig. 4.2, the energy for electrons is balanced by heat flux densities through the left and right faces of this cell, and the terms on the right hand side of Eq. (4.2):

$$\begin{aligned}&\frac{1}{h_x}\left[q_t-\left(\left(\frac{3}{2}n_eT_eu_{e\|}+q_{e\|}^{\mathrm{cond}}\right)b_x+q_{e\perp}b_z\right)_l\right]\\&=-n_eT_e\nabla_\|u_{e\|}-Q_\Delta+S_{Ee}\end{aligned}\tag{4.29}$$

where h_x is the poloidal length of cell $nx-1$ and the subscript l denotes the variable calculated at the left face of cell $nx-1$. As it can be clearly seen, the electron heat flux density at the left face still has the form of Eq. (2.74), but it is replaced with q_t, since the right face is the simulation boundary and the boundary heat flux density cannot be given self-consistently within the fluid model, as discussed in section 3.1.

Similarly, in the steady state of a KIPP run, the energy balance can be achieved based on Eq. (4.21) at the corresponding cell (cell *smax*), as shown in Fig. 4.3:

$$\begin{aligned}&\frac{1}{h_\|}\left[\left(q_{t\|}^{\mathrm{k}}-\Gamma_{t\|}T_{et}\right)-\left(q_{e\|}^{\mathrm{k}}-\Gamma_{e\|}T_e\right)_l\right]\\&=-n_e^{\mathrm{k}}T_e^{\mathrm{k}}\nabla_\|u_{e\|}^{\mathrm{k}}+Q_c^{\mathrm{k}}-u_{e\|}^{\mathrm{k}}R_{T_\|}^{\mathrm{k}}+S_{Ee}^{\mathrm{k}}\end{aligned}\tag{4.30}$$

The parallel length of the cell in KIPP $h_\|=h_x/b_x$. The subscript l denotes the variable calculated at the left face of this cell. The details of the geometry and the relation between the grids generated in SOLPS and KIPP can be found in sections 4.2.2 and 4.2.3. The subscript l means that the corresponding vector variable is calculated at the left face of the cell. For simplicity, one can specify:

$$q_t=b_x\left(q_{t\|}^{\mathrm{k}}-\Gamma_{t\|}T_{et}\right)\tag{4.31}$$

to incorporate the kinetic boundary condition into SOLPS. In the coupling steady state, the additional heat flux density $q_{e\perp}b_z$ in SOLPS at the left face will be included in the automatic source term S_{Ee}^{k} in KIPP. However, Eq. (4.31) is not used in this work since the term $q_{e\perp}b_z$ is present in Eq. (4.29), but not explicitly in Eq. (4.30). Eq. (4.31) would result in an inconsistent source term for cell *smax* in KIPP. An adaptation of the SOLPS boundary condition can avoid this.

Before describing the adaptation of SOLPS boundary condition, one point, which we did not mention above because it would only add difficulty to discussions, has to be clarified for following discussions. The boundary electron heat flux density mentioned above is defined, in the SOLPS code, as the same form as internal cells. It consists of two parts:

$$q_t=b_xq_{t\|}+b_zq_{t\perp}\tag{4.32}$$

where $q_{t\|}$ and $q_{t\perp}$ are defined in the same forms as those at the internal cell faces but calculated based on parameters in cell $nx-1$ and the guard cell nx. This will be discussed more in detail below.

Adaptation of the SOLPS boundary condition to the coupling scheme As mentioned above, the boundary electron heat flux density in SOLPS still has the form of Eq. (2.74): $q_t = b_x q_{t\parallel} + b_z q_{t\perp}$, as shown in Fig. 4.4. In order to be compatible with the internal cells and have the same form of the source (Eq. (4.23)), the SOLPS boundary is modified as (shown in Fig. 4.4):

$$b_x q_{t\parallel} + b_z q_{t\perp} = \gamma'_{e\parallel} b_x \Gamma_{t\parallel} Tet + b_z q_{t\perp} \tag{4.33}$$

$$\longrightarrow \qquad q_{t\parallel} = \gamma'_{e\parallel} \Gamma_{t\parallel} T_{et} \tag{4.34}$$

Instead of specifying the constant c_1 in Eq. (4.27), the new boundary condition only specifies the parallel part, and then the perpendicular part is determined automatically since $q_{t\perp}/q_{t\parallel} \propto b_z \chi_{e\perp}/b_x \chi_{e\parallel}$. The numerical details and the proof that this doesn't change the physics can be found in Appendix B.3.7. Subtracting $\nabla_\perp q_{e\perp}$ from both sides of Eq. (4.29) and substitution of Eq. (4.33) into it, leads to:

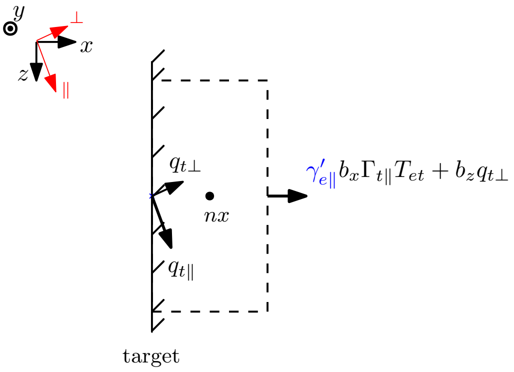


Figure 4.4: In the default boundary condition in Fig. 4.2, the boundary electron heat flux density is specified as $q_t = b_x q_{t\parallel} + b_z q_{t\perp} = \gamma'_e b_x \Gamma_{t\parallel} T_{et}$. However, in the adapted boundary condition here, it is specified as $q_t = b_x q_{t\parallel} + b_z q_{t\perp} = \gamma'_{e\parallel} b_x \Gamma_{t\parallel} T_{et} + b_z q_{t\perp}$.

$$\begin{aligned} &\frac{1}{h_\parallel}\left[\gamma'_{e\parallel}\Gamma_{t\parallel}T_{et} - \left(\frac{3}{2}n_e T_e u_{e\parallel} + q_{e\parallel}^{\text{cond}}\right)_l\right] \\ &= -n_e T_e \nabla_\parallel u_{e\parallel} - Q_\Delta + S_{Ee} - \nabla_\perp q_{e\perp} \end{aligned} \tag{4.35}$$

The adapted SOLPS boundary condition (specifying $\gamma'_{e\parallel}$) is used in the following coupling runs.

By comparing Eqs. (4.30) and (4.35), one can derive:

$$\begin{aligned} \boldsymbol{\gamma'_{\parallel,eff}} &= \frac{q_{t\parallel}^{\text{k}} - \Gamma_{t\parallel}T_{et}}{\Gamma_{t\parallel}T_{et}} \\ &= \frac{q_{t\parallel}^{\text{k}}}{\Gamma_{t\parallel}T_{et}} - 1 \\ &= \boldsymbol{\gamma_{eff}} - 1 \end{aligned} \tag{4.36}$$

where the effective electron sheath heat transmission coefficient is defined as:

$$\begin{aligned} \boldsymbol{\gamma_{eff}} &= \frac{q_{t\parallel}^{\text{k}}}{\Gamma_{t\parallel}T_{et}} \\ &= \frac{1}{2}m_e \int_{v_c}^{\infty} f_t^{\text{k}} v^2 v_\parallel d\vec{v} / (\Gamma_{t\parallel}T_{et}) \end{aligned} \tag{4.37}$$

In the coupling steady state, Eq. (4.23) is automatically fulfilled.

The difference of unity between $\gamma'_{\|,eff}$ and γ_{eff} in Eq. (4.36) is attributed to the fact that the "internal" electron energy conservation equation, as a result of subtracting $\nabla_{\|}\left(n_e T_e u_{e\|}\right)$ from both sides of the total electron energy conservation equation, is used in SOLPS. For simplicity, the following discussions in this work are based on the total electron energy conservation equation where the electron sheath heat transmission coefficient is $\gamma_{e\|}$, and $\frac{5}{2}$ is the coefficient for the convective heat flux (instead of $\gamma'_{e\|}$ and $\frac{3}{2}$):

$$\gamma_{e\|} = \gamma'_{e\|} + 1 \tag{4.38}$$

In a coupling run, $\gamma_{e\|}$ is replaced by γ_{eff}. The difference of unity between the electron sheath heat transmission coefficient $\gamma_{e\|}$ and the coefficient $\gamma'_{e\|}$ used for SOLPS "internal" electron energy conservation equation is explained numerically in Appendix B.3.7. Since the perpendicular terms have been regarded as source terms, in later discussions, heat flux density through the cell faces or the boundary only means the parallel part ($q_{e\|}$ and $q_{t\|}$).

The effective heat conduction coefficient c_{eff}, effective thermal force coefficient k_{eff}, effective potential drop $\Delta\phi_{eff}$ and effective sheath heat transmission coefficient γ_{eff} mentioned above are obtained when KIPP reaches a quasi-steady state, since a steady state like one in a fluid code is not possible in a kinetic code (this will be discussed in section 4.3.1), which here means that the heat flux through each cell face is not changing much with time, that is, the profile of c_{eff} and boundary γ_{eff} are not changing much with time. Then the effective coefficients are transferred back into SOLPS, replacing c_e, $k_{\|}$, $\Delta\phi$ and $\gamma_{e\|}$ in Eqs. (4.6), (4.7), (3.28) and (4.38). The numerical details of the iterative coupling scheme can be found in Appendix B.3.

4.2 1D geometry for the iterative coupling

4.2.1 The simulation geometry

Since we are currently mainly focusing on kinetic effects of electron parallel transport, 1D SOLPS (section 2.4) is used here in order to test the compatibility of the coupling algorithm with the KIPP code.

In ASDEX-Upgrade, major plasma radius $R_0 = 1.65$m, minor horizontal plasma radius $a = 0.5 \sim 0.52$m, minor vertical plasma radius is ~ 0.8m and the ellipticity is ~ 1.8. Therefore the perimeter of the plasma in the poloidal plane is ~ 4.5m. We assume that the average distance along the poloidal magnetic field between the two targets is ~ 5m. Half of the length is taken as the length of the simulation domain in our case since one target is assumed at one end, while the other is the stagnation point, $L_{pol} = 2.5$m. The magnetic field

$B = const$, poloidal $B_{pol} = 1.045285\times10^{-1}$T, toroidal $B_{tor} = 9.945219\times10^{-1}$T.

SOLPS and KIPP generate simulation grids separately, therefore, the grids in both codes should be generated based on the above mentioned parameters and made consistent with each other.

As mentioned in section 3.2.1, all parameters in KIPP are dimensionless. However, in later discussions, all parameters in KIPP and SOLPS will be given in dimensional form unless stated otherwise.

4.2.2 Grid cells generated in KIPP

In the KIPP code, the 1D grid cells are generated along the magnetic field based on parameters *EPSS* and L_{par}:

$$EPSS = \frac{ds(m)}{ds(m+1)} \qquad (0 \leq m < smax) \tag{4.39}$$

$$L_{par} = \sum_{m=0}^{smax} ds(m) \tag{4.40}$$

where m is the cell number and $smax$ is the highest cell number, hence there are $smax + 1$ cells. $ds(m)$ is the parallel length of cell m and L_{par} is the length of the simulation domain along the magnetic field line:

$$L_{par} = L_{pol} \times \frac{\sqrt{B_{pol}^2 + B_{tor}^2}}{B_{pol}} \tag{4.41}$$

The simulation domain in KIPP is shown as the red line in Fig. 4.5. The left end is the center of cell 0, the stagnation point, therefore only half of cell 0 belongs to the simulation domain, the other half is assumed to have mirror-reflected parameters. The height of the cells in Fig. 4.5 doesn't reflect the real grids since the real geometry in KIPP is only 1D with boundaries at the two ends. Electron temperature, density, velocity, electric field, volumetric power and particle sources or sinks are defined at cell centers, while all fluxes are defined at cell faces.

4.2.3 Grid cells generated in SOLPS

In contrast to KIPP, in SOLPS grid cells are generated along the poloidal direction. In order to be compatible with the KIPP grid, they are generated in the following way:

$$hx(0) = \frac{1}{2}ds(0) \cdot b_x \tag{4.42}$$

$$hx(m) = ds(m) \cdot b_x \qquad (0 \leq m \leq nx - 1) \tag{4.43}$$

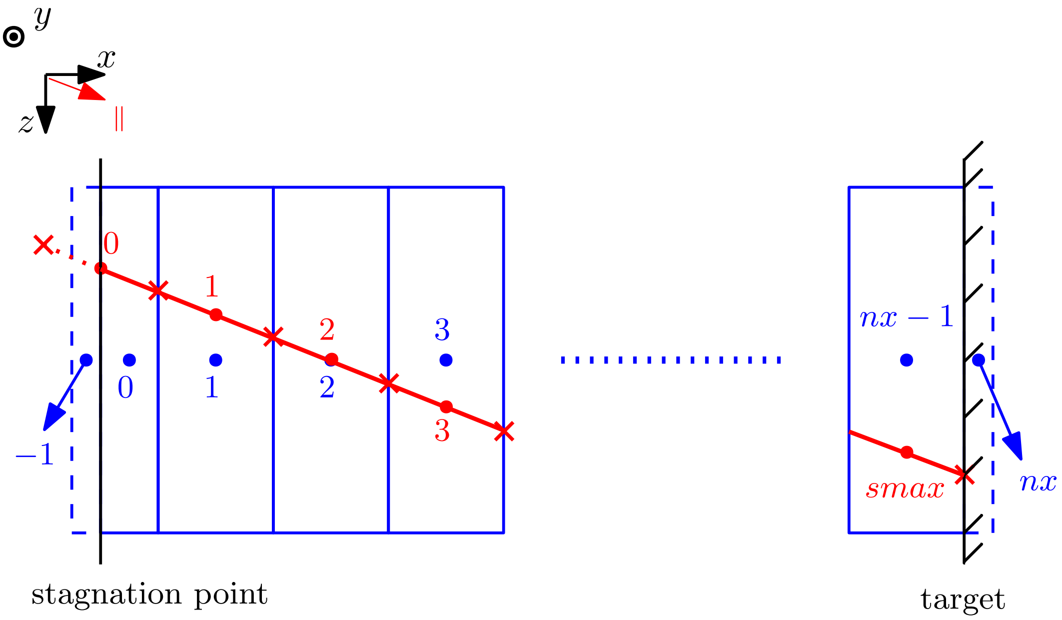


Figure 4.5: The grids generated in KIPP (red lines) and SOLPS (blue boxes). x and y are poloidal and radial directions respectively. The KIPP grid cells are generated along the direction parallel to the magnetic field with the center of cell 0 being the stagnation point and the right face of cell $smax$ being the target while the SOLPS grid cells are generated along the poloidal direction with two guard cells attached to the two ends. $nx = smax + 1$. The KIPP simulation domain is from the stagnation point (black solid line at the left end) to the target (black solid line at the right end).

where $hx(m)$ is the poloidal length of the cell m and $b_x = \frac{B_{pol}}{\sqrt{B_{pol}^2+B_{tol}^2}}$. nx is the number of cells generated in the simulation domain.

$$nx = smax + 1 \tag{4.44}$$

There are also two small guard cells attached to the two ends, numbered -1 and nx.

$$\begin{aligned} hx(-1) &= \frac{1}{1000} hx(0) && (4.45) \\ hx(nx) &= \frac{1}{1000} hx(nx-1) && (4.46) \end{aligned}$$

The guard cells are created only for implementing boundary conditions, and one should note that they are not included in the simulation domain. The comparison between the generated SOLPS and KIPP grids is shown in Fig. 4.5. The west and east boundaries are the left and right ends of the KIPP grid, respectively.

4.3 Testing the iterative coupling scheme

The simulation geometry is defined in sections 4.2.1, 4.2.2 and 4.2.3. The poloidal length of the simulation domain is 2.5m from the stagnation point to the target. The two ends in the poloidal direction are assumed to be the stagnation point and the target. The simulation domain physically corresponds to a flux tube from the upstream mid-plane position to the target with constant magnetic field projected onto the poloidal plane. The external power source is distributed evenly over cells 0 to 10 (from $x = 0$m to $x \approx 0.83$m). In order to systematically study the feasibility and performance of the iterative coupling algorithm, a series of cases with only deuterium plasma is run.

4.3.1 Coupling setup

A pure deuterium plasma is used for the convergence and density scan studies in sections 4.3.2 and 4.3.3 respectively. Carbon is introduced as an impurity to study the case with large temperature drop but comparatively low upstream collisionality in section 4.4.2. α denotes one kind of species. It can be D^0 or D^+ for pure deuterium plasma, while it can be D^0, D^+, C^0, C^+, C^{2+}, C^{3+}, C^{4+}, C^{5+} or C^{6+} for deuterium plasma with the carbon impurity. The recycling coefficient at the target is set as 0.999 for deuterium, 0 for carbon. At the stagnation point, a constant deuterium ion density is maintained by an automatic particle source while the deuterium and carbon neutral particle fluxes are specified as 0, meaning that this is a reflective boundary for neutrals. The quasi-neutrality condition is fulfilled at each time step, indicating that electron density is evolving along with the ion densities to satisfy Eq. (2.42). At the target, the velocity of each ion species satisfies the Bohm condition, but the speed of neutrals is assumed to be zero:

$$v_{\alpha\|}(nx) \begin{cases} \geq C_s & \text{if } \alpha \text{ is an ion} \\ = 0 & \text{if } \alpha \text{ is a neutral} \end{cases} \tag{4.47}$$

where $v_\alpha(nx)$ is the velocity of species α at the boundary and C_s is local acoustic speed [91]

$$C_s = \sqrt{\frac{\beta_i T_{it} + z_\alpha^2 \frac{n_\alpha}{n_e} T_{et}}{m_\alpha}} \tag{4.48}$$

where T_{it} and T_{et} are ion and electron temperatures at the boundary, β_i is a free parameter, with $\frac{5}{3}$ being the default value. Ambipolar flows are specified:

$$\begin{aligned} & j_\| &=& \ 0 & (4.49) \\ & j_\perp &=& \ 0 & (4.50) \\ \longrightarrow \quad & j_x &=& \ b_x j_\| + b_z j_\perp = 0 & (4.51) \end{aligned}$$

where j is the current density, resulting in equal ion and electron particle flux densities:

$$\begin{aligned} & \sum_\alpha z_\alpha \Gamma_{\alpha\|} &= \Gamma_{e\|} \qquad (4.52)\\ & \sum_\alpha z_\alpha \Gamma_{\alpha\perp} &= \Gamma_{e\perp} \qquad (4.53)\\ \longrightarrow \quad & \sum_\alpha z_\alpha \Gamma_{\alpha x} &= \Gamma_{ex} \qquad (4.54)\end{aligned}$$

The subscripts 'x', '$\|$', '$\perp$' have the same meaning as in section 2.4. Since only electrons are treated kinetically, the default boundary condition for the ion energy conservation equation is used by specifying the ion sheath heat transmission coefficient [91]:

$$\gamma_i = 2.5 \qquad (4.55)$$

This value is fixed during the entire run. As pointed out above, the modified boundary condition A.34 for electron energy conservation equation is used. In the initial iteration, we specify [91]:

$$\gamma_{e\|}^{\mathrm{ini}} = 1.9 + |e\Delta\phi^{\mathrm{ini}}|/T_{et} \qquad (4.56)$$

Afterwards, this value is replaced by γ_{eff} calculated from Eq. (4.37). The initial sheath potential drop $\Delta\phi^{\mathrm{ini}}$ is determined by Eq. (3.28) and then replaced by Eq. (4.25). Likewise, the coefficients for electron heat conduction and thermal force are initially specified as:

$$c_e^{\mathrm{ini}} = 3.16 \qquad (4.57)$$

$$k_\|^{\mathrm{ini}} = 0.71 \qquad (4.58)$$

and later replaced by c_{eff} and k_{eff} calculated by Eqs. (4.22) and (4.19).

Initial plasma parameter profiles in KIPP are transferred from the steady state of the initial SOLPS run. Electron distribution functions in each cell are specified initially as Maxwellian. They evolve during KIPP runs to reach a quasi-steady state. However, the concept of the quasi-steady state is not well defined, so it cannot be regarded as a convergence criteria. In practice, one can run KIPP for certain number of steps to reach a state at which the profile of an evolving parameter P is changing slowly with time, for example,

$$\left|\frac{dP^n}{dt}\right| \leq \beta \left|\frac{dP^1}{dt}\right| \qquad (4.59)$$

where dt is the time step, dP^n is the change of the parameter P during the time step n, and β is a free parameter typically. In order to get precise results in KIPP, the specified time step is chosen to be [86, 87]:

$$dt \leq 0.1\tau_0 \qquad (4.60)$$

where τ_0 is the Trubnikov collision time [81] in any cell. Because the collision time is typically smallest at the target, the condition 4.60 becomes

$$dt \leq 0.1\tau_t \tag{4.61}$$

The evolving parameter P here can be electron heat flux through cell faces or kinetic factors, but it cannot be electron density or temperature since these profiles are maintained in KIPP, not evolving with time. If $\beta = 0$, it means that a complete steady state is reached, however, this is not possible for a kinetic code. Nevertheless, based on the simulation results shown in section 4.3.3, the exact value of β seems not to influence the coupling steady state profiles, it rather influences the efficiency of reaching the coupling steady state.

4.3.2 Convergence study

Sensitivity to coupling schemes

In SOLPS the electron heat conduction coefficient c_e is defined at cell centers (details can be found in appendix B.3) while the conductive heat flux density is defined at cell faces and then calculated with a hybrid regime based on interpolations of c_e on cell faces as described in detail in appendix A. Since the numerical scheme implemented in SOLPS has been extensively studied and optimized, one should keep it as intact as possible. In KIPP the distribution function f_e is defined at cell centers while the flux term is defined at cell faces (see section 3.2.1). One must therefore make changes to the numerical schemes of either SOLPS or KIPP. Three coupling schemes with increasing changes in SOLPS are investigated (numerical details can be found in appendix B.3):

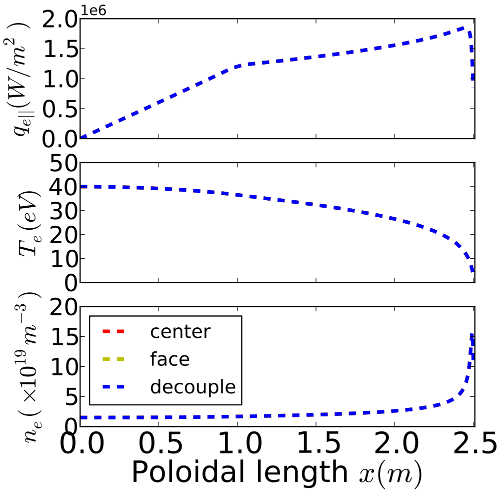


Figure 4.6: Three coupling schemes are compared. 'center', 'face', 'decouple' denote the coupling Schemes A, B, and C respectively.

Scheme A *'center'*: replace c_e in cell centers in the SOLPS part with the effective c_{eff} calculated in cell centers in the KIPP part.

Scheme B *'face'*: replace the interpolated c_e on cell faces in the SOLPS part with the effective c_{eff} calculated at cell faces in the KIPP part.

Scheme C *'decouple'*: replace the electron heat flux formula in the SOLPS part with decoupled convective and conductive pieces, making sure that the electron heat flux at each cell face is the same as the one calculated in KIPP in the coupling steady state.

The pure deuterium plasma with the stagnation point density $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$ is used to test the coupling schemes. The steady state profiles from the three coupling schemes are compared in Fig. 4.6. As expected, they almost coincide with each other. In the following test simulations, the Scheme C is used to force the parallel electron heat flux densities in SOLPS and KIPP to have the same form in order to make the testing as simple as possible. However, in future applications, the Scheme B is proposed because it attempts to keep the numerical form of the heat flux density in SOLPS intact.

Sensitivity to initial conditions

In a coupling run, the KIPP part depends on profiles provided by the SOLPS part. If the iterative coupling scheme works, different initial conditions with different specifications in the SOLPS part are supposed to give the same profiles in the coupling steady state. There are two main free parameters in SOLPS determining electron profiles:

Electron heat flux limiter $\boldsymbol{\alpha_e}$ modifying the electron parallel heat flux density according to:

$$q_{e\|} = \frac{5}{2}\Gamma_{e\|}T_e - \frac{1}{1+\frac{c_e\lambda_e}{\alpha_e L_{T_e}}}c_e\frac{n_eT_e\tau_e}{m_e}\nabla_\| T_e \tag{4.62}$$

where λ_e is electron mean free path and L_{T_e} is electron temperature scale length.

Electron sheath heat transmission coefficient $\boldsymbol{\gamma_{e\|}}$ determining the boundary electron heat flux density as discussed above.

Four cases with different combinations of initial values of α_e and $\gamma_{e\|}$ are run to investigate coupling properties:

Case A Initial run with

$$\gamma_{e\|} = 1.9 + |e\Delta\phi^{\mathrm{ini}}|/T_{et} \tag{4.63}$$

but without heat flux limiter ($\alpha_e = \infty$).

Case B Initial run with

$$\gamma_{e\|} = 4.1 + |e\Delta\phi^{\mathrm{ini}}|/T_{et} \tag{4.64}$$

but without heat flux limiter ($\alpha_e = \infty$).

Case C Initial run with

$$\gamma_{e\|} = 1.9 + |e\Delta\phi^{\mathrm{ini}}|/T_{et} \tag{4.65}$$

and with heat flux limiter $\alpha_e = 0.3$.

Case D Initial run with

$$\gamma_{e\|} = 4.1 + |e\Delta\phi^{\text{ini}}|/T_{et} \tag{4.66}$$

and with heat flux limiter $\alpha_e = 0.3$.

The heat flux limiter is removed after the initial run of SOLPS. The coupling iterations converge to the same steady state profiles as expected, as shown in Fig. 4.7. This means that results of the iterative coupling scheme doesn't depend on initial conditions. However, the convergence efficiency is better when initial profiles are closer to the steady state ones (see Fig. 4.8). In the following simulations, the initial specifications of Case A are used.

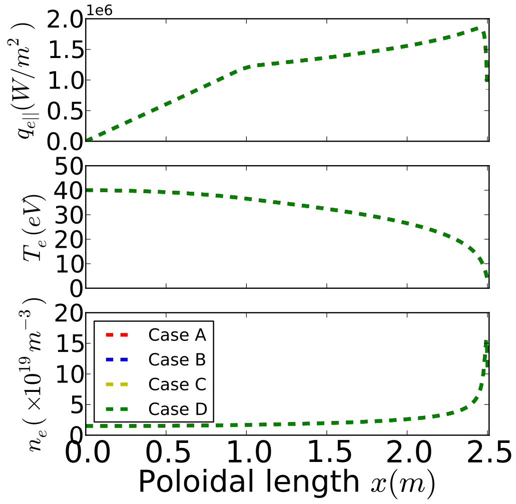


Figure 4.7: Four cases: A, B, C, D (see text for details) with the different initial specifications finally converge to the same steady state profiles. However the coupling efficiency varies (see Fig. 4.8).

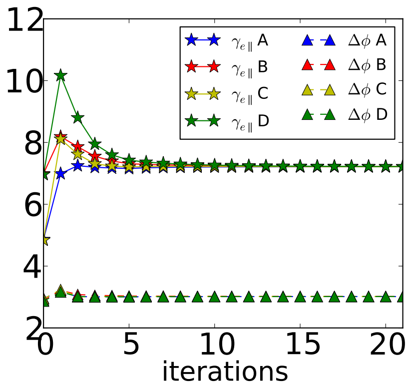


Figure 4.8: Evolution of $\gamma_{e\|}$ and $\Delta\phi$ with the coupling iterations for the four cases. $Iteration = 0$ means the values used in the initial run of SOLPS; iteration=1 means the values obtained after the initial run of SOLPS and then KIPP, and the same for $iteration = 2, 3, \cdots$. They converge to the same values (7.2 and 3.0 respectively).

4.3.3 Deuterium density scan study

As pointed out in section 3.2.2 and in reference [43], the upstream collisionality $\nu*$ is the critical parameter that determines the role of kinetic effects of parallel electron propagation. Since $\nu* \propto T_u^2/n_u$, in the coupling algorithm, the easiest way of scanning upstream collisionality is to vary upstream plasma density while keeping the same power input (keeping the upstream temperature approximately constant). In this section, a pure deuterium plasma is assumed. A series of runs scanning the stagnation point ion density from $n_u = 0.5\times10^{19}\text{m}^{-3}$ to $n_u = 2.5 \times 10^{19}\text{m}^{-3}$ is carried out. The recycling coefficient and boundary conditions are the same as those described in section 4.3.1.

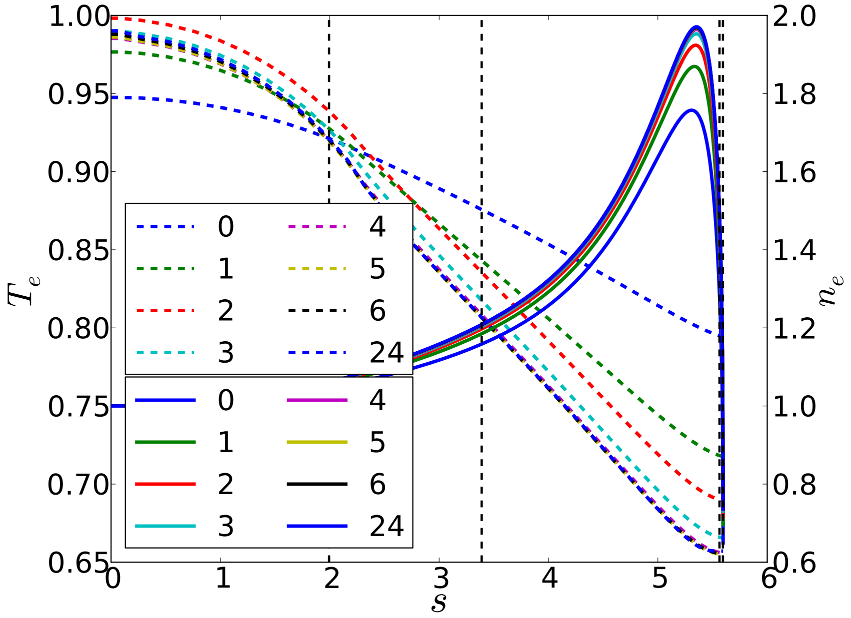


Figure 4.9: Evolution of electron density and temperature profiles for the case with the stagnation point density $n_u = 0.5 \times 10^{19}\text{m}^{-3}$. All parameters are normalized by the reference parameters (density $n_0 = 0.5 \times 10^{19}\text{m}^{-3}$, the temperature $T_0 = 47\text{eV}$, Coulomb logarithm $\Lambda_0 = 15$). The horizontal axis s is the parallel distance from the stagnation point (normalized by the electron mean free path λ_0 calculated based on the reference parameters). The colorful dashed (solid) lines are the profiles of electron temperature (density) evolving with coupling iterations. Profiles denoted by '0' means the convergent profiles after the initial SOLPS run based on default inputs. '1' means the convergent profiles achieved in SOLPS with the effective c_{eff}, γ_{eff}, k_{eff}, $\Delta\phi_{eff}$ which are calculated in KIPP based on the profiles with '0'. The four vertical black dashed lines from left to right denote locations 1, 2, 3, 4, respectively.

