

# Verification and validation procedures with applications to plasma-edge turbulence simulations

THÈSE N° 7789 (2017)

PRÉSENTÉE LE 14 JUILLET 2017  
À LA FACULTÉ DES SCIENCES DE BASE  
SPC - THÉORIE  
PROGRAMME DOCTORAL EN PHYSIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

**Fabio RIVA**

acceptée sur proposition du jury:

Prof. V. Savona, président du jury  
Prof. P. Ricci, directeur de thèse  
Dr M. Greenwald, rapporteur  
Prof. V. Naulin, rapporteur  
Prof. A. Fasoli, rapporteur



ÉCOLE POLYTECHNIQUE  
FÉDÉRALE DE LAUSANNE

Suisse  
2017



Alla mia famiglia





# Acknowledgements

The work discussed in the present thesis is the result of a journey started four years ago, but my passion for science, and in particular for physics, starts well before the beginning of my Ph.D. During my doctoral studies, as well as during my entire studies, I was helped, supported, and inspired by a number of people that I would like to thank here. Without them, it would not be possible to be where I am, and for this I would like to express my sincere gratitude.

First of all, I would like to thank my thesis advisor Prof. Paolo Ricci, without whom this work would have not be possible. Thank you Paolo for your help, for your guidance, and your careful revision of all of my manuscripts. You always had time for discussions with me, and not only of subjects related to plasma physics. Thank you for having guided me through the process of becoming a scientist, sharing your knowledge and your expertise. Being a teaching assistant for Physique Générale II and Plasma Physics II helped me a lot in understanding and enjoying physics. Thank you also for all the brunches you organized at your home, it was always a great moment to meet outside the office.

I am very thankful to Christoph, with whom I shared the office for almost four years, and who was always available to discuss issues related to GBS and to answer questions about the submission of the thesis. I would like to thank also all the others current and former colleagues of the plasma-edge theory group: Sébastien and Joaquim, who introduced me to GBS and to the compilation of numerical codes on clusters, and helped me a lot at the beginning of my Ph.D.; Annamaria, who helped me understanding the linear code; Federico, for his help in running GBS simulations, and all the others, Paola, Carrie, and André. Thank you all, it was a great pleasure and honour to be part of the same group! Interacting with you was always stimulating and pleasant.

A special thanks goes to Ivo Furno, my Master thesis advisor, who helped and supported me during my Master project. Your energy is contagious! Thank you to all the other TORPEX group members, in particular Fabio Avino and Alexandre Bovet, who helped me a lot to carry out plasma discharges in TORPEX during my Master studies.

I am also very thankful to Stefano Coda, who organized for me the TCV discharges necessary to validate GBS simulations against experimental measurements, and to Cedric Tsui, who helped me in collecting and analysing the TCV experimental measurements.

## Acknowledgements

---

Thank you for your time and your help!

I would like to thank all the students whom I worked with, in particular Emmanuel, Lucio, and Carrie. Working with you was extremely stimulating. I learnt a lot discussing all our projects with you.

Thanks to all the Ph.D. students at CRPP and SPC. You make the SPC department a very pleasant place to work. Thank you for all the nice moments, and the beers, shared together. I would like to thank also all the CRPP and SPC members, who are always available to share their expertise and to provide stimulating discussions. It was a pleasure to work with you. I am also thankful to all the collaborators whom I worked with, Fulvio Militello, Nicola Vianello, Monica Spolaore, and all the others.

A big thanks goes to all my friends, here in Lausanne, in Ticino, and all around the world. Thank you for having supported me, in particular during my Ph.D., providing very pleasant discussions and enjoyable moments. A special thanks goes to my special friend Shaila, it is always a pleasure to discuss with you about physics, to make fun of engineers, to debate about feminism, and to discuss all other subjects, both the frivolous and the more serious ones. Thank you for tolerating me, supporting me, and having repeated me many times during the past year “yes, you’ll make it”.

Finally, and most importantly, I would like to thank my family. Thank you! Whatever I am and whatever I managed to do, I owe to you. I had the privilege to study what I wanted, being always supported and motivated. Grazie mamma, grazie papà. Grazie Luca, grazie Giulia. Grazie a tutta la mia famiglia!

F. R.

# Abstract

Understanding the plasma dynamics in tokamaks is of fundamental importance to reliably predict the performances of future fusion devices. Because of the complex phenomena at play, state-of-the-art simulation codes are needed to investigate the plasma dynamics. Consequently, errors affecting such numerical tools can have far reaching consequences. The present thesis focuses on developing and applying rigorous verification and validation (V&V) procedures to plasma turbulence simulations, ultimately improving our understanding of the plasma dynamics.

First, a rigorous code verification methodology for grid-based simulation codes is presented and used to assess the correct implementation of a physical model in a plasma simulation code. It consists of using the method of manufactured solutions and executing an order-of-accuracy test, assessing the rate of convergence of the numerical solution to the manufactured one. The methodology is then generalized to particle-in-cell (PIC) codes, accounting for numerical schemes intrinsically affected by statistical noise and providing a suitable measure of the distance between continuous, analytical distribution functions, and finite samples of computational particles. The proposed procedure is successfully applied to verify GBS, a simulation code used to study plasma turbulence in the tokamak scrape-off layer (SOL), and to verify a one-dimensional PIC code.

Even if the physical model is correctly implemented, simulations are always affected by numerical errors. A rigorous solution verification methodology for estimating the numerical statistical and discretization errors affecting plasma turbulence simulations is discussed in the present thesis. The estimate of the numerical statistical error, such as the one due to the finite number of particles used in PIC simulations, is based on repeating the simulation with different pseudorandom number generator seeds. For the discretization error, the Richardson extrapolation is used to provide an approximation of the exact solution. The solution verification methodology is then applied to quantify the numerical error affecting GBS and PIC simulation results.

A further source of uncertainty affecting the results of plasma turbulence simulations is given by the use of input parameters that are not precisely known or accurately measured. A methodology based on a decomposition of the model equation solution in terms of Chebyshev polynomials along the input parameter, time, and spatial coordinates is proposed. This methodology is then successfully employed to investigate the influence of input parameter variations on the results of a two-dimensional drift-reduced Braginskii model.

By carrying out a set of validation exercises, the final part of the present thesis is tar-

## Acknowledgements

---

geted to increase the reliability of our SOL modelling. First, seeded blob simulations carried out considering five different models are validated against measurements taken in the TORPEX basic plasma physics experiment. The comparison of simulations and experimental results sheds light on the dynamics of these structures that lead to large transport in the SOL. Furthermore, GBS simulations are validated against RFX-mod experimental measurements, providing interesting insights on the SOL plasma dynamics in this device. Finally, the impact of the shape of magnetic equilibrium on SOL turbulence is investigated through a rigorous validation of GBS simulations against TCV experimental measurements.

### **Keywords:**

plasma physics, code verification, solution verification, validation, controlled fusion, scrape-off layer, plasma turbulence, turbulent transport, fluid simulations, kinetic simulations.

# Résumé

Comprendre la dynamique des plasmas dans les tokamaks est d'importance cruciale pour pouvoir prédire les performances de futures expériences de fusion de façon fiable. En raison des phénomènes très complexes en jeu, des codes de simulation de pointe sont nécessaires pour étudier la dynamique des plasmas. Par conséquent, des erreurs affectant ces outils numériques peuvent avoir des conséquences énormes. Cette thèse est dédiée au développement et à l'application de procédures de vérification et de validation (V&V) rigoureuses pour des simulations de turbulence des plasmas, avec comme but final l'amélioration de notre compréhension de la dynamique des plasmas.

Tout d'abord, une méthodologie pour vérifier rigoureusement l'implémentation correcte d'un modèle physique dans un code de simulation des plasmas basé sur la méthode des différences finies est présentée. Cette méthodologie consiste à utiliser la méthode des *manufactured solutions* et à évaluer le taux de convergence de la solution numérique vers la solution exacte. La méthodologie est ensuite généralisée aux codes PIC (*particle-in-cell*), en tenant compte du fait que les schémas numériques sont intrinsèquement affectés par des incertitudes statistiques et en fournissant une mesure appropriée de la distance entre des fonctions de distribution continues et analytiques, et les particules de calcul utilisées pour les simulations. La procédure proposée est appliquée avec succès pour vérifier GBS, un code de simulation utilisé pour étudier la turbulence des plasmas dans la *scrape-off layer* (SOL) des tokamaks, et pour vérifier un code PIC unidimensionnel.

Même si le modèle physique est correctement implémenté, les simulations sont toujours affectées par des erreurs numériques. Une méthodologie rigoureuse pour estimer les erreurs numériques statistiques et de discrétisation affectant les simulations de turbulence des plasmas est discutée dans la présente thèse. L'estimation de l'erreur statistique, telle que celle due au nombre fini de particules utilisées dans les simulations PIC, repose sur la répétition de la simulation avec différentes graines aléatoires utilisés pour l'initialisation des générateurs de nombres pseudo-aléatoires. Pour l'erreur de discrétisation, l'extrapolation de Richardson est utilisée pour approximer la solution exacte. La méthodologie est ensuite appliquée pour quantifier l'erreur numérique affectant les résultats des simulations de GBS et des codes PIC.

Une autre source d'incertitude affectant les résultats des simulations de turbulence des plasmas est liée à l'utilisation de paramètres d'entrée qui ne sont pas connus ou mesurés avec précision. Une méthodologie basée sur une décomposition de la solution des équations en termes de polynômes de Tchebychev le long des paramètres d'entrée, du temps et des coordonnées spatiales est proposée. Cette méthodologie est ensuite appliquée avec

## Acknowledgements

---

succès pour étudier l'influence des variations des paramètres d'entrée sur les résultats d'un modèle *drift-reduced* de Braginskii bidimensionnel.

En effectuant un ensemble d'exercices de validation, la partie finale de la présente thèse vise à accroître la fiabilité de notre modélisation de la SOL. Tout d'abord, des simulations de *seeded blobs* effectuées en considérant cinq modèles différents sont validées avec des mesures prises dans l'expérience de physique des plasmas TORPEX. La comparaison des simulations avec des résultats expérimentaux permet de mieux comprendre le mouvement de ces structures, qui sont responsables d'une grande partie du transport observé dans la SOL. En outre, des simulations faites avec GBS sont validées avec des mesures expérimentales prises sur RFX-mod. Elles fournissent des informations intéressantes sur la dynamique du plasma dans la SOL de ce dispositif. Enfin, l'influence de la forme de l'équilibre magnétique sur la turbulence dans la SOL est étudiée grâce à une validation rigoureuse des simulations de GBS avec des mesures expérimentales de TCV.

**Mots clefs :**

**physique des plasmas, vérification, validation, fusion contrôlée, scrape-off layer, turbulence, transport turbulent, simulations fluides, simulations cinétiques.**

# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Abstract (English/Français)</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 The tokamak device . . . . .	4
1.2 Plasma modeling and numerical simulations . . . . .	6
1.3 Verification and validation (V&V) procedure . . . . .	8
1.4 Scope and outline of the present thesis . . . . .	10
<b>2 Code verification</b>	<b>13</b>
2.1 Rigorous verification of grid-based simulation codes . . . . .	14
2.1.1 Order of accuracy test . . . . .	14
2.1.2 The method of manufactured solutions . . . . .	15
2.2 Rigorous verification of PIC codes . . . . .	16
2.2.1 The PIC method . . . . .	17
2.2.2 Challenges in applying the MMS to PIC code verification . . . . .	18
2.2.3 Reducing the computational cost of a PIC verification . . . . .	21
2.3 Examples of application of the code verification methodology . . . . .	23
2.3.1 Application of the code verification methodology to GBS . . . . .	23
2.3.2 Application of the verification methodology to a PIC simulation code . . . . .	27
2.4 Conclusions . . . . .	31
<b>3 Solution verification</b>	<b>33</b>
3.1 Statistical error . . . . .	34
3.2 Discretization error . . . . .	35
3.3 Examples of application of the solution verification methodology . . . . .	37
3.3.1 Application of the solution verification methodology to GBS . . . . .	37
3.3.2 Application of the solution verification methodology to a PIC si- mulation . . . . .	40
3.4 Conclusions . . . . .	41

<b>4</b>	<b>Uncertainty propagation</b>	<b>43</b>
4.1	Chebyshev spectral method . . . . .	45
4.1.1	Weighted residual method . . . . .	45
4.1.2	Weighted residual method in Chebyshev spectral domain . . . . .	46
4.1.3	Operators in the Chebyshev spectral domain . . . . .	49
4.1.4	Initial condition . . . . .	50
4.1.5	Boundary conditions . . . . .	51
4.2	Uncertainty propagation through a plasma turbulence model . . . . .	54
4.2.1	Two-dimensional drift-reduced Braginskii equations . . . . .	54
4.2.2	Numerical implementation . . . . .	56
4.2.3	Simulation results . . . . .	57
4.2.4	Uncertainty propagation . . . . .	58
4.3	Conclusions . . . . .	61
<b>5</b>	<b>A multi-code validation : blob dynamics in TORPEX</b>	<b>63</b>
5.1	Experimental scenario . . . . .	64
5.2	The models and the simulation codes . . . . .	66
5.2.1	BOUT++3D . . . . .	66
5.2.2	BOUT++2D . . . . .	67
5.2.3	GBS . . . . .	68
5.2.4	HESEL . . . . .	69
5.2.5	TOKAM3X . . . . .	70
5.2.6	Summary of analogies and differences among the physical models	71
5.3	Seeded blob simulations . . . . .	73
5.4	Sensitivity studies . . . . .	77
5.5	Analysis and validation of the simulation results . . . . .	80
5.6	Conclusions . . . . .	85
<b>6</b>	<b>A validation in a tokamak : turbulence in the RFX-mod SOL</b>	<b>87</b>
6.1	The experimental setup . . . . .	88
6.2	GBS simulations of the RFX-mod SOL . . . . .	89
6.3	Identification of the instability driving the SOL transport . . . . .	91
6.3.1	Nonlinear simulations . . . . .	92
6.3.2	Linear instabilities . . . . .	94
6.4	Validation of the GBS simulations against experimental measurements .	96
6.5	Conclusions . . . . .	102
<b>7</b>	<b>A rigorous validation : plasma shaping effects in TCV</b>	<b>105</b>
7.1	The model . . . . .	106
7.1.1	Fluid moment equations . . . . .	106
7.1.2	Coordinate systems and differential operators . . . . .	107
7.1.3	Magnetic equilibrium . . . . .	108
7.2	Linear instabilities . . . . .	109
7.2.1	Resistive ballooning modes . . . . .	111



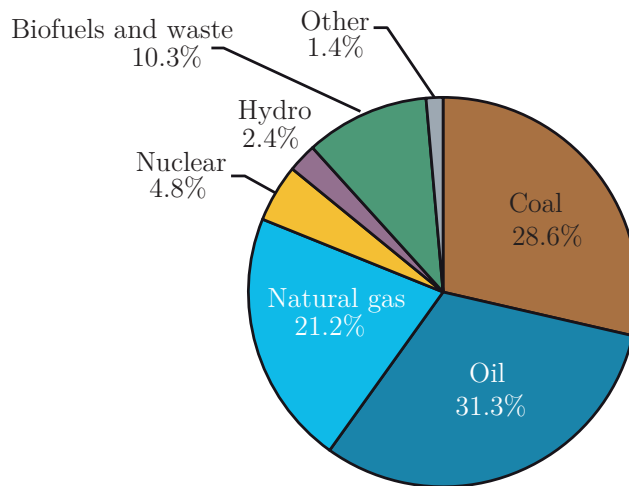
7.2.2	Resistive drift waves . . . . .	112
7.3	Estimate of the pressure gradient length . . . . .	113
7.4	Nonlinear simulations . . . . .	117
7.4.1	Implementation and numerics . . . . .	117
7.4.2	Simulation results . . . . .	117
7.5	Discussion of the theoretical results . . . . .	122
7.6	Rigorous validation against TCV measurements . . . . .	123
7.6.1	Validation methodology . . . . .	124
7.6.2	Experimental setup . . . . .	126
7.6.3	GBS simulations and validation results . . . . .	129
7.7	Conclusions . . . . .	133
<b>8</b>	<b>Conclusions and outlook</b>	<b>135</b>
<b>A</b>	<b>The SOL model and the GBS code</b>	<b>139</b>
A.1	Braginskii equations . . . . .	140
A.2	Drift-reduced Braginskii equations . . . . .	143
A.2.1	The semi-electrostatic limit . . . . .	145
A.2.2	Continuity and vorticity equations . . . . .	146
A.2.3	Generalized Ohm's law and parallel momentum equation . . . . .	147
A.2.4	Temperature equations . . . . .	148
A.3	The GBS code . . . . .	149
A.3.1	The GBS differential operators . . . . .	149
A.3.2	The GBS model . . . . .	150
A.3.3	Boundary conditions . . . . .	151
A.3.4	Geometry . . . . .	152
A.3.5	Numerics . . . . .	153
<b>B</b>	<b>Estimate of the equilibrium pressure gradient length</b>	<b>155</b>
<b>C</b>	<b>Differential operators for an axisymmetric magnetic field</b>	<b>157</b>
<b>D</b>	<b>Derivation of a shaped magnetic equilibrium</b>	<b>161</b>
<b>E</b>	<b>Curvature operator in the <math>\epsilon = 0</math> limit</b>	<b>163</b>
	<b>Bibliography</b>	<b>165</b>
	<b>Curriculum Vitae</b>	<b>179</b>



# CHAPTER 1

## Introduction

Because of the continuous growth of world population and energy per capita consumption, the overall energy production increased by more than a factor 20 in the past two centuries [1]. This allowed many people to enjoy an unprecedented level of mobility and comfort. However, about a quarter of the world population still lives today without electricity [2]. Providing the energy necessary to support the development of emerging



**Figure 1.1** – World total energy supply by source in 2014. Data taken from the International Energy Agency.

economies constitutes an enormous challenge for humankind. In fact, today's energy production is essentially based on fossil fuels such as oil, gas or coal (see Fig. 1.1). The reserves of these resources will inevitably deplete and are unequally distributed on Earth, giving rise to conflicts and social uncertainty. Even more important, the combustion of fossil fuels releases an enormous amount of greenhouse gases, ultimately leading to climate changes and global warming. To overcome these issues and ensure the entire world

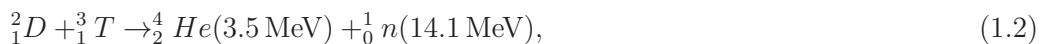
population an equitable access to low-price energy, there is a strong motivation in developing alternative sources of energy that are sustainable and abundant.