Low stagnation point density $n_u = 0.5 \times 10^{19}\text{m}^{-3}$

The case with the stagnation point density $n_u = 0.5 \times 10^{19}\text{m}^{-3}$ is run. The upstream collisionality (ν^*) of this case is ≈ 5.6, which is very low. The evolution of electron density and temperature profiles with coupling iterations is shown in Fig. 4.9. One complete coupling iteration means that SOLPS runs until the steady state is achieved and then KIPP runs to the quasi-steady state based on density and temperature profiles transferred from the steady state SOLPS solutions. The evolution details are not clearly seen in this figure. Four locations: 1, 2, 3, 4, at poloidal coordinates: $x \approx 0.890\text{m}$, $x \approx 1.514\text{m}$, $x \approx 2.485\text{m}$, $x \approx 2.499\text{m}$, denoted by the four black vertical dashed lines are chosen to show evolution trends of some parameters.

T_e at the four locations, the electron sheath heat transmission coefficient ($\gamma_{e\|}$) and the sheath potential drop ($\Delta\phi$) achieve the coupling steady state at *iteration* = 4, as shown in Figs. 4.10 and 4.11a. $\Delta\phi$ is insensitive to electron kinetic effects at this collisionality. $\gamma_{e\|}$ in the coupling steady state is not far

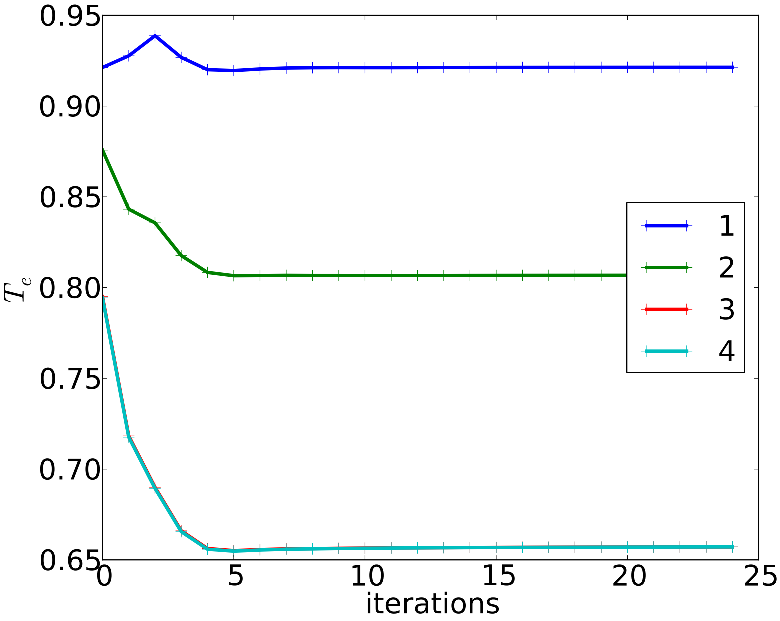
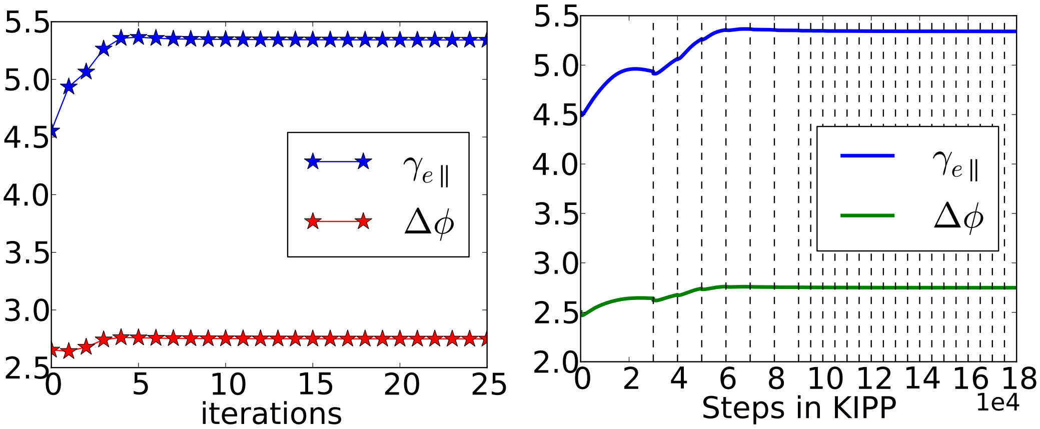


Figure 4.10: Evolution of the electron temperature with coupling iterations at the 4 locations: 1, 2, 3, 4 marked in Fig. 4.9.



(a) Evolution of $\gamma_{e\parallel}$ and $\Delta\phi$ with coupling iterations.

(b) Evolution of $\gamma_{e\parallel}$ and $\Delta\phi$ within KIPP.

Figure 4.11: Evolution of $\gamma_{e\parallel}$ and $\Delta\phi$ for the case with the stagnation point density $n_u = 0.5 \times 10^{19}\text{m}^{-3}$.

away from value 5. This is due to the fact that the electron temperature profile is rather flat ($T_0/T_t < 2$). The electron distribution function has a slightly extended high energy tail, as shown in Fig. 4.12, where the 1D distribution function is defined by integrating $f_e(v_\|, v_\perp)$ over perpendicular velocity space as:

$$F(v_\|) = \int f_e(v_\|, v_\perp) d\vec{v_\perp} \qquad (4.67)$$

The small deviation from the local Maxwellian at the high positive parallel velocity is responsible for the small increase (above 5) of the electron sheath heat transmission coefficient.

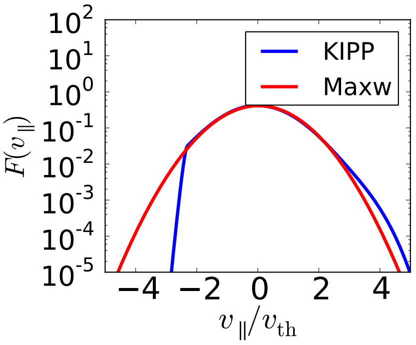
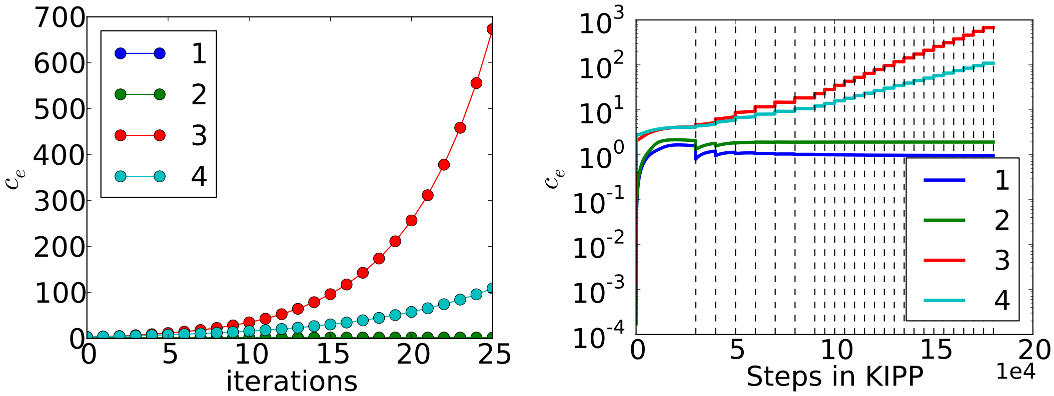


Figure 4.12: 1D electron distribution function at the cell adjacent to the target. The blue curve is the distribution function from KIPP while the red one is the local Maxwellian distribution function.

As mentioned in section 4.3.1, the quasi steady state is expected for each KIPP run. However, $\gamma_{e\|}$ doesn't fully reach the steady state in the second and third KIPP runs shown in Fig. 4.11b. The vertical dashed lines denote steps at which SOLPS is run and thus fixed density and temperature profiles in KIPP are updated. The values at the first dashed line are corresponding to the values at $iteration = 1$ in Fig. 4.13a, etc. It is not practical to set the quasi-steady state as the exit condition for KIPP since one has no prior knowledge about the evolution trend and the time required to reach the quasi-steady state. At least one should assign more time steps at the initial iteration where most significant changes are expected. The influence of the size and number of the time steps specified



(a) Evolution of c_e with coupling iterations at the 4 locations: 1, 2, 3, 4 marked in Fig. 4.9.

(b) Evolution of c_e within KIPP at the 4 locations.

Figure 4.13: Evolution of c_e within KIPP at the 4 locations for the case with the stagnation point density $n_u = 0.5 \times 10^{19}\text{m}^{-3}$.

in each KIPP run on the optimization of the coupling regime might require further study. It is inferred however that the coupling steady state does not depend on these details.

Although profiles of T_e and n_e, $\gamma_{e\|}$ and $\Delta\phi$ achieve the coupling steady state at *iteration* $= 4$, as discussed above, c_e at locations 3 and 4 (close to the target in Fig. 4.9) are increasing exponentially (numerical instability) with coupling iterations (see Fig. 4.13).

The numerical instability in the low upstream density case

An exponential increase in c_e reveals the limitation of the coupling scheme. Here we investigate the reason for this phenomenon, and a new coupling scheme is discussed later in section 4.3.4.

The vertical dashed lines in Fig. 4.13b have the same meaning as those in Fig. 4.11b. The exponential increases of c_e are seen at locations 3 and 4 (see Fig. 4.13a) after *iteration* $= 4$, although profiles of T_e and n_e, $\gamma_{e\|}$ and $\Delta\phi$ already achieve the coupling steady state. Changes in c_e only occur at the initial step, afterwards they stay unchanged for the following steps in each KIPP run. This results in a flatter electron temperature profile, which, in turn, leads to larger coefficients for the next KIPP run. With coupling iterations, the coefficients in this region keep increasing while electron temperature profile becomes increasingly flatter, however, $c_e \nabla_\| T_e \approx const.$ Since the electron temperature profile is very flat in the region near the target, a flatter profile has no impact on any other parameters. This explains why the coupling steady state profiles of T_e, n_e and $\gamma_{e\|}$, $\Delta\phi$ can be achieved, while the numerical instability still exists.

The instability might be explained by fact that the prerequisite of a fluid model is violated in the region near the target for such a low collisionality case since the distribution function in this region deviates far from a Maxwellian one due to the non-local transport [48]. The Braginskii formula for the electron conductive heat flux density completely fails. A new coupling scheme is proposed and discussed in section 4.3.4 in order to avoid such numerical instability in the low collisionality case.

Medium stagnation point density $n_u = 1.0 - 2.0 \times 10^{19}\mathrm{m}^{-3}$

Here we analyse the case with the stagnation point density $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$ which is representative in showing trends for cases with $n_u = 1.0 - 2.0 \times 10^{19}\mathrm{m}^{-3}$. Profiles of electron density and temperature reach the coupling steady state already at *iteration* $= 1$ (shown in Figs. 4.14 and 4.15), although c_e near the target, $\gamma_{e\|}$ and $\Delta\phi$ require more iterations (see Fig. 4.16).

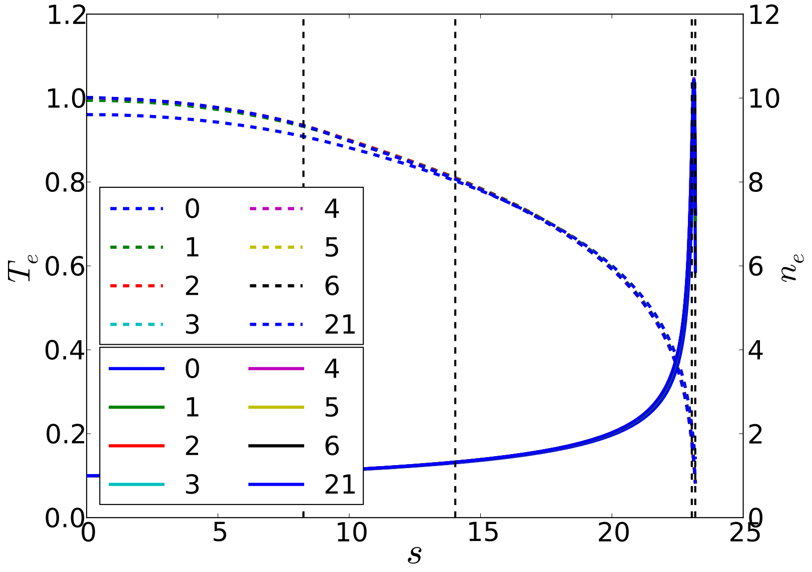


Figure 4.14: Evolution of n_e (solid) and T_e (dashed) profiles for the case with the stagnation point density $n_u = 1.5 \times 10^{19}\text{m}^{-3}$. The four vertical dashed lines from left to right denote locations: 1, 2, 3, 4, respectively, corresponding to the 4 positions: $x \approx 0.890$m, $x \approx 1.514$m, $x \approx 2.485$m, $x \approx 2.499$m in poloidal coordinate.

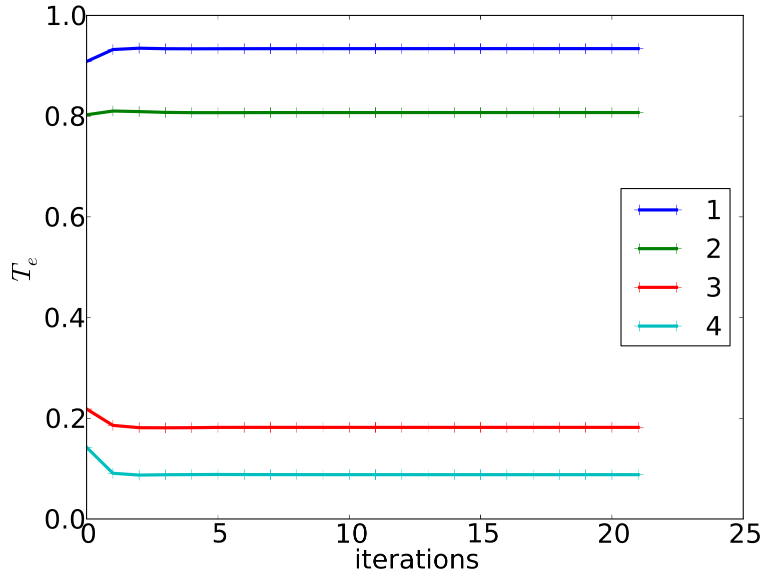
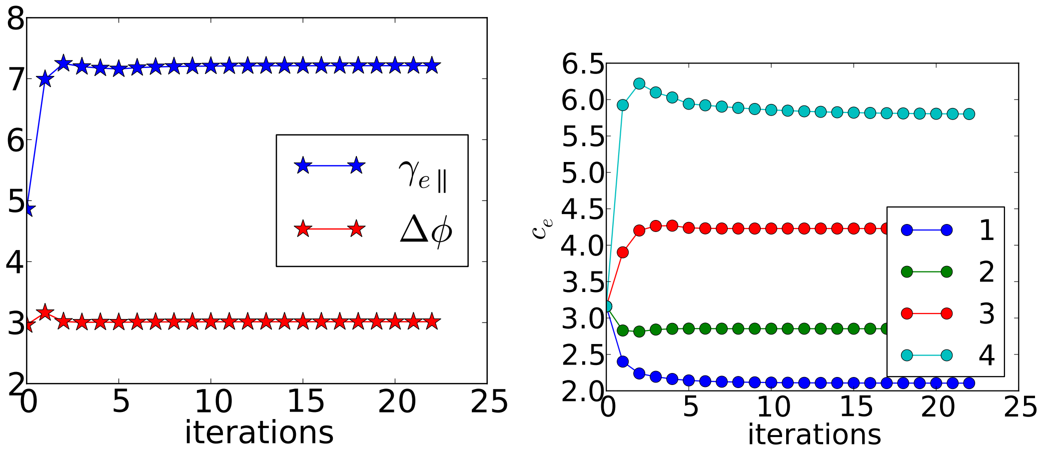


Figure 4.15: Evolution of T_e at the 4 locations: 1, 2, 3, 4 marked in Fig. 4.14.



(a) Evolution of $\gamma_{e\|}$ and $\Delta\phi$ with coupling iterations.

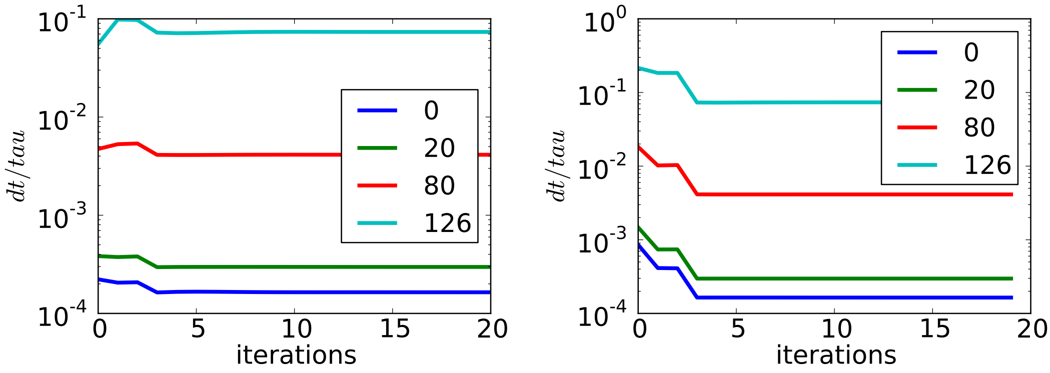
(b) Evolution of c_e at the 4 locations: 1, 2, 3, 4 marked in Fig. 4.14.

Figure 4.16: Evolution of $\gamma_{e\|}$, $\Delta\phi$ and c_e with coupling iterations for the case with the stagnation point density $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$.

Challenge posed by the steep temperature drop

When the stagnation point density increases up to $2.0 \times 10^{19}\mathrm{m}^{-3}$ and higher, c_e in the near target region converges quickly, however, the upstream parameters (heat conduction coefficient, electron temperature) evolve slowly because of the time step limitation posed by Eq. (4.61). The large difference between electron collision times upstream and downstream presents a challenge for the kinetic code by requiring large computation times. This provides a motivation for alternative ways of specifying the time step.

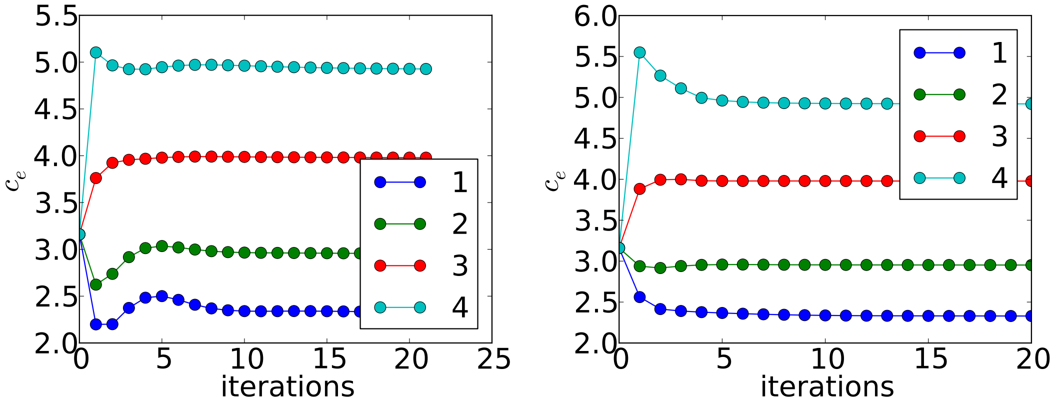
Since the initial KIPP run always requires more steps, we increase the time



(a)

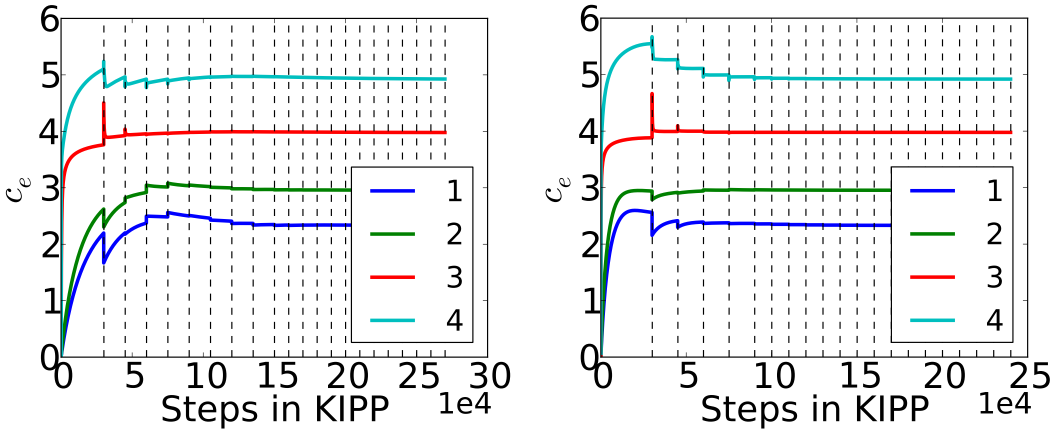
(b)

Figure 4.17: Evolution of the ratio of the specified time step to the Trubnikov collision time [81] calculated based on parameters at cell 0: the stagnation point, cell 20: upstream, cell 80: downstream, cell 126: the cell adjacent to the target for the (a) "original" and (b) "test" cases.



(a) Evolution of c_e at the 4 locations with coupling iterations in the "original" case.

(b) Evolution of c_e at the 4 locations with coupling iterations in the "test" case.



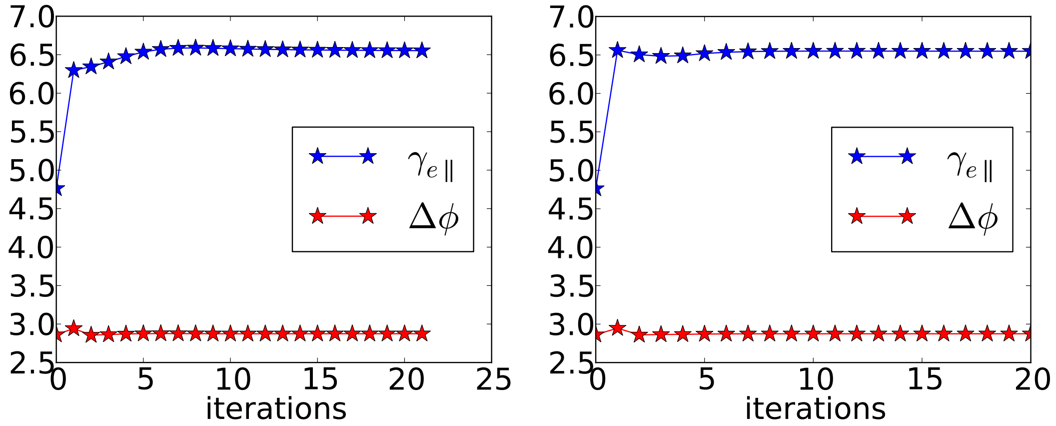
(c) Evolution of c_e at the 4 locations within KIPP in the "original" case.

(d) Evolution of c_e at the 4 locations within KIPP in the "test" case.

Figure 4.18: Comparison of the evolution of heat conduction coefficients between two cases (the "original" and "test" cases), with the stagnation point density $n_u = 2.0 \times 10^{19}\mathrm{m}^{-3}$.

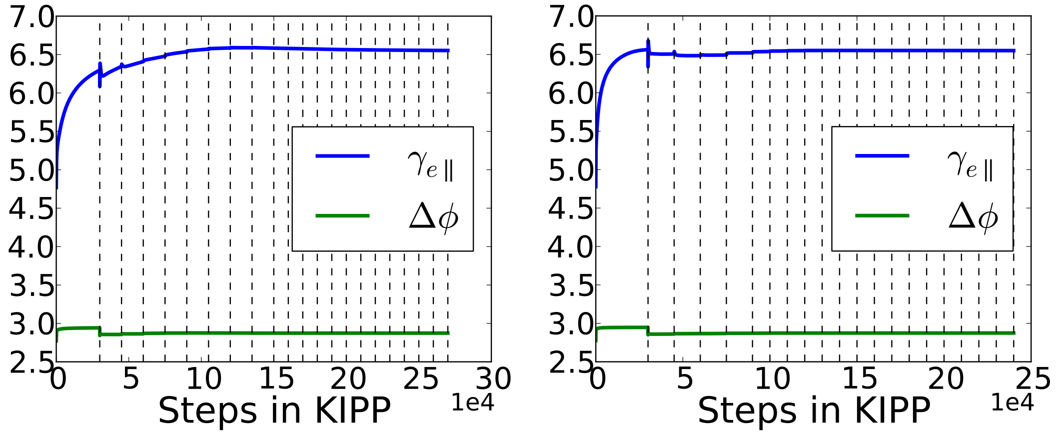
step in the initial KIPP run and then decrease it gradually to satisfy the condition given by Eq. (4.61) during last iterations. Such a case will be referred to as the "test" case below, with the standard case, with constant time step dt, referred to as the "original" case.

Two cases with $n_u = 2.0 \times 10^{19}\mathrm{m}^{-3}$: "original" and "test" cases, are run and compared below. The only difference between them is the size of time steps specified for the initial, second and third runs of KIPP (see Fig. 4.17 where the ratio of the specified time step to electron collision times calculated at cell 0: the stagnation point, cell 20: upstream, cell 80: downstream, cell 126: the cell adjacent to the target, is shown for the (a) "original" and (b) "test" cases). The two cases both reach the same coupling steady state, however, it can be clearly seen that in the "test" case, upstream c_e (at locations 1 and 2, corresponding to $x \approx 0.890$m, $x \approx 1.514$m) reaches the coupling steady state



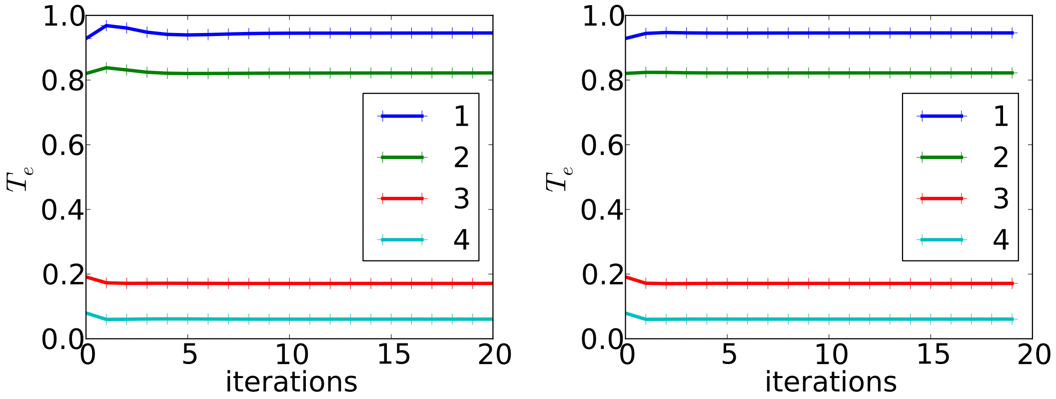
(a) Evolution of $\gamma_{e\|}$ and $\Delta\phi$ with coupling iterations in the "original" case.

(b) Evolution of $\gamma_{e\|}$ and $\Delta\phi$ with coupling iterations in the "test" case.



(c) Evolution of $\gamma_{e\|}$ and $\Delta\phi$ within KIPP in the "original" case.

(d) Evolution of $\gamma_{e\|}$ and $\Delta\phi$ within KIPP in the "test" case.



(e) Evolution of T_e at the four locations in the "original" case.

(f) Evolution of T_e at the four locations in the "test" case.

Figure 4.19: Comparison of the evolution of the electron heat transmission coefficients, sheath potential drops and electron temperature profiles between the two cases (the "original" and the "test" cases) with $n_u = 2.0 \times 10^{19}\text{m}^{-3}$.

.

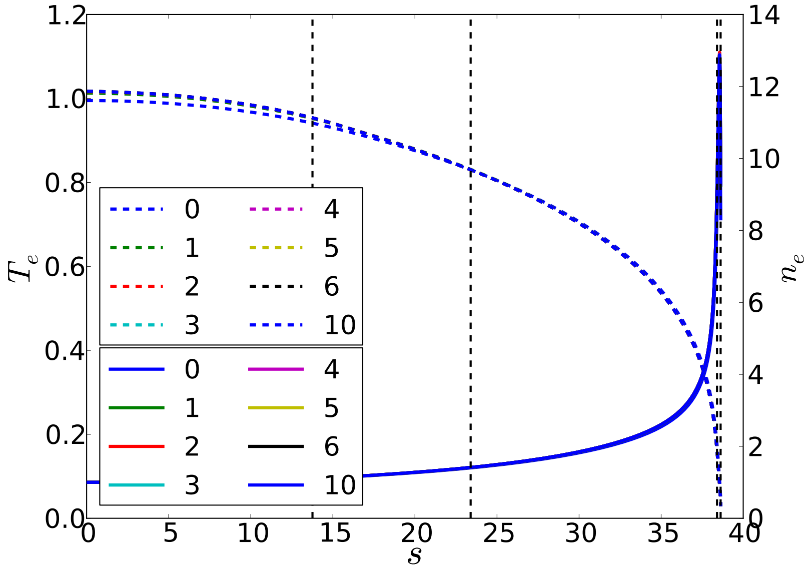


Figure 4.20: Evolution of n_e and T_e profiles. The profiles denoted by "1" are already in the coupling steady state.

faster, despite downstream c_e (at locations 3 and 4, corresponding to $x \approx 2.485$m, $x \approx 2.499$m) requires more iterations (compare Figs. 4.18b and 4.18a) than the "original" case due to larger time steps specified at the first three KIPP runs. The larger time steps in the "test" case speed up the evolution of upstream temperatures (locations 1 and 2) without impacting the downstream ones (locations 3 and 4, see Figs. 4.19e and 4.19f). It is surprising that $\gamma_{e\parallel}$ in the "test" case also converges faster (Figs. 4.19a and 4.19b, 4.19c and 4.19d) which is probably attributed to the fact that the electron heat flux transport through the boundary is only determined by the high energy electrons with $v_\parallel > v_c$, coming from the upstream region. The method adopted in the "test" case can be used for cases with substantial temperature drops.

High stagnation point density $n_u = 2.5 \times 10^{19} m^{-3}$

The "test" case method is used here for the case with the stagnation point density $n_u = 2.5 \times 10^{19}\text{m}^{-3}$. The upstream collisionality of this case is ~ 40, which is quite collisional even for Heat Carrying Electrons (HCE):

$$\nu^*_{\text{HCE}} \approx \frac{40}{25} = 1.6 > 1 \tag{4.68}$$

The profiles of n_e, T_e and $\gamma_{e\parallel}$, $\Delta\phi$ only require 2 iterations to achieve the coupling convergence as shown in Fig. 4.20. Although this case still shows heat flux limiting upstream and heat flux enhancement downstream (up to factor 1.5 near the target) shown in Fig. 4.21, the T_e profile in the coupling steady state is quite close to the fluid model result, with only a slight decrease in

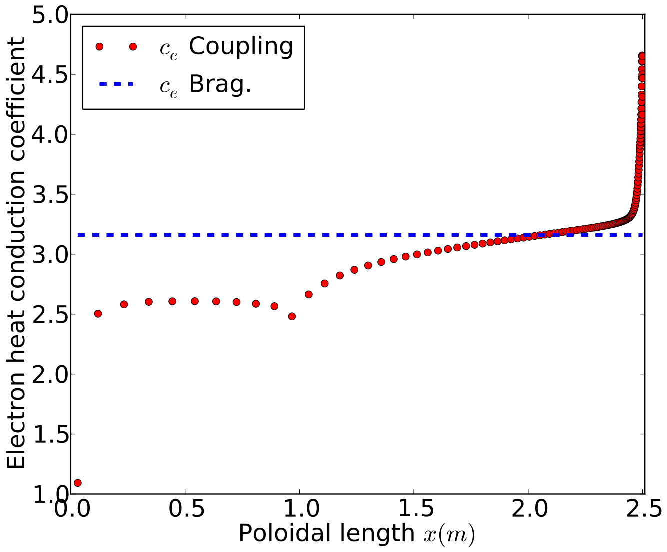


Figure 4.21: The coupling steady state profile of c_e for the case with $n_u = 2.5 \times 10^{19}\mathrm{m}^{-3}$.

target electron temperature ($\sim 25\%$, from 1.6eV to 1.15eV) and an increase in $\gamma_{e\|}$ (from 5 to 5.8). The electron distribution function f_e at the target shows a high energy tail, which however is not so pronounced to significantly impact the value of $\gamma_{e\|}$ (see section 4.4.1).

Coupling steady state profiles of n_e and T_e with higher stagnation point density are expected to converge to the fluid ones. However, cases with the electron temperature at the final cell *smax* smaller than $\sim$ 1eV will cause new numerical instability with coupling iterations. The profile of c_e will start to oscillate after several coupling iterations for reasons that are not yet clearly understood.

4.3.4 Limitations of the iterative coupling algorithm and a possible solution

The numerical instability of evolving c_e in the region near the target reveals the failure of the effort of describing conductive heat flux densities based on local temperature gradients. As shown in section 2.4, the electron parallel heat flux density in SOLPS consists of two contributions: the conductive heat flux density proportional to the local electron temperature gradient and the convective heat flux density $\frac{5}{2}\Gamma_{e\|}T_e$. Up to now, only the conductive piece was manipulated to incorporate the contribution of kinetic effects in a coupling run, which failed in the low upstream collisionality case.

As shown in Fig. 3.1, the heat flux density calculated in KIPP at a certain location consists of two parts: q_l and q_r. The electrons contributing to the downstreaming heat flux density q_l come from far upstream while those contributing to q_r come from the target reflection in very low collisional cases. The net heat flux density has obviously no direct relation to the local electron temperature gradient. Such an artificial linkage has led to an uncontrolled rise of c_e in the region near the target, as discussed earlier. Hence in this section a new numerical coupling scheme is proposed and investigated for cases with low upstream collisionality.

New scheme

Since the numerical instability occurs in the region where it shows flux enhancement ($c_{eff} > 3.16$) near the target, the main idea of the new scheme is to transfer the enhanced electron heat flux density, captured by KIPP, not only to the conductive heat flux density (by increasing c_e, causing the numerical instability in low density cases) but also to the convective piece (by increasing the coefficient, which was 2.5, in the convective heat flux formula). So the electron heat flux density in SOLPS is modified as:

$$\boldsymbol{\gamma_{v_e}} T_e \Gamma_{e\parallel} - \boldsymbol{c_{eff}} n_e \tau_e \frac{T_e}{m_e} \nabla_\parallel T_e \tag{4.69}$$

The conductive piece (the second term) is still retained because of numerical stability issue discussed later. Initially

$$\boldsymbol{\gamma_{v_e}} = 2.5 \tag{4.70}$$

$$\boldsymbol{c_{eff}} = 3.16 \tag{4.71}$$

are adopted as a default input. In a coupling run, the enhanced heat flux density calculated in KIPP is split into two parts by determining γ_{v_e} and c_{eff} in SOLPS:

$$\boldsymbol{\gamma_{v_e}} T_e \Gamma_{e\parallel} - \boldsymbol{c_{eff}} n_e \tau_e \frac{T_e}{m_e} \nabla_\parallel T_e = \frac{1}{2} m_e \int f_e^{\text{kipp}} v^2 v_\parallel d\vec{v} \tag{4.72}$$

with

$$\boldsymbol{\gamma_{v_e}} \geq \frac{5}{2} \tag{4.73}$$

$$\boldsymbol{c_{eff}} \leq c_{\text{lim}} \tag{4.74}$$

where c_{lim} is the user-specified limiting constant, determining the splitting ratio of the two parts (i.e. the values of γ_{v_e} and c_{eff}). In later discussions, we will show that the coupling steady state does not depend on the choice of c_{lim}. We define a coefficient c_e', which is the effective heat conduction coefficient when transferring all the enhanced part to the conductive piece, as:

$$c_e' = \left(\frac{1}{2} m_e \int f_e^{\text{kipp}} v^2 v_\parallel d\vec{v} - \frac{5}{2} T_e \Gamma_{e\parallel} \right) / \left(-n_e \tau_e \frac{T_e}{m_e} \nabla_\parallel T_e \right) \tag{4.75}$$

Then c_e' is compared to the limiting coefficient $c_{\rm lim}$. If c_e' exceeds the limiting coefficient $c_{\rm lim}$, we limit the effective heat conduction coefficient to $c_{\rm lim}$ instead of c_e' ($c_{eff} = c_{\rm lim}$), then transfer the further enhanced part to the convective part (i.e. γ_{v_e} is increased above 2.5). Otherwise, we transfer all the enhanced flux density to the conductive piece by specifying $c_{eff} = c_e'$ and $\gamma_{v_e} = \frac{5}{2}$. The analytical expression for determining γ_{v_e} and c_{eff} can be found from the following equations:

$$\text{if } c_e' \leq c_{\rm lim} \begin{cases} \gamma_{v_e} = \frac{5}{2} \\ c_{eff} = c_e' \end{cases} \tag{4.76}$$

$$\text{if } c_e' > c_{\rm lim} \begin{cases} \gamma_{v_e} = \frac{5}{2} + \left(c_e' - c_{\rm lim}\right) \frac{\tau_e \nabla_\parallel T_e}{m_e u_{e\parallel}} \\ c_{eff} = c_{\rm lim} \end{cases} \tag{4.77}$$

The conditions $\gamma_{v_e} = 2.5$, $c_{eff} = 3.16$ must be satisfied in the collisional limit. In the case of flux limiting (normally in the upstream region), c_{eff} is reduced, $\gamma_{v_e} = 2.5$. On the other hand, c_{eff} is increased in the case of flux enhancement (downstream near the target), however, its increase is limited to the threshold $c_{\rm lim}$, with any further enhancement moved to the convective piece.

The conductive term in Eq. (4.72) is still retained since it helps to avoid the numerical instability, although it is pointed out in section 4.3.3 that the formula of conductive heat flux density completely fails in the target region with low upstream collisionality. Removing completely the conductive term was found to cause numerical instabilities in SOLPS. The other kinetic factors: $k_\parallel$, $\gamma_{e\parallel}$ and $\Delta\phi$ are still calculated by Eqs. (4.19), (4.37) and (4.25). This coupling scheme will be investigated in the following part of this section.

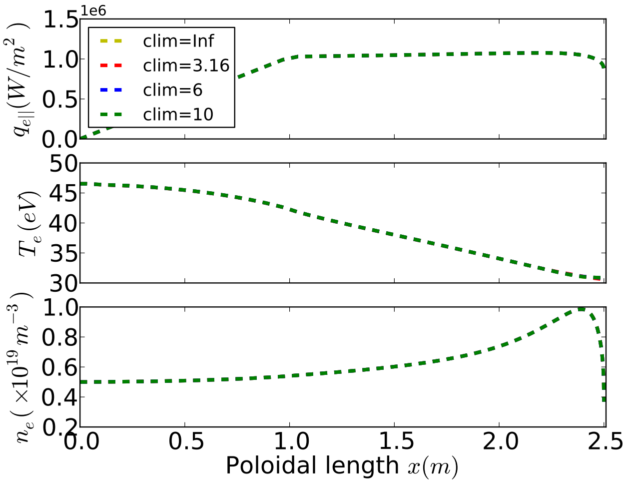
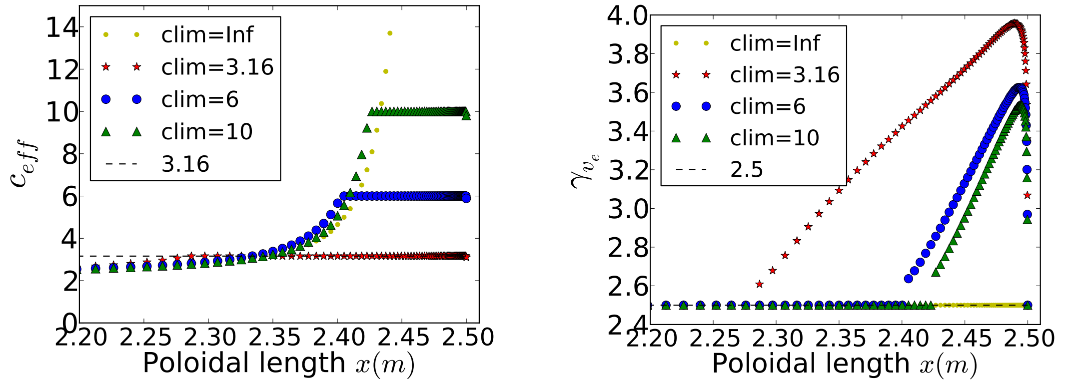
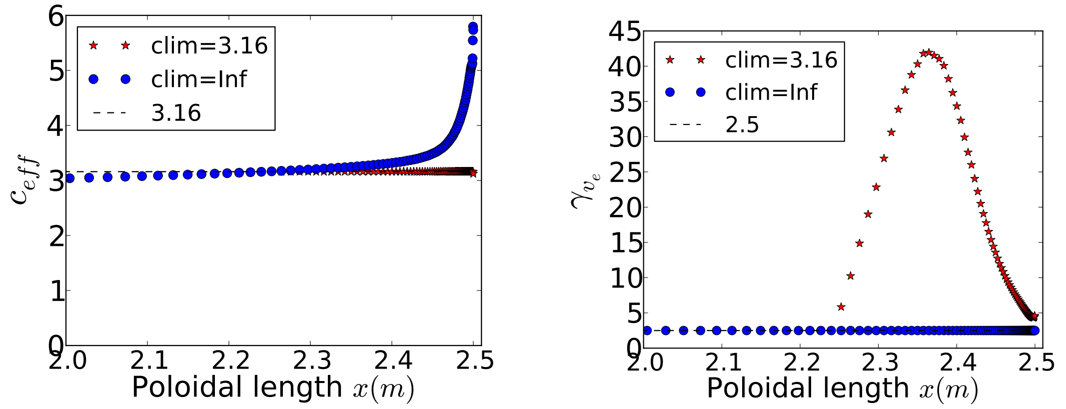


Figure 4.22: Coupling steady state profiles of $q_{e\parallel}$, T_e and n_e are compared among various limiting coefficients: $c_{\rm lim} = \infty$, $c_{\rm lim} = 3.16$, $c_{\rm lim} = 6$, $c_{\rm lim} = 10$, for the case with the stagnation point density $0.5 \times 10^{19}\text{m}^{-3}$.



(a) **(b)**

Figure 4.23: Coupling steady state profiles of c_{eff} (a) and γ_{v_e} (b) for various $c_{\lim}$ with the stagnation point density $n_u = 0.5 \times 10^{19}\mathrm{m}^{-3}$.



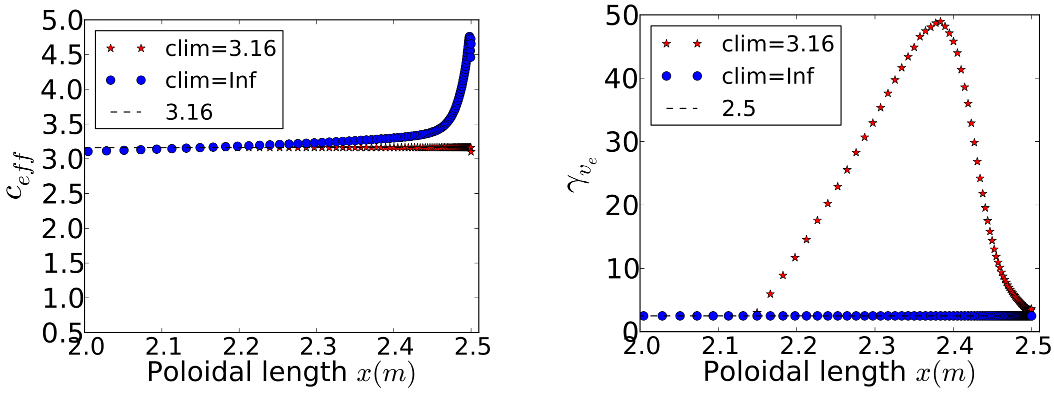
(a) **(b)**

Figure 4.24: Coupling steady state profiles of c_{eff} (a) and γ_{v_e} (b) for various $c_{\lim}$ with the stagnation point density $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$.

Testing the new scheme

Three limiting coefficients: $c_{\lim} = 3.16$, $c_{\lim} = 6$, $c_{\lim} = 10$ are tested for the case with $n_u = 0.5 \times 10^{19}\mathrm{m}^{-3}$ and then compared to the one in section 4.3.3 without the limiting coefficient ($c_{\lim} = \infty$). They all achieve the same steady state profiles as shown in Fig. 4.22. The coupling steady state profiles of c_{eff} and γ_{v_e} are compared among these three cases with various limiting coefficients in Fig. 4.23, where it can be clearly seen that c_{eff} is limited to $c_{\lim}$ near the target and γ_{v_e} in the corresponding region is increased above 2.5.

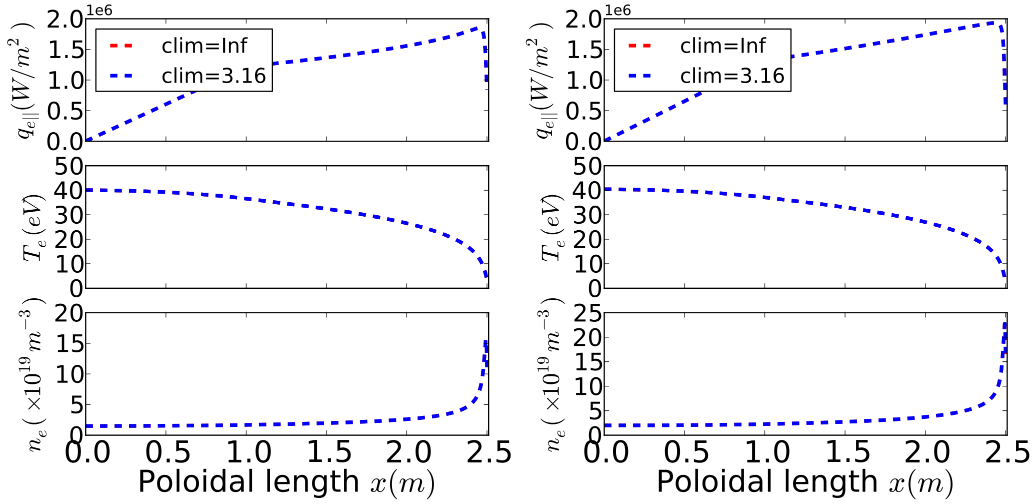
In order to further test the performance of this numerical scheme, $c_{\lim} = 3.16$ is also tested for the cases with the stagnation point densities $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$ and $n_u = 2.0 \times 10^{19}\mathrm{m}^{-3}$. The coupling steady state profiles of c_{eff} and γ_{v_e} with the limiting coefficient are compared with the ones without this limit ($c_{\lim} = \infty$) for the case with $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$ shown in Fig. 4.24



(a) **(b)**

Figure 4.25: Coupling steady state profiles of c_{eff} (a) and γ_{v_e} (b) for various c_{lim} with the stagnation point density $n_u = 2.0 \times 10^{19}\text{m}^{-3}$.

and for the case with $n_u = 2.0 \times 10^{19}\text{m}^{-3}$ shown in Fig. 4.25. As expected, the steady state profiles of n_e, T_e and $q_{e\|}$ are the same as shown in Fig. 4.26 despite a slight increase (5%) of $\gamma_{e\|}$ for the case with $n_u = 2.0 \times 10^{19}\text{m}$ and with $c_{\text{lim}} = 3.16$, resulting in slightly steeper electron temperature profile near the target.



(a) $n_u = 1.5 \times 10^{19}\text{m}^{-3}$ **(b)** $n_u = 2.0 \times 10^{19}\text{m}^{-3}$

Figure 4.26: Coupling steady state profiles of $q_{e\|}$, T_e and n_e are compared between $c_{\text{lim}} = \infty$ and $c_{\text{lim}} = 3.16$ for (a) case with $n_u = 1.5 \times 10^{19}\text{m}^{-3}$ and (b) case with $n_u = 2.0 \times 10^{19}\text{m}^{-3}$.

4.3.5 Summary on the coupling scheme

One may be suspicious of the idea that the information about non-local transport modelled by KIPP can be wrapped up into a local effective heat conduction coefficient c_{eff} in the coupling scheme. However, based on the analysis in section 4.1.2, achieving the coupling steady state of the iterative coupling scheme already means treating electrons fully kinetically. The coupling steady state profiles can always be achieved, and they seem to be unique despite the numerical instability on the evolution of heat conduction coefficients (however the heat flux density is already in the steady state since $c_{eff}\nabla_{\parallel}T_e \approx const$ with coupling iterations) in the region near the target in cases with low upstream collisionality.

The three coupling schemes: 'center' (Scheme A), 'face'(Scheme B), 'decouple'(Scheme C) give the same steady state profiles (see section 4.3.2) and Scheme B is suggested for future simulations.

The coupling steady state does not depend on initial conditions, however, the convergence efficiency is rather sensitive to the initial profile (see section 4.3.2). The initial setup with the default coefficients: $c_e = 3.16$, $k_{\parallel} = 0.71$ and $\gamma_{e\parallel} = 5$ is suggested.

For cases with the large electron temperature drop ($T_u/T_t > 10$), the necessary condition (Eq. (4.61)) limits the time step to a significantly smaller value compared to the upstream collision time τ_u:

$$dt < 0.1\tau_t \lesssim 6 \times 10^{-4}\tau_u \tag{4.78}$$

which leads to slow evolution of upstream parameters. Instead of specifying $dt < 0.1\tau_t$ for the entire coupling run, we tested the scheme with varying time step dt from $\sim \tau_t$ to $0.1\tau_t$ at the first one or two KIPP runs, which significantly accelerated the evolution of upstream parameters without affecting the downstream parameter evolution (see section 4.3.3).

The numerical instability occurring in the low density case can be avoided by splitting the enhanced heat flux into two parts and adding them to both the convective and conductive pieces in SOLPS, controlled by the limiting coefficient $c_{\lim}$ (see section 4.3.4). The most important point is that electron profiles of the coupling steady state are rather insensitive to $c_{\lim}$, further proving that the coupling solution is unique. In addition, it can be concluded that the electron temperature profile doesn't depend on the ratio of the convective to conductive heat flux densities. Instead, it is determined by the total heat flux density profile which is related to energy sources and sinks. This further justifies the conclusion that the choice of a coupling scheme is not critical: whether it wraps up non-local information into a local effective coefficient or by other possible ways, which only act as a bridge to transfer information from KIPP to SOLPS. The most important point is the achievement of the coupling steady state in the end.

Since the new scheme gives the same simulation results as the iterative coupling scheme even for the low collisionality case, later discussions in this work are based on the iterative coupling scheme (corresponding to the new scheme with $c_{\mathrm{lim}} = \infty$).

4.4 Kinetic effects in the coupling runs

4.4.1 Pure D cases

Effects of collisionality

In this section, $\gamma_{e\|}$ and $\Delta\phi$ in the coupling steady states are compared with the classical values $\gamma_{e\|} \approx 5.0$ and $-e\Delta\phi/T_e \approx 3.0$, calculated based on the Maxwellian distribution, for all scanned cases with various collisionalities, as shown in Fig. 4.27. The sheath potential drop is rather insensitive to the collisionality variation. The electron heat transmission coefficient is first increasing and then decreasing with collisionality [50, 52, 53, 99], close to the classical value for cases with low and high stagnation point densities, however, it can be $\sim 50\%$ higher than the classical value for cases with medium collisionalities. Accordingly the extended high energy tail of the distribution function downstream is pronounced at medium upstream collisionalities [48,49,51,53,75] (see Figs. 4.28f and 4.28g), but, it is not obvious at low and high collisionalities (see Figs. 4.28e and 4.28h) due either to a small temperature drop, so that upstream electrons have similar energies to those downstream, or to the plasma being too collisional causing fast maxwellization of the high energy tail appearing due to electron free-streaming [43].

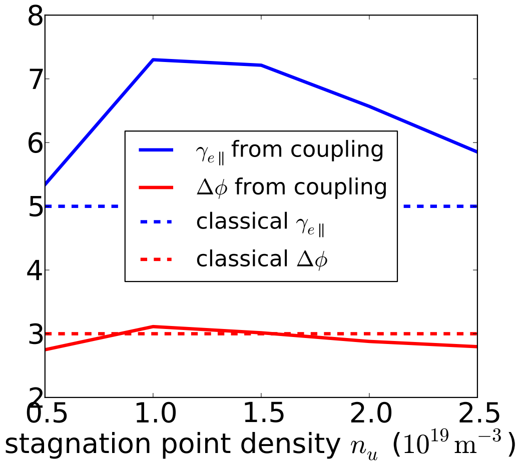


Figure 4.27: The comparison between the kinetic electron heat transmission coefficients, the sheath potential drops and the classical values.

Effects of the extended tail on the target temperature

The coupling steady state 1D distribution functions (defined by Eq. 4.67) for low ($n_u = 0.5 \times 10^{19}\mathrm{m}^{-3}$), medium ($n_u = 1.0 \sim 1.5 \times 10^{19}\mathrm{m}^{-3}$) and high ($n_u = 2.5 \times 10^{19}\mathrm{m}^{-3}$) stagnation point density cases are shown, at the cell adjacent to the target in Fig. 4.28: (e),(f),(g),(h), and at cell 10 in Fig. 4.28: (i),(j),(k),(l), respectively. We introduce a 1D heat flux distribution $q_{1\mathrm{D}}$, defined as a surface integral along a certain surface in the velocity space:

$$q_{1\mathrm{D}}(\varepsilon) = \frac{1}{2} m_e \oint_{v_\|'^2 + v_\perp^2 = 2\varepsilon} f_e v' v_\|' dS_v \tag{4.79}$$

where

$$v'_{\|} = v_{\|} - u_e \tag{4.80}$$

$$v' = \sqrt{v'^2_{\|} + v^2_{\perp}} \tag{4.81}$$

u_e is electron fluid velocity defined by Eq. 2.4. S_v is the surface satisfying $v'^2_{\|} + v^2_{\perp} = 2\varepsilon$ in the velocity space with ε being arbitrary kinetic energy. A dimensionless 1D heat flux distribution $\tilde{q}_{1\mathrm{D}}$ is thus defined as:

$$\tilde{q}_{1\mathrm{D}}(\varepsilon) = \frac{q_{1\mathrm{D}}(\varepsilon)}{q^{\max}_{1\mathrm{D}}} \tag{4.82}$$

where $q^{\max}_{1\mathrm{D}}$ is the largest value of $q_{1\mathrm{D}}(\varepsilon)$ with varying ε in the velocity space. Dimensionless 1D heat flux distributions against kinetic energy ε (normalized by local electron temperature) are calculated by the corresponding distribution functions, respectively, shown in Fig. 4.28 (red lines denote the ones calculated by the corresponding distribution functions at cell 10 and blue lines denote those at the cell adjacent to the target): (a),(b),(c),(d). By relating the extended high energy tails in Fig. 4.28:(a),(b),(c),(d) with the corresponding $\gamma_{e\|}$ for the four cases: 5.3, 7.3, 7.2, 5.8, it can be concluded that the extended high energy tails are mainly responsible for the increases of the electron heat transmission coefficient from the classical value (5.0).

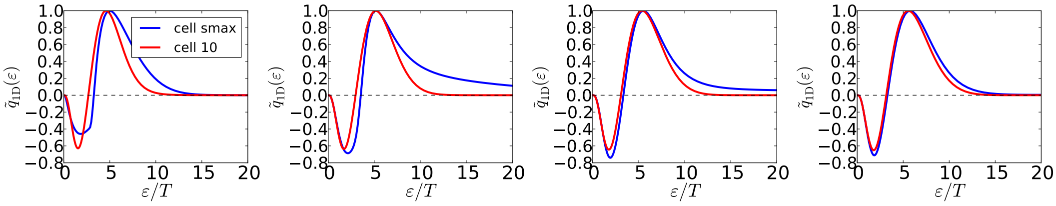
Fig. 4.29 shows the difference between the target electron temperature (T_{et}) in the coupling steady state case and that from the initial coupling iteration (which is actually the steady state in SOLPS only case without any kinetic effects) for the scanned cases. For the cases with the stagnation point density $1.0 \times 10^{19}\mathrm{m}^{-3}$ and $1.5 \times 10^{19}\mathrm{m}^{-3}$, the coupling steady state T_{et} drops by $\sim 45\%$ and $\sim 43\%$ while the sheath heat transmission coefficient increases to 7.30 (by $\sim 45\%$) and 7.21 (by $\sim 40\%$), respectively, compared to the ones without kinetic effects (see Fig. 4.30). The decrease of T_{et} due to kinetic effects seems to be related mostly to the deviation of the electron heat transmission coefficients from the classical values. This will be discussed further in section 5.1.

Summary

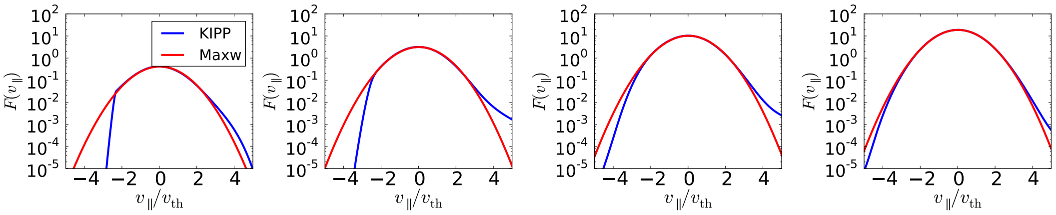
From the above discussions, it can be concluded that kinetic effects of electron parallel transport are primarily attributed to two factors [48]:

1. Non-local effects of high energy downstreaming electrons featured by the extended high energy f_e tail towards the target in medium or even highly collisional cases.
2. Asymmetric target sink characterized by a cut-off f_e in the region near the target in weakly collisional cases.

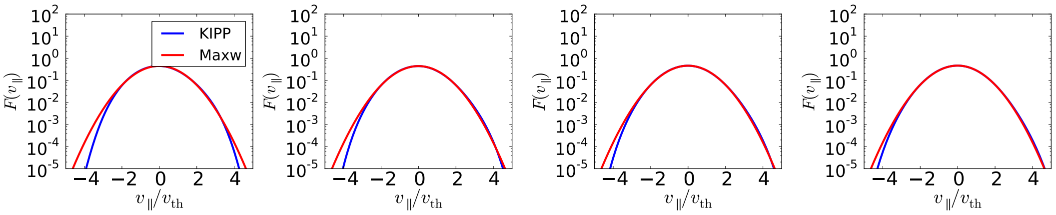
There are two key conditions that determine the importance of the non-local effects of high energy downstreaming electrons:



(a) $0.5 \times 10^{19}\mathrm{m}^{-3}$ **(b)** $1.0 \times 10^{19}\mathrm{m}^{-3}$ **(c)** $1.5 \times 10^{19}\mathrm{m}^{-3}$ **(d)** $2.5 \times 10^{19}\mathrm{m}^{-3}$



(e) $0.5 \times 10^{19}\mathrm{m}^{-3}$ **(f)** $1.0 \times 10^{19}\mathrm{m}^{-3}$ **(g)** $1.5 \times 10^{19}\mathrm{m}^{-3}$ **(h)** $2.5 \times 10^{19}\mathrm{m}^{-3}$



(i) $0.5 \times 10^{19}\mathrm{m}^{-3}$ **(j)** $1.0 \times 10^{19}\mathrm{m}^{-3}$ **(k)** $1.5 \times 10^{19}\mathrm{m}^{-3}$ **(l)** $2.5 \times 10^{19}\mathrm{m}^{-3}$

Figure 4.28: Dimensionless 1D heat flux distribution at the cell (cell *smax*) next to the target (in blue) and at cell 10 corresponding to the position at $\sim \frac{1}{3}$ of the simulation domain upstream (in red) are shown for a) $n_u = 0.5 \times 10^{19}\mathrm{m}^{-3}$, b) $n_u = 1.0 \times 10^{19}\mathrm{m}^{-3}$, c) $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$, d) $n_u = 2.5 \times 10^{19}\mathrm{m}^{-3}$. The corresponding 1D distribution functions against $v_\parallel$ are shown at cell *max* (in blue) in e,f,g,h and the ones at cell 10 are shown (in blue) in i,j,k,l, the red curves in e,f,g,h,i,j,k,l are local Maxwellian distribution functions.

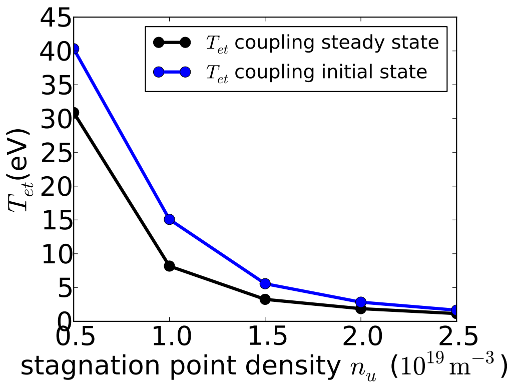


Figure 4.29: Target electron temperature of a coupling steady state (black) and of a SOLPS steady state (blue) for various cases with different stagnation point densities.