Inspired by nature and looking at the primary energy source of our planet, the Sun, researchers all around the world are investigating the possibility of using nuclear fusion reactions, continuously occurring in stars, as an alternative source of energy. Nuclear fusion is the process that allows two nuclei to fuse together, generating all the known elements heavier than hydrogen. If the total mass of the elements produced by the fusion reaction is smaller than the sum of the reactant masses, the produced nuclei have a higher binding energy per nucleon than the reactants, and the reaction is exothermic. In fact, according to Einstein's relation, the energy  $E = \Delta mc^2$  is released during the nuclear reaction, where

$$\Delta m = \sum_r m_r - \sum_p m_p, \quad (1.1)$$

with  $m_r$  and  $m_p$  the masses of the reactants and of the products. Because of the low probability of most of the fusion reactions to take place, among all the possible exothermic reactions only few are of interest for a fusion power plant.

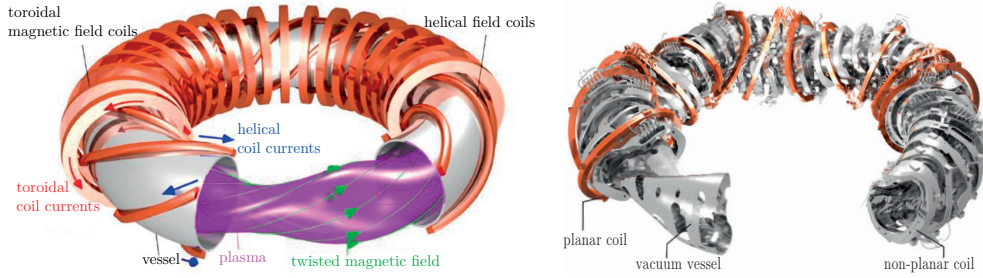
The most suitable nuclear reaction to be exploited for energy production is the fusion of a nucleus of deuterium ( $D$ ) with one of tritium ( $T$ ). This reaction, described by the following equation



requires a temperature of the reactants larger than 1 keV to occur. This temperature is necessary for deuterium and tritium nuclei to overcome Coulomb's repulsion and allow the strong nuclear force to fuse the nuclei together with a non-vanishing probability. The reaction in Eq. (1.2) releases approximately 350 GJ of energy per gram of fuel. Deuterium is abundant in oceans and can be easily extracted from water. On the other hand, tritium is a rare resource and should be produced by making the neutrons collide with lithium in the blanket of a fusion reactor.

Energies larger than 1 keV are well beyond the ionization energy. Therefore, the  $D - T$  reaction occurs when matter is completely ionized and in the plasma state. To be of interest for energy production, the plasma should be confined sufficiently well, so that the energy released by fusion reactions can heat the fresh fuel and keep the reactions going. In other words, the power lost from a fusion reactor,  $W/\tau_E$ , where  $W$  is the energy density in the plasma and  $\tau_E$  the energy confinement time, should be compensated by the power generated by fusion reactions. This results in a constraint on the triple product  $n_e T_i \tau_E$ , with  $n_e$  the electron plasma density and  $T_i$  the ion temperature. For  $D - T$  fusion reactions and  $T_i \gtrsim 14 \text{ keV}$ , it results that self-sustaining fusion reactions occur for  $n_e T_i \tau_E \gtrsim 3 \cdot 10^{21} \text{ keV s m}^{-3}$ , a condition derived by Lawson in 1957 [3].

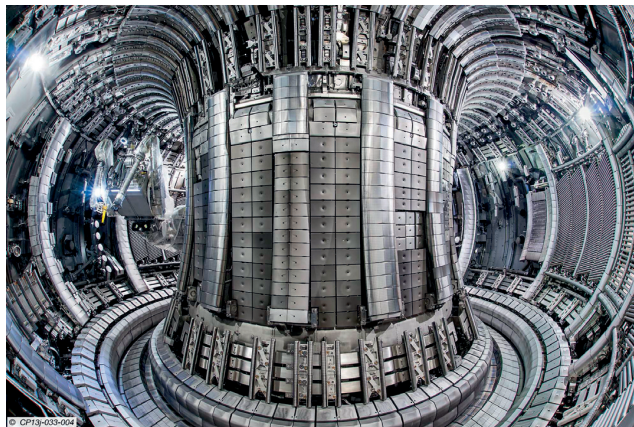
To satisfy Lawson's criterion, two main strategies are currently investigated, based on an inertial and a magnetic approach to the plasma confinement. The goal of the inertial confinement fusion (ICF) research is to obtain very high plasma densities ( $n_e \sim 10^{30} \text{ m}^{-3}$ ) by compressing pellets of fuel with high-energy lasers, electron or ion beams. The high



**Figure 1.2** – Left: Schematic representation of a stellarator. Right: Magnetic field coils of Wendelstein 7-X. Image source: Eurofusion website.

density allows to confine the plasma for a very short time, of the order  $\tau_E \sim 10^{-9}$  s. The largest ICF experiment currently operational is the National Ignition Facility (NIF) in Livermore, USA [4]. On the other hand, magnetic confinement fusion (MCF) research is targeted to develop a fusion reactor working at much lower densities ( $n_e \sim 10^{19} \text{ m}^{-3}$ ). This implies that much longer confinement times should be achieved ( $\tau_E \gtrsim 1$  s).

The stellarator and the tokamak are the two most promising concepts of MCF reactors, and are both based on confining the plasma inside a toroidal vacuum vessel by using twisted magnetic field lines. Stellarators use the magnetic field created by a complex set of coils to confine the burning plasma (see Fig. 1.2). The largest operational stellarator in the world, whose construction was recently completed in Greifswald, Germany, is Wendelstein 7-X. Its first Helium plasma was successfully produced at the end of 2015, showing encouraging preliminary results [5]. On the other hand, tokamaks generate twisted magnetic field lines by exploiting the high plasma conductivity and inducing a toroidal plasma current like a transformer. The largest tokamak in operation is JET, an European experiment located in Culham, UK (see Fig. 1.3). It is the fusion device that



**Figure 1.3** – Interior of JET, the largest tokamak operating in the world. Image source: Eurofusion website.

obtained the highest amplification factor  $Q$ , with  $Q$  the ratio between the fusion power generated in the reactor and the external heating power, reaching  $Q \simeq 0.7$  and a triple product  $n_e T_i \tau_E \simeq 8 \cdot 10^{20} \text{ keV s m}^{-3}$  [6]. The target of the ITER tokamak (*the way* in Latin), which is under construction in Cadarache, France, is to significantly improve the

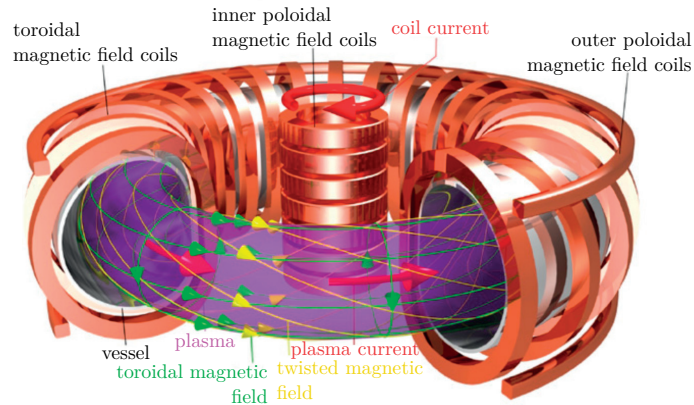
energy amplification and perform plasma discharges with  $Q = 10$  [7]. ITER operation is expected to start in 2026, and, if successful, it will “demonstrate the scientific and technological feasibility of fusion power for peaceful purposes”.

The tokamak is currently the most advanced device to exploit fusion as an energy source, and it is described in detail in Section 1.1. Then, the remainder of the present chapter is organized as follows. Because of the complex interactions between charged particles, solid surfaces, and electromagnetic fields, understanding and modelling the plasma dynamics in tokamaks is an extremely challenging task. As a result, state-of-the-art numerical codes are needed to investigate the physics at play. The discussion of these tools is the subject of Section 1.2. The procedure used to assess the reliability of plasma simulations is summarized in Section 1.3. Finally, the main contributions provided by the present thesis to the plasma physics community are illustrated in Section 1.4.

### 1.1 The tokamak device

A toroidal magnetic field can eliminate the losses of plasma charged particles due to their free streaming along the magnetic field lines. However, since particle drifts are induced by curvature and gradients of the magnetic field lines, a purely toroidal magnetic field is not sufficient to confine a plasma [8]. A poloidal magnetic field should be superimposed to the toroidal one and used to average out these drifts. In a tokamak, the toroidal field is produced by a set of external coils located around the vacuum vessel, whereas the poloidal field is generated by inducing a current in the plasma. This current, which also heats the plasma because of Ohmic dissipation, is induced by the action of a central solenoid that works as the primary circuit of a transformer. By varying the current in the solenoid, an electromotive force is applied to the plasma and, because of its high conductivity, a strong toroidal plasma current is induced, which in turn generates the desired poloidal field. The sum of the toroidal and the poloidal magnetic fields engenders twisted magnetic field lines that confine the hot plasma. The configuration thus generated results in magnetic field lines that wind around the torus, defining toroidally nested magnetic surfaces, called flux surfaces. A schematic representation of the tokamak concept is presented in Fig. 1.4.

Even if the magnetic field lines created in a tokamak can confine single charged particles, their collective response makes the plasma confinement very challenging. First, for the plasma to be macroscopically stable, plasma currents, densities, and plasma to magnetic pressure ratios cannot exceed certain values. In fact, because of the presence of curvature and gradients of the magnetic field, pressure gradients, and current densities, violent macroscopic events such as disruptions can occur if these limits are exceeded. These events should be avoided in a fusion reactor, as they might severely damage the device. Second, plasma particles can collide against each other by interacting through Coulomb’s force, giving rise to radial diffusion (the so-called collisional transport) that degrades the plasma confinement. Finally, because of the presence of

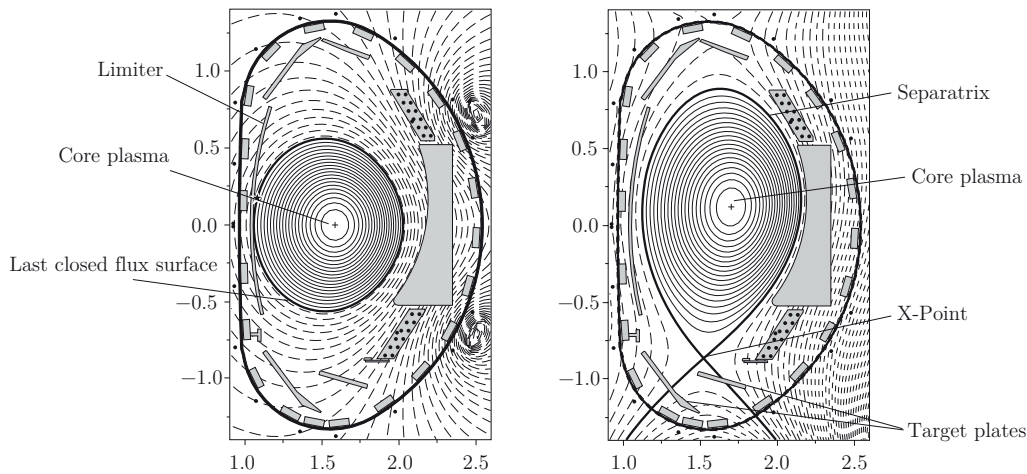


**Figure 1.4** – Schematic representation of a tokamak. The inner poloidal magnetic field coils, at the center of the device, are used to induce the plasma current and generate the poloidal magnetic field. The toroidal magnetic field coils generate the toroidal magnetic field. The sum of the two magnetic field components results in twisted magnetic field lines (in yellow), used to confine the plasma. Outer poloidal magnetic field coils are used to shape the plasma and control its stability. Image source: Eurofusion website.

free energy sources such as plasma pressure gradients and magnetic curvature, instabilities arise in tokamaks and develop into turbulent transport, the so-called anomalous transport, ultimately leading to radial particle and heat flows. It has been observed experimentally that anomalous transport exceeds by many orders of magnitude the expected collisional transport in tokamaks [9]. The physical mechanism that governs the anomalous transport is not entirely understood yet, undermining our capabilities to predict the performances of future fusion devices.

Because of turbulence, plasma is transported across the magnetic flux surfaces and eventually reaches the walls of the device. This leads to undesired particle and heat flows to the vacuum vessel, which can be damaged. To control the location where the particles and heat are deposited and, possibly, mitigate their effect, two different strategies are adopted. First, one can insert a solid rail in the vessel, the so-called limiter, or push the flux surfaces against a particular location of the vessel wall (the inboard wall in the case of Fig. 1.5, left). This defines a separation between the *closed flux surface region*, where the magnetic field lines wrap around the nested flux surfaces and the plasma is well confined (core plasma, indicated by solid lines in Fig. 1.5, left), and the region where magnetic field lines intercept a solid surface, the so-called open field line region (indicated by dashed lines in Fig. 1.5, left). The surface that defines the separation between these two regions is called the last close flux surface (LCFS). By flowing much more rapidly along the field lines than perpendicularly to them, the plasma is lost and the heat is exhausted at the limiter surfaces or in proximity of the contact point between the LCFS and the vessel, preventing therefore the plasma to reach the rest of the vessel walls. This is why the open field line region is referred to as the scrape-off layer (SOL). On the other hand, one can use a set of external coils to produce a magnetic topology such that the core plasma is kept away from the walls of the device, thus directing the particles and heat outflowing from the core to plates designed to sustain high heat fluxes.





**Figure 1.5** – Poloidal cross section of a tokamak for two different magnetic configurations. Left: Limiter configuration, the plasma touches the limiter and the contact point defines the LCFS. Right: Diverted configuration, the hot plasma core does not enter in direct contact with the wall and the magnetic field is diverted so that particles and heat flows are directed to the target plates. Source: Ref. [10].

This *diverted* configuration is represented in Fig. 1.5, right.

Plasma properties are extremely different in the closed flux surface region and in the SOL. In the confined region the plasma is well confined at high density and temperature, and micro-turbulence leads to density and temperature fluctuations usually small compared to the equilibrium profiles ( $\simeq 1\%$ ). On the other hand, the SOL plasma dynamics is determined by the interplay between the plasma outflowing from the core, the cross-field turbulent transport, and parallel flows along the magnetic field lines. In this region fluctuations are large, with amplitude comparable to the background values. A multiphysics approach is required to address the SOL plasma dynamics, since it results from the interaction between charged particles, neutrals, and solid surfaces. Despite being just a few centimeters thick, the SOL plays a crucial role in determining the performance of the entire fusion device, as it sets the boundary conditions for the core, it controls the impurity dynamics and the recycling level, and it is responsible of exhausting the tokamak power, thus determining the heat load at the vessel [11].

## 1.2 Plasma modeling and numerical simulations

Understanding the turbulent plasma dynamics in a tokamak is an extremely challenging task, although necessary to address some of the most crucial issues that the fusion program is facing today. The plasma charged particles are subject to the action of the Lorentz force and, at the same time, they are sources of electromagnetic fields, leading to complex nonlinear interactions. Moreover, plasma turbulence phenomena involve an extremely wide range of spatial and temporal scales, from the electron-orbit Larmor



---

## 1.2. Plasma modeling and numerical simulations

---

motion ( $\sim 10^{-11}$  s,  $\sim 10^{-5}$  m) to the macroscopic scales of a discharge ( $\sim 1$  s,  $\sim 1$  m) [12]. The study of plasmas in MCF devices is further compounded by the complex magnetic geometries involved. As a result, the models developed to describe the plasma dynamics are in general extremely difficult to solve analytically and state-of-the-art simulations need to be used to investigate the physics at play.

Because of the rich variety of plasma phenomena present in tokamaks and due to their extremely complex behaviour, a hierarchy of models was developed in the past to describe the plasma physics. The most direct description consists in solving the equations of motion of each particle together with the Maxwell equations. Although conceptually simple, such a model is computationally too demanding due to the large number of particles involved. To overcome this issue, a statistical approach based on Liouville's theorem was developed, leading to the Boltzmann equation, which is written as

$$\frac{\partial f_\alpha}{\partial t} + \mathbf{v} \cdot \frac{\partial f_\alpha}{\partial \mathbf{x}} + \frac{q_\alpha}{m_\alpha} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f_\alpha}{\partial \mathbf{v}} = \sum_\beta C(f_\alpha, f_\beta). \quad (1.3)$$

Equation (1.3) describes the evolution of the distribution functions of the  $\alpha$  species,  $f_\alpha$ , under the action of the electromagnetic fields and Coulomb's collisions, where  $\mathbf{x}$  and  $\mathbf{v}$  are the spatial and velocity phase-space coordinates,  $q_\alpha$  and  $m_\alpha$  the particle charge and mass,  $\mathbf{E}$  and  $\mathbf{B}$  the electric and magnetic fields, solution of the Maxwell equations, and  $C(f_\alpha, f_\beta)$  is an operator modelling Coulomb's collisions between  $\alpha$  and  $\beta$  plasma species. In the tokamak core, collisions are typically neglected due to the local high temperature, and it is not justified to assume that distribution functions are close to a Maxwellian. A full kinetic description of the plasma is therefore necessary. To decrease the computational cost of kinetic models, Eq. (1.3) is usually averaged over the fast-particle gyrorational motion, removing the fast cyclotron timescales from the system and reducing from six to five the dimensions of the phase-space, thus obtaining the so-called gyrokinetic model. Moreover, since the density and temperature fluctuations are generally much smaller than the background values, a separation between equilibrium profiles and fluctuations is often performed. Different numerical approaches were developed in the past targeted to solve the gyrokinetic equation, based both on particle-in-cell (PIC) algorithms, where a number of computational particles is evolved according to the equations of motion, and gyrokinetic Eulerian simulation codes, which make use of grid-based algorithms.

Unlike the core, the collisionality in the SOL is typically high enough to assume that distribution functions are close to Maxwellian. In this case, a fluid model that evolves a few moments of the distribution function is usually sufficient to describe the plasma dynamics. On the other hand, transport is highly intermittent in the SOL, and a separation between background profiles and fluctuations is not justified. The modeling of SOL dynamics is usually based on a set of two-fluid equations, such as the ones derived by Braginskii in 1965 [13]. These equations, which are often simplified thanks to the drift approximation [14], were implemented in a number of simulation codes that are currently used to investigate the SOL turbulence dynamics (see e.g. Refs. [15–20]). These simulation codes are providing insights into some of the fundamental physical mechanisms at play in the SOL, such as the dynamics of intermittent transport events [21], the value

of the electrostatic potential [22], the origin of the spontaneous SOL toroidal plasma rotation [23], or the equilibrium pressure gradient length [24].

### 1.3 Verification and validation (V&V) procedure

Errors affecting simulations of the plasma dynamics in a tokamak can have far reaching consequences, particularly at this stage of the fusion program, when we approach the ITER era and the conceptual design of DEMO. The reliability of numerical simulations is assessed by the verification and validation (V&V) procedure [25–27]. V&V is composed of three separated tasks: the *code verification* process, which is a mathematical issue targeted to assess that the physical model is correctly implemented in a numerical code; the *solution verification* procedure, used to estimate the numerical error affecting the results of a simulation; and the *validation*, used to assess the consistency of the code results, and therefore of the physical model, with experimental measurements. A schematic representation of the V&V procedure is shown in Fig. 1.6.

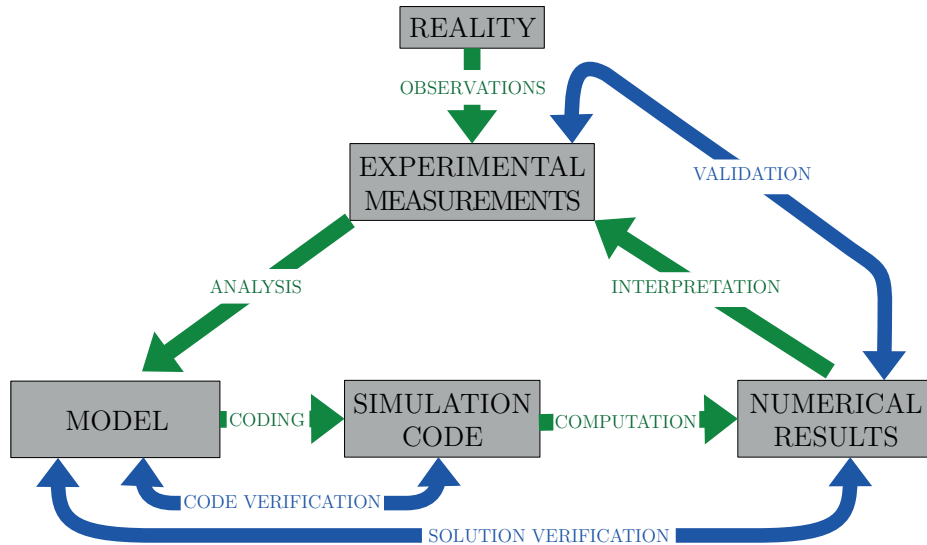
In plasma physics, code verification is usually based on performing code-to-code comparisons (see e.g. Refs. [28–32]). Simulations of very simple problems, such as the Landau damping of electron-plasma waves, for which an analytical solution is known, are also used. While valuable, these tests do not allow to rigorously verify the correct implementation of a model in a simulation code, since, for example, it is generally impossible to understand if differences between the numerical results of the two codes are due to the finite resolution of the grid used for the discretization or to errors in the implementation of the codes. On the other hand, to perform more rigorous verification procedures, an analytical solution is needed, but this does not generally exist. To overcome these issues, a systematic approach has been developed by the computational fluid dynamics community [26]. However, its use in plasma physics remained limited to single routines (see e.g. Refs. [33,34]).

The correct implementation of the model equations in the simulation code does not imply that numerical results are free from numerical errors. Sources of numerical errors are: rounding off, finite statistical sampling (e.g. using a finite number of particles randomly distributed to represent a distribution function), termination of an iterative scheme with a non-vanishing residue, and finite grid resolution [26]. All these have to be estimated in order to provide the numerical error affecting the simulation results. In plasma physics, numerical errors are usually quantified by performing grid-refinement-based analysis.

Furthermore, in order to perform a rigorous validation of the code results and to assess the reliability of the code predictions, it is necessary to estimate the uncertainties that affect simulation results due to the use of input parameters that are not precisely known or accurately measured. In plasma physics, these uncertainties are typically investigated by performing sensitivity scans.

Once the simulation code is rigorously verified and the uncertainties affecting the numerical results are correctly quantified, the simulations can be compared with experimental

### 1.3. Verification and validation (V&V) procedure



**Figure 1.6** – Schematic representation of the V&V procedure. By performing observations, experimental measurements are collected. Through their analysis, analytical models are derived and used to describe the dynamics of a physical system. The model equations are then discretized and implemented in a numerical code that is used to carry out simulations of the physical system. The numerical results are finally used to interpret the experimental measurements and understand the physics at play. Code verification procedures are used to assess if the algorithm is correctly implemented in the simulation code. The numerical error affecting simulation results is estimated by performing a solution verification. Finally, validation is used to assess the agreement between simulations and experimental measurements.

measurements, ultimately to assess the maturity of the considered physical model. An increasing validation effort is being carried out by the plasma physics community. A number of validation exercises have been performed in the recent past, for example focusing on gyrokinetic simulations of turbulence in the tokamak core (see e.g. Refs. [35–38]). Moreover, guidelines to perform a rigorous validation were ported to the plasma physics community from other domains, as described in Refs. [39, 40]. From a practical point of view, performing a rigorous validation requires to (i) identify the common quantities to compare between experiments and simulations, denoted as validation observables; (ii) organize the observables in a hierarchy based on the assumptions required for their evaluation; (iii) estimate the difference between experimental measurements and simulations for each individual observable; and (iv) quantify the agreement between numerical results and experiments through the use of an appropriate metric. Main goals of a validation procedure are: (i) compare different models to understand which one is better in describing a physical system; (ii) assess the maturity of a physical model in order to make reliable predictions; and (iii) make progress in our physics understanding by pointing out which are the key physics elements of a model. A review of validation in plasma physics was recently published [41].

## 1.4 Scope and outline of the present thesis

The goal of the present thesis is to develop a rigorous V&V procedure, suitable for numerical codes used for the simulation of the plasma dynamics, and apply it, ultimately, to improve our understanding of plasma turbulence in the tokamak SOL. More precisely, this thesis provides a rigorous code and solution verification procedure, as well as some examples of its application, both considering (i) GBS, a three-dimensional fluid code that is used to simulate plasma turbulence in basic plasma physics experiments and in the SOL of tokamaks; and (ii) a one-dimensional, electrostatic, collisionless PIC code, used to study the dynamics of the plasma sheath. A rigorous methodology to assess how model results are affected by input parameter variations is also presented. Finally, to improve our understanding of some of the key physics elements governing the SOL plasma dynamics, a validation procedure is applied to GBS and other plasma turbulence codes for different experimental scenarios.

This thesis is structured as follows. Bridging the gap between plasma physics and other scientific domains, in particular the computational fluid dynamics community, a general, rigorous, and simple-to-apply code verification procedure for grid-based simulation codes is presented in Chapter 2. The proposed code verification procedure consists in using the method of manufactured solutions [27, 42–44] and executing an order-of-accuracy test, assessing the rate of convergence of the numerical solution to the manufactured one. This methodology is then generalized to PIC codes. Finally, the proposed methodology is used to verify GBS and a one-dimensional, electrostatic, collisionless PIC code.

In Chapter 3 a rigorous solution verification methodology is proposed, which allows to quantify both statistical and discretization uncertainties affecting a simulation. The statistical uncertainty affecting a PIC simulation is estimated by repeating the simulation with different pseudorandom number generator seeds. For the discretization uncertainty, the Richardson extrapolation [45, 46] is used to provide an approximation of the analytical solution and the grid convergence index (GCI) [47] is used as an estimate of the relative discretization uncertainty. The procedure is applied to evaluate the numerical error affecting the simulations carried out with the codes verified in Chapter 2.

The study of the propagation of input parameter uncertainties through a simulation model is the subject of Chapter 4. To investigate the dependence of the model results on input parameter variations, the use of the weighted residual method (WRM) in the Chebyshev spectral domain is proposed. In particular, the model equation solution is decomposed in terms of Chebyshev polynomials along the parameter, time, and spatial coordinates, providing an approximated semi-analytical solution of the problem with explicit dependence on input parameters. The proposed methodology is then applied to a two-dimensional drift-reduced Braginskii model to assess the influence of input parameter variations on the model results.

The dynamics of blobs is investigated through a multi-code validation in Chapter 5. Blobs, also known as filaments, are structures with an excess of density and temperature relative to the surrounding plasma, which substantially contribute to the observed turbu-

lent transport in the tokamak SOL. Three-dimensional and two-dimensional seeded blob simulations are performed with five different fluid models, all based on the drift-reduced Braginskii equations, and the numerical results are compared among themselves and validated against experimental measurements provided by the TORPEX device [48,49]. The multi-code validation allows to point out the key elements at play in determining the blob motion, providing some interesting insights on the TORPEX plasma dynamics. In Chapter 6 the SOL plasma dynamics of the RFX-mod experiment [50] is investigated. Considering the experimental parameters of two inner-wall limited RFX-mod plasma discharges, the instability that drives most of the SOL turbulence transport in the RFX-mod device is identified using linear theory and nonlinear GBS simulations. Then, the numerical results are compared with experimental measurements, assessing the reliability of the GBS model in describing the RFX-mod SOL plasma dynamics.

The study of magnetic equilibrium effects on SOL turbulence transport is the subject of Chapter 7. First, an analytical model used to describe non-circular magnetic geometries is implemented in GBS. This model is then used to investigate theoretically the impact of the equilibrium shape on SOL linear instabilities. An analytical scaling for the equilibrium pressure gradient length in non-circular geometries is also derived. Finally, the GBS model is rigorously validated against TCV experimental measurements considering plasma discharges with different magnetic geometries.

The results discussed in the present thesis are finally summarized in Chapter 8, offering possible future perspectives for the development and the application of V&V procedures to plasma turbulence simulation codes. The discussion of the drift-Braginskii model and of the GBS code, the presentation of the gradient removal mechanism and its use in evaluating the equilibrium pressure gradient length, the computation of the metric coefficients used in evaluating GBS differential operators, the derivation of the magnetic equilibrium implemented in GBS, and the analytical derivation of the curvature operator used in Chapter 7, are illustrated in Appendices A, B, C, D, and E, respectively.



# CHAPTER 2

## Code verification

Code verification is the procedure used to assess the correct implementation of a physical model in a numerical simulation code. It can be carried out by [26]: (a) performing simple tests (e.g. energy conservation tests), (b) comparing simulation results with results from other codes (also known as code-to-code benchmark), (c) quantifying the numerical error with respect to an exact solution, (d) testing the convergence of the numerical solution to an exact solution, and (e) comparing the rate of convergence of the numerical solution to an exact solution with the expected order of the numerical scheme (order-of-accuracy tests). As the first two procedures, (a) and (b), do not require an exact solution of the model equations, they are simpler to perform. Indeed, code-to-code comparison is routinely performed to verify numerical codes used in plasma physics, including particle-in-cell (PIC) simulation codes [28, 30–32, 51, 52]. While valuable, this test does not allow to rigorously verify the considered codes. In fact, a fully verified code of reference implementing the same mathematical model is needed to use this method [25, 53] and, generally, it is very difficult to understand if a difference in the code results is due to discretization errors or to an incorrect implementation of the model. Moreover, performing a benchmark between two simulation codes can be tedious due to different choices in normalization, coordinates, etc. On the other hand, the last three procedures, (c)-(e), are more rigorous, but they require an exact solution of the model. While rigorous code verification procedures have been used in plasma physics [33, 34], their use remained limited to single routines, without approaching the full complexity of a simulation code. In the present chapter we focus on the order-of-accuracy tests, as those are the only tests able to ensure both the correct coding of the model equations and the correct implementation of the chosen numerical scheme [26]. Since an analytical solution is not available for most of the physical models used in plasma physics research, we employ a systematic approach to overcome this issue, that is the method of manufactured solutions (MMS) [27, 42–44]. This approach has been developed by the computational fluid dynamics community, with the idea of reversing the considered problem: instead of searching for the analytical solution of the model, we impose a manufactured solution

and we modify the equations by adding analytical terms with the goal of accommodating the manufactured solution.

The goal of the present chapter is threefold. First, in Section 2.1 we present a rigorous code verification methodology, developed in particular by the computational fluid dynamics community and based on the MMS, for the verification of grid-based simulation codes, bridging the gap between our community and other scientific domains, where considerable experience was developed in the last years on this subject. Then, in Section 2.2 we generalize this methodology for the verification of PIC codes, accounting for numerical schemes intrinsically affected by statistical noise and providing a measure of the distance between continuous, analytical distribution functions, and finite samples of computational particles. Finally, in Section 2.3 we apply these procedures to rigorously verify (i) a plasma turbulence code, namely GBS [20], used to simulate plasma turbulence in the tokamak scrape-off layer (SOL) and in basic plasma devices (see Appendix A for a detailed description of GBS), and (ii) a one-dimensional, electrostatic, collisionless PIC code, used to study the sheath plasma dynamics. The results discussed in the present chapter are published in Refs. [54, 55]. We remark that, following our work, other plasma turbulence simulation codes were rigorously verified, as reported for example in Refs. [56, 57].

## 2.1 Rigorous verification of grid-based simulation codes

In this section we present a rigorous code verification methodology for grid-based simulation codes. This methodology is based on performing an order-of-accuracy test. This test analyses the convergence of the numerical solution to a known analytical solution, also verifying that the discretization errors reduce at the rate expected for the numerical scheme, as the spatial mesh and the time step are refined. Since an exact solution of the model is needed to systematically evaluate the discretization error affecting the simulation results, the order-of-accuracy test is performed using the MMS approach, as detailed in the following.

### 2.1.1 Order of accuracy test

Formally, an order of accuracy test for a simulation code can be stated as follows. Given a theoretical model  $M$  with an analytical solution  $s$ , such that  $M(s) = 0$ , and the numerically discretized model of  $M$ ,  $M_h$ , with a numerical solution  $s_h$  that satisfies  $M_h(s_h) = 0$  ( $h$  is a parameter representing the degree of refinement of the mesh), the error affecting the numerical results is expressed as  $\epsilon_h = \|s_h - s\|$ , where  $\|\cdot\|$  denotes a designed norm. The theoretical order of accuracy,  $p$ , associated with the numerically discretized operator  $M_h$ , represents the rate at which the numerical solution  $s_h$  converges to the analytical solution  $s$  as the mesh is refined. The numerical error, in fact, satisfies the



---

## 2.1. Rigorous verification of grid-based simulation codes

relation  $\epsilon_h = C_p h^p + O(h^{p+1})$ , where  $C_p$  is independent of  $h$ , and  $p$  is the order of accuracy of the numerical scheme, typically evaluated through its Taylor expansion [26, 27, 44]. Having the two numerical solutions of  $M_h$  and  $M_{rh}$ , i.e.  $s_h$  and  $s_{rh}$ , where  $rh$  indicates coarsening the  $h$  mesh by a factor  $r$ , one can evaluate an observed order of accuracy,  $\hat{p}$ , using

$$\hat{p} = \frac{\ln(\epsilon_{rh}/\epsilon_h)}{\ln(r)}. \quad (2.1)$$

If  $\hat{p}$  converges to  $p$  for  $h \rightarrow 0$ , i.e. when the discretization error is dominated by the lowest order term in the expansion (the so-called asymptotic regime), we can state that the code is verified and the equations are correctly solved, with the order of accuracy expected for the numerical scheme.

### 2.1.2 The method of manufactured solutions

The evaluation of the numerical error  $\epsilon_h$ , necessary to obtain  $\hat{p}$ , requires that  $s$  is a known function. Unfortunately,  $s$  is unknown in most cases, in particular for complex plasma models. The MMS was developed to overcome this issue, and approaches the problem as follows [27, 42–44]: instead of solving  $M$  analytically, an arbitrary function  $s_M$  is imposed as a solution to the model (the so-called manufactured solution), and the model equations are modified to accommodate the imposed solution; the modified model is then solved numerically to compute the numerical error. More precisely, for a given model  $M$ , we choose an analytical function  $s_M$  and compute a source term,  $S = M(s_M)$ , which is subsequently subtracted from  $M$  to obtain a new analytical model  $G = M - S$ . The analytical solution of  $G$  is  $s_M$ :  $G(s_M) = M(s_M) - S = 0$ . It is then straightforward to compute the discretization of  $G$ ,  $G_h = M_h - S$ , which can be solved numerically to obtain  $s_{M,h}$ . Since the source term  $S$  is computed analytically, we do not add any new source of numerical error to the original numerical model, and the numerical error  $\epsilon_h = \|s_M - s_{M,h}\|$  satisfies

$$\epsilon_h = C' h^p + \mathcal{O}(h^{p+1}), \quad (2.2)$$

where  $C'$  is a constant independent of  $h$ . By showing that  $\hat{p} \rightarrow p$  for  $h \rightarrow 0$ , one verifies the simulation code.

To conclude the description of the MMS, we note that the initial condition and the boundary conditions have to be imposed to  $s_{M,h}$ . Regarding the initial condition, we impose  $s_{M,h}|_{t=0} = s_M|_{t=0}$ . When Neumann boundary conditions are considered, we enforce  $(\mathbf{n} \cdot \nabla)_h s_{M,h}|_{boundary} = \mathbf{n} \cdot \nabla s_M|_{boundary}$ , where  $\mathbf{n}$  is the unit vector perpendicular to the boundary and the operator  $(\mathbf{n} \cdot \nabla)_h$  is the discretized derivative used by the code. In the case of Dirichlet boundary conditions, we require  $s_{h,M}|_{boundary} = s_M|_{boundary}$ . In some cases, for example in GBS, more elaborated boundary conditions are used, which require the computation of further source terms (see Section 2.3.1 for a concrete example).

The idea behind the MMS is trivial. However, its implementation requires to consider some subtleties. As a matter of fact, the manufactured solution should satisfy the following requirements [26]: (i) be sufficiently smooth and not singular, (ii) satisfy the code constraints (e.g. positivity for the density or the temperature), (iii) be general enough to excite all terms present in the equations, and (iv) ensure that the different terms composing the equations are of the same order of magnitude so that no term dominates the others. Due to these constraints, the manufactured solutions are usually built as a combination of trigonometric and/or hyperbolic functions. We remark that the code verification is a purely mathematical issue and therefore the choice of  $s_M$  is independent of the physical solution of  $M$ . We also remark that the MMS cannot be applied to codes used to model singularities, shocks or discontinuities; the verification of these codes is still an open issue [26]. Finally, care must be taken computing the source terms and applying the boundary conditions, the use of symbolic computational software could result necessary for this purpose.

## 2.2 Rigorous verification of PIC codes

Originally developed to simulate fluid flows in two dimensions [58], the PIC algorithm is now a valuable tool to solve the Vlasov-Maxwell system of equations [59–63]. The PIC algorithm approximates the distribution function with a set of computational particles that are evolved in time according to Newton’s laws, and computes self-consistently the electric and magnetic fields acting on the particles by solving Maxwell’s equations. While conceptually simple in their basic formulations, the development of PIC simulation methods has significantly increased their range of applicability, accuracy, and performance. Energy, momentum, and charge conserving algorithms have been developed [64–66], which, also within an implicit-time discretization [67–71], allowed progress from the solution of one-dimensional, electrostatic models, to the simulation of complex and realistic three-dimensional electromagnetic systems. Thanks to PIC simulations, significant progress has been made in the understanding of fundamental plasma phenomena, such as collisionless shocks (see e.g. Refs. [72–74]), magnetic reconnection (see e.g. Refs. [30, 75, 76]), laser-plasma interactions (see e.g. Refs. [77–79]), and the plasma-wall transition (see e.g. Refs. [80, 81]).