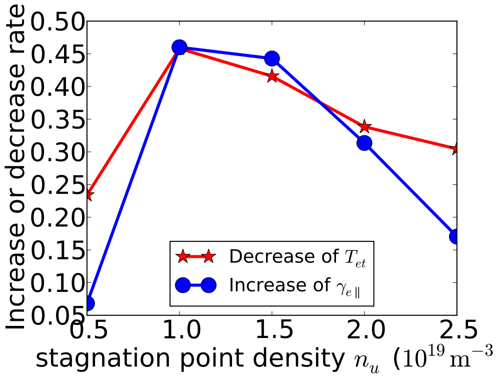


Figure 4.30: The degree of the decrease in the target Te and the increase in $\gamma_{e\parallel}$ compared to the SOLPS results (without coupling with KIPP) for various coupling cases with different n_u.

- The upstream collisionality $\nu^* = \frac{L_{par}}{\lambda_u}$.
- The electron temperature drop T_{eu}/T_{et}

Medium upstream collisionality cases with pure deuterium plasmas already show observable kinetic effects. In order to achieve stronger electron temperature drops but with unchanged upstream collisionalities, we will increase the radiation downstream by introducing impurities. Carbon impurities, used as the radiator, will be added in the following section 4.4.2.

4.4.2 D+C case

In a future fusion reactor e.g. ITER [16], the high confinement mode [114,115] with highly radiative divertor regime [116] is planned. Accordingly, in the SOL, the upstream temperature will be high, improving confinement, with a very low temperature at the target, reducing impurity sputtering, i.e. the upstream collisionality ν^* will be low and the temperature drop will be substantial, so that electron kinetic effects are expected to play a significant role. In order to investigate such a situation with low upstream collisionality but a strong temperature drop, carbon, used as a radiator, is introduced in the coupling simulation as an impurity.

Simulation setup

The upstream deuterium ion density is set at $2.2 \times 10^{19}\text{m}^{-3}$, and power input is set to match the power flux through the separatrix with constant flux density 0.132MW/m^2 from the stagnation point to $\sim \frac{1}{4}$ of the simulation domain (corresponding to the first 7 cells, from $x = 0\text{m}$ to $x \approx 0.68\text{m}$). The cells with power input are hence separated from the region where the radiation power is concentrated (downstream). Chemical and physical impurity sputtering models in SOLPS are activated to produce carbon impurities. The constant sputtering yield with deuterium sputtering coefficient 0.095 is prescribed for the chemical sputtering model. TRIM database is used for the physical sputtering model. The recycling coefficient of carbon particles (including all charged states and neutrals) is set to zero while the deuterium recycling coefficient is set to 0.999.

One problem for the 1D geometry with carbon impurity is that carbon ions, especially highly charged ions (particularly C^{4+}), tend to concentrate upstream due to thermal force since there are no sinks for them. This concentration would make the evolution unstable and cause the cases to crash even if the chemical sputtering coefficient is prescribed at a very small value. To avoid this problem, an artificial carbon ion particle sink is introduced along the simulation domain, mimicking the radial loss out of a 2D geometry. The sink loss rate is specified as:

$$S_{C^{i+}} = -r_i n_{C^{i+}} \sqrt{\frac{T_i}{m_i}} \qquad i = 0, 6 \tag{4.83}$$

where $n_{C^{i+}}$ is particle density of species C^{i+}. Carbon neutrals have $i = 0$, and $S_{C^{i+}}$ is particle loss rate of species C^{i+}. r_i is loss rate coefficient, a free dimensionless parameter to be prescribed by the user. In this case, in order to efficiently remove highly charged ions upstream, the loss rate coefficient r_i is prescribed to be proportional to the local pressure of C^{i+}, as:

$$r_i = \frac{n_{C^{i+}}(x) T_i(x) L_{pol}}{\int n_{C^{i+}}(x) T_i(x) dx} \qquad i = 3, 6 \tag{4.84}$$

$$r_i = 0 \qquad i = 0, 2 \tag{4.85}$$

where L_{pol} is poloidal length of the simulation domain and x is poloidal coordinate.

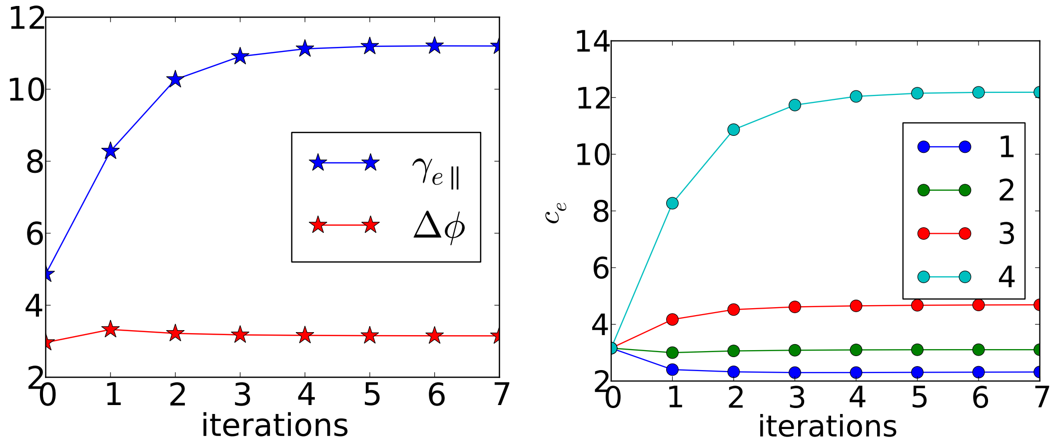
Coupling test with carbon impurity

Evolution of n_e and T_e profiles, $\gamma_{e\|}$ and $\Delta\phi$ with coupling iterations are shown in Fig. 4.31. The upstream collisionality is $\sim$ 18, comparable to the case with $n_u = 1.0 \times 10^{19}\mathrm{m}^{-3}$ of pure deuterium plasmas (section 4.3.3), however, the electron temperature drop $T_{eu}/T_{et} \approx 12.20$ here is larger by about factor 2. Such a medium upstream collisionality, but with the substantial electron temperature drop, results in an extended high energy tail of the 1D distribution function at the cell adjacent to the target, as shown in Fig. 4.32a. The contour plot of the electron conductive heat flux density and dimensionless 1D heat flux distribution at the same cell are shown in Fig. 4.33. It can be clearly seen that the contribution of high energy electrons is increased dramatically because of their non-local transport since they experience significantly fewer collisions when moving from upstream to downstream.

The extended high energy tail leads to a rather high electron sheath heat transmission coefficient $\gamma_{e\|} \approx 11.20$, more than 100% increase from the classical value, while it does not contribute much to the sheath potential drop, $\Delta\phi \sim 3.15\frac{T_e}{e}$. Accordingly, the target electron temperature is decreased by $\sim$ 60%, while the stagnation electron temperature is not much affected (see Fig. 4.31c), indicating that the kinetic boundary conditions are important in the modification of the profile of electron temperature (the reason can be found in section 5.1).

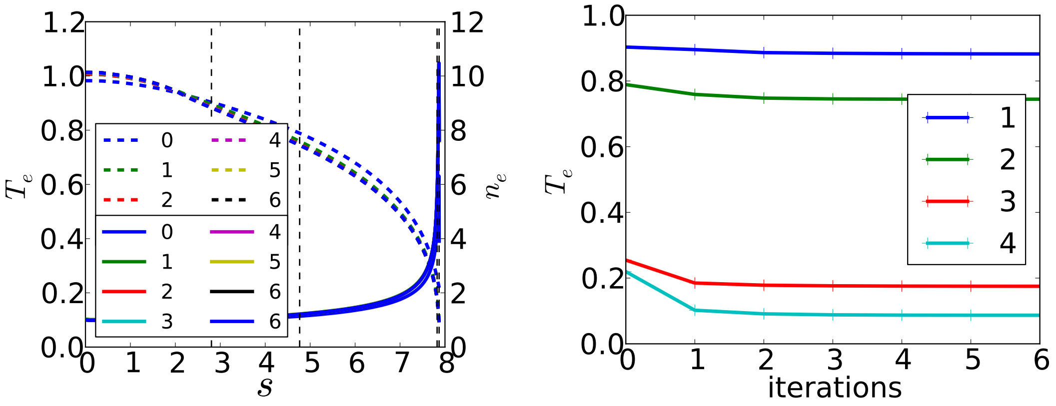
4.5 Summary

The iterative coupling algorithm has been tested and proven to be working well for various upstream collisionalities. The numerical instability, occurring



(a) Evolution of $\gamma_{e\parallel}$ and $\Delta\phi$ with coupling iterations.

(b) Evolution of c_e at the 4 locations with coupling iterations.



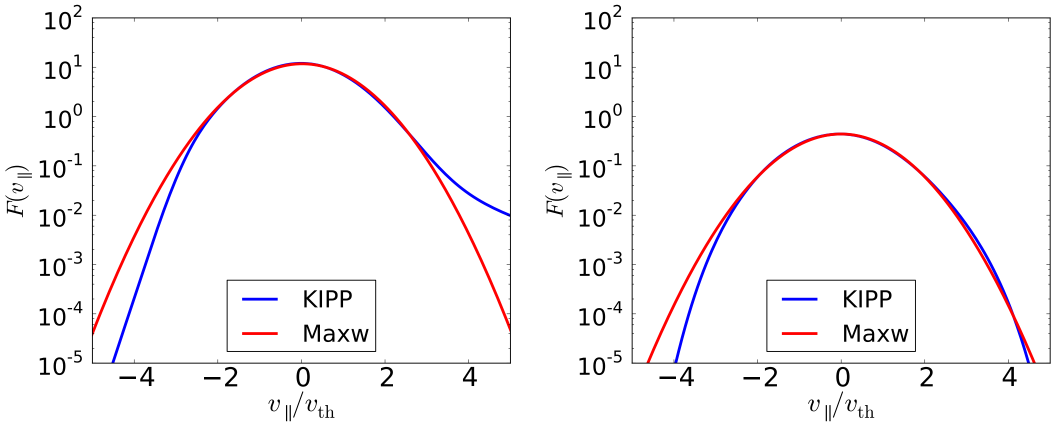
(c) Evolution of n_e and T_e profiles.

(d) Evolution of T_e at the 4 locations with coupling iterations.

Figure 4.31: Evolution of n_e and T_e profiles, c_e, $\gamma_{e\parallel}$ and $\Delta\phi$ with coupling iterations for the case with carbon impurity, with the stagnation point density $n_u = 2.2 \times 10^{19}\mathrm{m}^{-3}$. The four vertical dashed lines in (c) from left to right denote locations: 1, 2, 3, 4, respectively, where the evolution of c_e (b) and T_e (d) is shown.

in the case with low collisionalities, can be removed by the new scheme with a limiting coefficient c_{lim}. Additionally, the coupling steady state profiles of n_e, T_e and $q_{e\parallel}$ from the new scheme with various c_{lim} are found to be exactly the same as those from the iterative coupling scheme.

As was expected (see e.g. [43]), density scan cases with pure deuterium plasmas show strong kinetic effects in cases with medium upstream collisionalities. However, in real situations, radiation sinks due to impurities are always present in the divertor region. Hence the upstream collisionality ν^* is not the only free parameter to determine the role of kinetic effects. The T_e drop is also important. A case with lower collisionality and stronger temperature drop is expected to have stronger kinetic effects. This can be concluded by comparing



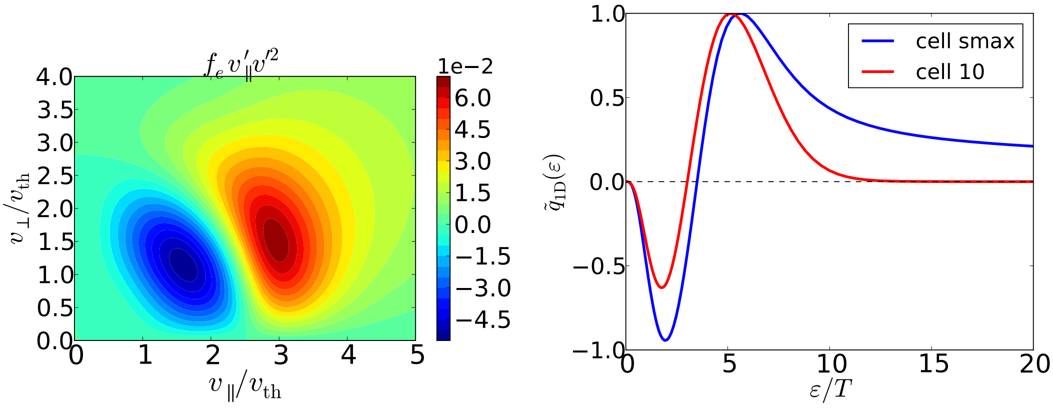
(a) Cell next to the target.

(b) Cell 10.

Figure 4.32: 1D distribution functions (Eq. (4.67)) at the cell adjacent to the target (a) and the cell 10 (b). Blue curves denote the distribution function calculated by KIPP while red ones denote local Maxwellian.

the case with the carbon impurity and the case with $n_u = 1.0 \times 10^{19}\mathrm{m}^{-3}$ with pure deuterium, which have similar upstream collisionalities $\nu^* \approx 16 - 18$ but different electron temperature drops.

For all cases run in this section, the extended high energy tail of the distribution function for the downstreaming electrons at the target results in the increase of $\gamma_{e\parallel}$ compared to the classical value. This increase, instead of



(a) Contour plot of the electron conductive heat flux density in velocity space at the cell adjacent to the target.

(b) Dimensionless 1D heat flux distribution against kinetic energy ε (normalized by local electron temperature) at the cell adjacent to the target (blue line) and at cell number number 10 (red line) corresponding to the point at the first dashed line in Fig.4.31c.

Figure 4.33: 2D and 1D electron conductive heat flux contribution from different velocity.

impacting target electron heat flux density, leads to the decrease of target electron temperature (T_{et}) from the value in the initial coupling state to the final coupling steady state (see Fig. 4.29 discussed in section 4.4.1). The relation between $\gamma_{e\|}$ and T_{et} is investigated in section 5.1.

Chapter 5

Comparison of the coupling results with SOLPS results

In Chapter 4 we demonstrated that the iterative coupling algorithm allows one to include kinetic electrons into the sophisticated fluid code, SOLPS, by treating electron parallel transport fully kinetically with KIPP and wrapping up non-local transport information into 'local' effective heat conduction coefficients.

The following nomenclature will be adopted in this section. Similar to definitions given in [52, 53], we classify the four kinetic factors into two groups: boundary factors ($\gamma_{e\|}$ and $\Delta\phi$) and flux factors (c_e and $k_\|$). The kinetic factors in a coupling steady state are indicated as $\gamma_{e\|}^{\mathrm{ss}}$, $\Delta\phi^{\mathrm{ss}}$, c_e^{ss}, $k_\|^{\mathrm{ss}}$, while those determined by Eqs. (3.29), (3.23), (2.40), (2.26) and commonly used for the fluid model, are indicated as $\gamma_{e\|}^{\mathrm{c}}$, $\Delta\phi^{\mathrm{c}}$, c_e^{c}, $k_\|^{\mathrm{c}}$. We will call $\gamma_{e\|}^{\mathrm{ss}}$ and $\Delta\phi^{\mathrm{ss}}$ "kinetic boundary factors", c_e^{ss} and $k_\|^{\mathrm{ss}}$ "kinetic flux factors", $\gamma_{e\|}^{\mathrm{c}}$ and $\Delta\phi^{\mathrm{c}}$ "classical boundary factors", and c_e^{c} and $k_\|^{\mathrm{c}}$ "fluid flux factors".

Based on the coupling simulations in chapter 4, we conclude the so-called flux enhancement downstream ($c_e^{\mathrm{ss}} > c_e^{\mathrm{c}}$) tends to flatten electron temperature rather than enhancing the heat flux. Instead of influencing the target heat flux density, the deviation of $\gamma_{e\|}^{\mathrm{ss}}$ from $\gamma_{e\|}^{\mathrm{c}}$ causes the decrease of the target electron temperature (see Fig. 4.30). This indicates, counter-intuitively, that, by flattening the electron temperature profile, non-local transport tends to reduce the target electron temperature.

In order to explain the counter-intuitive phenomena pointed out above and investigate the difference between the contributions of the two groups of kinetic factors to the modification of the T_e profile, the boundary and flux factors are separately applied to SOLPS. Then the results are compared with those obtained by applying all four kinetic factors: $\gamma_{e\|}^{\mathrm{ss}}$, $\Delta\phi^{\mathrm{ss}}$, c_e^{ss}, $k_\|^{\mathrm{ss}}$, to SOLPS.

5.1 Density scan cases: pure deuterium

Alongside the coupling cases with pure deuterium plasmas in section 4.3.3 for each stagnation point density, four SOLPS cases with and without applying kinetic boundary factors and the heat flux limiter are run:

Case A running SOLPS with fluid flux factors ($c_e = c_e^{\mathrm{c}}$, $k_\| = k_\|^{\mathrm{c}}$), classical boundary factors ($\gamma_{e\|} = \gamma_{e\|}^{\mathrm{c}}$, $\Delta\phi = \Delta\phi^{\mathrm{c}}$) and without heat flux limiter ($\alpha_e = \infty$).

Case B running SOLPS with fluid flux factors ($c_e = c_e^{\mathrm{c}}$, $k_\| = k_\|^{\mathrm{c}}$), classical boundary factors ($\gamma_{e\|} = \gamma_{e\|}^{\mathrm{c}}$, $\Delta\phi = \Delta\phi^{\mathrm{c}}$) and with heat flux limiter ($\alpha_e = 0.3$).

Case C running SOLPS with fluid flux factors ($c_e = c_e^{\mathrm{c}}$, $k_\| = k_\|^{\mathrm{c}}$), kinetic boundary factors ($\gamma_{e\|} = \gamma_{e\|}^{\mathrm{ss}}$, $\Delta\phi = \Delta\phi^{\mathrm{ss}}$) and without heat flux limiter ($\alpha_e = \infty$).

Case D running SOLPS with fluid flux factors ($c_e = c_e^{\mathrm{c}}$, $k_\| = k_\|^{\mathrm{c}}$), kinetic boundary factors ($\gamma_{e\|} = \gamma_{e\|}^{\mathrm{ss}}$, $\Delta\phi = \Delta\phi^{\mathrm{ss}}$) and with heat flux limiter ($\alpha_e = 0.3$).

Low stagnation point density $n_u = 0.5 \times 10^{19}\mathrm{m}^{-3}$

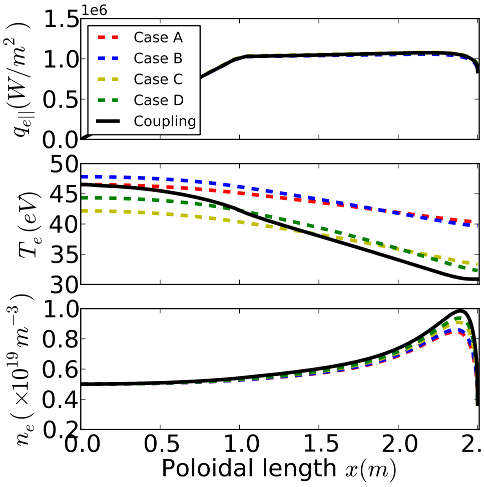


Figure 5.1: Profiles of $q_{e\|}$, T_e and n_e with the stagnation point density $n_u = 0.5 \times 10^{19}\mathrm{m}^{-3}$ are compared for the 5 cases: cases A, B, C, D and the coupling case (see text for details).

The comparison of profiles of electron density n_e, electron temperature T_e, electron heat flux density $q_{e\|}$ among the 5 cases: A, B, C, D and the coupling case with $n_u = 0.5 \times 10^{19}\mathrm{m}^{-3}$, is shown in Fig. 5.1. As pointed out above, the upstream collisionality of these cases is quite low, ~ 6. Interestingly enough, profiles of $q_{e\|}$ from all cases are the same, even the target electron heat flux densities ($q_{t\|}$) are the same, although $\gamma_{e\|}$ for cases A and B is different from that for cases C, D and the coupling case. By comparing T_e profiles from cases A, C and the coupling case, it can be inferred that the deviation of $\gamma_{e\|}^{\mathrm{ss}}$ from $\gamma_{e\|}^{\mathrm{c}}$ is mainly responsible for decreasing the target electron temperature. The flux limiting upstream ($c_e^{\mathrm{ss}} < c_e^{\mathrm{c}}$) slightly increases the stagnation point temperature while the flux enhancement ($c_e^{\mathrm{ss}} > c_e^{\mathrm{c}}$) downstream flattens the electron temperature profile.

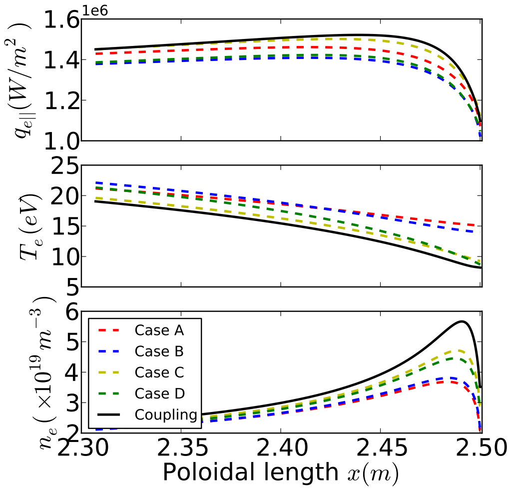


Figure 5.2: Profiles of $q_{e\parallel}$, T_e and n_e with the stagnation point density $n_u = 1.0 \times 10^{19}\mathrm{m}^{-3}$ are compared for the 5 cases: cases A, B, C, D and the coupling case (see text for details).

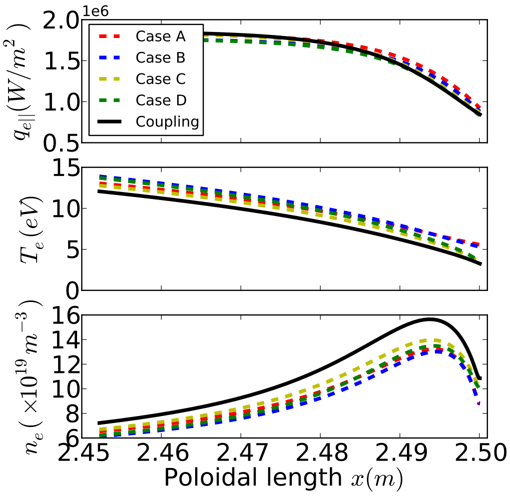


Figure 5.3: Profiles of $q_{e\parallel}$, T_e and n_e with the stagnation point density $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$ are compared for the 5 cases: cases A, B, C, D and the coupling case (see text for details).

Medium stagnation point densities

The comparison of profiles of n_e, T_e and $q_{e\parallel}$ among the 5 cases with $n_u = 1.0 \times 10^{19}\mathrm{m}^{-3}$ and $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$ are shown in Figs. 5.2 (a blow-up of the region $x = 2.30\mathrm{m} - 2.50\mathrm{m}$) and 5.3 (a blow-up of the region $x = 2.45\mathrm{m} - 2.50\mathrm{m}$), respectively. As one can see by comparing the 5 cases for $n_u = 1.0 \times 10^{19}\mathrm{m}^{-3}$ and $n_u = 1.5 \times 10^{19}\mathrm{m}^{-3}$, $q_{e\parallel}$ is insensitive to the kinetic boundary factors and heat flux limiter, however, target electron temperatures with the kinetic boundary factor (cases C, D and the coupling case) are almost half of those with the classical boundary factor (cases A and B). This strengthens the argument that the deviation of $\gamma_{e\parallel}^{\mathrm{ss}}$ from $\gamma_{e\parallel}^{\mathrm{c}}$ tends to decrease the target electron temperature rather than to modify the target heat flux. From Fig. 5.4, where the stagnation point and target electron temperatures (T_{eu} and T_{et}) with increasing n_u are shown for the 5 cases, one can conclude that the deviation of T_{et} in the coupling case from that in case A is mainly attributed to the deviation of $\gamma_{e\parallel}^{\mathrm{ss}}$ from $\gamma_{e\parallel}^{\mathrm{c}}$. The reason for this is investigated in the following subsection.

Relation between $\gamma_{e\parallel}$ and T_{et}

To explain the relation between $\gamma_{e\parallel}$ and T_{et} discussed in section 4.4.1 (see Fig. 4.30) and above, a simple analysis is given here. The target electron heat flux density is specified as:

$$q_{t\parallel} = \gamma_{e\parallel} n_t T_{et} C_s \propto \gamma_{e\parallel} n_t T_{et}^{3/2} \tag{5.1}$$

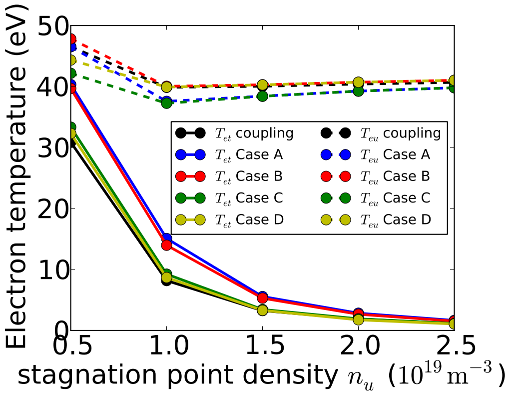


Figure 5.4: Comparison of the stagnation point electron temperature (T_{eu}, dashed) and the target electron temperature (T_{et}, solid) among the five cases: Case A, B, C, D and the coupling case for various n_u.

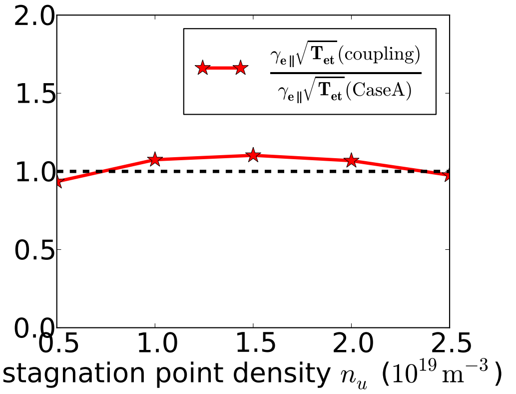


Figure 5.5: The ratio of $\gamma_{e\parallel}^{\mathrm{cpl}}\sqrt{T_{et}^{\mathrm{cpl}}}$ to $\gamma_{e\parallel}^{\mathrm{A}}\sqrt{T_{et}^{\mathrm{A}}}$ for various stagnation point densities. The horizontal black dashed line corresponds to value 1.0. This ratio is ~ 1.

where $n_t C_s$ is the target electron particle flux density. Upstream and downstream pressures can be related to each other by the two-point model [6]:

$$n_u T_{eu} \approx 2 n_t T_{et} \tag{5.2}$$

Hence the target electron heat flux density can be transformed as:

$$q_{t\parallel} \propto \gamma_{e\parallel} n_u T_{eu} T_{et}^{1/2} \tag{5.3}$$

We compare case A and the coupling case at a certain n_u. As pointed out above (see Figs. 5.1, 5.2 and 5.3), the target electron heat flux densities for case A and the coupling case:

$$q_{t\parallel}^{\mathrm{A}} \propto \gamma_{e\parallel}^{\mathrm{A}} n_u T_{eu}^{\mathrm{A}} \sqrt{T_{et}^{\mathrm{A}}} \tag{5.4}$$

$$q_{t\parallel}^{\mathrm{cpl}} \propto \gamma_{e\parallel}^{\mathrm{cpl}} n_u T_{eu}^{\mathrm{cpl}} \sqrt{T_{et}^{\mathrm{cpl}}} \tag{5.5}$$

are more or less the same:

$$q_{t\parallel}^{\mathrm{A}} \approx q_{t\parallel}^{\mathrm{cpl}} \tag{5.6}$$

where the variables with the superscript "A" are for the Case A and "cpl" for the coupling case. So,

$$\gamma_{e\parallel}^{\mathrm{A}} = \gamma_{e\parallel}^{\mathrm{c}} \tag{5.7}$$

$$\gamma_{e\parallel}^{\mathrm{cpl}} = \gamma_{e\parallel}^{\mathrm{ss}} \tag{5.8}$$

As can be seen in Fig. 5.4, $T_{eu}^{\mathrm{A}} \approx T_{eu}^{\mathrm{cpl}}$, since the upstream electron temperature depends mainly on the upstream power input. Therefore:

$$\frac{q_{t\parallel}^{\mathrm{A}}}{q_{t\parallel}^{\mathrm{cpl}}} = \frac{\gamma_{e\parallel}^{\mathrm{A}}\sqrt{T_{et}^{\mathrm{A}}}}{\gamma_{e\parallel}^{\mathrm{cpl}}\sqrt{T_{et}^{\mathrm{cpl}}}} \approx 1 \tag{5.9}$$

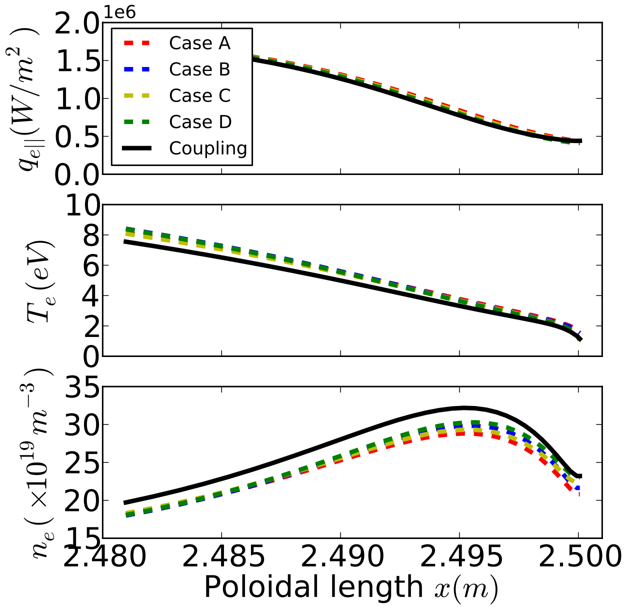


Figure 5.6: Profiles of $q_{e\|}$, T_e and n_e with the stagnation point density $n_u = 2.5 \times 10^{19}\mathrm{m}^{-3}$ are compare among the 5 cases: cases A, B, C, D and the coupling case (details can be found in the text).

in agreement with the profile in Fig. 5.5. Hence

$$\frac{T_{et}^{\mathrm{A}}}{T_{et}^{\mathrm{cpl}}} \approx \left(\frac{\gamma_{e\|}^{\mathrm{cpl}}}{\gamma_{e\|}^{\mathrm{A}}}\right)^2 \tag{5.10}$$

This above equation clearly explains the result shown in Fig. 4.30.