Considering the widespread use of PIC codes, we propose a methodology to rigorously assess their correct implementation. We note that the rigorous code verification methodology presented in Section 2.1 is not directly applicable to PIC codes. Therefore, in the present section we discuss how to generalize the MMS for the verification of PIC codes, accounting for numerical schemes intrinsically affected by statistical noise and providing a measure of the distance between continuous, analytical distribution functions, and finite samples of computational particles.

### 2.2.1 The PIC method

The PIC algorithm represents the distribution function of plasma species as a set of computational particles (also known as superparticles or markers), whose position in the phase space is evolved according to Newton's laws. The forces acting on the particles are obtained by solving Maxwell's equations, having assigned to a numerical grid the charge and the current carried by the particles [59–63].

As the goal of this section is to introduce a rigorous methodology for the verification of PIC simulation codes, we consider the simplest kinetic model describing a one-dimensional, electrostatic, collisionless plasma in a periodic domain. The generalization to the collisional, electromagnetic, three-dimensional case does not present conceptual difficulties. The model we consider is written

$$\frac{\partial f_\alpha}{\partial t} + v \frac{\partial f_\alpha}{\partial x} + \frac{q_\alpha}{m_\alpha} E \frac{\partial f_\alpha}{\partial v} = 0, \quad (2.3)$$

$$\frac{\partial E}{\partial x} = \frac{\rho}{\epsilon_0}, \quad (2.4)$$

where  $f_\alpha(x, v, t)$  is the distribution function for the  $\alpha$  species ( $\alpha = e$  for electrons and  $\alpha = i$  for ions),  $q_\alpha$  and  $m_\alpha$  are the particle charge and mass,  $\rho(x, t) = \sum_\alpha q_\alpha \int_{-\infty}^{+\infty} f_\alpha(x, v, t) dv$  is the total charge distribution and  $E(x, t)$  is the electric field. As  $m_i \gg m_e$ , ions can be assumed at rest as a first approximation, with the ion plasma density  $n_i = \int_{-\infty}^{+\infty} f_i dv$  constant in time and uniform along  $x$ . In the remainder of this section we use this approximation and we consider only the evolution of the electron distribution function (we drop the  $\alpha$  index).

The PIC method solves numerically Eqs. (2.3)-(2.4) by performing the following steps. (i) At  $t = 0$ ,  $N$  computational particles are randomly distributed in the phase space according to a distribution function  $f_0(x, v)$ , and a weight  $w_p$  is assigned to each particle, with  $w_p = f(x_p, v_p, t = 0)/f_0(x_p, v_p)$  [if  $f_0(x, v) = f(x, v, t = 0)$  all markers have the same weight]. (ii) The particle charge is assigned to a numerical grid with spacing  $\Delta x$ , to obtain the charge distribution at each grid point. (iii) Poisson's equation, Eq. (2.4), is solved and the electric field  $E$  is computed on the grid. (iv)  $E$  is interpolated from the grid to the particle positions, to obtain the electric field  $E_p$  acting on each particle. (v) The equations of motion of the computational particles

$$\frac{dw_p}{dt} = 0, \quad \frac{dx_p}{dt} = v_p, \quad \frac{dv_p}{dt} = \frac{q}{m} E_p, \quad (2.5)$$

are numerically integrated in time to  $t = \Delta t$ , with  $\Delta t$  the step of the time integration scheme. The distribution function is now known at  $t = \Delta t$  and, following the steps (ii)-(v), the system is advanced until the final time of the simulation is reached.

Noting that the error associated with a statistical representation of the distribution function is expected to decrease as  $N^{-1/2}$  [82, 83], the numerical error affecting quantities

that result from a simulation such as  $f$  and  $E_p$  is

$$\epsilon = C_1 \Delta x^\alpha + C_2 \Delta t^\beta + C_3 N^{-1/2} + \text{higher order terms.} \quad (2.6)$$

where  $C_1$ ,  $C_2$ , and  $C_3$  are constants independent of  $\Delta x$ ,  $\Delta t$ , and  $N$ ;  $\alpha$  denotes the order of accuracy of the spatial operators in the interpolation between particles and grid positions and in the solution of Poisson's equation; and  $\beta$  is the order of accuracy of the time integration scheme.

To simplify the expression of the numerical error, we write the degree of refinement  $h$  as

$$h^p = \left( \frac{\Delta x}{\Delta x_0} \right)^\alpha = \left( \frac{\Delta t}{\Delta t_0} \right)^\beta = \left( \frac{N}{N_0} \right)^{-1/2}, \quad (2.7)$$

where  $\Delta x_0$ ,  $\Delta t_0$ , and  $N_0$  are reference parameters for a standard simulation. Consequently, from Eq. (2.6) we obtain

$$\epsilon_h = C_p h^p + \mathcal{O}(h^{p+1}), \quad (2.8)$$

where  $C_p$  is a constant independent of  $h$ . In the following, we consider  $p = \alpha$ , i.e. we define the theoretical order of accuracy of the algorithm as the order of accuracy of the spatial discretization scheme.

## 2.2.2 Challenges in applying the MMS to PIC code verification

The verification of PIC codes with the MMS is not straightforward, as it implies the comparison of a continuous, analytical distribution function with a sample of computational particles affected by statistical noise. In this section we propose a methodology to perform this comparison.

First, the manufactured solutions  $E_M$  and  $f_M$  are chosen, and the corresponding source terms to be added on the right-hand side of Eqs. (2.3)-(2.4) are computed according to

$$S_f(x, v, t) = \frac{\partial f_M}{\partial t} + v \frac{\partial f_M}{\partial x} + \frac{q E_M}{m} \frac{\partial f_M}{\partial v} \quad (2.9)$$

and

$$S_E(x, t) = \frac{\partial E_M}{\partial x} - \frac{\rho}{\epsilon_0}, \quad (2.10)$$

with  $S_E = 0$  if  $E_M$  is chosen consistently with  $f_M$ . While adding  $S_E$  in Poisson's equation does not present any conceptual difficulty, adding a source term to the Vlasov equation requires the evolution in time of the computational particle weights,  $w_p$ , and the modification of Eq. (2.5) accordingly [84]. More precisely, the particles are initially distributed with a pseudorandom number generator according to a chosen distribution function

$f_0(x, v)$  and the initial weights are set as  $w_p(0) = f_M[x_p(0), v_p(0), 0] / f_0[x_p(0), v_p(0)]$ . The weights  $w_p$  are then evolved according to

$$\frac{dw_p}{dt} = \frac{S_f[x_p(t), v_p(t), t]}{f_0[x_p(0), v_p(0)]}. \quad (2.11)$$

We remark that, in the presence of a collision operator, the marker distribution is not conserved along particle trajectories and Eq. (2.11) should be generalized according to Ref. [84]. We also note that, to avoid altering the convergence properties of the numerical scheme, Eq. (2.11) has to be integrated in time by using a numerical scheme with order of accuracy  $\beta$  or greater.

We now define the norms used to compute the numerical error affecting the simulation results. For the electric field, this does not present any particular difficulty, and we indicate the distance between the numerical and the manufactured solution as

$$\epsilon(E_p) = \max_t \max_{p=1, \dots, N} |E_p(t) - E_M[x_p(t), t]|. \quad (2.12)$$

On the other hand, the definition of the norm used to quantify the numerical error affecting  $f_M$  requires to measure the distance between a continuous analytical distribution function and a set of  $N$  computational particles.

The comparison of a data set of  $N$  elements to a distribution function is a general mathematical issue that appears in many research fields [85, 86]. For a one-dimensional probability density function  $g(x)$ , a data set can be compared to  $g$  considering the cumulative distribution function (CDF)  $G(x) = \int_{-\infty}^x g(x') dx'$  and the empirical distribution function (EDF) related to the data set  $G_N(x) = \sum_{p=1}^N \theta(x - x_p) / N$ , where  $x_p$  are the elements of the data set,  $p = 1, \dots, N$  is the particle index and  $\theta(x)$  is the Heaviside step function [ $\theta(x) = 0$  if  $x < 0$ , and  $\theta(x) = 1$  otherwise]. Under the null hypothesis, i.e.  $\{x_p\}_{p=1, \dots, N}$  is a set of  $N$  random realizations of the distribution function  $g$ , and in the limit  $N \rightarrow \infty$ , the distance  $D_N = \sup_{x \in \mathbb{R}} |G(x) - G_N(x)|$  converges to 0 as  $\mathcal{O}(N^{-1/2})$  [87], where the supremum is used rather than the maximum since  $G_N(x)$  is a piecewise continuous function.

To generalize this result to  $d > 1$  dimensions, Peacock developed a method, detailed in Ref. [88], which is used to evaluate the distance between a multidimensional distribution function and an observed sample of  $N$  elements. For a two-dimensional distribution function  $f_M(x, v, t)$  and a data set of  $N$  elements  $\{x_p(t), v_p(t)\}_{p=1, \dots, N}$  of equal weight, at a given time  $t$  (in the reminder of this section we drop the  $t$  dependence to simplify the notation), Peacock's methodology requires one to define the four CDFs

$$\begin{aligned} F^1(x, v) &= \frac{1}{n} \int_{-\infty}^x \int_{-\infty}^v f_M(x', v') dx' dv', & F^2(x, v) &= \frac{1}{n} \int_x^{+\infty} \int_{-\infty}^v f_M(x', v') dx' dv', \\ F^3(x, v) &= \frac{1}{n} \int_x^{+\infty} \int_v^{+\infty} f_M(x', v') dx' dv', & F^4(x, v) &= \frac{1}{n} \int_{-\infty}^x \int_v^{+\infty} f_M(x', v') dx' dv', \end{aligned} \quad (2.13)$$

and the four EDFs

$$\begin{aligned} F_N^1(x, v) &= \sum_{p=1}^N \frac{1}{N} \theta(x - x_p) \theta(v - v_p), & F_N^2(x, v) &= \sum_{p=1}^N \frac{1}{N} \theta(x_p - x) \theta(v - v_p), \\ F_N^3(x, v) &= \sum_{p=1}^N \frac{1}{N} \theta(x_p - x) \theta(v_p - v), & F_N^4(x, v) &= \sum_{p=1}^N \frac{1}{N} \theta(x - x_p) \theta(v_p - v), \end{aligned} \quad (2.14)$$

and compute the largest difference between  $F^i$  and  $F_N^i$  ( $i = 1, 2, 3, 4$ ),

$$d_i^P = \sup_{(x,v) \in \mathbb{R}^2} |F^i(x, v) - F_N^i(x, v)|. \quad (2.15)$$

The distance between  $f_M(x, v)$  and  $\{x_p, v_p\}_{p=1, \dots, N}$  is thus defined as

$$\epsilon_P(f_M) = \max(d_1^P, d_2^P, d_3^P, d_4^P). \quad (2.16)$$

Reference [88] shows empirically that  $\epsilon_P(f_M)$  decreases as  $\mathcal{O}(N^{-1/2})$ , irrespective of the choice of  $f_M$ , if  $\{x_p, v_p\}_{p=1, \dots, N}$  is a set of random realizations of  $f_M$ .

To verify a PIC simulation code with the MMS, one has to account for arbitrary values of  $w_p$ , and the definition of the  $F_N^i$ , Eq. (2.14), should be modified as

$$\begin{aligned} F_N^1(x, v) &= \sum_{p=1}^N \hat{w}_p \theta(x - x_p) \theta(v - v_p), & F_N^2(x, v) &= \sum_{p=1}^N \hat{w}_p \theta(x_p - x) \theta(v - v_p), \\ F_N^3(x, v) &= \sum_{p=1}^N \hat{w}_p \theta(x_p - x) \theta(v_p - v), & F_N^4(x, v) &= \sum_{p=1}^N \hat{w}_p \theta(x - x_p) \theta(v_p - v), \end{aligned} \quad (2.17)$$

with  $\hat{w}_p = w_p / \sum_{p=1}^N w_p$ . We show empirically (see Section 2.2.3) that, if one defines the EDFs according to Eq. (2.17),  $\epsilon_P(f_M)$  still decreases as  $N^{-1/2}$  for  $N \rightarrow \infty$ .

We remark that  $\epsilon_P(f_M)$  is affected by statistical uncertainty due to the random initialization of the markers. Consequently, the observed order of accuracy  $\hat{p}$  obtained using  $\epsilon_h = \epsilon_P(f_M)$  in Eq. (2.1) is also affected by statistical uncertainty. To perform an order of accuracy test, it is therefore necessary to carry out a number,  $n_s$ , of simulations with different pseudorandom number generator seeds, and compute the numerical error  $\epsilon_{h,i} = \epsilon_P(f_M)$  for each simulation, with  $i = 1, \dots, n_s$ . Then, following the methodology discussed in Section 3.1, it is possible to approximate the expected value of  $\epsilon_h$  with

$$\epsilon_h \simeq \frac{1}{n_s} \sum_{i=1}^{n_s} \epsilon_{h,i} \quad (2.18)$$

and the corresponding statistical uncertainty with

$$\Delta \epsilon_h = 1.96 \frac{\sigma_{\epsilon_h}}{\sqrt{n_s}}, \quad (2.19)$$

where  $\sigma_{\epsilon_h} = \sqrt{\sum_{i=1}^{n_s} (\epsilon_h - \epsilon_{h,i})^2 / (n_s - 1)}$  is the standard deviation corresponding to the distribution of  $\epsilon_{h,i}$ . Finally, the expected value of  $\hat{p}$  is computed combining Eq. (2.18)

with Eq. (2.1), and the corresponding statistical uncertainty is obtained as

$$\Delta\hat{p} = \frac{1}{\ln(r)} \sqrt{\left(\frac{\Delta\epsilon_h}{\epsilon_h}\right)^2 + \left(\frac{\Delta\epsilon_{rh}}{\epsilon_{rh}}\right)^2}. \quad (2.20)$$

### 2.2.3 Reducing the computational cost of a PIC verification

The evaluation of  $\epsilon_P(f_M)$  is computationally expensive for a data set with a large number of elements. In fact, since  $F_N^i(x, v)$  is a discontinuous function, the classical methods applied to compute the maximum value of a continuous function are not suitable. Moreover, the local maxima of the difference  $|F^i(x, v) - F_N^i(x, v)|$  are found at the  $N^2$  points  $(x_{p_j}, v_{p_k})$ , with  $p_j = 1, \dots, N$  and  $p_k = 1, \dots, N$ . Therefore, to compute  $d_i^P$ , one has to evaluate the limits

$$d_{j,k}^i = \lim_{x \rightarrow x_{p_j}^\pm} \lim_{v \rightarrow v_{p_k}^\pm} |F^i(x, v) - F_N^i(x, v)| \quad (2.21)$$

for all  $(x_{p_j}, v_{p_k})$  and then impose  $d_i^P = \max_{j,k} d_{j,k}^i$ .

Reference [89] shows that  $F_N^i(x, v)$ , where  $(x, v)$  is a general point of the phase-space, can be evaluated with a brute force algorithm, or partitioning the points  $(x_p, v_p)$  in a k-d tree or in a range-counting tree. It results that the complexity of computing  $d_i^P$  according to Eq. (2.15) is:  $\mathcal{O}(N^3)$  with the brute force algorithm,  $\mathcal{O}(N^{5/2})$  with a k-d tree partitioning, and  $\mathcal{O}(N^2 \log N)$  with a range-counting tree partitioning. The memory used to partition the  $N$  points with a range-counting tree scales as  $\mathcal{O}(N \log N)$ , while it scales as  $\mathcal{O}(N)$  for a k-d tree partitioning or brute-force algorithm. Therefore, all the evaluations of the EDF used to obtain the results presented in this chapter are performed using a k-d tree partitioning, which in our opinion is the best compromise between computational cost and memory needs.

To decrease the computational cost of computing  $d_i^P$ , Fasano and Franceschini propose an alternative approach [90], which approximates  $d_i^P$  as

$$d_i^P \simeq d_i^{FF} = \max_{p=1, \dots, N} \lim_{x \rightarrow x_p^\pm} \lim_{v \rightarrow v_p^\pm} |F^i(x, v) - F_N^i(x, v)|, \quad (2.22)$$

where the  $F_N^i$  are evaluated according to Eq. (2.14). Reference [90] shows empirically that the value of  $\epsilon_{FF}(f_M)$  decreases as  $\mathcal{O}(N^{-1/2})$  if we define  $\epsilon_{FF}(f_M) = \max(d_1^{FF}, d_2^{FF}, d_3^{FF}, d_4^{FF})$ , where  $\{x_p, v_p\}_{p=1, \dots, N}$  is a set of random realizations of  $f_M$ . The computational cost of evaluating  $d_i^{FF}$  is reduced by a factor  $N$  with respect to  $d_i^P$ .

As the computational cost of evaluating  $d_i^{FF}$  remains very demanding for high values of  $N$ , we discuss here an alternative method used to approximate  $d_i^P$ . Instead of maximizing  $|F^i(x, v) - F_N^i(x, v)|$  over all points  $(x_p, v_p)$ , as done according to Fasano and Franceschini's approach, one can generate  $M$  random points  $(x_j, v_j)$ , with  $j = 1, \dots, M$ ,

and approximate  $d_i^P$  with

$$d_i^P \simeq d_i^{MC} = \max_{j=1,\dots,M} |F^i(x_j, v_j) - F_N^i(x_j, v_j)|. \quad (2.23)$$

This approximation is a true equality in the limit  $M \rightarrow \infty$ , and corresponds to evaluating Eq. (2.15) with the Monte-Carlo method. We can therefore compute the distance between  $f_M(x, v)$  and  $\{x_p, v_p\}_{p=1,\dots,N}$  as

$$\epsilon_{MC}(f_M) = \max(d_1^{MC}, d_2^{MC}, d_3^{MC}, d_4^{MC}). \quad (2.24)$$

The evaluation of  $d_i^{MC}$  is computationally  $N^2/M$  times less demanding than  $d_i^P$  and  $N/M$  times less demanding than  $d_i^{FF}$ .

To further reduce the computational cost of performing a PIC code verification, we also investigate the comparison of  $F^i(x, v)$  with  $F_N^i(x, v)$  only at  $x = \pm\infty$  and  $v = \pm\infty$ , i.e. evaluating the supremum of the difference  $|F^i(x, v) - F_N^i(x, v)|$  only over the boundaries of the phase-space domain. More precisely, we define the two errors

$$\epsilon_x(f_M) = \sup_{x \in \mathbb{R}} \left| \int_{-\infty}^x \left[ \int_{-\infty}^{+\infty} f_M(x', v) dv \right] dx' - \sum_{p=1}^N \hat{w}_p \theta(x - x_p) \right|, \quad (2.25)$$

$$\epsilon_v(f_M) = \sup_{v \in \mathbb{R}} \left| \int_{-\infty}^v \left[ \int_{-\infty}^{+\infty} f_M(x, v') dx \right] dv' - \sum_{p=1}^N \hat{w}_p \theta(v - v_p) \right|, \quad (2.26)$$

and assess whether they decrease according to the order of accuracy expected for the numerical scheme.