By comparing profiles of n_e among the 5 cases both in Figs. 5.2 and 5.3, one may notice that the deviation of n_e in the coupling case from that in case A near the target is mainly due to the fact that kinetic and fluid flux factors are implemented respectively for the two cases (case A and coupling case). This is clarified in section 5.2.

High stagnation point density

Despite the difference between the target electron temperature ($\sim$ 1.6eV) for cases A, B and that ($\sim$ 1.15eV) for cases C, D and the coupling case, the profiles of n_e, T_e and $q_{e\|}$ in the coupling case start to converge to the ones with running only SOLPS (cases A and B) as shown in Fig. 5.6 (a blow-up of the region $x = 2.48\mathrm{m} - 2.50\mathrm{m}$).

5.2 D+C case

Similar to pure D cases in section 5.1, cases A, B, C, D are run with carbon impurity with the same setup as in section 4.4.2. However, instead of applying

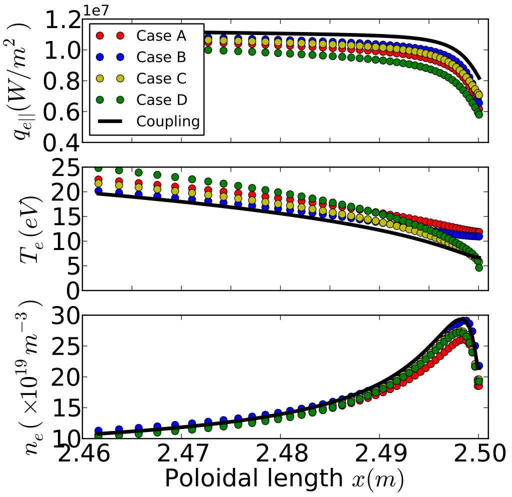


Figure 5.7: Profiles of $q_{e\|}$, T_e and n_e with the stagnation point density $n_u = 2.2 \times 10^{19}\text{m}^{-3}$ with carbon impurity are compared among the 5 cases: cases A, B, C, D and the coupling case (details can be found in the text). A blowup of the region ($x = 2.40\text{m} - 2.50\text{m}$).

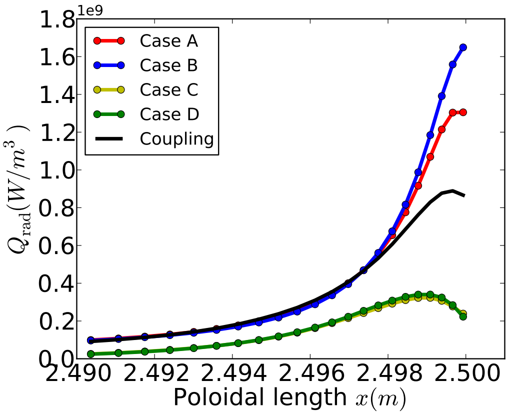


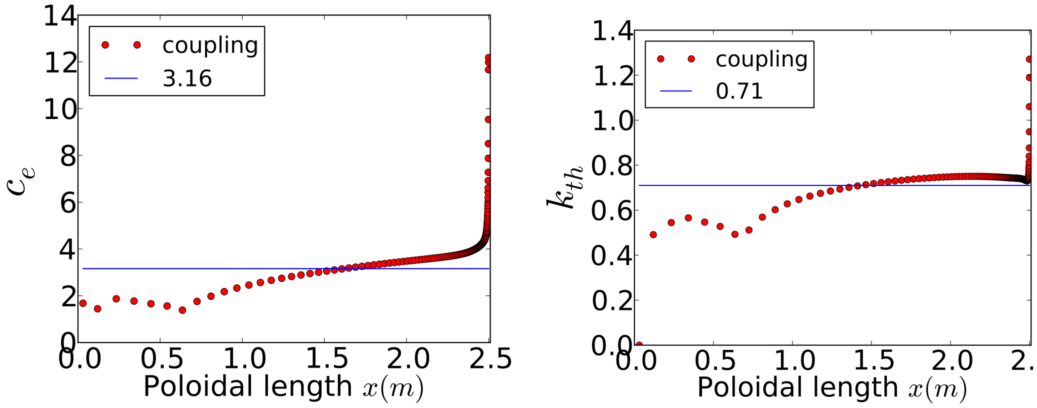
Figure 5.8: Profiles of the electron radiation energy sink density in the steady state for the five cases. Only the region near the target is shown (from $x = 2.49$m to $x = 2.50$m).

a flux limiter $\alpha_e = 0.3$, the case B is run with implementing the kinetic flux factors ($c_e = c_e^{\rm ss}$) in order to investigate their effect on the n_e profile. Then the four cases are compared with the coupling case with the carbon impurity from section 4.4.2.

Fig. 5.7 shows the comparison of n_e, T_e and $q_{e\|}$ profiles among the five cases (only a blow-up of the region near the target $x = 2.46 - 2.50$m is shown). As mentioned above, by comparing case A (or B), case C (or D) and the coupling case, it is clear again that the deviation of $\gamma_{e\|}^{\rm ss}$ from $\gamma_{e\|}^{\rm c}$ is mainly responsible for decreasing the target electron temperature, however, the kinetic flux factors have little impact on T_t, as can be concluded by comparing cases A and B.

n_e near the target in the coupling case can be matched by running SOLPS only with the kinetic flux factors (case B), as seen in Fig. 5.7, consistent with the discussions in section 5.1.

Nevertheless, in difference to the results in section 5.1, the electron heat flux densities near and at the target are distinctly different among the five cases. It can be inferred from Fig. 5.8 that the discrepancy between the electron heat flux densities is mainly attributed to differences in electron energy sinks due to differences in electron temperatures and densities in this region, rather than to kinetic effects on the ionization rate coefficients, since kinetic atomic rates were not calculated here.



(a) **(b)**

Figure 5.9: Profiles of (a) the electron heat conduction coefficient ($c_e = c_e^{\rm ss}$) and (b) the thermal force coefficient ($k_\parallel = k_\parallel^{\rm ss}$) in the steady state coupling case with the carbon impurity.

Both kinetic flux and boundary factors play an important role in determining profiles of n_e, T_e and therefore $q_{e\parallel}$. The significant deviation of $c_e^{\rm ss}$ and $k_e^{\rm ss}$ from $c_e^{\rm c}$ and $k_e^{\rm c}$ near the target (see Fig. 5.9) tend to increase n_e, and, as discussed above, the deviation of $\gamma_e^{\rm ss}$ from $\gamma_e^{\rm c}$ tends to decrease the target electron temperature, which together results in the modification of the radiation density in the near target region.

5.3 Summary

Coupling simulations successfully reproduce the expected effect of heat flux limiting ($c_e^{\rm ss} < c_e^{\rm c}$) upstream and enhancement ($c_e^{\rm ss} > c_e^{\rm c}$) downstream [43]. For cases with pure deuterium plasmas, based on our simulations, the flux limiting and enhancement factors were found to have no impact on the profile of electron heat flux, although, in the region near the target, the deviation of $c_e^{\rm ss}$ from $c_e^{\rm c}$ is significant (e.g. by factor 3 or 4 for medium upstream collisionalities). The flux enhancement only tends to flatten the electron temperature profile without affecting the absolute value much in the near target region. We conclude that the kinetic boundary factor $\gamma_e^{\rm ss}$ is mainly responsible for the electron target temperature decrease compared to a fluid simulation, which leads to the change in the electron temperature profile. The SOLPS case with fluid flux factors ($c_e = c_e^{\rm c}$, $k_\parallel = k_\parallel^{\rm c}$) and kinetic boundary factors ($\gamma_{e\parallel} = \gamma_{e\parallel}^{\rm ss}$, $\Delta\phi = \Delta\phi^{\rm ss}$) is able to produce profiles close to those from the coupling simulation.

This, however, is not true for the case with the carbon impurity. The combined effect of a remarkable decrease of the downstream electron temperature, attributed to the deviation of the kinetic boundary factors from the fluid boundary factors, and a strong density increase, caused by the difference between the kinetic flux factors and the fluid flux factors in the same region, strongly affect the ionization radiation power, thus modifying the electron heat flux profile as a result.

We would like to conclude that the inclusion of kinetic boundary factors is always important for a fluid model to correctly predict the target electron temperature for all collisionalities we have investigated in this chapter. Instead of influencing the boundary heat flux, the kinetic sheath heat transmission coefficient γ_e^{ss} rather modifies the target electron temperature. The increase of the upstream electron temperature due to flux limiting upstream is always small and hence can be neglected. The heat flux profile is determined by the energy sources and sinks rather than being directly affected by the flux limiting or enhancement. However, for cases with the carbon impurity where the radiation concentrates downstream, the electron heat flux profile near the target is substantially modified by kinetic effects of electron parallel transport as a result of the modification of radiation power profile due to the change of electron density and temperature in that region.

Chapter 6

Summary and conclusions

Power exhaust is one of the critical issues for future fusion reactors. The power flux from the core plasma to the SOL deposits in the divertor region. This power is carried mostly by parallel transport in SOL. The parallel heat flux density can be up to $\sim 1\text{GWm}^{-2}$ for ITER [20], which is way above the limit of divertor materials (10MWm^{-2}). A detached divertor regime is planned for ITER operation and some novel divertor geometries, e.g. snow-flake divertor [117], small angle slot divertor [118], have been tested to avoid high power depositions. The careful divertor design, which requires highly precise heat flux calculation, is the key for successful fusion energy realization.

SOLPS is the main tool for the prediction of SOL and divertor conditions in the future fusion device ITER, where parallel kinetic effects in the SOL will play an important role. The default version of SOLPS solves the Braginskii (fluid) equations which don't take kinetic effects into account. The present work has enabled SOLPS in its 1D version to incorporate kinetic effects of parallel electron transport by coupling it with KIPP. We fully tested the iterative coupling algorithm, in which SOLPS solves the fluid equations with effective kinetic factors for electrons transferred from KIPP and KIPP solves the Vlasov-Fokker-Planck equation maintaining plasma profiles transferred from SOLPS. The iteration was made as an automatic process.

The coupling scheme used in the coupling algorithm takes advantage of the strong points of SOLPS which uses a highly sophisticated fluid model, with self-consistent recycling and physical and chemical sputtering models, as well as atomic physics, while treating electrons kinetically in order to consider the most important kinetic effects of parallel electron transport with the rest (ion transport, atomic physics, etc.) tackled by the fluid model. It has been demonstrated that typically only 2 or 3 iterations between SOLPS and KIPP are necessary to achieve a coupling steady state (see section 4.3.3).

In some circumstances where the electron temperature profile from the stagnation point to the target is flat hence T_e is still quite high at the target, there exists an exponential increase in c_e near the target region with coupling iterations, although this increase has no influence on the already convergent ion and electron parameter profiles. This numerical instability only occurred in low collisionality cases. It can be avoided by the new scheme which applies a limiting coefficient c_{lim} to conductive heat flux density (see section 4.3.4). The new scheme modifies the formula of parallel electron heat flux density $q_{e\|}$ by limiting $c_e \leq c_{\mathrm{lim}}$ and allows the enhanced flux transferred not only to the conductive piece but also to the convective piece. The achieved c_{lim}-independent coupling steady states for various collisionalities imply that the coupling scheme allows one to obtain the unique solution with the kinetic treatment of electrons by simply maintaining profiles of electron density, temperature and particle flux density in KIPP and then transferring kinetic coefficients from KIPP back to SOLPS.

A fairly low sensitivity of the electron heat flux density profile to electron kinetic effects in cases with pure deuterium plasmas can be inferred from the comparison between the coupling and SOLPS only simulations (see chapter 5). The deviation of the coupling steady state sheath heat transmission coefficient $\gamma_{e\|}^{\mathrm{ss}}$ from the classical value $\gamma_{e\|}^{\mathrm{c}}$, attributed to non-local transport of upstream hot electrons, is mainly responsible for the difference of the target electron temperature in a coupling steady state and the one achieved only by SOLPS. The electron heat conduction coefficient c_e and the electron sheath heat transmission coefficient $\gamma_{e\|}$ in a fluid code are specified to give the electron heat flux densities through cell faces and the boundary. However, instead of influencing the profile of electron heat flux density and the target electron heat flux density (see section 5.1) they tend to modify the electron temperature profile. This phenomenon may be explained by the fact that, for a self-consistent complete model with the Vlasov-Fokker-Planck equation or the infinite number of moment equations, the profiles of electron heat flux and temperature are essentially determined by energy sources and sinks. Nevertheless, for the Braginskii model where only the first three moment equations are present, the electron temperature profile is imposed by the closure equation (Eq. (4.6)).

Similar to fully kinetic simulations, the iterative coupling algorithm self-consistently achieves the profiles of electron temperature and heat flux density determined by sources, but with the help of the fluid model to provide information that cannot be obtained with only electron kinetic equation. The effective kinetic factors c_{eff}, k_{eff}, γ_{eff}, $\Delta\phi_{eff}$ connects the kinetic electron equation and the fluid equations in the coupling scheme. For example, c_{eff} acts as a free parameter, matching the relation between electron temperature and heat flux density that is used in the fluid model. It indicates that the specific form of the equation for the relation between electron temperature and heat flux density (the closure equation 4.6) doesn't influence the simulation results with the iterative coupling scheme, which only acts as a bridge to transfer information from KIPP to SOLPS (as has been elucidated in section 4.3.4).

The achievement of the coupling steady state is the most important point. However, the choice of the closure equation should not introduce numerical instabilities, e.g. removing the conductive piece from the closure equation can introduce numerical instability to SOLPS and hence the coupling steady state will never be reached (see section 4.3.4).

In order to simulate the case with low upstream collisionality but a strong temperature drop which is expected in ITER, we switched on carbon impurity sputtering. The carbon was introduced as a radiator in the coupling simulation. One problem for 1D geometry with carbon impurity is that highly charged ions tend to concentrate near the stagnation point, which causes cases to crash. The artificial carbon ion particle sink (see section 4.4.2) was applied to avoid this problem. In difference to pure deuterium cases, however, the comparison between the coupling and SOLPS only cases with the carbon impurity as a radiator shows that the heat flux density profile can be affected indirectly by parallel kinetic effects (see section 4.4.2), since the modifications of temperature and density profiles near the target change rate coefficients for atomic physics.

Electron kinetic effects are expected to have significant impacts on divertor detachment regime [72, 73, 75]. Based on 1D SOLPS simulations, detachment occurs when $T_i \approx 1$eV and $T_e < 1$eV. However, the present work has shown that there exists a new numerical instability for the iterative coupling scheme when simulating cases with target electron temperature $T_{et} \lesssim 1$eV. For these cases, c_e profile oscillated in the near target region and the amplitude of this oscillation grew with coupling iterations. Above a certain amplitude, negative values appeared in the c_e profile, ruining the simulation. The reason is not yet clearly understood. This may require further study in future.

Acknowledgement

Foremost I would like to express my sincere gratitude to my direct supervisors Dr. Alex Chankin and Dr. David Coster. Alex guided me through the whole project. I had enormous fruitful discussions with Alex on a daily basis. I wouldn't have completed this project without his help. Particularly during writing he dedicated to polishing my expressions, improving my logic and rearranging the structure, which has significantly improved my writing skills. Thanks to David, I got the 1D SOLPS setup and gained a deep knowledge of SOLPS, which is the essential prerequisite for this work to be done. Advice and comments given by David have been a great help in completion of this project.

I would also like to thank my academic supervisor Prof. Dr. Ulrich Stroth who has provided a lot of helpful suggestions for this work, Prof. Dr. Frank Jenko who has polished this thesis and arranged the thesis committee and Dr. Alejandro Banon-Navarro.

Discussions with Prof. Martine Baelmans at 2016 PSI conference and the idea from Dr. Kristel Ghoos helped me solve the problem of 1D SOLPS with the carbon impurity.

My deep appreciation also goes to the small but warm Chinese community at IPP: Dr. Wei Zhang, Dr. Linming Shao, Dr. Lingyan Xiang, Dr. Liang Gao, Dr. Hao Zhu, Ou Pan, Jingbo Chen, Baoguo Wang, etc.

I have obtained inspiring ideas from my sincere fellow PhD students: Patrick Astfalk, Alexander Ross, Ivan Erofeev.

Last but not least, I am particularly grateful to my parents for their love and encouragements which have been supporting my life in Germany.

Appendix A

Numerical details of SOLPS

A.1 Numerical implementation of B2.5 equations

In SOLPS, the 2D B2.5 equations (Eqs. (2.41-2.46)) are discretized using the finite volume method. Since it is very important for the coupling, the electron energy conservation equation (Eq. (2.46)) for cell i_x, i_y, taken as an example, is discretized (the change of the average energy content in one cell is due to the heat flux flowing in/out of the cell faces and the volumetric sources/sinks inside this cell) as:

$$\frac{3}{2}V\frac{\Delta(n_eT_e)}{\Delta t} = Q_l - Q_r + Q_b - Q_u - Q_\Delta V + \left(S_{Ee} - n_eT_e\nabla\cdot\vec{u}_e + \frac{1}{en_e}\vec{j}\cdot\vec{R}_e\right)V \tag{A.1}$$

As shown in Fig. A.1, Q_l, Q_r, Q_b, Q_u are the numerical electron heat fluxes through the left, right, bottom and top faces of cell ix, iy, respectively. V is the volume of this cell. A hybrid scheme is used for convective and conductive heat fluxes (numerical details can be found in Appendix B). For example, the heat flux through the left face Q_l has the form:

$$Q_l = Q_l^v + Q_l^d \tag{A.2}$$

Q_l^v is the convective piece and Q_l^d is the conductive piece:

$$\begin{aligned} Q_l^v &= f_l\cdot T_e^{\mathrm{up}}\cdot A_l \\ &= f_lA_l\frac{1}{2}\left[T(i_x-1,i_y)+T(i_x,i_y)\right] - \frac{1}{2}\left|f_lA_l\right|\left[T(i_x,i_y)-T(i_x-1,i_y)\right] \end{aligned} \tag{A.3}$$

$$Q_l^d = \left(\sqrt{D_l^2+f_l^2/4}-\left|f_l\right|/2\right)\frac{T(i_x,i_y)-T(i_x-1,i_y)}{\left(hx(i_x,i_y)+hx(i_x-1,i_y)\right)/2}\cdot A_l \tag{A.4}$$

where $T(i_x,i_y)$ is the electron temperature at cell i_x, i_y. $hx(i_x,i_y)$ is the length of cell i_x, i_y in the x direction. A_l is the effective area of the left face (exactly the left face area if the cell is rectangular), f_l is the poloidal convective piece

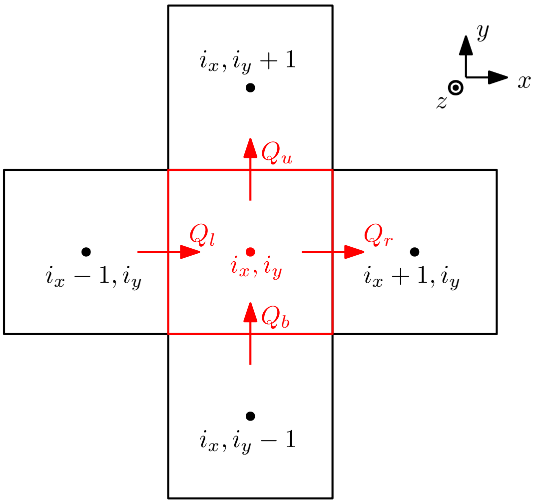


Figure A.1: Energy balance of cell i_x, i_y is related to the four adjacent cells.

coefficient (Eq. (2.62)) interpolated onto the cell left face and D_l is the poloidal conductive diffusivity (Eq. (2.65)) interpolated onto the cell left face:

$$f_l = F_{ex}|_l \tag{A.5}$$

$$D_l = b_x^2\, c_e \frac{n_e T_e \tau_e}{m_e}\bigg|_l + b_z^2\, \chi_e^E n_e\big|_l \tag{A.6}$$

where "$|_l$" denotes the linear interpolation onto the left face of cell i_x, i_y, by defining:

$$F_l = f_l \cdot A_l \tag{A.7}$$

$$C_l = \frac{2\left(\sqrt{D_l^2 + f_l^2/4} - |f_l|/2\right) \cdot A_l}{hx(i_x, i_y) + hx(i_x-1, i_y)} + \frac{1}{2}|F_l| \tag{A.8}$$

The heat fluxes through the four faces are transformed to consist of two components:

$$Q_l = F_l \frac{1}{2}\left[T(i_x-1, i_y) + T(i_x, i_y)\right] - C_l\left[T(i_x, i_y) - T(i_x-1, i_y)\right] \tag{A.9}$$

$$Q_r = F_r \frac{1}{2}\left[T(i_x+1, i_y) + T(i_x, i_y)\right] - C_r\left[T(i_x+1, i_y) - T(i_x, i_y)\right] \tag{A.10}$$

$$Q_b = F_b \frac{1}{2}\left[T(i_x, i_y-1) + T(i_x, i_y)\right] - C_b\left[T(i_x, i_y) - T(i_x, i_y-1)\right] \tag{A.11}$$

$$Q_u = F_u \frac{1}{2}\left[T(i_x, i_y+1) + T(i_x, i_y)\right] - C_u\left[T(i_x, i_y+1) - T(i_x, i_y)\right] \tag{A.12}$$

The variables with the subscriptions "r, b, u" have the same forms of Eq. (A.7) and (A.8), but, D_b and D_u are radial diffusivities:

$$D_b = \chi_e^E n_e\big|_b \tag{A.13}$$

$$D_u = \chi_e^E n_e\big|_u \tag{A.14}$$

The heat exchange term Q_Δ is described in the same way as in Eq. (2.37). The sixth term on the right hand side of Eq. (A.1), with the time-dependent term moved to the right, is regarded as electron energy source and is written in the form:

$$\left(S_{Ee} - n_e T_e \nabla \cdot \vec{u}_e + \frac{1}{en_e}\vec{j} \cdot \vec{R}_e - \frac{3}{2}\frac{\Delta(n_e T_e)}{\Delta t}\right) V = S_0 + S_1 T(i_x, i_y) + S_2 n_e + S_3 n_e T(i_x, i_y) \tag{A.15}$$

with S_0, S_1, S_2 determined based on electron density and temperature. The variable remained to be solved is the profile of the electron temperature. The implicit numerical scheme is used to describe the evolution of the electron temperature by solving the following equation at every time step:

$$\begin{aligned}
&\left\{F_l \frac{1}{2}\left[T^{n+1}(i_x - 1, i_y) + T^{n+1}(i_x, i_y)\right] - C_l \left[T^{n+1}(i_x, i_y) - T^{n+1}(i_x - 1, i_y)\right]\right\} \\
-&\left\{F_r \frac{1}{2}\left[T^{n+1}(i_x + 1, i_y) + T^{n+1}(i_x, i_y)\right] - C_r \left[T^{n+1}(i_x + 1, i_y) - T^{n+1}(i_x, i_y)\right]\right\} \\
+&\left\{F_b \frac{1}{2}\left[T^{n+1}(i_x, i_y - 1) + T^{n+1}(i_x, i_y)\right] - C_b \left[T^{n+1}(i_x, i_y) - T^{n+1}(i_x, i_y - 1)\right]\right\} \\
-&\left\{F_u \frac{1}{2}\left[T^{n+1}(i_x, i_y + 1) + T^{n+1}(i_x, i_y)\right] - C_u \left[T^{n+1}(i_x, i_y + 1) - T^{n+1}(i_x, i_y)\right]\right\} \\
-&Q_\Delta^{n+1} V + S_0 + S_1 T^{n+1}(i_x, i_y) + S_2 n_e + S_3 n_e T^{n+1}(i_x, i_y) = 0
\end{aligned} \tag{A.16}$$

Here T^{n+1} describes the unknown temperature profile to be solved in the above equation:

$$T^{n+1}(i_x, i_y) = T^n(i_x, i_y) + dT(i_x, i_y) \tag{A.17}$$

with T^n (the solution from the previous time step) already known at time step n. The other variables: F_l, F_r, F_b, F_u, C_l, C_r, C_b, C_u, S_0, S_1, S_2 are calculated based on profiles of electron density and temperature at time step n, which are already known from the solution of the previous time step. Now the unknown variable becomes the profile of dT. Substitution of Eq. (A.17) into Eq. (A.16), leads to:

$$\begin{aligned}
&aa_b(i_x, i_y) * dT(i_x, i_y - 1) + aa_u(i_x, i_y) * dT(i_x, i_y + 1) \\
+&aa_l(i_x, i_y) * dT(i_x - 1, i_y) + aa_r(i_x, i_y) * dT(i_x + 1, i_y) \\
+&aa_c(i_x, i_y) * dT(i_x, i_y) = \mathrm{res}^n(i_x, i_y)
\end{aligned} \tag{A.18}$$

where

$$
\begin{aligned}
aa_b(i_x,i_y) &= -F_b - C_b && \text{(A.19)}\\
aa_u(i_x,i_y) &= F_u - C_u && \text{(A.20)}\\
aa_l(i_x,i_y) &= -F_l - C_l && \text{(A.21)}\\
aa_r(i_x,i_y) &= F_r - C_r && \text{(A.22)}\\
aa_c(i_x,i_y) &= -S_1 - S_3 n_e + V\frac{\partial Q_\Delta}{\partial T_e}\bigg|^n \\
&\quad - aa_b(i_x,i_y) - aa_u(i_x,i_y)\\
&\quad - aa_l(i_x,i_y) - aa_r(i_x,i_y) && \text{(A.23)}
\end{aligned}
$$

and

$$
\begin{aligned}
&\mathrm{res}^n(i_x,i_y) =\\
&\quad\left\{F_l\frac{1}{2}\left[T^n(i_x-1,i_y)+T^n(i_x,i_y)\right] - C_l\left[T^n(i_x,i_y)-T^n(i_x-1,i_y)\right]\right\}\\
&\quad-\left\{F_r\frac{1}{2}\left[T^n(i_x+1,i_y)+T^n(i_x,i_y)\right] - C_r\left[T^n(i_x+1,i_y)-T^n(i_x,i_y)\right]\right\}\\
&\quad+\left\{F_b\frac{1}{2}\left[T^n(i_x,i_y-1)+T^n(i_x,i_y)\right] - C_b\left[T^n(i_x,i_y)-T^n(i_x,i_y-1)\right]\right\}\\
&\quad-\left\{F_u\frac{1}{2}\left[T^n(i_x,i_y+1)+T^n(i_x,i_y)\right] - C_u\left[T^n(i_x,i_y+1)-T^n(i_x,i_y)\right]\right\}\\
&\quad-Q^n_\Delta V + S_0 + S_1 T^n(i_x,i_y) + S_2 n_e + S_3 n_e T^n(i_x,i_y) \qquad \text{(A.24)}
\end{aligned}
$$

As one may note, there are five points involved in each cell. Eq. (A.18) is the discretized electron energy conservation equation only for cell i_x, i_y. We have a mesh with $n_x + 2$ cells poloidally from -1 to n_x, $n_y + 2$ cells radially from -1 to n_y. The discretized electron energy conservation equations for all cells can be written as a matrix equation:

$$\boldsymbol{A}\cdot\boldsymbol{dT} = \mathbf{Res} \tag{A.25}$$

$\boldsymbol{A}$ is a square matrix with $(n_x + 2)(n_y + 2) \times (n_x + 2)(n_y + 2)$ elements; $\boldsymbol{dT}$ is a vector, remained to be solved, with $(n_x + 2) \times (n_y + 2)$ elements, while **Res** is the right hand side vector with $(n_x + 2) \times (n_y + 2)$ elements. One has to note that the boundary cells (e.g. the right boundaries with $i_x = n_x$) are ghost cells for specifying boundary conditions. The flux terms in Eq. (A.16) at the right face of cells with $i_x = n_x$ are specified as:

$$
\begin{aligned}
F_r &= 0 && \text{(A.26)}\\
C_r &= 0 && \text{(A.27)}
\end{aligned}
$$

In these ghost cells, the boundary conditions, e.g. the electron heat flux out of the simulation domain, are regarded as sinks specified in the source term S_{Ee}

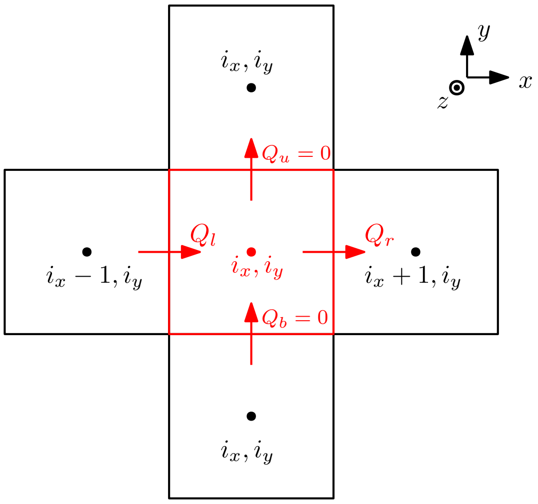


Figure A.2: Radial heat fluxes through bottom and top faces of cell i_x, i_y are 0.

in Eq. (A.1). The same numerical schemes are used for the other ghost cells. LU decomposition is then performed on the matrix $\boldsymbol{A}$:

$$\boldsymbol{A} = \boldsymbol{LU} \tag{A.28}$$

and hence the equation $\boldsymbol{LU} \cdot \boldsymbol{dT} = \mathbf{Res}$ can be easily solved iteratively.

A.2 1D adaptation

The 1D geometry (see section 2.4) is used in this work. The radial transport is switched off by implementing periodic boundary conditions in the y direction (see Fig. A.2), so the radial heat fluxes are:

$$\Gamma_b = \Gamma_u = D_y(i_x, i_y) \times \frac{n(i_x, i_y) - n(i_x, i_y)}{hy} = 0 \tag{A.29}$$

$$Q_b = Q_u = \frac{5}{2}\Gamma_b T(i_x, i_y) + \kappa_y(i_x, i_y) \times \frac{T(i_x, i_y) - T(i_x, i_y)}{hy} = 0 \tag{A.30}$$

Coefficients with subscripts "u" and "b" in the discretized electron energy balance equation (Eq. (A.16)) can be dropped, therefore, the discretized 5-point energy balance equation (Eq. (A.18)) becomes essentially a 3-point equation:

$$\begin{aligned} &aa_l(i_x, i_y) * dT(i_x - 1, i_y) + aa_r(i_x, i_y) * dT(i_x + 1, i_y) \\ &+aa_c(i_x, i_y) * dT(i_x, i_y) = \mathrm{res}^n(i_x, i_y) \end{aligned} \tag{A.31}$$

All the variables have the same forms as those in Eq. (A.18) but with terms with the subscript "u" or "b" being dropped.