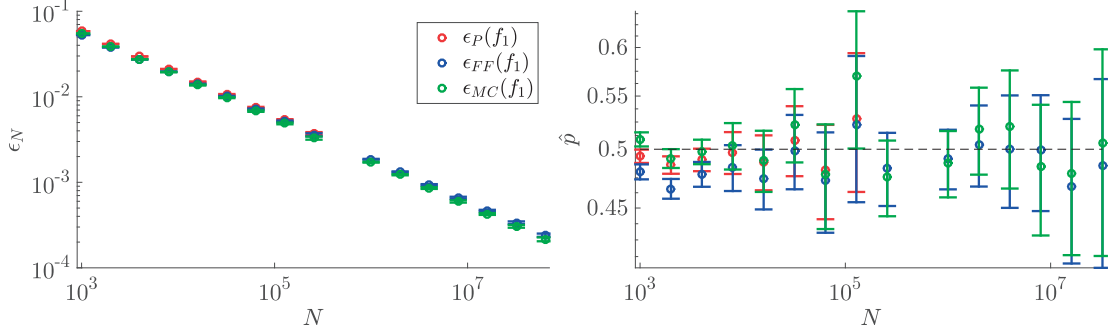
To empirically show that  $\epsilon_P(f_M)$ ,  $\epsilon_{FF}(f_M)$ , and  $\epsilon_{MC}(f_M)$  decrease as  $N^{-1/2}$  for  $N \rightarrow \infty$  if  $\{x_p, v_p\}_{p=1,\dots,N}$  is a set of random realizations of  $f_M$  and  $w_p \neq 1$ , we proceed as follows. First, we generate  $N$  points  $(x_p, v_p)$  according to  $f_0(v) = e^{-|v|}/(2L)$ , with  $p = 1, \dots, N$ ,  $x_p \in [0, L]$ , and  $v_p \in ]-\infty, \infty[$ , and we set  $w_p = f_1(x_p, v_p)/f_0(v_p)$ , where

$$f_1(x, v) = \frac{1}{L} \frac{2}{\sqrt{\pi}} v^2 e^{-v^2}. \quad (2.27)$$

Then, we compute  $\epsilon_P(f_1)$ ,  $\epsilon_{FF}(f_1)$ , and  $\epsilon_{MC}(f_1)$  [for  $\epsilon_{MC}(f_1)$ ,  $M = 10^6$ ]. We apply this procedure for different  $N$ , and, for each  $N$ , we repeat the process a number of times, changing the pseudorandom number generator seed. We compute the averaged value of  $\epsilon_P(f_1)$ ,  $\epsilon_{FF}(f_1)$ , and  $\epsilon_{MC}(f_1)$  according to Eq. (3.1), and the corresponding statistical uncertainties according to Eq. (3.3). Finally, defining  $h = 1/N$ , we estimate  $\hat{p}$  and its statistical uncertainty by applying Eqs. (2.1) and (2.20). The results thus obtained are presented in Fig 2.1. We observe that the distance between  $f_1(x, v)$  and the data sets decreases as  $N^{-1/2}$  for  $N \rightarrow \infty$ , with a similar value, for all the three norms.



## 2.3. Examples of application of the code verification methodology



**Figure 2.1** – Values of  $\epsilon_P(f_1)$ ,  $\epsilon_{FF}(f_1)$ , and  $\epsilon_{MC}(f_1)$  averaged over the performed set of simulations (left panel) and corresponding values of  $\hat{p}$  (right panel) for  $h = 1/N$ . The error bars represent the statistical uncertainty affecting the results.

## 2.3 Examples of application of the code verification methodology

We illustrate two concrete examples of application of the code verification methodologies discussed above by applying the procedure described in Section 2.1 to the GBS code and the methodology illustrated in Section 2.2 to a one-dimensional, electrostatic, collisionless PIC code. These two codes constitute an ideal test bed for the proposed verification methodologies.

### 2.3.1 Application of the code verification methodology to GBS

The GBS code [20] was developed at the Swiss Plasma Center (SPC) in the last few years with the goal of simulating plasma turbulence in basic plasma physics experiments and in the tokamak SOL. The details of the model solved by GBS, as well as its numerical implementation, are described in Appendix A. For the purpose of the present chapter, we consider a limited tokamak configuration with circular magnetic flux surfaces in the infinite aspect ratio limit and a toroidal limiter on the high-field side equatorial midplane, with no magnetic shear. Moreover, the Boussinesq [91–93], the cold ion ( $T_i = 0$ ), and the electrostatic approximations are used. Under these assumptions the GBS model

equations, Eqs. (A.65)-(A.70), reduce to

$$\partial_t n = -R_0 \{\phi, n\} + 2[C(p_e) - nC(\phi)] - \nabla_{\parallel} (nv_{\parallel e}) + D_n \nabla_{\perp}^2 n + S_n, \quad (2.28)$$

$$\partial_t \omega = -R_0 \{\phi, \omega\} + \frac{2}{n} C(p_e) - v_{\parallel i} \nabla_{\parallel} \omega + \frac{1}{n} \nabla_{\parallel} j_{\parallel} + \frac{1}{3n} C(G_i) + D_{\omega} \nabla_{\perp}^2 \omega, \quad (2.29)$$

$$\begin{aligned} \partial_t v_{\parallel e} = & -R_0 \{\phi, v_{\parallel e}\} + \frac{m_i}{m_e} \left[ \nabla_{\parallel} \phi - \frac{1}{n} \nabla_{\parallel} p_e - 0.71 \nabla_{\parallel} T_e + \nu j_{\parallel} - \frac{2}{3n} \nabla_{\parallel} G_e \right] \\ & - v_{\parallel e} \nabla_{\parallel} v_{\parallel e} + D_{v_{\parallel e}} \nabla_{\perp}^2 v_{\parallel e}, \end{aligned} \quad (2.30)$$

$$\partial_t v_{\parallel i} = -R_0 \{\phi, v_{\parallel i}\} - v_{\parallel i} \nabla_{\parallel} v_{\parallel i} - \frac{1}{n} \nabla_{\parallel} p_e - \frac{2}{3n} \nabla_{\parallel} G_i + D_{v_{\parallel i}} \nabla_{\perp}^2 v_{\parallel i}, \quad (2.31)$$

$$\begin{aligned} \partial_t T_e = & -R_0 \{\phi, T_e\} + \frac{4}{3} T_e \left[ \frac{7}{2} C(T_e) + \frac{T_e}{n} C(n) - C(\phi) \right] - v_{\parallel e} \nabla_{\parallel} T_e + \nabla_{\parallel} (\chi_{\parallel e} \nabla_{\parallel} T_e) \\ & + \frac{2}{3} T_e \left[ 0.71 \nabla_{\parallel} v_{\parallel i} - 1.71 \nabla_{\parallel} v_{\parallel e} + 0.71 \left( \frac{v_{\parallel i} - v_{\parallel e}}{n} \right) \nabla_{\parallel} n \right] + D_{T_e} \nabla_{\perp}^2 T_e + S_{T_e}, \end{aligned} \quad (2.32)$$

where the differential operators  $\mathcal{C}$ ,  $\{\phi, -\}$ ,  $\nabla_{\parallel}$  and  $\nabla_{\perp}^2$  are detailed in Section A.3.4. The system is closed by the Poisson's equation  $\nabla_{\perp}^2 \phi = \omega$ .

The GBS model equations are completed by a set of boundary conditions that describe the plasma properties at the magnetic pre-sheath entrance [94], as described in Appendix A. Within the assumptions used in this section, they are written as

$$v_{\parallel i} = \pm c_s \quad (2.33)$$

$$v_{\parallel e} = \pm c_s \exp(\Lambda - \phi/T_e) \quad (2.34)$$

$$\partial_y T_e = 0 \quad (2.35)$$

$$\partial_y n = \mp \frac{n}{c_s} \partial_y v_{\parallel i} \quad (2.36)$$

$$\omega = - \left( \partial_y v_{\parallel i} \right)^2 \mp c_s \partial_y^2 v_{\parallel i} \quad (2.37)$$

$$\partial_y \phi = \mp c_s \partial_y v_{\parallel i} \quad (2.38)$$

where  $\Lambda = 3$ . Here the upper signs apply to the case of magnetic field directed towards the wall, while the lower ones apply to the opposite case. While the numerical scheme used to solve Eqs. (2.28)-(2.32) and (2.33)-(2.38) is described in Section A.3.5, we note that the expected orders of accuracy characterizing the numerical model are  $p_s = 2$ , in the spatial directions, and  $p_t = 4$ , for the time discretization. Defining  $h = \Delta y / \Delta y_0 = \Delta x / \Delta x_0 = \Delta z / \Delta z_0 = (\Delta t / \Delta t_0)^2$ , we expect an overall  $p = 2$  for the numerical scheme. In order to verify the implementation of Eqs. (2.28)-(2.32) and (2.33)-(2.38) in GBS, the methodology illustrated in Section 2.1 is applied as follows. First, the discretization scheme used to solve Eqs. (2.28)-(2.32) and the Poisson's equation is analyzed, using, for simplicity, Dirichlet (for  $v_{\parallel i}$ ,  $v_{\parallel e}$  and  $\omega$ ) and Neumann (for  $n$ ,  $T_e$  and  $\phi$ ) boundary conditions. Dirichlet boundary conditions are applied at the grid points (therefore, no numerical error results), and Neumann boundary conditions are discretized with a second-order numerical scheme. Second, we study the order of accuracy characterizing the discretization scheme of the boundary conditions, Eqs. (2.33)-(2.38), decoupling these

---

### 2.3. Examples of application of the code verification methodology

from the solution of Eqs. (2.28)-(2.32). Finally, the two sets of equations [Eqs. (2.28)-(2.32) and Eqs. (2.33)-(2.38)] are coupled to complete the verification of the overall GBS code. The code verification methodology is divided in these three steps to simplify the investigation of possible implementation errors; in the present section we show only the final results [i.e. verification of Eqs. (2.28)-(2.32) coupled with Eqs. (2.33)-(2.38)], which summarize the verification results obtained for GBS. We note that the methodology for the code verification allowed us to find and correct a minor bug, related to the discretization of the  $G_i$  and  $G_e$  terms at the boundaries. We tested that, luckily, the generated numerical error was very small, and its influence on the previous GBS results completely negligible.

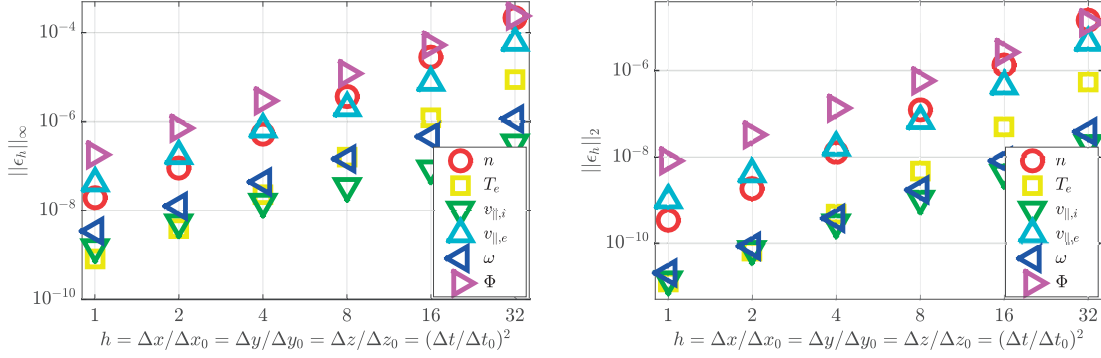
To verify the implementation of the drift-reduced Braginskii equations into GBS and to satisfy the requirements given in Section 2.1.2, we choose to manufacture the model solution as the combination of trigonometric functions. More precisely, the functions used to represent the six fields appearing in Eqs. (2.28)-(2.32) are expressed as

$$s_M(y, x, z, t) = A_s \left\{ B_s + \sin \left[ C_s \left( z - \frac{q}{a} y \right) \right] \sin(D_s y) \sin(E_s t + F_s x) \right\}, \quad (2.39)$$

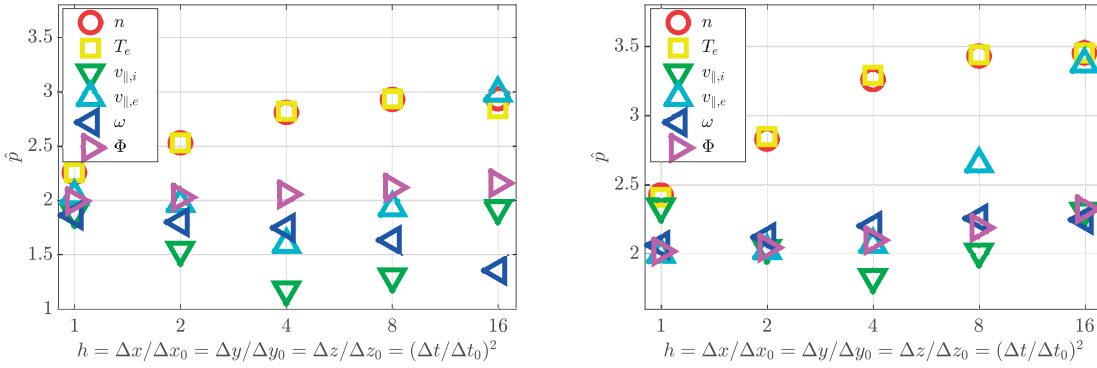
where  $A_s$ ,  $B_s$ ,  $C_s$ ,  $D_s$ ,  $E_s$  and  $F_s$  are arbitrary constants and  $s_M = n, T_e, v_{||i}, v_{||e}, \omega, \phi$  are the fields present in the GBS equations. The  $B_s$  coefficients are used to ensure the positivity of  $n$  and  $T_e$ , the other coefficients to calibrate the amplitude of the errors in order to guarantee that there is no dominating term in the equations. This means that the amplitude of the coefficients is chosen such that, for the used meshes, the simulations are in the asymptotic regime and the errors affecting the different terms of Eqs. (2.28)-(2.32) are of the same magnitude. As GBS is developed to simulate turbulent modes mainly aligned to the field lines, we impose the dependence on  $y$  and  $z$  as the product of two terms: the first one perfectly aligned to the field lines (the term containing  $C_s$ ) and a second term (containing  $D_s$ ) representing a perturbation in the poloidal direction (i.e. along the  $y$  coordinate), chosen small, not to have the discretization error on the parallel derivative dominating over the others. The  $E_s$  and  $F_s$  terms introduce the time and radial dependencies. We note that  $C_s$  must be an integer to satisfy the periodicity of the system along the  $z$  coordinate.

The computation of the source terms is trivial. It consists in plugging the analytical functions presented in Eq. (2.39) into Eqs. (2.28)-(2.32) and in Poisson's equation to obtain the source term  $S$ . This process is particularly tedious, but it involves only straightforward algebraic manipulations with no conceptual difficulties. As the results of these computations do not present any theoretical interest, we do not present those herein. We just mention that we compute the source terms using the symbolic manipulation software *Mathematica* [95], which allows the direct translation into Fortran90 language. This enables the implementation of the obtained expressions in GBS, without any significant difficulty and reducing the possibilities of mistakes.

The verification of the boundary conditions described in Eqs. (2.33)-(2.38) requires the computation of additional source terms. In fact, the manufactured solutions given in Eq. (2.39) do not satisfy the boundary conditions. Consequently, as done for the equa-



**Figure 2.2** – Norm of the numerical error affecting the discretization scheme used in GBS, plotted as function of the refinement degree  $h$ , for the two norms  $L_\infty$  (left panel) and  $L_2$  (right panel).



**Figure 2.3** – Observed order of accuracy,  $\hat{p}$ , characterizing the discretization scheme of GBS, computed applying Eq. (2.1) and plotted as function of the refinement degree  $h$ , for the two norms  $L_\infty$  (left panel) and  $L_2$  (right panel).

tions governing the physics of the SOL region, we insert the manufactured solution into Eqs. (2.33)-(2.38) and we add the resulting source terms to the boundary condition equations.

For the computation of the error and to estimate  $\hat{p}$ , we consider the two norms,  $L_2$  (i.e.  $\|\mathbf{f}\|_2 = \sqrt{\sum_i^N f_i^2/N}$ ) and  $L_\infty$  (i.e.  $\|\mathbf{f}\|_\infty = \max |f_i|$ );  $L_2$  is appropriate to ensure the correct global convergence of the results, while  $L_\infty$  is used to assess the local convergence in all points of the domain. Figures 2.2 and 2.3 summarize the GBS verification results. Six simulations are performed with  $h = 1, 2, 4, 8, 16, 32$  and the corresponding errors, computed using the  $L_2$  and  $L_\infty$  norms, are shown in Fig. 2.2. We observe that the numerical error clearly decreases when refining the mesh (i.e. decreasing the value of  $h$ ); on a logarithmic scale, the numerical error decreases linearly, with slope  $p$ , as expected. We also note that our scan leads to a reduction of the numerical error by at least three orders of magnitude, this gives confidence that there are not subdominant errors decreasing at a rate different than the expected one. The estimate of the observed order of accuracy, evaluated according to Eq. (2.1), is plotted as a function of the parameter  $h$  in Fig. 2.3. Clearly, with the refinement of the meshes,  $\hat{p}$  tends to  $p$  for all the fields, as expected, although the rate of convergence is field dependent. This is due to the fact that the coefficients of the Taylor expansion of the numerical error are different for each field.

---

### 2.3. Examples of application of the code verification methodology

Consequently, we demonstrate that Eqs. (2.28)-(2.32), the Poisson's equation, and the boundary conditions [Eqs. (2.33)-(2.38)] are correctly coded in GBS, with a numerical scheme that satisfies the theoretical order of convergence.

#### 2.3.2 Application of the verification methodology to a PIC simulation code

To exemplify the methodology presented in Section 2.2, we consider the following PIC algorithm. Equations (2.3)-(2.4) are solved on a periodic spatial domain that extends from  $x = 0$  to  $x = L$ . A numerical grid  $x_i = i\Delta x$  is used to discretize the  $x$  coordinate, with  $\Delta x = L/M$  the grid spacing ( $i = 0, \dots, M - 1$  and  $M$  the number of grid points), and a time step  $\Delta t$  is introduced for the integration of the equations of motion. The charge of the particles is assigned to the grid using a first-order weighting scheme, known as cloud-in-cell (CIC) scheme [60], i.e.  $\rho(x_i, t) = \sum_{p=1}^N qI[x_i - x_p(t)]w_p(t)$ , with  $I$  the interpolation function given by

$$I(x) = \begin{cases} 0 & \text{if } |x| > \Delta x \\ -\frac{|x|}{\Delta x} + 1 & \text{if } |x| \leq \Delta x. \end{cases} \quad (2.40)$$

Poisson's equation  $\partial_x^2 \phi(x, t) = -\rho/\epsilon_0$  is solved by using a second order centered finite difference scheme and imposing the boundary condition  $\phi(x = 0) = 0$ . The electric field  $E_p$  is computed according to  $E(x, t) = -\partial_x \phi(x, t)$  by using a second order centered finite difference scheme and interpolating from the grid onto the particle positions using again the CIC scheme. Finally, the equations of motion, Eq. (2.5), are integrated in time with a second order Leapfrog scheme. This numerical scheme is second order in  $\Delta x$  and  $\Delta t$ , i.e.  $\alpha = \beta = 2$ . All quantities are normalized to (tilde denotes a physical quantity in SI units):  $x = \tilde{x}/\tilde{\lambda}_D$ ,  $t = \tilde{t}\tilde{\omega}_{pe}$ , where  $\tilde{\lambda}_D = \sqrt{\epsilon_0 \tilde{T}_{e0}/(\tilde{n}_0 e^2)}$  is the Debye length and  $\tilde{\omega}_{pe} = \sqrt{\tilde{n}_0 e^2/(\epsilon_0 m_e)}$  is the plasma frequency, with  $\tilde{n}_0$  and  $\tilde{T}_{e0}$  a reference density and electron temperature, respectively. The simulation code is written in Fortran90 and parallelized using a domain cloning approach, implemented within an hybrid message passing interface (MPI) and OpenMP environment.

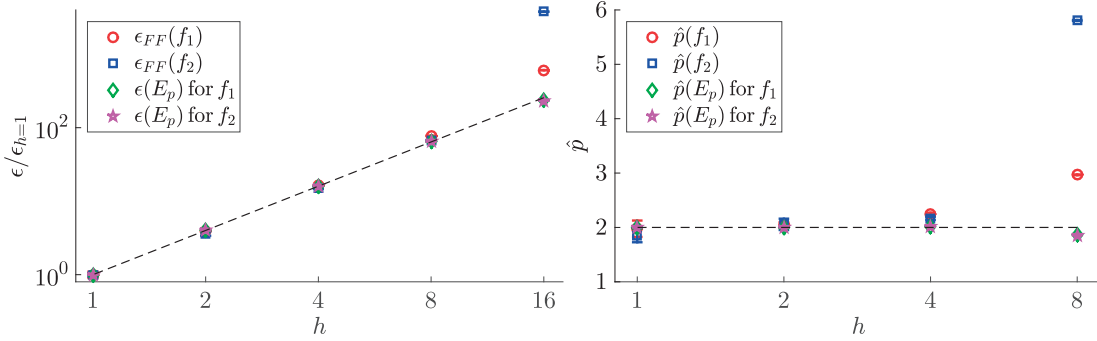
To apply the code verification methodology discussed in Section 2.2, we choose the following manufactured solutions

$$E_M(x, t) = 2\pi k_x L \sin(\pi t) \sin\left(k_x \frac{2\pi}{L} x\right), \quad (2.41)$$

$$f_M(x, v, t) = f_x(x, t) f_v(v), \quad (2.42)$$

where

$$f_v(v) = \frac{2}{\sqrt{\pi}} v^2 e^{-v^2} \quad (2.43)$$



**Figure 2.4** – Values of  $\epsilon_{FF}(f_M)$  and  $\epsilon(E_p)$  averaged over the performed set of simulations (left panel) and corresponding  $\hat{p}$  (right panel) for the two distribution functions  $f_1$  and  $f_2$ , and for  $h = 1, 2, 4, 8, 16$ . Each error is normalized to its value at  $h = 1$ , and the statistical uncertainties are represented with error bars. The dashed lines represent  $h^2$  (left panel) and  $\hat{p} = 2$  (right panel).

and we make use of two functions for  $f_x(x, t)$ ,

$$f_{x1}(x, t) = [1 - \sin(\pi t) \cos(2\pi x/L)] / L, \quad (2.44)$$

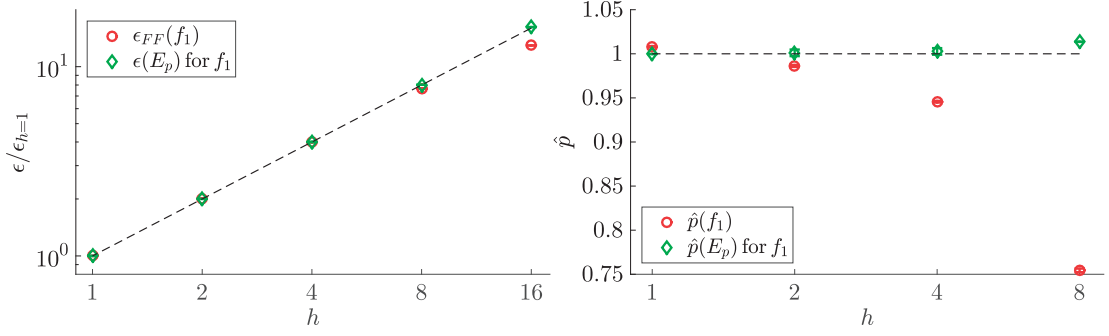
$$f_{x2}(x, t) = \frac{1}{L} + \frac{1}{2} [1 - \cos(\pi t)] \left\{ \frac{\lambda [(L-x)^{-2} + x^{-2}]}{2 \cosh^2[\lambda(L-x)^{-1} - \lambda x^{-1}]} - \frac{1}{L} \right\}, \quad (2.45)$$

to ensure empirically that the results discussed are valid for different choices of  $f_M$ . We denote  $f_1 = f_{x1}f_v$  and  $f_2 = f_{x2}f_v$ . We remark that the manufactured solutions satisfy the requirements listed in Section 2.1.2. In particular, the parameters  $k_x$ ,  $L$ , and  $\lambda$  allow us to calibrate the numerical error, so that the magnitude of the different terms in the Vlasov-Poisson system are of the same order of magnitude (we use  $L = 2$  and  $k_x = 2$  for  $f_1$ , and  $k_x = 5$ ,  $L = 5$ , and  $\lambda = 20$  for  $f_2$ , and for all the simulations we evolve the computational particles for 2 time units). We note that  $f_v(v)$  is chosen different from a Maxwellian distribution in  $v$  to ensure that a numerical solution does not converge to  $f_M$  because of numerical dissipation. Finally, we note that the computational particles are initially distributed according to the probability distribution function  $f_0(v) = e^{-|v|}/(2L)$  and the initial weights are computed as  $w_p(0) = f_M[x_p(0), v_p(0), 0]/f_0[v_p(0)]$ .

For the verification of the PIC code, we refine at the same time the grid size and the time step, while increasing the number of particles. Defining  $h = \Delta x/\Delta x_0 = \Delta t/\Delta t_0 = (N/N_0)^{-1/4}$ , we perform five sets of simulations, with respectively  $h = 1, 2, 4, 8, 16$ , for both  $f_1$  and  $f_2$ . We perform simulations with  $h = 8, 16$  thousands of times, simulations with  $h = 4$  hundreds of times, and simulations with  $h = 1, 2$  a few times, and for each value of  $h$  we compute the average of  $\epsilon_{FF}(f_M)$  and  $\epsilon(E_p)$  and the corresponding uncertainty. The observed order of accuracy  $\hat{p}$  and the corresponding uncertainty are computed applying Eqs. (2.1) and (2.20), respectively. We note that, while  $\epsilon(E_p)$  is computed considering all the time steps of the simulations,  $\epsilon_{FF}(f_M)$  is estimated at  $t = 2$  due to the high computational cost of its evaluation.

The results obtained from these simulations are represented in Fig. 2.4. Both  $\epsilon_{FF}(f_M)$  and  $\epsilon(E_p)$  clearly decrease for  $h \rightarrow 0$ . Moreover, the observed order of accuracy  $\hat{p}$  converges to 2 when decreasing  $h$ , proving that the PIC algorithm is correctly implemented

### 2.3. Examples of application of the code verification methodology



**Figure 2.5** – Values of  $\epsilon_{FF}(f_1)$  and  $\epsilon(E_p)$  averaged over the performed set of simulations (left panel) and corresponding  $\hat{p}$  (right panel) for  $h = 1, 2, 4, 8, 16$ . Each error is normalized to its value at  $h = 1$ , and the statistical uncertainties are represented with error bars. The dashed lines represent  $h$  (left panel) and  $\hat{p} = 1$  (right panel).

in the code, and the equations are verified.

As a further proof of the capabilities of the code verification methodology illustrated herein, we perform the same verification with a zero-order weighting scheme (the so-called nearest-grid-point scheme, or NGP scheme) when interpolating the electric field. This corresponds to use an interpolation function defined as

$$I(x) = \begin{cases} 1 & \text{if } |x| \leq \frac{\Delta x}{2} \\ 0 & \text{if } |x| > \frac{\Delta x}{2} \end{cases} \quad (2.46)$$

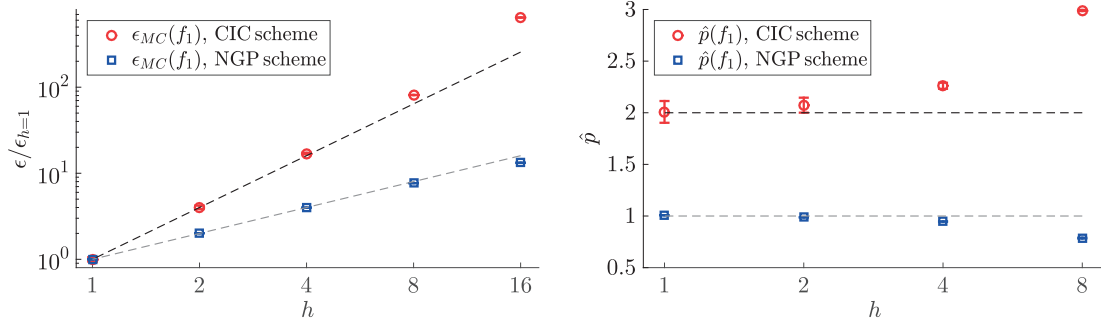
when interpolating the electric field from the grid onto the marker positions. Since the accuracy of the numerical scheme is reduced, the error affecting the results is expected to satisfy

$$\epsilon = C''h + \mathcal{O}(h^2), \quad (2.47)$$

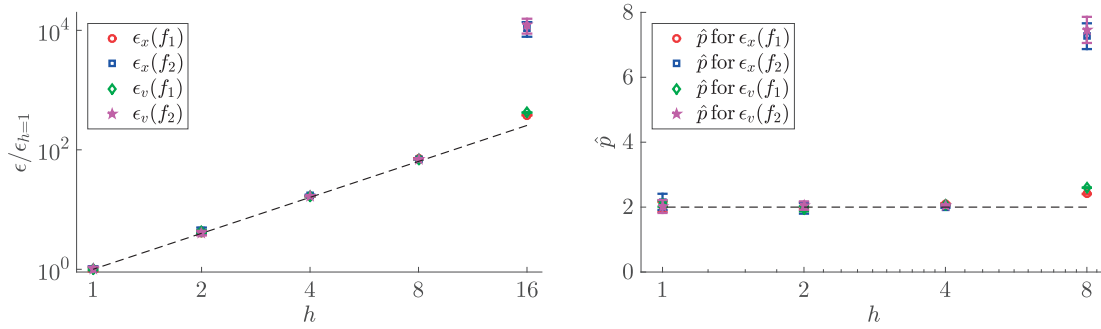
where  $C''$  is a constant independent of  $h$ . The results are presented in Fig. 2.5 (only  $f_1$  is considered for this test). The code verification methodology is able to identify this change in the numerical scheme. In fact, while both  $\epsilon_{FF}(f_1)$  and  $\epsilon(E_p)$  decrease as  $h \rightarrow 0$ , the observed order of accuracy converges to 1. Therefore, the proposed code verification methodology not only ensures that the numerical solution converges to the exact solution, but it also correctly identifies the convergence rate.

To investigate the applicability of the distance  $\epsilon_{MC}(f_M)$  for the verification of PIC simulation codes, we consider the same set of simulations presented in Figs. 2.4 and 2.5, and we evaluate the difference between  $f_M(x, v, t)$  and the sample of computational particles according to Eq. (2.24) at  $t = 2$ , for  $M = 10^6$ . The results thus obtained are shown in Fig. 2.6. We observe that the error decreases for  $h \rightarrow 0$  as expected both for the CIC and NGP schemes, with  $\hat{p} \rightarrow 2$  for the CIC scheme, and  $\hat{p} \rightarrow 1$  for the NGP scheme. Moreover, we note that the errors computed according to Fasano and Franceschini's method (see Figs. 2.4 and 2.5) and according to Eq. (2.24) are very similar. This means that the norm  $\epsilon_{MC}$  is suitable for the verification of PIC simulation codes.

Finally, we consider the same set of simulations presented in Figs. 2.4 and 2.5 and we



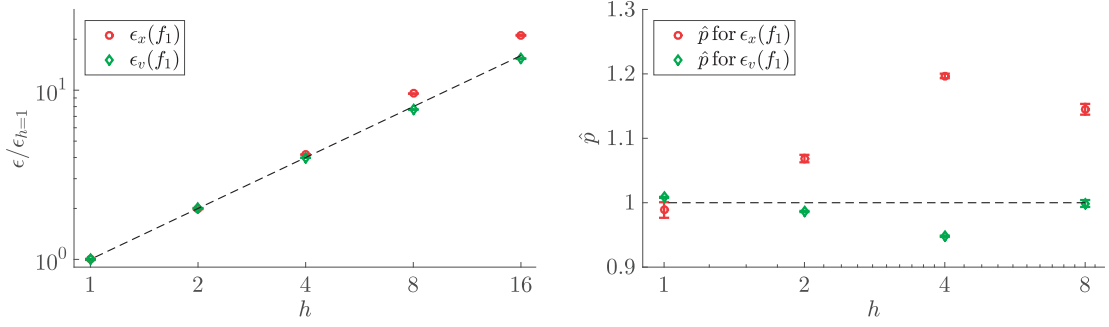
**Figure 2.6** – Values of  $\epsilon_{MC}(f_1)$  averaged over the performed set of simulations (left panel) and corresponding  $\hat{p}$  (right panel) for  $h = 1, 2, 4, 8, 16$ , interpolating the electric field onto the marker positions both with the CIC (red circles) and the NGP (blue squares) interpolation schemes. Each error is normalized to its value at  $h = 1$ , and the statistical uncertainties are represented with error bars. The dashed lines represent  $h$  and  $h^2$  (left panel) and  $\hat{p} = 1$  and  $\hat{p} = 2$  (right panel).



**Figure 2.7** – Values of  $\epsilon_x(f_M)$  and  $\epsilon_v(f_M)$  averaged over the performed set of simulations (left panel) and corresponding  $\hat{p}$  (right panel) for the two distribution functions  $f_1$  and  $f_2$ , and for  $h = 1, 2, 4, 8, 16$ . Each error is normalized to its value at  $h = 1$ , and the statistical uncertainties are represented with error bars. The dashed lines represent  $h^2$  (left panel) and  $\hat{p} = 2$  (right panel). The electric field is interpolated from the grid onto the particle positions using the CIC scheme.

evaluate  $\epsilon_x(f_M)$  and  $\epsilon_v(f_M)$ . The results thus obtained are presented in Figs. 2.7 and 2.8 for the CIC and NGP weighting schemes, respectively. We observe that the error decreases for  $h \rightarrow 0$  as expected for both the weighting schemes, with  $\hat{p} \rightarrow 2$  for the CIC scheme, and  $\hat{p} \rightarrow 1$  for the NGP scheme. Therefore, also the norms defined in Eqs. (2.25)-(2.26) are suitable for the verification of PIC simulation codes with the MMS. We note that, as the computational cost of evaluating  $\epsilon_x(f_M)$  and  $\epsilon_v(f_M)$  is considerably decreased with respect to  $\epsilon_{FF}(f_M)$ , this evaluation is performed for all  $t = 0.08j$ , with  $j = 0, \dots, 25$ , and the maximum between the resulting values is computed. We finally remark that this last approach is easily generalized to a d-dimensional distribution function, without increasing significantly the computational cost of performing the order-of-accuracy test.





**Figure 2.8** – Values of  $\epsilon_x(f_1)$  and  $\epsilon_v(f_1)$  averaged over the performed set of simulations (left panel) and corresponding  $\hat{p}$  (right panel) for  $h = 1, 2, 4, 8, 16$ . Each error is normalized to its value at  $h = 1$ , and the statistical uncertainties are represented with error bars. The dashed lines represent  $h$  (left panel) and  $\hat{p} = 1$  (right panel). The electric field is interpolated from the grid onto the particle positions using the NGP scheme.

## 2.4 Conclusions

In this chapter a rigorous methodology for plasma simulation code verification is discussed. This methodology is general, rigorous, simple-to-apply, and does not present any conceptual difficulties. Code verification requires to choose an adequate manufactured solution which satisfies some reasonable assumptions. Then, the source terms to be added to the model equations, as well as the boundary conditions, are readily evaluated. At this point, it is possible to compute  $\hat{p}$  [Eq. (2.1)] by performing a number of simulations corresponding to more and more refined meshes. If  $\hat{p} \rightarrow p$  for  $h \rightarrow 0$ , the code is verified.

The methodology is then generalized to PIC algorithms. This is done by accounting for the statistical noise that intrinsically affects the simulation results, and providing a measure of the distance between continuous, analytical distribution functions, and finite samples of computational particles. In particular, the value of  $\epsilon_h$  is estimated averaging over several simulations carried out with different pseudorandom number generator seeds, and the statistical uncertainty affecting  $\epsilon_h$  and  $\hat{p}$  is quantified. Then, a distance to account for time-evolving marker weights if provided, proving empirically that  $\epsilon_P$  and  $\epsilon_{FF}$  still decrease as  $N^{-1/2}$  for  $N \rightarrow \infty$  when  $w_p \neq 1$ . Moreover, since the proposed norms are extremely demanding in terms of computational resources when large number of computational particles are considered, the value of  $d_i^P$  is approximated with a Monte-Carlo approach and  $\epsilon_{MC}$  is used in verifying the PIC simulation code, allowing us to considerably decrease the computational cost of a PIC code verification. Finally, the norms  $\epsilon_x$  and  $\epsilon_v$  are introduced, showing that it is possible to consider independently each coordinate of the phase-space when performing a PIC code verification. The latter approach is easily generalized to phase-space in more dimensions, without increasing the computational cost considerably.

The application of the proposed procedure to the GBS code allowed us to find and correct a minor bug that was generating very small numerical errors, with completely negligible influence on the previous GBS results. This shows the power of the proposed method-

ology. The final result of the study described herein is that the implementation of the physical model in the GBS code has been completely and rigorously verified, ensuring the correct solution of the model equations. As a matter of fact, the verification exercise largely increases the confidence on the numerical results obtained using the GBS code. Finally, the application of the code verification procedure to a one-dimensional, electrostatic, collisionless PIC simulation code allowed us to investigate the peculiarities of the proposed methodology, showing how to perform a rigorous PIC code verification. This methodology can be easily generalized to more complex geometries and more realistic systems, providing the basis to perform a rigorous verification of complex PIC codes.