A.3 Adaptation of boundary conditions in SOLPS to the coupling scheme

A.3.1 The default boundary condition

As discussed in section 4.1.2, in order to be compatible with the KIPP target boundary conditions (technically the sheath edge, but referred to as "boundary" or "target" in later discussions) where the parallel heat flux density is the only contributor (see Eq. (4.37)), a modification is made to the boundary condition for the electron energy conservation equation in SOLPS. As pointed out in section 4.2.3, there are two ghost cells at each end of the simulation grid for implementation of boundary conditions. Cell nx is adjacent to the target (sheath edge). As shown in Fig. A.3a, the electron heat flux through the simulation boundary (denoted as "target" in the figure) is defined as Q_t, which consists of two components:

$$\begin{aligned} Q_t &= b_x q_{t\|} A + b_z q_{t\perp} A \\ &= Q_{t\|} + Q_{t\perp} \end{aligned} \tag{A.32}$$

where $q_{t\|}$ and $q_{t\perp}$ are electron parallel and perpendicular heat flux densities through the boundary and A is the boundary area. Since the ghost cell is attached to the simulation boundary, Q_t is also the heat flux entering the ghost cell. The boundary condition with specified γ_e' is applied to the ghost cell (actually $\gamma_e' = c_1 + |e\phi|/T_{et}$ with c_1 being the free parameter, but, for simplicity, here we only discuss γ_e', see discussions in section 4.1.2) as an energy sink to balance Q_t. So, in the steady state, it leads to:

$$Q_{t\|} + Q_{t\perp} = \gamma_e' b_x \Gamma_{t\|} A T_{et} \tag{A.33}$$

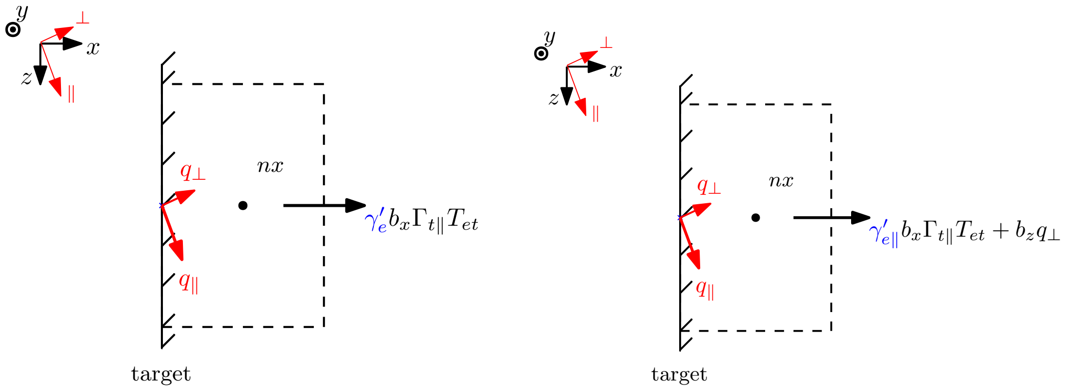
where $\Gamma_{t\|}$ is the parallel electron particle flux density through the boundary and T_{et} is the electron temperature interpolated at the boundary. γ_e' is specified for both parallel and perpendicular parts through the boundary, although the perpendicular one is normally substantially smaller ($\lesssim 10\%$) than the parallel one. The perpendicular heat flux density is treated as a source in KIPP even in the final cell, hence the divergence term $\nabla_\perp q_{e\perp}$ in SOLPS (details can be found in section 4.1.2) is already included automatically in the energy source term. Therefore, the effective heat transmission coefficient γ_{eff} from Eq. (4.37) obtained in KIPP, related to the parallel transport, corresponds to the parallel part $Q_{t\|}$ in Eq. (A.33).

A.3.2 The adapted boundary condition

The modified boundary condition

The boundary condition for the electron energy conservation in SOLPS is modified as:

$$Q_{t\|} + Q_{t\perp} = \gamma_{e\|}' b_x \Gamma_{t\|} A T_{et} + Q_{t\perp} \tag{A.34}$$



(a) The default boundary condition for the electron energy conservation equation. γ'_e is the user specified parameter.

(b) The modified boundary condition for the electron energy conservation equation. $\gamma'_{e\parallel}$ is the user specified parameter. In a coupling simulation, this value will be replaced by the effective one calculated in KIPP.

Figure A.3: Schematic of the modification of the boundary condition in SOLPS.

where $\gamma'_{e\parallel}$ (instead of γ'_e) is a free parameter to be specified or replaced by γ_{eff} (Eq. (4.36)). The comparison between the default and modified boundary conditions is illustrated in Fig. A.3.

The purpose of the boundary modification is to adapt SOLPS boundary condition to the coupling scheme. It is not aimed at altering the physics or numerics of SOLPS.

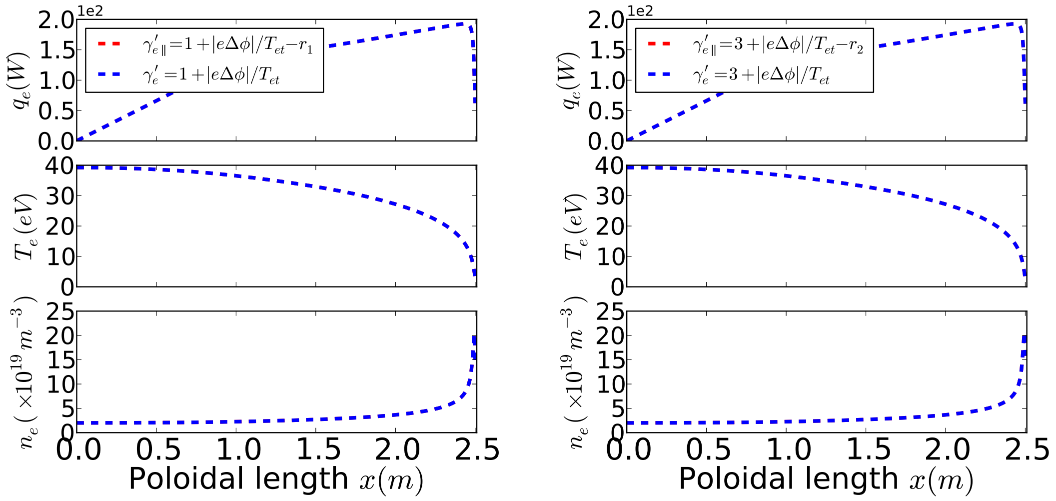
Testing the modified boundary condition

Theoretically, the modified boundary condition is supposed to give the same boundary electron heat flux relative to the default one if one specifies $\gamma'_{e\parallel}$ and γ'_e which satisfies the following relation:

$$\gamma'_e b_x \Gamma_{t\parallel} A T_{et} = \gamma'_{e\parallel} b_x \Gamma_{t\parallel} A T_{et} + Q_{t\perp} \tag{A.35}$$

In this section, we test whether the modified boundary condition gives the same result as the default one both numerically and physically. Two cases with the default boundary conditions of two different γ'_e (γ'_1, γ'_2) and two cases with the modified boundary condition of two corresponding $\gamma'_{e\parallel}$ ($\gamma'_{1\parallel}$, $\gamma'_{2\parallel}$) are run with the same power input. The stagnation point density is fixed as $2.0 \times 10^{19} m^{-3}$. Each pair of γ'_e and $\gamma'_{e\parallel}$ satisfies Eq. (A.35). Note that the term $Q_{t\perp}$, or the relation between $Q_{t\parallel}$ and $Q_{t\perp}$, is not known at the beginning, therefore, the cases with the default boundary conditions:

$$\gamma'_1 = 1 + |e\Delta\phi|/T_{et} \tag{A.36}$$
$$\gamma'_2 = 3 + |e\Delta\phi|/T_{et} \tag{A.37}$$



(a) Comparison of the cases with $\gamma'_{1\parallel}$ and γ'_1.

(b) Comparison of the cases with $\gamma'_{2\parallel}$ and γ'_2.

Figure A.4: The modified boundary condition works in exactly same way both numerically and physically as the default one.

are run first. Then the ratios of r_1 and r_2:

$$r_1 = \frac{Q_{1\perp}}{Q_1} \tag{A.38}$$

$$r_2 = \frac{Q_{2\perp}}{Q_2} \tag{A.39}$$

are obtained. $Q_{i\perp}$ and Q_i are perpendicular and total heat fluxes through the boundary for $i = 1, 2$, when the case reaches the steady state. Then the two cases with the modified boundary condition are run with specifying:

$$\gamma'_{1\parallel} = 1 + |e\Delta\phi|/T_{et} - r_1 \tag{A.40}$$

$$\gamma'_{2\parallel} = 3 + |e\Delta\phi|/T_{et} - r_2 \tag{A.41}$$

The steady state profiles are thus compared in Fig. A.4. It can be clearly seen that the profiles of electron density, temperature and heat flux density from the cases with the modified boundary condition are exactly the same as those with the default boundary condition. Additionally, the same boundary electron heat flux and potential drop are achieved, indicating that the modified boundary condition is behaving both physically and numerically the same way as the default one. The modified boundary condition is used in the coupling test.

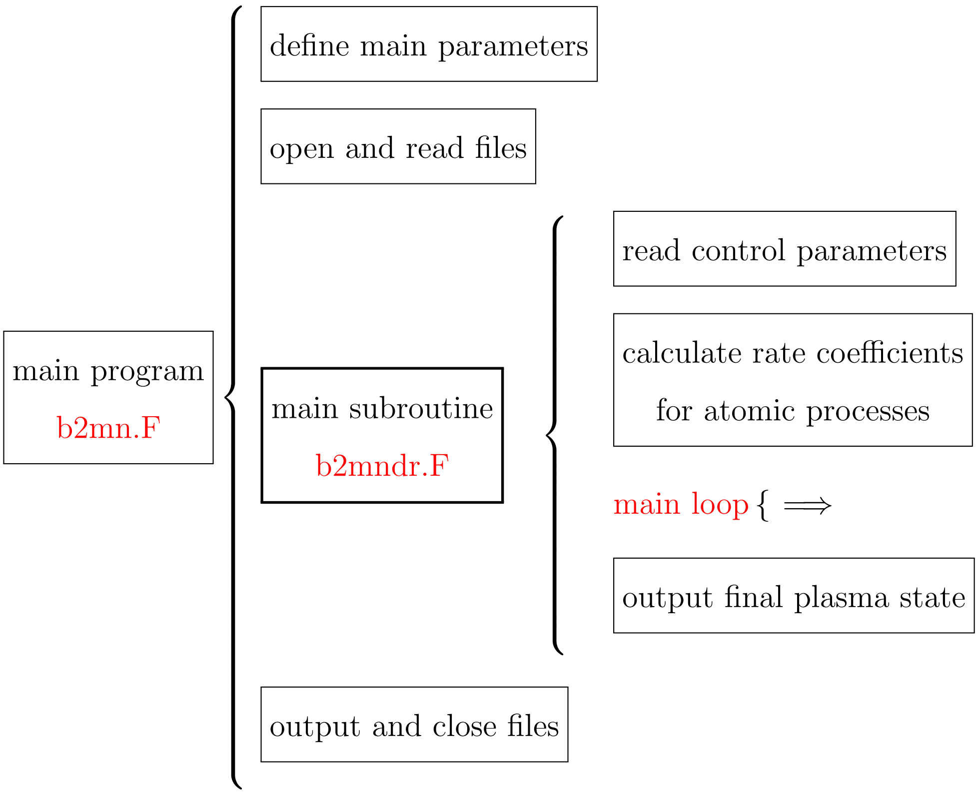
Appendix B

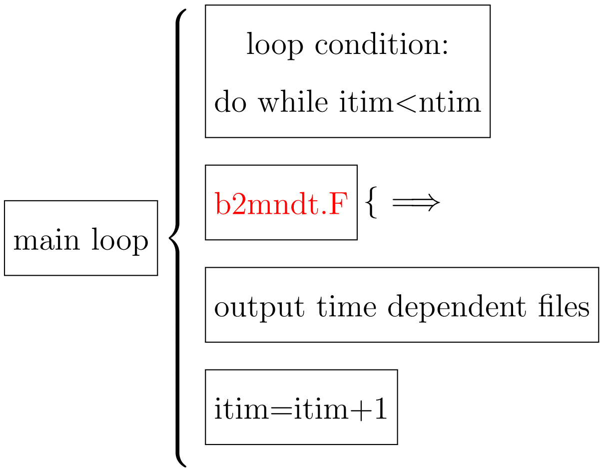
Details of the iterative coupling

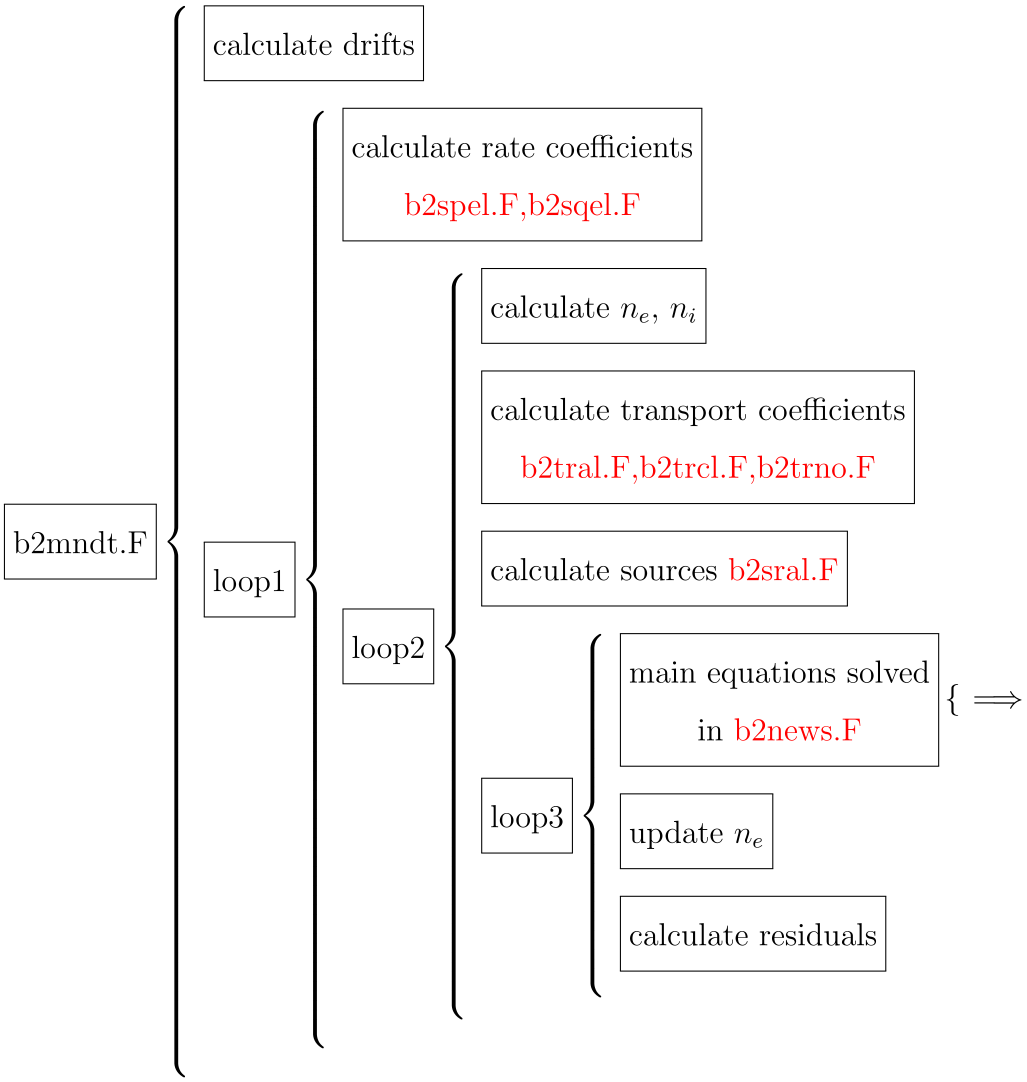
KIPP is a parallelized code while SOLPS only uses one core. The main program *b2mn.F* in SOLPS is modified to become a subroutine in the KIPP main program *Maxi.f90*. This appendix is arranged as follows. Sections B.1.1 and B.1.2 list the main subroutines of SOLPS. Those in red are related to the coupling scheme. The structure of the coupled version of SOLPS and KIPP is described in section B.2. Section B.3 describes the details of the numerical implementation of the coupling scheme.

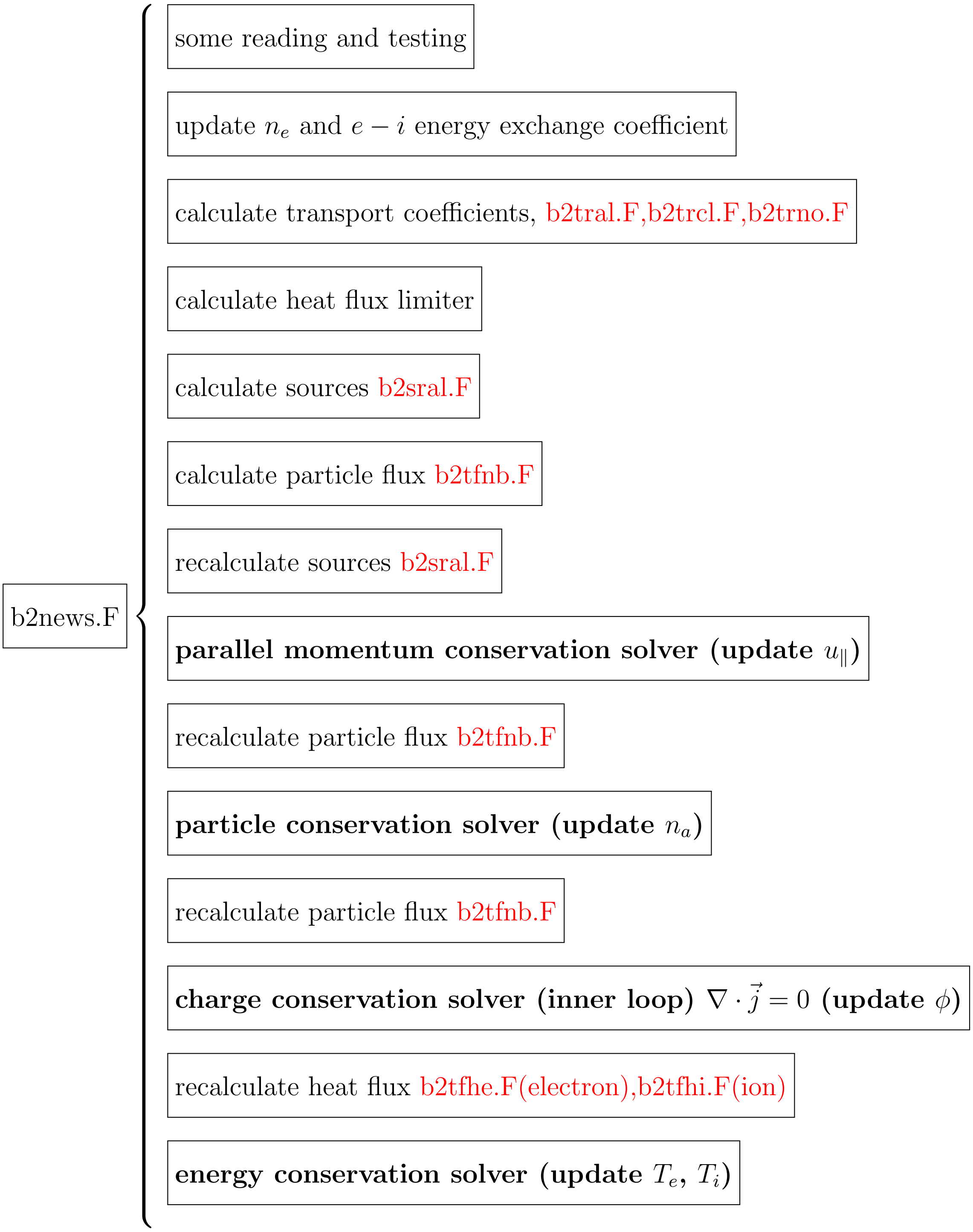
B.1 The structure of SOLPS and KIPP

B.1.1 The structure of SOLPS

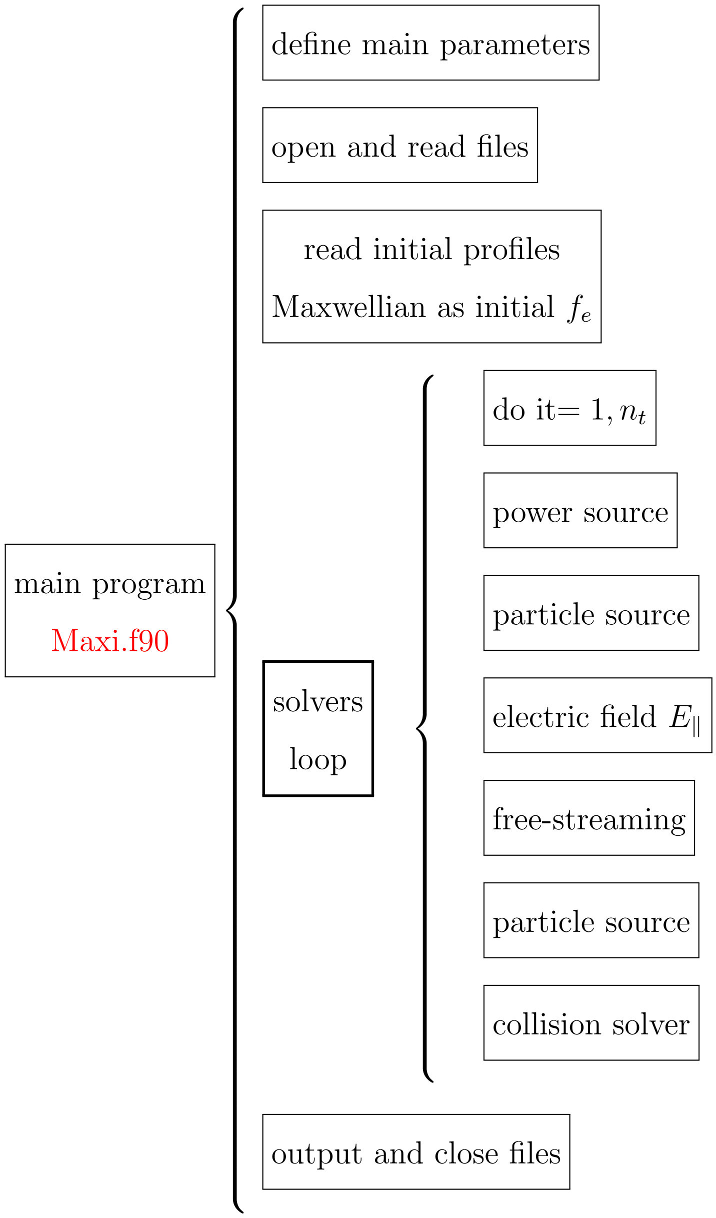



main loop
loop condition:
do while itim<ntim
b2mndt.F
{ ⟹
output time dependent files
itim=itim+1

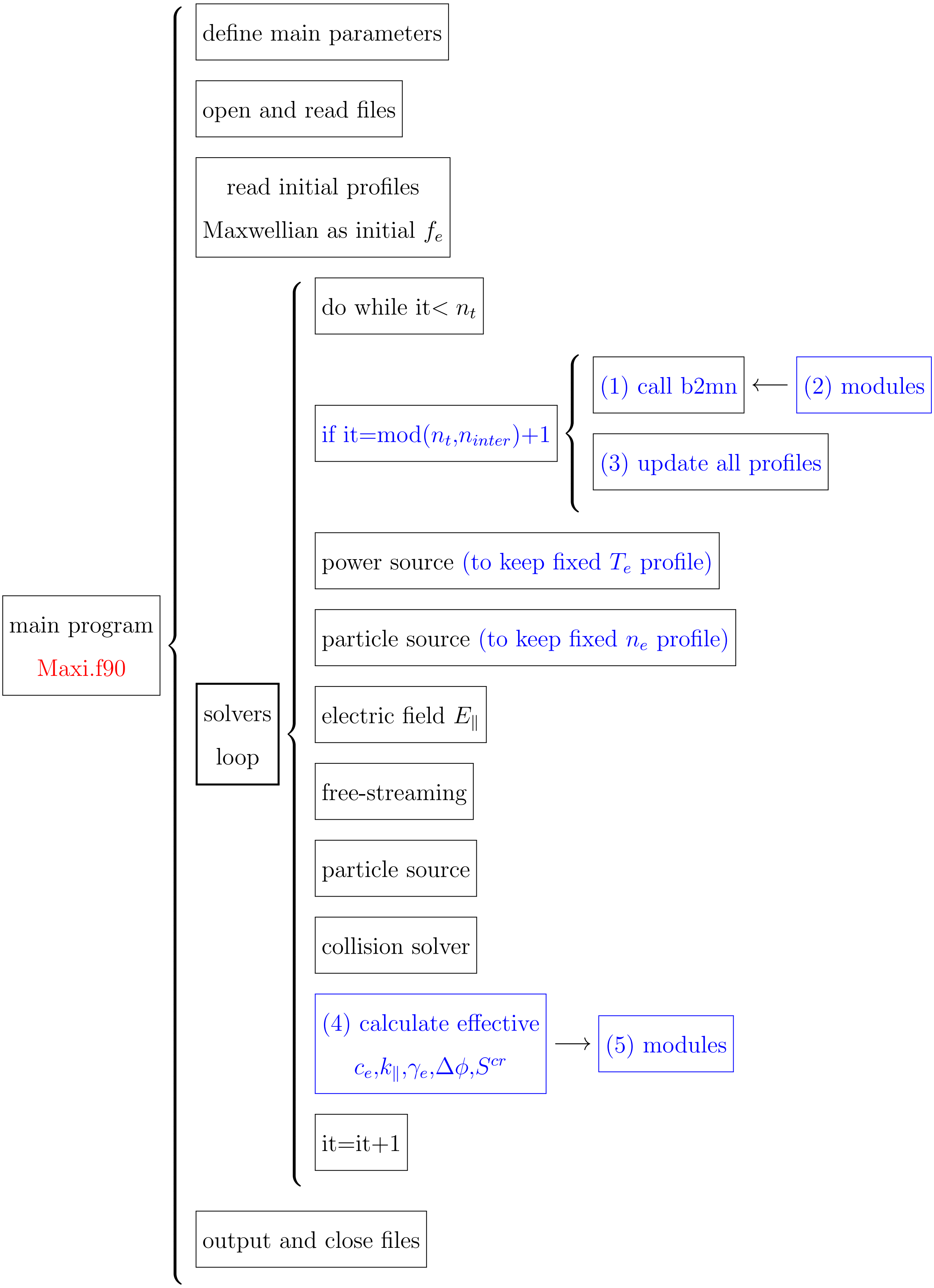

b2mndt.F
calculate drifts
loop1
calculate rate coefficients
b2spel.F,b2sqel.F
loop2
calculate n_e, n_i
calculate transport coefficients
b2tral.F,b2trcl.F,b2trno.F
calculate sources b2sral.F
loop3
main equations solved
in b2news.F
{ ⟹
update n_e
calculate residuals


b2news.F
some reading and testing
update n_e and $e-i$ energy exchange coefficient
calculate transport coefficients, b2tral.F,b2trcl.F,b2trno.F
calculate heat flux limiter
calculate sources b2sral.F
calculate particle flux b2tfnb.F
recalculate sources b2sral.F
parallel momentum conservation solver (update $u_\parallel$)
recalculate particle flux b2tfnb.F
particle conservation solver (update n_a)
recalculate particle flux b2tfnb.F
charge conservation solver (inner loop) $\nabla \cdot \vec{j} = 0$ (update ϕ)
recalculate heat flux b2tfhe.F(electron),b2tfhi.F(ion)
energy conservation solver (update T_e, T_i)

B.1.2 The structure of KIPP



B.2 Structure of the coupling algorithm



For simplicity, the coupling version of KIPP and SOLPS is split into 3 parts:

KIPP part Files used by KIPP

SOLPS part Files used by SOLPS (e.g. *b2mn.F* and its subroutines)

common part Modules (2) and (5) that are used to transfer common parameters, including files:

- ***ini_2_SOLPS.F*** transfer some control parameters from the KIPP part to the SOLPS part.
- ***KIPP_2_SOLPS.F*** transfer effective c_e, $k_\parallel$, γ_e, $\Delta\phi$, S^{cr} and related parameters, as well as some control parameters from the KIPP part to the SOLPS part.
- ***SOLPS_2_KIPP.F*** transfer profiles of electron and ion densities and temperatures, as well as other parameters to the KIPP part.
- ***KIPP_norm.F*** specify reference parameters (n_0, T_0, Λ_0) in the input file *runpar.dat*

The module *KIPP_2_SOLPS.F* transfers the effective c_e, $k_\parallel$, γ_e, $\Delta\phi$, S^{cr} to the corresponding subroutines in the SOLPS part, and the module *SOLPS_2_KIPP.F* transfers the required steady state profiles to the KIPP part. Afterwards, the subroutine (3) updates all the required profiles in one single time step in the KIPP part.

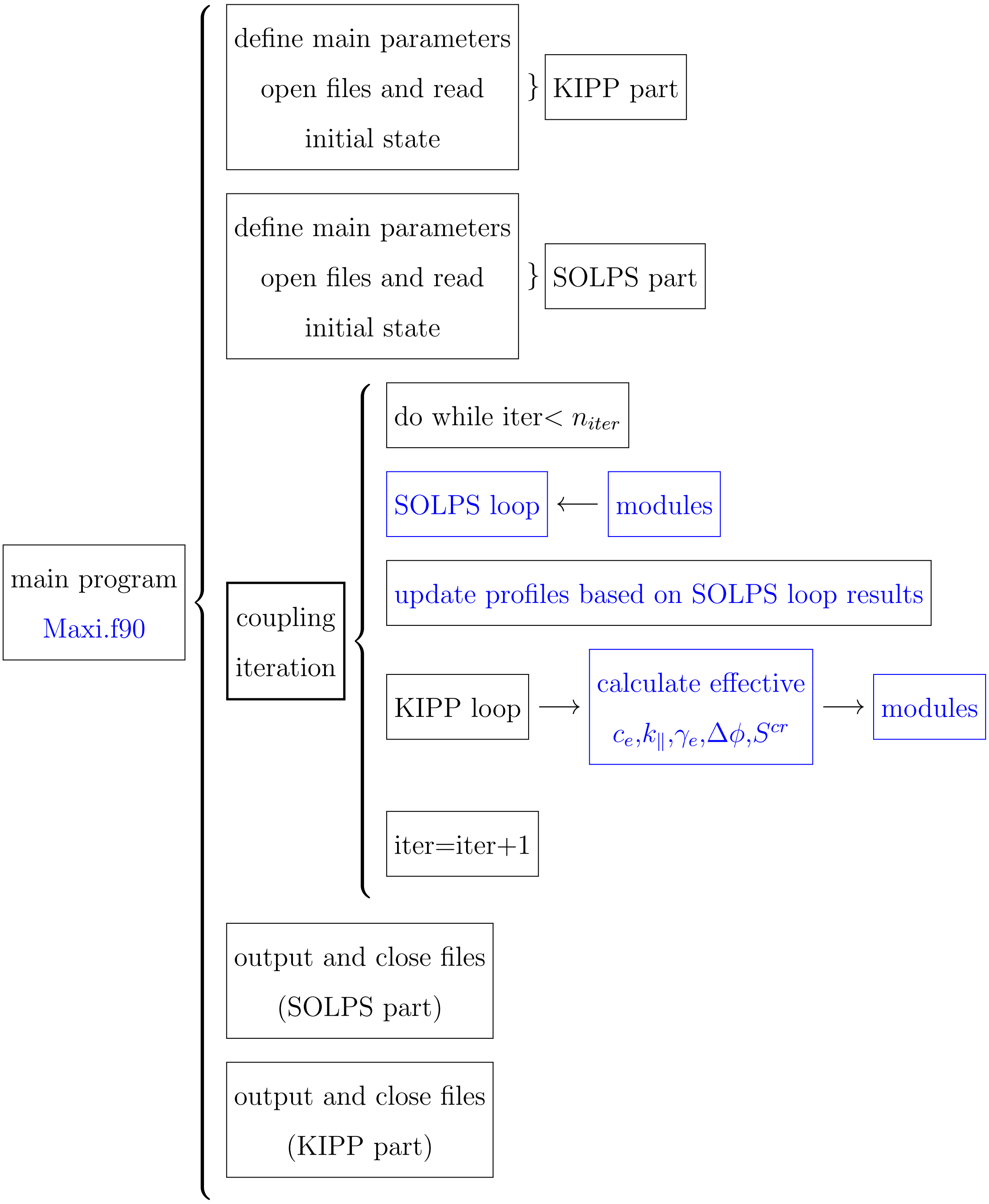
The SOLPS part (1) remains the main structure of SOLPS, however, the main program *b2mn.F* is modified to become a subroutine which is inserted in the "loop" in KIPP. The following parameters are introduced to describe the coupling:

$\mathbf{n_{iter}}$ The number of coupling iterations

$\mathbf{n_{inter}}$ The number of KIPP time steps within each coupling iteration (described in the following part)

$\mathbf{n_t}$ The total number of time steps specified in KIPP, $n_t = n_{iter} \times n_{inter}$

The structure of the coupling algorithm is shown below:



As one can see, there are essentially 3 loops in the coupling algorithm:

SOLPS loop The time step *dtim* and the number of time steps *ntim* within one coupling iteration are specified in the input file *b2mn.dat*.

KIPP loop The time step dt, the number of time steps within one coupling iteration n_{inter} and the total number of time steps nt for one coupling run are specified in the input file *runpar.dat*.

coupling iteration one coupling iteration means one SOLPS loop + one KIPP loop.

B.3 Numerical implementation of the coupling algorithm

B.3.1 Geometry

The standard SOLPS version has 2D grid cells, in poloidal (x) and radial (y) directions. However, it is the 1D SOLPS version (section 2.4) that is used in the coupling algorithm. The simulation grid cells are generated separately in SOLPS and KIPP. There are two lines of ghost cells attached to north and south boundaries of the simulation domain created originally to implement boundary conditions in 2D SOLPS version, but they are not used in 1D simulations and will, therefore, be omitted in the following descriptions. The two grids conform to each other, as shown in Fig. 4.5. The SOLPS grid has $nx+2$ cells from cell -1 (ghost cell) to cell nx (ghost cell) while the KIPP grid has $smax+1$ cells from cell 0 to $smax$, hence:

$$nx = smax + 1 \tag{B.1}$$

Since SOLPS has two ghost cells attached to the two ends of the simulation domain, but not included in it, the simulation domain in SOLPS is from the left face of cell 0 (stagnation point) to the right face of cell $nx-1$ (target), corresponding to the simulation domain in KIPP, which is from the center of cell 0 (the stagnation point) to the right face of cell $smax$ (the target). Cells generated in SOLPS are cubic boxes (in the 1D geometry), areas of all cell faces perpendicular to the poloidal direction x are the same, A, and the length in the poloidal direction is hx_i for cell i. Hence the volume of cell i is $V_i = hx_i A$. Accordingly, in the KIPP grid, the length of cell i is ds_i and the corresponding relation between hx_i and ds_i is described in section 4.2.1 and shown here again:

$$hx_0 = \frac{1}{2} ds_0 \cdot b_x \tag{B.2}$$

$$hx_m = ds_m \cdot b_x \qquad (0 \leq m \leq nx-1) \tag{B.3}$$

B.3.2 Definition of variables at cell centers and faces

In the SOLPS part, electron density n_e, ion density n_a of species a, electron and ion temperatures T_e, T_i, electron and ion (species a) fluid velocities u_e, u_a, as well as particle, momentum and energy sources are defined at cell centers, but all fluxes e.g. particle and heat fluxes (or flux densities) are defined at cell faces. In the rest of this section, we will use the following notations:

$\boldsymbol{Q_i}$ $i = -1, 0, \cdots, nx$ indicates variable Q at the center of cell i, Q can be density, temperature, velocity, heat conduction coefficient, or a volumetric source.

$\boldsymbol{Q_{if}}$ $i = 0, \cdots, nx$ indicates the interpolation of $\boldsymbol{Q_i}$ on the left face of cell i.

$\boldsymbol{P_{if}}$ $i = -1, 0, \cdots, nx$ indicates poloidal flux (or flux density) variable P at the left face of cell i, P can be particle, momentum and heat flux (or flux density). $P_{-1f} = 0$ since the ghost cell particle and power balance is maintained between the flux through the boundary and the source terms due to boundary conditions (boundary conditions are specified as "source" terms in ghost cells).

As discussed in chapter 4, for the 1D SOLPS geometry, a poloidal flux (or flux density) variable P_{if} is the sum of projections of the parallel and perpendicular fluxes on the poloidal direction:

$$P_{if} = b^x_{if} P_{\|if} + b^z_{if} P_{\perp if} \tag{B.4}$$

where

$$b^x_{if} = \frac{B_{pol}}{\sqrt{B^2_{pol} + B^2_{tor}}} = b_x \tag{B.5}$$

$$b^z_{if} = \frac{B_{tor}}{\sqrt{B^2_{pol} + B^2_{tor}}} = b_z \tag{B.6}$$

In the 1D SOLPS geometry, B is assumed to be constant. Since the flux variable appears in the form $\nabla_x P_{if}$ (see Eq.s (2.41) and (2.46)), $\nabla_x P_{if} = b_x \nabla_x P_{\|if} + b_z \nabla_x P_{\perp if}$. In the KIPP part, $b_z \nabla_x P_{\perp if}$ will be automatically included in the particle and power sources (see section 4.1.2), therefore, only $P_{\|if}$ is discussed in the following sections.

B.3.3 First SOLPS loop with default coefficients

For the initial coupling run, with the logical parameter *continue_case* specified as $false$ in the input file *runpar.dat*, the SOLPS loop will start with the Braginskii coefficients:

$$c_e = 3.16 \tag{B.7}$$

$$k_\| = 0.71 \tag{B.8}$$

and the classical boundary conditions:

$$\Delta\phi = -3T_e/e \tag{B.9}$$

$$\gamma_e = e|\Delta\phi|/T_e + const + 1 \tag{B.10}$$

The *const* in Eq. (B.10) is specified in the input file *b2.boundary.parameters*. The reason for mentioning "1" here will be discussed in section B.3.7.

For the second and following coupling runs, the SOLPS loop will start with the effective coefficients and boundary conditions transferred from the modules. Chapter 4 shows the analytical description of the coupling scheme, while here the numerical implementation is described.

B.3.4 Transferring plasma profiles from the SOLPS part to the KIPP part

In the final time step of a SOLPS loop (when the steady state has been reached), profiles of:

$\boldsymbol{n_{a,i}}$ Density of species a at the center of cell i. The subscript a denotes the species, which can be D^0, D^+ for a pure deuterium plasma and D^0, D^+, C^0, C^+, C^{2+}, C^{3+}, C^{4+}, C^{5+}, C^{6+} for a case with the carbon impurity. The subscript a has the same meaning for other parameters listed below.

$\boldsymbol{u_{a\|,i}}$ Parallel velocity of species a at the center of cell i.

$\boldsymbol{\Gamma_{a\|,if}}$ Parallel particle flux density of species a at the left face of cell i

$\boldsymbol{T_{i,i}}$ Ion temperature at the center of cell i. It is assumed that ion temperatures of all species are the same.

$\boldsymbol{T_{e,i}}$ Electron temperature at the center of cell i.

$\boldsymbol{z_a}$ Charge of species a.

and other relevant variables (e.g. radiation power), as well as some control parameters, are transferred to the KIPP part via the module *SOLPS_2_KIPP.F* and a subroutine *output_for_kipp.F*. Then, in the KIPP part, the profiles are updated accordingly in the subroutines *ion_update.f90*, *electron_update.f90*, *inial_power_b.f90*, for electron density at cell centers:

$$\begin{aligned} n_{e,i}^{\text{kipp}} &= \sum_a z_a n_{a,i} \qquad \text{for } i = 1, 2, \cdots, smax \\ n_{e,0}^{\text{kipp}} &= \sum_a z_a n_{a,-1} \\ n_{e,t}^{\text{kipp}} &= \sum_a z_a n_{a,nx} \end{aligned} \tag{B.11}$$

for electron temperature at cell centers:

$$\begin{aligned} T_{e,i}^{\text{kipp}} &= T_{e,i} \qquad \text{for } i = 1, 2, \cdots, smax \\ T_{e,0}^{\text{kipp}} &= T_{e,-1} \\ T_{e,t}^{\text{kipp}} &= T_{e,nx} \end{aligned} \tag{B.12}$$

for electron particle flux density at cell faces:

$$\begin{aligned}
\Gamma^{\text{kipp}}_{e\|,if} &= \sum_a z_a \Gamma_{a\|,if} \qquad \text{for } i = 1, 2, \cdots, smax \\
\Gamma^{\text{kipp}}_{e\|,t} &= \sum_a z_a \Gamma_{a\|,nxf} \qquad (\text{B.13})
\end{aligned}$$

for ion temperature (used for the collision operator) at cell centers:

$$\begin{aligned}
T^{\text{kipp}}_{i,i} &= T_{i,i} \qquad \text{for } i = 1, 2, \cdots, smax \\
T^{\text{kipp}}_{i,0} &= T_{i,-1} \\
T^{\text{kipp}}_{i,t} &= T_{i,nx} \qquad (\text{B.14})
\end{aligned}$$

The subscript 't' denotes variables at the right face of cell $smax$ (the target boundary). The superscript 'kipp' denotes variables specified in the KIPP part which will be indicated as $n^{\text{k}}_{e,i}$, $\Gamma^{\text{k}}_{e\|,if}$, $T^{\text{k}}_{e,i}$, $T^{\text{k}}_{i,i}$ in the following discussions for clarity. Then the KIPP loop starts solving the distribution function $f_{e,i}$ by maintaining profiles of macroscopic parameters. These profiles are maintained constant by automatic sources where perpendicular transport $P_{\perp if}$ in the SOLPS part is automatically included as a source in the KIPP part.

B.3.5 Effective heat transport coefficient c_e^{eff} calculated in the KIPP part

In the SOLPS part, parallel electron heat conduction coefficients and relevant parameters are specified in the subroutines *b2tqce.F*, *b2trcl.F*. By default, the electron heat conduction coefficient and parallel heat diffusivity are calculated at cell centers. For cell i:

$$c_{e,i} = 3.16 \qquad (\text{B.15})$$

$$\begin{aligned}
D_{e\|,i} &= c_{e,i} n_{e,i} \chi_{e\|,i} \\
&= c_{e,i} \frac{n_{e,i} \tau_{e,i} T_{e,i}}{m_e} \qquad (\text{B.16})
\end{aligned}$$

As discussed in Appendix A, the hybrid scheme is used for the heat flux density. The electron parallel heat flux density is calculated in *b2tfhe.F*:

$$q_{e\|,if} = q^{\text{conv}}_{e,if} + q^{\text{cond}}_{e,if} \qquad (\text{B.17})$$

where the convective and conductive pieces are:

$$\begin{aligned}
q^{\text{conv}}_{e,if} &= \frac{3}{2}\Gamma_{e\|,if} \frac{T_{e,i-1} + T_{e,i}}{2} + \left| \frac{3}{2}\Gamma_{e\|,if} \right| \frac{T_{e,i-1} - T_{e,i}}{2} \qquad (\text{B.18}) \\
q^{\text{cond}}_{e,if} &= C_{e\|,if} \frac{(T_{e,i} - T_{e,i-1})}{(hx_{i-1} + hx_i) / (2b_x)} \qquad (\text{B.19})
\end{aligned}$$

where

$$C_{e\|,if} = \left(\sqrt{D^2_{e\|,if} + F^2_{e\|,if}} - F_{e\|,if}\right) \tag{B.20}$$

$$F_{e\|,if} = \left|\frac{3}{4}\Gamma_{e\|,if}\right| \tag{B.21}$$

$\Gamma_{e\|,if}$ is the electron parallel particle flux density at the left face of cell i calculated in subroutine *b2tfnb.F*. $D_{e\|,if}$ is the left face interpolation of $D_{e\|,i}$:

$$D_{e\|,if} = \frac{V_{i-1}D_{\|e,i-1} + V_i D_{\|e,i}}{V_{i-1} + V_i} \tag{B.22}$$

Eq. (B.22) can be rewritten as:

$$D_{e\|,if} = c_{e,if} \left.\frac{n_e \tau_e T_e}{m_e}\right|_{if} \tag{B.23}$$

where cell-face interpolated heat conduction coefficient $c_{e,if}$ and the term with $|_{if}$ are defined as:

$$c_{e,if} = \frac{V_{i-1}c_{e,i-1} + V_i c_{e,i}}{(V_{i-1} + V_i)} \tag{B.24}$$

$$\left.\frac{n_e \tau_e T_e}{m_e}\right|_{if} = \frac{n_{e,i-1}\tau_{e,i-1}T_{e,i-1}V_{i-1} + n_{e,i}\tau_{e,i}T_{e,i}V_i}{m_e\left(V_{i-1} + V_i\right)} \tag{B.25}$$

with $|_{if}$ denoting the interpolation of a variable onto the left face of cell i. The interpolation scheme is defined by Eq. (B.25).

In KIPP, electron distribution functions are specified at cell centers, e.g. $f_{e,i}$ is the electron distribution function at the center of cell i. At the same time, the numerical flux (for numerical implementation of $f_e v_\|$ see section 3.2.1) is specified at cell faces, e.g. F^{k}_{if} (denoted as $F^n_{i-\frac{1}{2}}$ in section 3.2.1) is the numerical flux at the left face of cell i. Hence the particle and heat flux densities at cell faces can be calculated:

$$\Gamma^{\mathrm{k}}_{e\|,if} = \int F^{\mathrm{k}}_{if} d\vec{v} \tag{B.26}$$

$$q^{\mathrm{k}}_{e\|,if} = \frac{1}{2}m_e \int F^{\mathrm{k}}_{if} v^2 d\vec{v} \tag{B.27}$$

One of the three coupling schemes can be selected to calculate the effective heat transport coefficient in the KIPP part, in the subroutine *kapae.f90*:

Scheme A "*center*": replaces the heat conduction coefficients at cell centers (Eq. (B.15)) in the SOLPS part with the ones calculated based on distribution functions $f_{e,i}$ at the cell centers in the KIPP part, $s = smax$

and $i = 1, 2, \cdots, smax - 1$:

$$c_{e,i}^{\mathrm{eff}} = \frac{1}{2} m_e \int f_{e,i} v'^2 v'_{\|} d\vec{v} / \left(\frac{n_{e,i}^{\mathrm{k}} \tau_{e,i}^{\mathrm{k}} T_{e,i}^{\mathrm{k}}}{m_e} \cdot \nabla_{\|} T_{e_i}^{\ \mathrm{k}} \right) \tag{B.28}$$

$$c_{e,s}^{\mathrm{eff}} = \frac{1}{2} m_e \int f_{e,s} v'^2 v'_{\|} d\vec{v} / \left(\frac{n_{e,s}^{\mathrm{k}} \tau_{e,s}^{\mathrm{k}} T_{e,s}^{\mathrm{k}}}{m_e} \cdot \nabla_{\|} T_{e_s}^{\ \mathrm{k}} \right) \tag{B.29}$$

where the electron temperature gradients are:

$$\begin{aligned} \nabla_{\|} T_{e_i}^{\ \mathrm{k}} &= \frac{ds_{i+1} + ds_i}{ds_{i-1} + 2ds_i + ds_{i+1}} \nabla_{\|} T_{e_{if}}^{\ k} \\ &+ \frac{ds_{i-1} + ds_i}{ds_{i-1} + 2ds_i + ds_{i+1}} \nabla_{\|} T_{e_{i+1f}}^{\ \mathrm{k}} \end{aligned} \tag{B.30}$$

$$\begin{aligned} \nabla_{\|} T_{e_s}^{\ \mathrm{k}} &= \frac{ds_s}{ds_{s-1} + 2ds_s} \nabla_{\|} T_{e_{sf}}^{\ \mathrm{k}} \\ &+ \frac{ds_{s-1} + ds_s}{ds_{s-1} + 2ds_s} \nabla_{\|} T_{e_t}^{\ k} \end{aligned} \tag{B.31}$$

Scheme B "*face*": replaces heat conduction coefficients at cell faces (Eq. (B.24)) in the SOLPS part with the ones calculated based on particle and heat flux densities at the cell faces in the KIPP part, $i = 1, 2, \cdots, smax$:

$$c_{e,if}^{\mathrm{eff,b}} = \left(q_{e\|,if}^{\mathrm{k}} - \frac{5}{2} \Gamma_{e\|,if}^{\mathrm{k}} T_{e,if}^{\mathrm{k}} \right) / \left(\frac{n_{e,if}^{\mathrm{k}} \tau_{e,if}^{\mathrm{k}} T_{e,if}^{\mathrm{k}}}{m_e} \cdot \nabla_{\|} T_{e_{if}}^{\ \mathrm{k}} \right) \tag{B.32}$$

$$c_{e,t}^{\mathrm{eff,b}} = \left(q_{e\|,t}^{\mathrm{k}} - \frac{5}{2} \Gamma_{e\|,t}^{\mathrm{k}} T_{e,if}^{\mathrm{k}} \right) / \left(\frac{n_{e,t}^{\mathrm{k}} \tau_{e,t}^{\mathrm{k}} T_{e,t}^{\mathrm{k}}}{m_e} \cdot \nabla_{\|} T_{e_t}^{\ \mathrm{k}} \right) \tag{B.33}$$

where the electron temperature gradients are:

$$\nabla_{\|} T_{e_{if}}^{\ \mathrm{k}} = 2 \frac{T_{e,i}^{\mathrm{k}} - T_{e,i-1}^{\mathrm{k}}}{ds_i + ds_{i-1}} \tag{B.34}$$

$$\nabla_{\|} T_{e_t}^{\ \mathrm{k}} = 2 \frac{T_{e,t}^{\mathrm{k}} - T_{e,smax}^{\mathrm{k}}}{ds_{smax}} \tag{B.35}$$

Scheme C "*decouple*": replaces $C_{e\|,if}$ (B.20) from the hybrid scheme and decouples the conductive and convective pieces to force the Eq. (B.19) having the same heat conduction coefficients as calculated based on particle and heat flux densities at the cell faces in KIPP. In the KIPP part, at the cell faces, $i = 1, 2, \cdots, smax$:

$$c_{e,if}^{\mathrm{eff,c}} = \left(q_{e\|,if}^{\mathrm{k}} - \frac{5}{2} \Gamma_{e\|,if}^{\mathrm{k}} T_e^{\mathrm{k,up}} \right) / \left(\left. \frac{n_e^{\mathrm{k}} \tau_e^{\mathrm{k}} T_e^{\mathrm{k}}}{m_e} \right|_{if} \cdot \nabla_{\|} T_{e_{if}}^{\ \mathrm{k}} \right) \tag{B.36}$$

$$c_{e,t}^{\mathrm{eff,c}} = \left(q_{e\|,t}^{\mathrm{k}} - \frac{5}{2} \Gamma_{e\|,t}^{\mathrm{k}} T_e^{\mathrm{k,up}} \right) / \left(\left. \frac{n_e^{\mathrm{k}} \tau_e^{\mathrm{k}} T_e^{\mathrm{k}}}{m_e} \right|_{t} \cdot \nabla_{\|} T_{e_t}^{\ \mathrm{k}} \right) \tag{B.37}$$

where $\nabla_{\parallel} T_{e_{if}}^{\mathrm{k}}$ and $\nabla_{\parallel} T_{e_t}^{\mathrm{k}}$ are taken from Eqs. (B.34) and (B.35). $T_e^{\mathrm{k,up}}$ is the upstream electron temperature. In the coupling test, $T_e^{\mathrm{k,up}} = T_{e,i-1}^{\mathrm{k}}$ when $\Gamma_{e\parallel,if}^{\mathrm{k}} > 0$, $T_e^{\mathrm{k,up}} = T_{e,i}^{\mathrm{k}}$ when $\Gamma_{e\parallel,if}^{\mathrm{k}} < 0$. $\left.\frac{n_e^{\mathrm{k}}\tau_e^{\mathrm{k}}T_e^{\mathrm{k}}}{m_e}\right|_{if}$ and $\left.\frac{n_e^{\mathrm{k}}\tau_e^{\mathrm{k}}T_e^{\mathrm{k}}}{m_e}\right|_{t}$ have the same form as in Eq. (B.25).

After the KIPP loop, the effective heat transport coefficients at cell centers from Scheme A based on Eqs. (B.28) and (B.29), and at cell faces from Scheme B based on Eqs. (B.32) and (B.33), and those from Scheme C based on Eqs. (B.36) and (B.37) are transferred to the SOLPS part via the module *KIPP_2_SOLPS.F* and the subroutine *output_for_SOLPS.f90*.

In the SOLPS part, the choice of a coupling scheme is determined by the control parameter *’b2tqce_transp_KIPP’* in the input file *b2mn.dat*. If *’b2tqce_transp_KIPP’* is set to 0, the default values will be used. Otherwise, if *’b2tqce_transp_KIPP’* is set to 1, effective heat conduction and thermal force coefficients (described in the following subsection) will be used in the SOLPS loop. Control parameters *’tempro_flat’* and *’force_match’* in the file *runpar.dat* determine which coupling scheme will be used in the SOLPS loop:

if *’tempro_flat’* is set to 0, the cell ”*center*” coupling regime (Scheme A) will be used in a coupling run. Replace Eq. (B.15) with Eqs. (B.28) and (B.29) in the subroutine *b2tqce.F*,

$$\begin{aligned} c_{e,i} &= c_{e,i}^{\mathrm{eff}} \qquad \text{for } i = 1, 2, \cdots, nx-1 \\ c_{e,-1} &= c_{e,1}^{\mathrm{eff}} \\ c_{e,0} &= c_{e,1}^{\mathrm{eff}} \\ c_{e,nx} &= c_{e,nx-1}^{\mathrm{eff}} \end{aligned} \tag{B.38}$$

Therefore the conductive heat flux density is calculated by Eqs. (B.19), (B.20), (B.22), (B.16) and (B.38).

if *’tempro_flat’* is set to 1 and *’force_match’* is set to 0, the cell ”*face*” coupling (Scheme B) will be used in a coupling run. Replace Eq. (B.24) with Eqs. (B.32) and (B.33) in the subroutine *b2trcl.F*,

$$\begin{aligned} c_{e,if} &= c_{e,if}^{\mathrm{eff,b}} \qquad \text{for } i = 1, 2, \cdots, nx-1 \\ c_{e,0f} &= c_{e,1f}^{\mathrm{eff,b}} \\ c_{e,nxf} &= c_{e,t}^{\mathrm{eff,b}} \end{aligned} \tag{B.39}$$

Therefore the conductive heat flux density is calculated by Eqs. (B.19), (B.20), (B.22), (B.23) and (B.39)

if *’tempro_flat’* is set to 1 and *’force_match’* is set to 1, the ”*decouple*” coupling (Scheme C) will be used in a coupling run to decouple the conductive and convective pieces. Replace Eq. (B.20) with the following formulas in the subroutine *b2tfhe.F*:

$$\begin{aligned}
C_{e\|,if} &= c^{\text{eff,c}}_{e,if} \left.\frac{n_e\tau_e T_e}{m_e}\right|_{if} \qquad \text{for } i = 1, 2, \cdots, nx-1 \\
C_{e\|,0f} &= c^{\text{eff,c}}_{e,1f} \left.\frac{n_e\tau_e T_e}{m_e}\right|_{0f} \\
C_{e\|,nxf} &= c^{\text{eff,c}}_{e,t} \left.\frac{n_e\tau_e T_e}{m_e}\right|_{nxf}
\end{aligned} \tag{B.40}$$

Therefore the conductive heat flux density is calculated by Eqs. (B.19) and (B.40).

One should note that, for Scheme C, the coefficient entering the electron energy conservation solver (Eq. (A.8)) should be modified accordingly. The convective piece in the SOLPS part has the coefficient $\frac{3}{2}$ while it is $\frac{5}{2}$ in the KIPP part. This is because the "internal" energy conservation equation is used in SOLPS where part of the convective heat flux is moved to the right hand side and treated as a source term, while in KIPP the third moment of the distribution function is the total heat flux instead of the internal heat flux. At the boundary, the discrepancy originating from this difference will be discussed in detail in B.3.7. In chapter 4, the three coupling schemes are first compared, giving the same results, then in the later analysis the Scheme C is adopted.

B.3.6 Effective thermal force coefficient $k_\|$ calculated in the KIPP part

Similar to the above discussion, the thermal force coefficient and thermal electric coefficient are specified in the subroutine *b2tqce.F* in the SOLPS part with the default value:

$$k_{\|,i} = -0.71 \qquad \text{for } i = -1, 0, \cdots, nx \tag{B.41}$$

$$\alpha_{\|,i} = k_{\|,i}\frac{\sigma_{e\|,i}}{e} \tag{B.42}$$

where $\sigma_{e\|,i}$ is parallel electric conductivity at the center of cell i. $\alpha_{\|,i}$ is then used to calculate electric currents and solve the charge conservation equation. Electric field consists of two parts if ambipolar flows are assumed:

$$E_\| = E_{\text{thr}} + E_{p_e} \tag{B.43}$$

where the terms due to thermal force and electron pressure gradients are given by:

$$E_{\text{thr}} = \frac{k_\|}{e}\nabla_\| T_e \tag{B.44}$$

$$E_{p_e} = -\frac{1}{en}\nabla_\| p_e \tag{B.45}$$

In the KIPP part, the ambipolar flow is maintained at each time step by adjusting:

$\boldsymbol{E}^{\mathrm{k}}_{\boldsymbol{p_e,i}}$ the part of the electric field due to electron pressure gradient at the center of cell i, for $i = 1, 2, \cdots, smax$.

$\boldsymbol{E}^{\mathrm{k}}_{\mathbf{thr},\boldsymbol{i}}$ the part of the electric field due to thermal force at the center of cell i, for $i = 1, 2, \cdots, smax$.

$E^{\mathrm{k}}_{p_e,i}$ corresponds to the free-streaming term (the pressure gradient term in the frame of a fluid model), while $E^{\mathrm{k}}_{\mathrm{thr},i}$ corresponds to the collision term. Effective thermal force coefficients at cell centers are calculated in the subroutine *kapae.f90*:

$$
\begin{aligned}
k^{\mathrm{eff}}_{\|,i} &= eE^{\mathrm{k}}_{\mathrm{thr},i}/\nabla_{\|}T_{e_i}^{\mathrm{k}} && \text{for } i = 1, 2, \cdots, samx - 1 && \text{(B.46)}\\
k^{\mathrm{eff}}_{\|,s} &= eE^{\mathrm{k}}_{\mathrm{thr},s}/\nabla_{\|}T_{e_s}^{\mathrm{k}} && \text{for } s = smax && \text{(B.47)}
\end{aligned}
$$

where $\nabla_{\|}T_{e_i}^{k}$ and $\nabla_{\|}T_{e_s}^{k}$ are taken from Eqs. (B.30) and (B.31). The effective thermal force coefficients are then transferred to the SOLPS part by the module *KIPP_2_SOLPS.F* and the subroutine *output_for_SOLPS.f90*.

In the SOLPS part, as mentioned above, the use of the effective thermal force coefficients is also determined by the control parameter *'b2tqce_transp_KIPP'* in the input file *b2mn.dat* specified by the user. If *'b2tqce_transp_KIPP'* is set to 1, the SOLPS loop will be run with the effective thermal force coefficients:

$$
\begin{aligned}
k_{\|,i} &= k^{\mathrm{eff}}_{\|,i} && \text{for } i = 1, 2, \cdots nx - 1\\
k_{\|,-1} &= k^{\mathrm{eff}}_{\|,1}\\
k_{\|,0} &= k^{\mathrm{eff}}_{\|,1}\\
k_{\|,nx} &= k^{\mathrm{eff}}_{\|,nx-1} && && \text{(B.48)}
\end{aligned}
$$

B.3.7 Effective electron sheath heat transmission coefficient and sheath potential drop

The modified boundary condition of SOLPS (numerical details can be found in Appendix A.3) is used in coupling runs as discussed above and in section 4. In the following discussion, the boundary condition for the electron energy conservation equation in SOLPS is referred to as $\gamma'_{e\|}$ (instead of γ'_e which is for the default SOLPS boundary condition).