# CHAPTER 3

## Solution verification

Due to the finite computational power available to perform simulations and, consequently, the finite precision achievable, the simulation results are always affected by numerical errors, even if the model equations are implemented correctly. Estimating the amplitude of these errors is a crucial issue, not only to ensure the reliability of the numerical results, but also to quantify the uncertainty of the simulations when performing a rigorous validation of the physical model against experimental results (see e.g. Chapter 7). The evaluation of the numerical error affecting the simulation results is the objective of the solution verification procedure [26, 27, 96, 97].

Simulation results are affected by round-off, iterative, statistical sampling, and discretization errors [26]. The sum of these errors constitute the numerical uncertainty affecting the simulation results. Round-off errors are due to the finite number of digits that computers use when representing numerical values. Assuming that all the computations are performed in double precision, round-off errors are usually negligible with respect to the other sources of errors (we assume that this is the case in the remainder). Iterative errors are due to the use of iterative numerical schemes terminated with a finite residue. This source of error can be reduced by increasing the number of iterations and it is neglected here. The statistical sampling errors entering, for example, in the evaluation of time-averaged quantities used for code validation can be reduced or eliminated performing averages on steady-state simulations over a sufficiently long time interval. In the following we assume that simulations are in steady-state and that long enough time are considered, and we consequently neglect these errors. On the other hand, statistical sampling errors affecting PIC simulations due to the random initialization of markers cannot be neglected. Also discretization errors introduced by the numerical scheme used to discretize the physical model are usually not negligible, both for grid-based and PIC simulation codes, and they should be rigorously estimated.

We note that the analysis of the simulation results is generally performed using post-processing tools (e.g. the linear growth rate of an instability is usually obtained with an exponential fit). The solution verification procedure should also quantify the numerical

uncertainty introduced by these tools.

As an a priori study of the numerical scheme to obtain an analytical expression characterizing the behavior of the error is, most of the time, extremely complex to perform, we use an a posteriori method to compute the numerical error affecting the simulations. This requires an estimate of the analytical solution, which in most cases is not known. In Section 3.1 we discuss a rigorous methodology for estimating the statistical uncertainties affecting the numerical results. Then, in Section 3.2, using the Richardson extrapolation [45] as a higher order estimator of the exact solution and Roache's grid convergence index (GCI) [47] as a relative numerical uncertainty estimate, we provide an evaluation of the discretization error. The methodology is then applied in Section 3.3 to GBS and to the one-dimensional PIC code presented in Chapter 2 to provide an estimate of the numerical error affecting the simulations carried out with these two codes. The results discussed in the present chapter are published in Refs. [54,55].

### 3.1 Statistical error

While for grid-based codes the statistical sampling error can be reduced to negligible values by performing sufficiently long steady-state simulations (note that this is true only if the average of a quantity of interest is well defined, we assume that it is the case in the following), one should rigorously evaluate the statistical error introduced in PIC codes by a finite number of computational particles initialized using pseudorandom number generators, or by operators based on pseudorandom number generators (e.g. when a collision term is added to the Vlasov equation). To estimate the statistical uncertainty affecting  $X_h$ , where  $X_h$  is a point-by-point solution value or a solution functional evaluated from a PIC simulation with discretization parameter  $h$ , we proceed as follows. We repeat the simulation  $n_s$  times with the same  $h$ , but changing the pseudorandom number generator seed, and we define

$$\bar{X}_h = \frac{1}{n_s} \sum_{i=1}^{n_s} X_{h,i}, \tag{3.1}$$

where  $X_{h,i}$  is the  $i$ -th evaluation of  $X_h$  and  $i = 1, \dots, n_s$ . Assuming that the  $X_{h,i}$  are randomly distributed from an unknown probability distribution with unknown but finite mean  $\mu_{X,h}$  and variance  $\sigma_{X,h}^2$ , then  $\bar{X}_h \rightarrow \mu_{X,h}$  for  $n_s \rightarrow \infty$ . Moreover, according to the central limit theorem, the distribution of  $\bar{X}_h$  converges to the normal distribution with mean  $\mu_{X,h}$  and variance  $\sigma_{X,h}^2/n_s$  for  $n_s \rightarrow \infty$ . Therefore, for  $n_s \rightarrow \infty$ ,

$$\mu_{X,h} - \frac{1.96\sigma_{X,h}}{\sqrt{n_s}} \leq \bar{X}_h \leq \mu_{X,h} + \frac{1.96\sigma_{X,h}}{\sqrt{n_s}} \tag{3.2}$$

with probability equal to 0.95. As a consequence,  $\bar{X}_h$  can be used as an estimator of  $X_h$ , and we compute the uncertainty on this value as

$$\Delta X_h^{stat} = \frac{1.96\sigma_{X,h}}{\sqrt{n_s}}. \quad (3.3)$$

We remark that the unknown  $\sigma_{X,h}^2$  can be estimated according to

$$\sigma_{X,h}^2 = \frac{1}{n_s - 1} \sum_{i=1}^{n_s} (X_{h,i} - \bar{X}_h)^2. \quad (3.4)$$

Equations (3.1) and (3.3) provide a rigorous estimate of  $X_h$  and of its statistical uncertainty. However, due to the high computational cost of PIC simulations,  $n_s$  is typically low. To still have a realistic estimate of the statistical uncertainty, one can run  $n_s$  simulations with a smaller number of particles,  $N' < N$ , and evaluate the corresponding variance,  $(\sigma_{X,h'})^2$ , according to Eq. (3.4). Then, assuming that the statistical uncertainty is proportional to  $N^{-1/2}$ , the statistical error for a single simulation carried out with  $N$  particles can be estimated as

$$\Delta X_h^{stat} = 1.96\sigma_{X,h'}\sqrt{\frac{N'}{N}}. \quad (3.5)$$

## 3.2 Discretization error

In the early 20th century, Richardson developed a method [45,46], later extended [98,99], to accelerate the rate of convergence of a numerical sequence. This method is based on the use of two numerical solutions obtained using two different meshes,  $X_h$  and  $X_{rh}$ , to compute a new solution that presents a convergence rate that is, in general, one order higher than the original solution. Defining the Richardson extrapolation as

$$\hat{X} = \bar{X}_h + \frac{\bar{X}_h - \bar{X}_{rh}}{r^p - 1}, \quad (3.6)$$

then  $|X - \hat{X}| = \mathcal{O}(h^{p+1})$  [45,46], where  $X$  is the exact solution of the physical model and  $p$  is the order of accuracy of the numerical scheme defined in Chapter 2 (i.e.  $\hat{X}$  converges to  $X$  faster than  $\bar{X}_h$  for  $h \rightarrow 0$ ). Consequently, we can use  $\hat{X}$  as higher order estimator of  $X$  and approximate the discretization error as

$$\Delta X_h^{disc} \simeq \|\bar{X}_h - \hat{X}\| = \left\| \frac{\bar{X}_{rh} - \bar{X}_h}{r^p - 1} \right\|, \quad (3.7)$$

and the relative discretization error (RDE) as

$$RDE = \frac{\bar{X}_h - X}{X} \simeq \frac{\bar{X}_h - \hat{X}}{\hat{X}} = \frac{\bar{X}_{rh} - \bar{X}_h}{\bar{X}_h r^p - \bar{X}_{rh}}. \quad (3.8)$$

We remark that, for  $\hat{X}$  to be a reasonable estimate of  $X$ , several assumptions should be satisfied [26]. First, the Richardson extrapolation method requires that the degree of mesh refinement can be represented solely by the parameter  $h$ . Second, the simulations used to evaluate  $\hat{X}$  should be in the asymptotic regime, that is  $p \simeq \hat{p}$ , where

$$\hat{p} = \frac{\ln \left[ \left( \bar{X}_{r^2h} - \bar{X}_{rh} \right) / \left( \bar{X}_{rh} - \bar{X}_h \right) \right]}{\ln(r)}. \quad (3.9)$$

This may result in computationally very expensive simulations, due to the potential need for very fine meshes. Third, it is required that the solutions are smooth enough and do not present singularities and/or discontinuities. More precisely, to allow the expansion of the numerical error in powers of the parameter  $h$ , the derivatives of the analytical solution should exist and be continuous. Finally, we note that we do not have any guarantee that the Richardson extrapolated solution will meet the same governing equations satisfied by either the numerical solution or the analytical solution. Consequently we use this extrapolation for the computation of the numerical error only.

Since it may be demanding to satisfy the requirement of being in the asymptotic regime, Ref. [47] introduces the GCI, defined as

$$\text{GCI} = \frac{F_s}{r^{\tilde{p}} - 1} \left| \frac{\bar{X}_{rh} - \bar{X}_h}{\bar{X}_h} \right|, \quad (3.10)$$

that represents another estimate of the relative discretization error affecting the simulation results. The GCI is obtained by approximating in Eq. (3.8)  $\bar{X}_h r^p - \bar{X}_{rh} \simeq (r^{\tilde{p}} - 1) \bar{X}_h$ . The parameters  $F_s$  and  $\tilde{p}$  ensure that the GCI is larger than the numerical discretization error in 95% of the cases, and are defined as follows: if the difference between  $p$  and  $\hat{p}$  is less than 10%, the simulations are assumed to be in the asymptotic regime and  $F_s = 1.25$  and  $\tilde{p} = p$ . If the difference between  $p$  and  $\hat{p}$  is larger than 10%, a more conservative factor of safety,  $F_s = 3$ , is used and  $\tilde{p} = \min[\max(0.5, \hat{p}), p]$ . If  $\hat{p}$  is not evaluated (for example, if only two solutions are available),  $F_s = 3$  and  $\tilde{p} = p$ . We note that there is still an ongoing discussion in the verification community about the generality of these estimates.

To conclude our presentation of the discretization error estimate methodology, we discuss a few details. First of all, we draw the attention to the fact that the presented procedure can be applied not only to point-by-point solution values, but also to solution functionals. This is important for the use of this methodology to estimate the numerical error affecting the observables used in the validation of the physical model [27]. Second, as  $X_h$  and  $X_{rh}$  are in general computed on different meshes, the results on the coarser mesh have to be interpolated on the finest grid, using an interpolation scheme whose order is equal or higher than the order of the numerical scheme used by the code. A complete discussion of this topic is found in Ref. [98]. Finally, we illustrate a useful propriety of the GCI, that is the possibility of computing the overall GCI analysing each coordinate of the problem independently. As it can result numerically very expensive to perform a uniform refinement of the grid along all the coordinates at the same time, it is

---

### 3.3. Examples of application of the solution verification methodology

---

possible to refine separately each coordinate of the mesh by a factor  $r_i$ , where the index  $i$  refers to the coordinate under investigation. This allows us to compute a  $GCI_i$  and a  $\tilde{p}_i$  for the  $i$  coordinate, and obtain the overall GCI as  $GCI = \sum_i GCI_i$ .

## 3.3 Examples of application of the solution verification methodology

Since estimating the numerical error affecting a simulation is needed not only to ensure the reliability of the numerical results, but also to perform a rigorous validation of the physical model, the quantification of the numerical error constitutes a fundamental process of the verification methodology. In this section we apply the methodology presented above to GBS and to the PIC code illustrated in Section 2.3.2 to exemplify the procedure and assess the reliability of the simulation results.

### 3.3.1 Application of the solution verification methodology to GBS

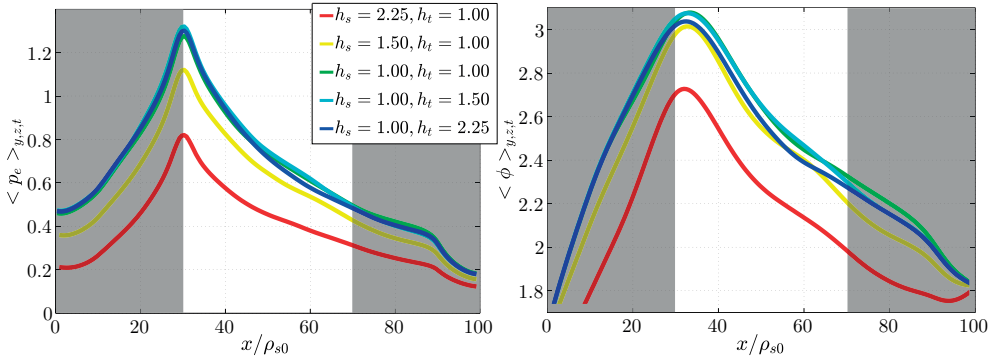
In the present section we focus on simulating the SOL plasma turbulence of a high-field side limited tokamak, considering the electrostatic, cold ion, GBS model discussed in Section 2.3.1, with  $q = 4$ ,  $\nu = e^2 n_0 R_0 / (m_i \sigma_{\parallel} c_{s0}) = 0.1$ ,  $m_e/m_i = 200$ ,  $L_y = 400$ ,  $R_0 = 500$ , and  $\chi_{\parallel e} = 0$  (see Ref. [100] for the physical investigation of those results). Since the GBS code is used to study quantities like temporal and spatial averages of  $v_{\parallel i}$  and  $v_{\parallel e}$ ,  $\langle v_{\parallel i} \rangle$  and  $\langle v_{\parallel e} \rangle$ , time-averaged radial profiles, e.g. of  $p_e = nT_e$  and  $\phi$ , and the equilibrium pressure gradient length,  $L_p = -p_e / \nabla p_e$  (see e.g. Chapters 6 and 7), in the following we evaluate the numerical errors affecting these quantities. We note that their time-average is done in the time interval  $40 \leq t \leq 80$ , during which the turbulence is in a quasi-steady state (except  $n$ , which still shows a secular trend, although relatively weak, and  $v_{\parallel i}$  and  $v_{\parallel e}$ , which present an even weaker secular trend). This allows to neglect the statistical uncertainty affecting the simulation results, and to focus our attention on discretization errors.

To apply the methodology described in Section 3.2, we analyse separately the spatial and the temporal coordinates. More precisely, in order to obtain the Richardson extrapolation, Eq. (3.6), and to compute the observed order of accuracy, Eq. (3.9), we execute five simulations using five different meshes: starting from the most refined mesh (in space and time) we perform two subsequent spatial grid coarsenings by  $r_s = 1.5$ ; the remaining two meshes result from the subsequent multiplication of the time step by 1.5 ( $r_t = 1.5$ ), without any change of the spatial grid. Hence, the meshes of the five simulations, listed in the first column of Table 3.1, are characterized by two parameters,  $h_s = \Delta y / \Delta y_0 = \Delta x / \Delta x_0 = \Delta z / \Delta z_0$ , that describes the discretization in the spatial coordinates, and  $h_t = \Delta t / \Delta t_0$ , that defines the degree of refinement of the time step. It

### Chapter 3. Solution verification

**Table 3.1** – Values of  $L_p$ ,  $\langle v_{\parallel i} \rangle$  and  $\langle v_{\parallel e} \rangle$  computed on five different meshes, as characterized by  $h_s$  and  $h_t$ .

Grid ( $n_y \times n_x \times n_z$ )	Time Step	$h_s$	$h_t$	$L_p$	$\langle v_{\parallel i} \rangle$	$\langle v_{\parallel e} \rangle$
192x80x24	$3.00 \cdot 10^{-5}$	2.25	1.00	25.56	-0.039	0.066
288x120x36	$3.00 \cdot 10^{-5}$	1.50	1.00	27.22	-0.067	0.091
432x180x54	$3.00 \cdot 10^{-5}$	1.00	1.00	27.22	-0.070	0.100
432x180x54	$4.50 \cdot 10^{-5}$	1.00	1.50	26.67	-0.071	0.100
432x180x54	$6.75 \cdot 10^{-5}$	1.00	2.25	23.89	-0.069	0.107



**Figure 3.1** – Radial profiles of  $p_e$  (left panel) and  $\phi$  (right panel), averaged over time and along the toroidal and poloidal directions, for five meshes, as characterized by  $h_s$  and  $h_t$ .

is  $\Delta y_0 = 0.93$ ,  $\Delta x_0 = 0.56$ ,  $\Delta z_0 = 0.12$ , and  $\Delta t_0 = 3.0 \cdot 10^{-5}$ .

For the five simulations considered, the quantities of interest are listed in Table 3.1 ( $L_p$ ,  $\langle v_{\parallel i} \rangle$ , and  $\langle v_{\parallel e} \rangle$ ) and shown in Fig. 3.1 (radial profiles of  $p_e$  and  $\phi$ ). We note that  $\langle v_{\parallel i} \rangle$  and  $\langle v_{\parallel e} \rangle$  are computed taking the average of the parallel velocities over the entire spatial domain of interest. The radial profiles of  $p_e$  and  $\phi$  are obtained taking the average of these quantities along the poloidal and toroidal directions; the equilibrium pressure gradient length is computed as the radial distance between the maximum value of the radial profile of  $p_e$  and the half of its maximum value.

The results presented in Table 3.1 and in Fig. 3.1 show that the differences of the various quantities computed on the meshes characterized by  $h_t = 1.00, 1.50, 2.25$  are very small, if compared to the changes due to the spatial discretization. The only quantity presenting a meaningful dependence on the time step is  $L_p$ ; for this quantity we apply the methodology described in Section 3.2, finding  $\hat{p}_t = 3.97$  and  $\text{GCI}_t = 0.6\%$  (here,  $\text{GCI}_t$  is referred to the most refined mesh; the  $\text{GCI}_t$  value relative to the other meshes is obtained by multiplying the  $\text{GCI}_t$  of the most refined mesh by  $h_t^4$ , and similarly for the spatial discretization). The observed order of accuracy is remarkably close to the expected  $p_t = 4$ , and the resulting numerical error is very small. Therefore, in the following, we neglect the numerical error associated to the time discretization with respect to the one due to spatial discretization.

The evaluation of the numerical error due to the spatial discretization affecting the quantities of interest is summarized in Table 3.2. We start our analysis by considering  $\langle v_{\parallel i} \rangle$



### 3.3. Examples of application of the solution verification methodology

**Table 3.2** – Values of  $GCI_s$  and  $RDE$  computed using the parameters  $\hat{p}_s$ ,  $\tilde{p}_s$  and  $F_s$ , valid for the finest mesh [ $h_s = h_t = 1.0$ ].