Without coupling with KIPP, the boundary conditions in SOLPS are specified in the subroutine *b2stbc_phys.F* (based on the same collisional plasma theory as discussed in Chapter 4), $\Delta\phi$ and $\gamma'_{e\parallel}$ are then determined by:

$$e\sum_a z_a \Gamma_{a\parallel,nxf} = e\frac{1}{4}n_{e,nxf}\overline{c_{e,nxf}}\exp^{e\Delta\phi/T_{e,nxf}} \tag{B.49}$$

$$\gamma'_{e\parallel} = c_1 + \frac{e|\Delta\phi|}{T_{e,nxf}} \tag{B.50}$$

The first equation assumes equal ion and electron particle flux densities across the simulation boundary. c_1 is a constant specified by the user (default value is 0.9) in the input file *b2.boundary.parameters*. The subscript nxf indicates the left face of cell nx, corresponding to the simulation boundary. $\Gamma_{a\parallel,nxf}$ is the particle flux density of species a at the left face of cell nx:

$$\Gamma_{a\parallel,nxf} = n_{a,nxf}u_{a,nxf} \tag{B.51}$$

and $\overline{c_{e,nxf}}$ is the electron average thermal speed of a Maxwellian, given by:

$$\overline{c_{e,nxf}} = \sqrt{\frac{8T_{e,nxf}}{\pi m_e}} \tag{B.52}$$

where $u_{a,nxf}$ is determined by the boundary condition for the momentum conservation equation based on the Bohm's criteria:

$$u_{a,nxf} \geq \sqrt{\frac{\gamma^i T_{i,nxf} + z_a^2 n_{a,nxf}T_{e,nxf}/n_{e,nxf}}{m_i}} \tag{B.53}$$

where γ^i is the adiabatic coefficient, with the default value $\frac{5}{3}$.

We would like to remind that the internal energy conservation equation is implemented in SOLPS, meaning that the parallel convective heat flux density numerically defined in SOLPS is with coefficient $\frac{3}{2}$ instead of $\frac{5}{2}$, where the part $\left(\frac{5}{2}-\frac{3}{2}\right)$ has been moved to the right hand side of the equation and regarded as a source. Therefore, a numerical mismatch arises between the boundary heat flux term in the SOLPS part and that in the KIPP part. Hence one has to be careful in transferring the effective heat transmission coefficient from the KIPP part to the SOLPS part. We will discuss this below.

The relation between $\gamma'_{e\parallel}$ and the electron sheath heat transmission coefficient $\gamma_{e\parallel}$ in SOLPS

The internal energy equation solved in SOLPS is

$$\frac{\partial}{\partial t}\left(\frac{3}{2}n_e T_e\right) + \nabla_x\left[b_x\left(\frac{3}{2}\Gamma_{e\parallel}T_e + q_{e\parallel}^{\mathrm{cond}}\right) + b_z q_{e\perp}\right] \\ = S_{Ee} - n_e T_e \nabla_x b_x u_{e\parallel} - Q_\Delta \tag{B.54}$$

In the ghost cell nx (shown in Fig. A.3b) energy is balanced between the heat flux flowing through the left face and the term due to the boundary condition (Eq. (A.34)):

$$\begin{aligned} & b_x \left(\tfrac{3}{2}\Gamma_{e\|,nxf} T_{e,nxf} + q^{\text{cond}}_{e\|,nxf}\right) A + Q_{e\perp,nxf} \\ & = \left(c_1 + \tfrac{e|\Delta\phi|}{T_{e,nxf}}\right) b_x \Gamma_{e\|,nxf} T_{e,nxf} A + Q_{e\perp,nxf} \end{aligned} \tag{B.55}$$

where the left side is the numerical heat flux through the boundary and the right side is the boundary condition specified. If one then looks at the energy balance of cell $nx-1$ (the energy flowing out of the right face of this cell is the one that flows into cell nx) in steady state, the balance equation is:

$$\begin{aligned} & \left[b_x \left(\frac{3}{2}\Gamma_{e\|,nxf} T_{e,nxf} + q^{\text{cond}}_{e\|,nxf}\right) A + Q_{e\perp,nxf}\right] \\ - & \left[b_x \left(\frac{3}{2}\Gamma_{e\|,nx-1f} T_{e,nx-1f} + q^{\text{cond}}_{e\|,nx-1f}\right) A + Q_{e\perp,nx-1f}\right] \\ = & \left(S_{Ee} - Q_\Delta - n_e T_e \nabla_x b_x u_{e\|}\right) V_{nx-1} \end{aligned} \tag{B.56}$$

However, the third term on the right hand side is not the real physical source in this cell. It has been partly cancelled as follows:

$$n_e T_e \nabla_x b_x u_{e\|} V_{nx-1} = b_x \left(n_{e,nxf} T_{e,nxf} u_{e\|,nxf} - n_{e,nx-1f} T_{e,nx-1f} u_{e\|,nx-1f}\right) A - b_x \nabla_x n_e T_e \tag{B.57}$$

where

$$n_{e,nxf} u_{e\|,nxf} \approx \Gamma_{e\|,nxf} \tag{B.58}$$

$$n_{e,nx-1f} u_{e\|,nx-1f} \approx \Gamma_{e\|,nx-1f} \tag{B.59}$$

Substitution of Eq. (B.57) into Eq. (B.56) leads to:

$$\begin{aligned} & \left[b_x \left(\frac{5}{2}\Gamma_{e\|,nxf} T_{e,nxf} + q^{\text{cond}}_{e\|,nxf}\right) A + Q_{e\perp,nxf}\right] \\ - & \left[b_x \left(\frac{5}{2}\Gamma_{e\|,nx-1f} T_{e,nx-1f} + q^{\text{cond}}_{e\|,nx-1f}\right) A + Q_{e\perp,nx-1f}\right] \\ = & \left(S_{Ee} - Q_\Delta + b_x u_{e\|} \nabla_x n_e T_e\right) V_{nx-1} \end{aligned} \tag{B.60}$$

The above equation is the full electron energy conservation balance equation. Therefore, combined with Eq. (B.55), the parallel heat flux density flowing out of this cell through its right face becomes

$$\frac{5}{2}\Gamma_{e\|,nxf} T_{e,nxf} + q^{\text{cond}}_{e\|,nxf} \approx \left(c_1 + 1 + \frac{e|\Delta\phi|}{T_{e,nxf}}\right) \Gamma_{e\|,nxf} T_{e,nxf} \tag{B.61}$$

Hence $\gamma_{e\|}$ $(= \gamma'_{e\|} + 1)$ is the actual electron heat transmission coefficient ($\gamma'_{e\|}$ is only for the internal electron energy balance equation, however, the second moment of the Vlasov-Fokker-Planck equation solved in KIPP gives the total electron energy balance equation).

Calculating and transferring effective boundary coefficients

In the KIPP part, the effective electron heat transmission coefficient γ_e^{eff} and the sheath potential drop $\Delta\phi^{\text{eff}}$ are calculated in the subroutine *heat_coefficient_tar.f90* by solving the following equation:

$$\int_{v_c} \left(f_{e,s} + \frac{1}{2} \left(ds_s - v_\parallel dt \right) \sigma_s(v_\parallel, v_\perp) \right) v_\parallel d\vec{v} = \Gamma^{\text{k}}_{e\parallel,t} \qquad \text{for } s = smax \quad \text{(B.62)}$$

which follows from the logical sheath condition, from which v_c can be determined. $\sigma_s(v_\parallel, v_\perp)$ is calculated based on the Lax-Wendorff method with the modified MC limiter (see section 3.2.1). Then the effective boundary conditions are determined by:

$$\Delta\phi^{\text{eff}} = -\frac{1}{2}\frac{m_e v_c^2}{T^{\text{k}}_{e,t}} \quad \text{(B.63)}$$

$$\gamma_e^{\text{eff}} = \frac{1}{2}\frac{m_e \int_{v_c} \left(f_{e,s} + \frac{1}{2} \left(ds_s - v_\parallel dt \right) \sigma_s(v_\parallel, v_\perp) \right) v_\parallel v^2 d\vec{v}}{\Gamma^{\text{k}}_{e\parallel,t} T^{\text{k}}_{e,t}} \quad \text{(B.64)}$$

The effective boundary coefficients are then transferred to SOLPS via the module *KIPP_2_SOLPS.F* and the subroutine *output_for_SOLPS.f90*.

The use of the effective boundary conditions in the SOLPS part is determined by the control parameter *'b2stbc_boundary_KIPP'* in the input file *b2mn.dat*. If *'b2stbc_boundary_KIPP'* is set to '1', the SOLPS loop will run with the effective boundary conditions from Eqs. (B.63) and (B.64) by specifying:

$$\Delta\phi = \Delta\phi^{\text{eff}} \quad \text{(B.65)}$$

$$\gamma'_{e\parallel} = \gamma_e^{\text{eff}} - 1 \quad \text{(B.66)}$$

One should note that in a coupling run the effective boundary conditions are only used when *'BCENE'* is set to '3' in the input file *b2.boundary.parameters* which is the default value for the target boundaries.

Summary on γ'_e, $\gamma'_{e\parallel}$, $\gamma_{e\parallel}$, γ_e^{eff}

There are several coefficients discussed above relevant to electron sheath heat transmission coefficient which might be confusing to readers. For clarity, here we list all of them together:

γ'_e is the coefficient specified for the SOLPS default boundary condition for the "internal" electron energy conservation equation.

$\gamma'_{e\parallel}$ is the coefficient specified for the adapted SOLPS boundary condition (details can be found in Appendix A.3.2) for the "internal" electron energy conservation equation.

$\gamma_{e\|}$ is electron sheath heat transmission coefficient. $\gamma_{e\|} = \gamma'_{e\|} + 1$.

γ_e^{eff} is effective electron sheath heat transmission coefficient calculated in the KIPP part.

Since the SOLPS boundary condition has been adapted to the coupling scheme, γ'_e is not used in this work, instead $\gamma'_{e\|}$ is used as a free parameter for the SOLPS part. In coupling runs, we calculate γ_e^{eff} in the subroutine *heat_coefficient_tar.f90* in the KIPP part and then specify $\gamma'_{e\|} = \gamma_e^{\text{eff}} - 1$ in the subroutine *b2stbc.F* in the SOLPS part. $\gamma_{e\|}$ does not appear in the code, but is introduced for discussions.

B.4 Added control parameters and switches

B.4.1 in *runpar.dat*

'fully_coupling' = integer

0 (default): Iterative coupling.

1 (not used): Real time coupling.

'internal_kipp_step' = integer

specifies the number of time steps in the KIPP loop.

'n_norm' = double precision

specifies reference density (normally the value near or at the stagnation point) for normalization.

'T_norm' = double precision

specifies reference temperature (normally the value near or at the stagnation point) for normalization.

'lambda_norm' = double precision

15 (default): specifies reference Coulumb Logarithm (normally the value near or at the stagnation point) for normalization.

'source_1d' = double precision

1.0×10^6 **(default):** specifies the power source into the simulation domain. The power is evenly distributed over the region from cell 0 to cell s, which is given by the parameter *'b2sral_1Dsource_end'* in *'b2mn.dat'*. The power input density into each cell is given by:

$$S_E = \mathit{source\_1d} \times \frac{A}{\sum_0^s hx_i}$$

'use_e_sub' = integer

0 (default): uses subroutine *energy_input_g.f90* for power input.

1: uses subroutine *energy_input_h.f90* for power input.

'exun' = integer

0 (default): uses subroutine *energy_input_g.f90* to specify radiation power (uniformly).

1: uses subroutine *e_cooling_ion_coupling.f90* to specify radiation power based on the ground state ionization cross section.

2: uses subroutine *e_cooling_simple.f90* to specify radiation power by subtracting it only from the electrons with energy above the value *'exun_thresh'*.

3: uses subroutine *e_cooling_inelastic.f90* to calculate the radiation power based on the inelastic collision operator.

'exun_thresh' = double precision

0.0 (default): specifies the threshold when *'exun'* is set to 2.

'solps_force_continue' = integer

0 (default): if *'continue_case'* is set to 'false' in *runpar.dat*, in SOLPS part, it runs with default coefficients and boundary conditions and KIPP part will run starting with initial Maxwellian distributions.

1: if *'continue_case'* is set to 'false' in *runpar.dat*, in SOLPS part, it runs with saved effective coefficients and boundary conditions under the directory './kipp_out/continue_case/' and KIPP part will run starting with initial Maxwellian distributions.

'tempro_flat' = integer

0 (default): cell center coupling scheme (Scheme A) of effective heat conduction coefficients is used.

1: cell face coupling scheme (Scheme B) of effective heat conduction coefficients is used.

'all_conv' = integer

0: does nothing.

1 (default): saves the evolution of the effective coefficients and boundary conditions in the KIPP part to './kipp_out/continue_case/iterative'.

'con_test' = integer

0 (default): does nothing.

1: calculates coefficients γ_{ve}, γ_{vth} and print

$$\gamma_{ve,if} = \frac{q^{\mathrm{k}}_{e\|,if}}{\Gamma^{\mathrm{k}}_{e\|,if} T^{\mathrm{k}}_{e,if}} \tag{B.67}$$

$$\gamma_{vth,if} = \frac{q^{\mathrm{k}}_{e\|,if} - \frac{5}{2}\Gamma^{\mathrm{k}}_{e\|,if} T^{\mathrm{k}}_{e,if}}{n^{\mathrm{k}}_{e,if}\sqrt{T^{\mathrm{k}}_{e,if}/m_e}\, T^{\mathrm{k}}_{e,if}} \tag{B.68}$$

'on_flux_correc' = integer

0 (default): does particle flux correction after free-streaming calculations in the KIPP part.

1 (for testing): does nothing.

'force_match' = integer

0 (default): does nothing.

1: if *'tempro_flat'* is set to 1, the cell face coupling scheme with decoupled convective and conductive electron parallel heat fluxes (Scheme C) is used.

'conv_only' = integer (this parameter is used to study the divergence phenomenon in low density cases)

0 (default): does nothing.

1 (doesn't work due to numerical instability): the entire electron parallel heat flux calculated in KIPP is transferred to SOLPS and treated as the convective piece.

2 (not used):

3 (not used):

4: limits the heat conduction coefficient to *'ce_thresh'*.

'ce_thresh' = double precision

0.0 (default): limits the heat conduction coefficient to the specified value. The rest of the heat flux is moved to the convective piece.

'conv_only4_extrem' = integer

0 (default): does nothing.

1 (doesn't work due to numerical instability): fixes heat conduction coefficient $c_e = 3.16$ and manipulates the convective piece to incorporate flux limiting or enhancement. This is not working well due to the fact that in the upstream region with typically flux limiting, parallel heat is mostly carried by the conductive piece. Modification of the convective piece to incorporate this flux limiting would result in negative coefficients (instead of 2.5) and cause numerical instability in SOLPS.

'tbrag_ttrub' = integer

0 (default): assumes the ratio of Braginskii electron collision time to Trubnikov one is $\tau_B/\tau_T = 3.76$.

1: assumes $\tau_B/\tau_T = \frac{3}{2}\sqrt{2\pi}$ (slightly higher precision).

B.4.2 in *b2mn.dat*

'b2mndr_internal_nt' = integer

specifies the number of time steps in the SOLPS loop.

'b2sral_1Dsource_end' = integer

-1 (default): specifies an integer s. The power is introduced evenly with constant power density along the region from cell 0 to cell s. If set to -1, the power will be introduced evenly along the whole simulation domain.

'b2tqce_transp_KIPP' = integer

0 (default): runs SOLPS loop with default electron heat conduction and e-i thermal force coefficients.

1: runs SOLPS loop with effective electron heat conduction and e-i thermal force coefficients.

'b2tqce_transp_force_evolv' = integer

0 (default): does nothing.

1: if *'b2stbc_boundary_KIPP'* is set to 0 but the subroutine *b2mndr.F* is called the second time or further, run SOLPS loop with effective heat conduction and e-i thermal force coefficients.

'b2tqce_no_kippalf' = integer

0 (default): does nothing.

1: runs the SOLPS loop with the default e-i thermal force coefficient even if *'b2tqce_transp_KIPP'* is set to 1.

'b2trcl_test_tar_ce' = integer

0 (default): does nothing.

1: if *'tempro_flat'* is set to 1, uses the boundary heat conduction coefficient $c_{e,nxf} = 3.16$.

'b2stbc_boundary_KIPP' = integer

0 (default): runs SOLPS loop with default values of the electron heat transmission coefficient and sheath potential drop.

1: runs SOLPS loop with effective values of the electron heat transmission coefficient and sheath potential drop.

'b2stbc_boundary_force_evolve' = integer

0 (default): does nothing.

1: if *'b2stbc_boundary_KIPP'* is set to 0 but the subroutine *b2mndr.F* is called the second time or further, run SOLPS loop with effective values of the electron heat transmission coefficient and sheath potential drop.

'b2sral_pow_from_KIPP' = integer

0 (default): power input density is specified by the parameter *'source_1d'* in the input file *runpar.dat.*

1 (not used): power input density is specified from reading files.

'b2sral_1dC_radialoss' = integer

0 (default): does not introduce additional sinks for carbon.

1 (necessary for D+C cases): introduces particle sinks for carbon particles. The sinks are controlled by the following control parameters:

'b2sral_1dC_lossfac_neutral' = double precision
'b2sral_1dC_lossfac_ion' = double precision
'b2sral_1dC1_lossfac2' = integer
'b2sral_1dC2_lossfac2' = integer
'b2sral_1dC3_lossfac2' = integer
'b2sral_1dC4_lossfac2' = integer
'b2sral_1dC5_lossfac2' = integer
'b2sral_1dC6_lossfac2' = integer
'b2sral_1dCi_uloss' = integer
'b2sral_1dC1_loss_rec' = double precision
'b2sral_1dC2_loss_rec' = double precision
'b2sral_1dC3_loss_rec' = double precision
'b2sral_1dC4_loss_rec' = double precision
'b2sral_1dC5_loss_rec' = double precision
'b2sral_1dC6_loss_rec' = double precision

'b2sral_1dD_radialoss' = integer

0 (default): does not introduce additional sinks for deuterium.

1: introduces sinks for deuterium particles. The sinks are controlled by the following control parameters:

'b2sral_1dD_lossfac_neutral' = double precision
'b2sral_1dD_lossfac_ion' = double precision

'b2sral_1dener_radialoss' = integer

0 (default): does not introduce energy sinks corresponding to the particle sinks mentioned above.

1: introduces the energy sinks controlled by parameters *'b2sral_1dC_radialoss'* and *'b2sral_1dD_radialoss'*.

'b2out_potential_for2D' = integer

0 (default): only 1D parallel transport is considered.

1: poloidal transport is assumed to be the sum of poloidal projections of parallel and perpendicular transport. The modified boundaries are used, as described in Chapter 4.

'b2news_poteq' = integer

3: if *'b2tfhe_no_current'* is set to 1 (switches off currents), 1D potential profile is calculated by the subroutine *potential_1D_KIPP*.

Bibliography

[1] S. Anwar, *Encyclopedia of Energy Engineering and Technology* (CRC Press, ADDRESS, 2014).

[2] F. F. Chen, *Introduction To Plasma Physics and Controlled Fusion* (Plenum Press, New York and London, 1984).

[3] R. W. Conn *et al.*, Nuclear Fusion **30**, 1919 (1990).

[4] J. D. Lawson, Proceedings of the Physical Society 6 (1957).

[5] J. Wesson, *Tokamaks* (Clarendon Press, Oxford, United Kingdom, 2004).

[6] P. C. Stangeby, *The Plasma Boundary of Magnetic Fusion Devices* (Institute of Physics, Bristol, UK, 2000).

[7] L. A. Artsimovitsh, Nuclear Fusion **12**, 215 (1972).

[8] R. J. Goldston, Plasma Physics and Controllel Fusion **26**, 87 (1984).

[9] B. B. Kadomtsev and E. W. Laing, *Tokamak plasma, a complex physical system* (IOP, Bristol, 1992).

[10] A. Kallenbach *et al.*, Nuclear Fusion **57**, 102015 (2017).

[11] X. Litaudon *et al.*, Nuclear Fusion **57**, 102001 (2017).

[12] J.-G. Kwak *et al.*, Nuclear Fusion **53**, 104005 (2013).

[13] B. N. Wan *et al.*, Nuclear Fusion **57**, 102019 (2017).

[14] J. L. Luxon *et al.*, Fusion Science and Technology **48**, 807 (2005).

[15] E. D. Pietro *et al.*, Fusion Engineering and Design **89**, 2128 (2014).

[16] E. D. Pietro *et al.*, Nuclear Fusion **41**, 1301 (2001).

[17] D. E. Post and R. Behrisch, *Physics of Plasma-Wall Iteractions in Controlled Fusion* (Plenum Press, New York and London, 1984).

[18] M. Kaufmann and R. Neu, Fusion Engineering and Design **82**, 521 (2007).

[19] A. Kallenbach *et al.*, Plasma Physics and Controlled Fusion **47**, (2005).

[20] A. Loarte *et al.*, Nuclear Fusion **47**, (2007).

[21] R. Schneider *et al.*, Contrib. Plasma Phys. **46**, 3 (2006).

[22] S. I. Braginskii, in *Reviews of Plasma Physics* (Consultants Bureau, New York, 1965), Vol. 1, Chap. Transport Processes In A Plasma, p. 205.

[23] R. Chodura, Contrib. Plasma Phys. **28**, 303 (1988).

[24] R. Chodura, Contrib. Plasma Phys. **28**, 325 (1988).

[25] R. Chodura, Contrib. Plasma Phys. **30**, 153 (1990).

[26] R. Chodura, Contrib. Plasma Phys. **32**, 219 (1992).

[27] S. I. Krasheninnikov, Sov. Phys. JETP **67**, 2483 (1988).

[28] S. I. Krasheninnikov, Physics of Plasmas **5**, 74 (1993).

[29] J. R. Albritton, E. A. Williams, I. B. Bernstein, and K. P. Swartz, Phys. Rev. Lett. **57**, 1887 (1986).

[30] A. M. Mirza *et al.*, Physics Letters A **141**, 56 (1989).

[31] G. Murtaza *et al.*, Physics Letters A **144**, 164 (1990).

[32] P. J. Catto and K. Yeoh, Contrib. Plasma Phys. **38**, 207 (1998).

[33] P. J. Catto and M. Grinneback, Physics Letters A **277**, 323 (2000).

[34] E. Zawaideh, F. Najmabadi, and R. W. Conn, Physics of Fluids **29**, 463 (1986).

[35] E. Zawaideh, F. Najmabadi, and R. W. Conn, Physics of Fluids **29**, 3702 (1986).

[36] E. Zawaideh and N. S. Kim, Physics of Fluids **31**, 3280 (1988).

[37] G. J. Radford, Contrib. Plasma Phys. **32**, 297 (1992).

[38] G. J. Radford *et al.*, Contrib. Plasma Phys. **36**, 187 (1996).

[39] G. J. Radford *et al.*, Contrib. Plasma Phys. **38**, 183 (1998).

[40] Y. L. Igitkhanov and P. N. Yushmanov, Contrib. Plasma Phys. **28**, 341 (1988).

[41] Y. L. Igitkhanov and A. Y. Pigarov, Journal of Nuclear Materials **176**, 557 (1990).

[42] A. Bergmann *et al.*, Contrib. Plasma Phys. **36**, 192 (1996).

[43] W. Fundamenski, Plasma Phys. Control. Fusion **47**, (2005).

[44] J. F. Luciani, P. Mora, and J. Virmont, Phys. Rev. Lett. **51**, 18 (1983).

[45] J. R. Albritton *et al.*, Physical Review Letters **57**, 1887 (1986).

[46] B. P. Schurtz, P. D. Nicolai, and M. Busquet, Physics of Plamsas **7**, 4238 (2000).

[47] J. T. Omotani and B. D. Dudson, Plasma Phys. Control. Fusion **55**, 055009 (2013).

[48] M. Zhao, A. V. Chankin, and D. P. Coster, Nuclear Materials and Energy **12**, 819 (2017).

[49] O. V. Batishchev *et al.*, Phys. Plasmas **4**, 1672 (1997).

[50] A. Froese, T. Takizuka, and M. Yagi, Contrib. Plasma Phys. **52**, 534 (2012).

[51] K. Kupfer *et al.*, Phys. Plasmas **3**, 3644 (1966).

[52] D. Tskhakaya *et al.*, Contrib. Plasma Phys. **48**, 89 (2008).

[53] D. Tskhakaya *et al.*, Journal of Nuclear Material **390**, 335 (2009).

[54] C. K. Birdsall and A. B. Langdon, *Plasma Physics via Computer Simulation* (McGraw-Hill, New York, 1985).

[55] R. J. LeVeque, *Finite-Volume Method for Hyperbolic Problems* (Cambridge University Press, Cambridge, United Kingdom, 2004).

[56] T. Takizuka and H. Abe, J. Comput. Phys. **25**, 205 (1977).

[57] T. Takizuka, K. Tani, M. Azumi, and K. Shimizu, J. Nucl. Mater. **128**, 104 (1984).

[58] T. Takizuka, Plasma Science and Technology **12**, 316 (2011).

[59] T. Takizuka, Plasma Physics and Controlled Fusion **59**, 034008 (2017).

[60] A. Froese, Ph.D. thesis, Kyushu University, Kyushu, 2010.

[61] A. Froese, T. Takizuka, and M. Yagi, Contrib. Plasma Phys. **50**, 273 (2010).

[62] A. Froese, T. Takizuka, and M. Yagi, Plasma and Fusion Research **5**, 26 (2010).

[63] A. Froese, T. Takizuka, and M. Yagi, Plasma and Fusion Research **5**, (2010).

[64] M. Hosokawa *et al.*, Plasma and Fusion Research **11**, 1403104 (2016).

[65] D. Tskhakaya and R. Schneider, Journal of Computational Physics **225**, 829 (2007).

[66] D. Tskhakaya *et al.*, Contib. Plasma Phys. **47**, 563 (2007).

[67] D. Tskhakaya *et al.*, Contib. Plasma Phys. **48**, 89 (2008).

[68] D. Tskhakaya, Contib. Plasma Phys. **52**, 490 (2012).

[69] D. Tskhakaya *et al.*, Journal of Nuclear Materials **438**, (2013).

[70] D. Tskhakaya *et al.*, Journal of Nuclear Materials **463**, 624 (2015).

[71] D. Tskhakaya *et al.*, Physics of Plasmas **23**, 032128 (2016).

[72] D. Tskhakaya, Contrib. Plasma Phys. **56**, 698 (2016).

[73] D. Tskhakaya, Plasma Phys. Control. Fusion **59**, 114001 (2017).

[74] O. V. Batishchev *et al.*, Contrib. plasma Phys. **34**, 436 (1994).

[75] O. V. Batishchev *et al.*, Phys. Plasmas **3**, 3386 (1996).

[76] O. V. Batishchev *et al.*, Contributions to Plasma Physics **36**, 225 (1996).

[77] M. A. Dorf *et al.*, Contrib. Plasma Phys. **52**, 518 (2012).

[78] M. A. Dorf *et al.*, Contrib. Plasma Phys. **54**, 517 (2014).

[79] M. A. Dorf *et al.*, Phys. Plasmas **23**, 056102 (2016).

[80] P. L. Bhatnagar, E. P. Gross, and M. Krook, Physical Review **94**, 511 (1954).

[81] B. A. Trubnikov, in *Reviews of Plasma Physics* (Consultants Bureau, New York, 1965), Vol. 1, Chap. Particle Interactions in A Fully Ionized Plasma, p. 105.

[82] A. S. Kukushkin and A. M. Runov, Contrib. Plasma Phys. **34**, 204 (1994).

[83] Y. L. Igitkhanov, A. S. Kukushkin, and A. M. Runov, Contrib. Plasma Phys. **34**, 216 (1994).

[84] Y. L. Igitkhanov and A. M. Runov, Contrib. Plasma Phys. **34**, 221 (1994).

[85] Y. L. Igitkhanov and A. M. Runov, Contrib. Plasma Phys. **34**, 204 (1994).

[86] A. V. Chankin, D. P. Coster, and G. Meisl, Contrib. Plasma Phys. **52**, 500 (2012).

[87] A. V. Chankin and D. P. Coster, Contrib. Plasma Phys. **54**, 493 (2014).

[88] A. V. Chankin and D. P. Coster, Journal of Nuclear Materials **463**, 498 (2015).

[89] G. Meisl, A. V. Chankin, and D. P. Coster, Journal of Nuclear Materials **438**, (2013).

[90] D. Coster *et al.*, Physica Scripta **2006**, 9 (2006).

[91] SOLPS user manual.

[92] D. P. Coster, Journal of Nuclear Materials **390**, 826 (2009).

[93] D. P. Coster, Contrib. Plasma Phys. **56**, 790 (2016).

[94] D. Bohm, *The Characteristics of Electrical Discharges in Magnetic Fields* (McGraw-Hill, New York, 1949).

[95] R. Chodura, Physics of Fluids **25**, 1628 (1982).

[96] J. A. Wesson, Plasma Phys. Control. Fusion **37**, 1459 (1995).

[97] P. C. Stangeby, Physics of Plasmas **2**, 702 (1995).

[98] P. C. Stangeby, Plasma Phys. Control. Fusion **37**, 1031 (1995).

[99] Y. L. Igitkhanov and A. M. Runov, Contributions to Plasma Physics **32**, 308 (1992).

[100] J. A. Tagle, P. C. Stangeby, and S. K. Erents, Plasma Phys. Control. Fusion **29**, 297 (1987).

[101] J. Horacek *et al.*, Journal of Nuclear Materials **313**, 931 (2003).

[102] D. Tskhakaya *et al.*, Journal of Nuclear Materials **415**, (2011).

[103] P. C. Stangeby, Journal of Nuclear Materials **128**, 969 (1984).

[104] M. A. Jaworski *et al.*, Fusion Engineering and Design **87**, 1711 (2012).

[105] J. G. Watkins *et al.*, Journal of Nuclear Materials **266**, .

[106] S. Marsen *et al.*, Journal of Nuclear Materials **438**, (2013).

[107] I. Duran *et al.*, Journal of Nuclear Materials **463**, 432 (2015).

[108] M. M. Shoucri and R. GagnÃ©, Journal of Computational Physics **27**, 315 (1978).

[109] O. V. Batishchev *et al.*, Journal of Plasma Physics **61**, 347 (1999).

[110] Z. Xiong *et al.*, Journal of Computational Physics **227**, 7192 (2008).

[111] G. Meisl, Master's thesis, Technische Universitaet Muenchen, Muenchen, 2011.

[112] S. E. Parker, R. J. Procassini, C. K. Birdsall, and B. I. Cohen, Journal of Computational Physics **104**, 41 (1993).

[113] A. Abdulle *et al.*, Acta Numerica **21**, 1 (2012).

[114] F. Wagner *et al.*, Physical Review Letters **49**, 1408 (1982).

[115] F. Wagner, Plasma Physics and Controlled Fusion **49**, (2007).

[116] V. A. Soukhanovskii, Plasma Physics and Controlled Fusion **59**, 064005 (2017).

[117] D. D. Ryutov and V. A. Soukhanovskii, Physics of Plasmas **22**, 110901 (2015).

[118] H. Y. Guo *et al.*, Nuclear Fusion **57**, 044001 (2017).