Field	$\hat{p}_s$	$\tilde{p}_s$	$F_s$	$GCI_s(\%)$	$RDE(\%)$
$\langle v_{\parallel i} \rangle$	3.41	2	3	20.7	-6.5
$\langle v_{\parallel e} \rangle$	2.87	2	3	20.5	-6.4
$p_e$	1.86	2	1.25	12.0	-8.8
$\phi$	3.08	2	3	7.3	-2.4

and  $\langle v_{\parallel e} \rangle$ . By applying Eq. (3.9) to these quantities, we obtain a value of  $\hat{p}_s$  larger than  $p_s$ . The difference between  $\hat{p}_s$  and  $p_s$  is probably due to the fact that, in the present scenarios, the parallel velocities average to very small quantities, if compared to the local value of  $v_{\parallel i}$  and  $v_{\parallel e}$  (even one order of magnitude lower). Therefore,  $\langle v_{\parallel i} \rangle$  and  $\langle v_{\parallel e} \rangle$  are very sensitive quantities; this can lead, not surprisingly, to a difference between  $\hat{p}_s$  and  $p_s$ . Using Eq. (3.10) and the conservative value  $F_s = 3$ , the resulting  $GCI_s$  are relatively large. The analysis of the radial profiles of  $p_e$  and  $\phi$  is very similar: depending on the difference between  $\hat{p}_s$  and  $p_s$  we choose the corresponding value of  $F_s$  and  $\tilde{p}_s$ , by which we compute the  $GCI_s$ . At a mesh similar to the one generally used for GBS simulations, we find that the numerical error affecting these quantities is of the order of 20 – 25%. Finally, we note that the differences between the values of  $L_p$  computed on the three meshes characterized by  $h_s = 1.00, 1.50, 2.25$  are of the same order of the spatial grid size ( $\Delta x = 0.56$  for  $h_s = 1.0$ ) and, therefore, below the numerical error necessary to perform the Richardson extrapolation (in fact,  $\Delta x$  is the intrinsic uncertainty on  $L_p$ ; therefore, it is not possible to distinguish between two values of  $L_p$  whose difference is below or equal to  $2\Delta x$ ). It follows that we can assume the numerical error affecting  $L_p$  comparable to the spatial grid size.

Several observations should be addressed to our results. First, the quantities are clearly converging at a rate that is typically not very different from the expected one. Second, our analysis allows us to estimate the numerical error affecting the different quantities of interest; this will be used in the future to choose the mesh refinement necessary to achieve the desired accuracy. In any case, it is reassuring that, the correct qualitative behaviour is retrieved by all GBS simulations even at the coarser meshes. Third, as the value of  $F_s = 3$  is generally thought to be conservative, our estimate of the numerical error is quite safe. Finally, throughout our solution verification, we assumed that statistical errors are negligible. As a matter of fact, the most refined simulation is computationally extremely expensive and we could not verify this assumption. Moreover, as previously pointed out,  $n$  is not in perfect steady state; it is possible that the value of the GCI is reduced by considering longer time intervals.

### 3.3.2 Application of the solution verification methodology to a PIC simulation

In order to illustrate a practical example of application of the solution verification methodology discussed above to a PIC simulation, we consider here the two-stream instability. This textbook plasma instability is ideally studied by using PIC codes.

We consider the distribution function  $f(x, v) = f_x(x)f_v(v)$ , where  $f_x(x) = 1/L$  and  $f_v(v) = [\delta(v - v_0) + \delta(v + v_0)]/2$ , with  $\delta(v)$  the Dirac function. The dispersion relation associated with small amplitude perturbations is

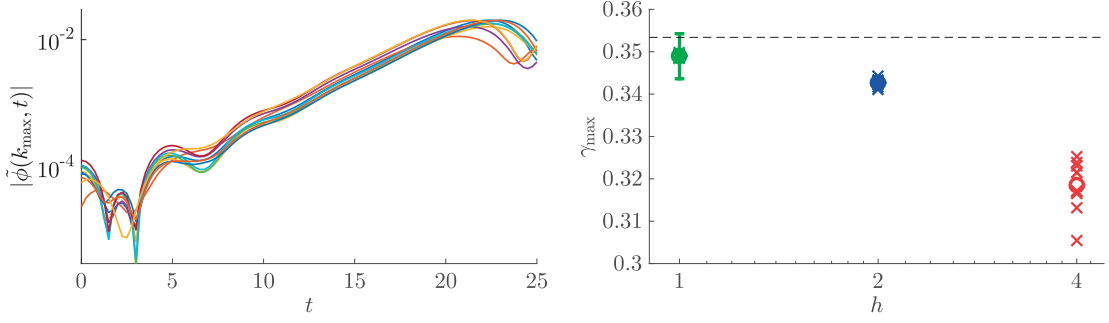
$$D(\omega, k) = 1 - \frac{1}{2} \left[ \frac{1}{(\omega - kv_0)^2} + \frac{1}{(\omega + kv_0)^2} \right]. \quad (3.11)$$

Since for  $0 < k < 1/v_0$  the  $\omega$  solution of  $D(\omega, k) = 0$  is complex, the system is affected by an instability called two-stream instability. As a consequence, if the system is perturbed, small amplitude modes can grow exponentially, before saturating due to nonlinear effects. The fastest growing mode, with growth rate  $\gamma_{\max} = 1/\sqrt{8}$ , is obtained for  $k_{\max} = \sqrt{3/(8v_0^2)}$ .

To numerically compute the linear growth rate of the two-stream instability, we proceed as follows. First, we initialize the system according to a distribution function  $f = [f_x + A \cos(k_{\max}x)] f_v$ , where  $A \ll 1$  is used to seed the perturbation, and we carry out a simulation with the PIC code detailed in Section 2.3.2. Second, we compute the Fourier transform of  $\phi(x, t)$  along  $x$ , thus obtaining  $\tilde{\phi}(k, t)$ . Third, we identify the time interval during which the mode  $\tilde{\phi}(k_{\max}, t)$  grows exponentially. Finally, over the identified time interval, we fit the amplitude of the mode,  $|\tilde{\phi}(k_{\max}, t)|$ , with an exponential curve to evaluate  $\gamma_{\max}$ .

We now apply the solution verification methodology discussed above to rigorously estimate  $\gamma_{\max}$  and its numerical uncertainty  $\Delta\gamma_{\max}$ . We perform three sets of ten simulations for  $h = \Delta x/\Delta x_0 = \Delta t/\Delta t_0 = (N/N_0)^{-1/4} = 1, 2, 4$ , with different pseudorandom number generator seed. We consider  $L = 2\pi$ ,  $v_0 = 0.2$ ,  $\Delta x_0 = L/128$ ,  $\Delta t_0 = 1/16$ , and  $N_0 = 2.048 \cdot 10^9$ , for which we expect  $k_{\max} \simeq 3$  and  $\gamma_{\max} \simeq 0.353$ . The time evolution of  $|\tilde{\phi}(k_{\max}, t)|$  is shown in Fig. 3.2 (left panel) for  $h = 4$ . After an initial transient, the mode grows until  $t \simeq 18$ , before saturating because of nonlinear effects. We exponentially fit each profile in the time interval  $15 \leq t \leq 17$  to obtain the growth rates  $\gamma_{\max}$  that we plot in Fig. 3.2 (right panel, red crosses). The same process is repeated for  $h = 1$  (green crosses) and  $h = 2$  (blue crosses). It is noticeable that the spreading of the growth rates is smaller at smaller  $h$  (i.e. larger  $N$ ).

To compute  $\Delta\gamma_{\max}$  we estimate separately the uncertainty introduced by the post-processing (i.e. the exponential fit),  $\Delta\gamma_{\max}^{fit}$ , the statistical uncertainty,  $\Delta\gamma_{\max}^{stat}$ , and the discretization error,  $\Delta\gamma_{\max}^{disc}$ . First, the uncertainty introduced by the exponential fit is  $\Delta\gamma_{\max,1}^{fit} \simeq 0.003$ , that is the confidence interval of the fit for the simulations with  $h = 1$ . Second, applying Eq. (3.1), we compute the values of  $\bar{\gamma}_{\max,h}$  averaged over the set of simulations for  $h = 1, 2, 4$ , which are represented in Fig. 3.2, right panel, as circles.



**Figure 3.2** – Time evolution of  $|\tilde{\phi}(k_{\max}, t)|$  for  $h = 4$  (left panel) and growth rates of the two-stream instability for  $h = 1, 2, 4$  (right panel). The circles in the right panel represent  $\bar{\gamma}_{\max, h}$ , while the error bars represent  $\Delta\gamma_{\max}$ . The dashed line represents the expected value of  $\gamma_{\max}$

We note that, as discussed in Section 3.1, the variance of  $\gamma_{\max, 1}$  can be estimated as  $\sigma_{\gamma, 1} \approx \sigma_{\gamma, 4}/16 = 3.7 \cdot 10^{-4}$ . This is close to  $\sigma_{\gamma, 1} = 3.9 \cdot 10^{-4}$ , obtained with Eq. (3.4). We therefore obtain  $\Delta\gamma_{\max, 1}^{stat} = 0.0002$ .

Third, the discretization error is obtained by applying the methodology discussed in Section 3.2. In particular, using the three estimates of  $\gamma_{\max, h}$  obtained by averaging over the 10 simulations, which are  $\bar{\gamma}_{\max, h} = 0.349, 0.343, 0.318$  for  $h = 1, 2, 4$ , respectively, we compute the Richardson extrapolation  $\bar{\gamma}_{\max} = \bar{\gamma}_{\max, 1} + (\bar{\gamma}_{\max, 1} - \bar{\gamma}_{\max, 2})/3 = 0.351$  according to Eq. (3.6). We also compute the observed order of accuracy according to Eq. (3.9), obtaining  $\hat{p} = 1.96$ , thus ensuring that the Richardson extrapolation is a reasonable estimate of the exact solution. The discretization error is thus computed according to Eq. (3.7), obtaining  $\Delta\gamma_{\max, 1}^{disc} = 0.002$ .

Finally,  $\Delta\gamma_{\max}$  is obtained by summing up the uncertainty introduced by the exponential fit, the statistical uncertainty, and the discretization error,  $\Delta\gamma_{\max} = \Delta\gamma_{\max, 1}^{fit} + \Delta\gamma_{\max, 1}^{stat} + \Delta\gamma_{\max, 1}^{disc}$ , obtaining  $\Delta\gamma_{\max} \simeq 0.005$ . Comparing the value of  $\bar{\gamma}_{\max, 1} = 0.349$  with the expected value  $\gamma_{\max} \simeq 0.353$ , it results that the numerical evaluation of  $\gamma_{\max}$  is consistent with the exact solution within the numerical uncertainty.

## 3.4 Conclusions

In the present chapter we discuss the methodology for performing a rigorous solution verification of plasma simulations, proposing a concrete approach for quantifying the statistical uncertainty and the discretization error. Since PIC simulations are intrinsically affected by non-negligible statistical uncertainties, these are quantified by repeating the simulation with different pseudorandom number generator seeds.

The procedure we propose for estimating the discretization error, based on the use of the Richardson extrapolation as an estimate of the exact solution, is definitely valid for simulations belonging to the asymptotic regime. For simulations not belonging to the asymptotic regime, the GCI still allows to estimate the numerical uncertainty. The total numerical uncertainty affecting the simulation results is then computed by summing up

the different contributions.

The application of the proposed procedure to GBS allowed us to estimate the amplitude of the numerical error affecting GBS simulations, useful not only to ensure the reliability of the simulations, but also to perform the validation of the code results against experimental data (see e.g. Chapter 7). The application of the proposed procedures to a one-dimensional, electrostatic, collisionless PIC simulation allowed us to exemplify the procedure used for the quantification of statistical uncertainties.

# CHAPTER 4

## Uncertainty propagation

In order to assess the reliability of a simulation and rigorously validate it against experimental measurements, it is crucial to estimate the uncertainties affecting the numerical results. The uncertainties stem from numerically solving the model equations with finite precision, or from the use of input parameters that are not precisely known or accurately measured. In Chapter 3 a methodology to evaluate the numerical uncertainty is discussed. In the present chapter we focus on the propagation of input uncertainties through the model equations.

An analytical study of uncertainty propagation is unfeasible for complex physical models such as the ones describing plasma turbulence. In these cases uncertainty propagation is approached numerically. The simplest strategy to study uncertainty propagation is based on the assumption that the uncertainty on an input parameter is described by a probability distribution [101]. A sample of input parameters is then randomly generated according to such a distribution and a simulation is performed for each input of the sample. A distribution of simulation results is thus obtained. From this distribution it is possible to evaluate the uncertainty affecting the point-by-point solution values or solution functionals. While conceptually simple, this approach is usually not applicable to plasma turbulence simulations because of the high computational cost and of the large number of input parameters typically involved. Despite the fact that sophisticated procedures have been developed to predict the response of the model to variation of input parameters using the smallest possible number of simulations, such as Bayesian analysis [101], multifidelity Monte-Carlo estimations [102], and the response surface methodology [103], to our knowledge they have never been employed by the plasma physics community. Rather, uncertainty propagation is typically investigated in plasma physics by performing sensitivity scans (see e.g. Ref. [104]). More precisely, the input parameters,  $x_i$ , which the code is more sensitive to, are identified. Then, one simulation is performed by using the input parameters at their reference values  $x_i = \bar{x}_i$ . In addition, two more simulations are run for each identified input parameter, using  $x_i = \bar{x}_i \pm \epsilon_i$ , with  $\epsilon_i$  the uncertainty on the reference value  $\bar{x}_i$ . Finally, the uncertainty is identified

assuming a linear dependence of the simulation results on the input parameters. While computationally less demanding than a Bayesian analysis, this approach is still considerably expensive, particularly for plasma turbulence simulations involving a large number of input parameters. Moreover, the solution of differential equations practically never depends linearly on the input parameters [101]. A more rigorous approach for analysing uncertainty propagation is therefore needed.

To overcome these issues and estimate the dependence of the code results on the input parameters, we develop a methodology based on a decomposition of the model equation solution in terms of Chebyshev polynomials along the input parameter, time, and spatial coordinates. More precisely, a series of Chebyshev polynomials is used to represent the solution of a differential equation and to express its dependence on all the temporal, spatial, and input variables. A weighted residual method (WRM) is then employed to deduce a set of algebraic equations, thus making it possible to numerically evaluate the coefficients appearing in the Chebyshev decomposition and obtaining a semi-analytical expression for the solution with explicit dependence on the input parameters. This allows determining the parametric dependence of the solution, avoiding to perform a set of simulations for different input values. Moreover, thanks to the *minmax* property characterizing Chebyshev polynomials [105], this approach allows estimating the parametric dependence of the model equation solution with a small number of spectral terms and, therefore, with reduced computational cost. By applying the proposed methodology to a two-dimensional drift-reduced Braginskii model previously used to investigate the plasma dynamics in the TORPEX device [104] and in the tokamak SOL [15, 106], and employed in Chapters 5 and 7 to study the TORPEX and the TCV SOL plasma dynamics, we assess the influence of input parameter uncertainty on the model results.

We note that the use of fully spectral methods to solve differential equations is far from new, as they have been widely employed by the computational fluid dynamics community (see e.g. Ref. [107]). However, fully spectral codes were rarely used to investigate plasma physics problems and their use was limited to the study of plasma flows and linear stability analysis (see e.g. Refs. [108, 109]). In fact, to our knowledge, fully spectral methods were never applied to the study of uncertainty propagation in nonlinear plasma turbulence simulations. This motivates the study illustrated in the present chapter.

This chapter is structured as follows. In Section 4.1 the spectral method we employ to solve differential equations is presented. More precisely, we discuss the approximation of the model equation solution with Chebyshev polynomials, the application of the WRM, the approximation of differential and nonlinear operators, and the treatment of initial and boundary conditions in the Chebyshev spectral domain. Then, in Section 4.2 we discuss the two-dimensional drift-reduced Braginskii model and we apply the Chebyshev spectral method to study the uncertainty propagation through the model equations. The conclusions follow.

## 4.1 Chebyshev spectral method

In this section we discuss the use of the WRM to approximate a differential equation with a set of algebraic equations in the Chebyshev spectral domain. First, we introduce the WRM for an arbitrary set of basis functions. Then, we discuss some interesting features of the WRM when Chebyshev polynomials are used as basis functions. Finally, we discuss the approximation of differential and nonlinear operators, and the treatment of the initial and the boundary conditions in the Chebyshev spectral domain.

### 4.1.1 Weighted residual method

The WRM is used to solve differential equations by approximating their solution with a linear combination of linearly independent basis functions. A set of algebraic equations for the coefficients of each corresponding basis function is then deduced, whose solution globally minimizes the distance between the exact solution and the linear combination of basis functions for a given norm [110].

Formally, we consider an initial value parabolic or hyperbolic partial differential equation

$$\frac{\partial \mathbf{u}}{\partial t} = D\{\mathbf{u}\} + \mathbf{S}, \quad (4.1)$$

with the exact solution  $\mathbf{u} = \mathbf{u}(t, \mathbf{x}; \mathbf{p})$  that depends on time  $t \in [t_0, t_1]$ , space coordinates  $\mathbf{x} \in [\mathbf{x}_0, \mathbf{x}_1]$ , and a set of parameters  $\mathbf{p} \in [\mathbf{p}_0, \mathbf{p}_1]$  (the indexes 0 and 1 referring to the lower and upper boundaries of the considered domains). Here  $D$  is a linear or nonlinear differential operator acting on  $\mathbf{u}$  and  $\mathbf{S} = \mathbf{S}(t, \mathbf{x}; \mathbf{p})$  a given source term. Equation (4.1) is completed by an initial condition  $\mathbf{u}(t_0, \mathbf{x}; \mathbf{p}) = \mathbf{u}_0(\mathbf{x}; \mathbf{p})$ , with  $\mathbf{u}_0$  a given function, and a set of boundary conditions for the spatial domain. In order to use the WRM to solve Eq. (4.1), we approximate

$$\mathbf{u}(t, \mathbf{x}; \mathbf{p}) \simeq \tilde{\mathbf{u}}(t, \mathbf{x}; \mathbf{p}) = \sum_{k=0}^K \sum_{l=0}^L \sum_{m=0}^M \mathbf{a}_{klm} \xi_k^t(t) \xi_l^x(\mathbf{x}) \xi_m^p(\mathbf{p}), \quad (4.2)$$

where  $\xi_k^t$ ,  $\xi_l^x$ , and  $\xi_m^p$  are the basis functions chosen for the expansion of  $\mathbf{u}$ ,  $\mathbf{a}_{klm}$  constant coefficients, and  $K+1$ ,  $L+1$ ,  $M+1$  the numbers of basis functions used for the expansion in time, real space, and input parameter space, respectively. Integrating Eq. (4.1) in time, i.e. writing

$$\mathbf{u}(t, \mathbf{x}; \mathbf{p}) = \mathbf{u}_0(\mathbf{x}; \mathbf{p}) + \int_{t_0}^t (D\{\mathbf{u}\} + \mathbf{S}) dt', \quad (4.3)$$

we define the residual  $\mathbf{R}$  as

$$\mathbf{R}(t, \mathbf{x}; \mathbf{p}) = \tilde{\mathbf{u}}(t, \mathbf{x}; \mathbf{p}) - \left[ \mathbf{u}_0(\mathbf{x}; \mathbf{p}) + \int_{t_0}^t (D\{\tilde{\mathbf{u}}\} + \mathbf{S}) dt' \right]. \quad (4.4)$$

For estimating the coefficients  $\mathbf{a}_{klm}$ , the equation

$$\int_{t_0}^{t_1} \int_{\mathbf{x}_0}^{\mathbf{x}_1} \int_{\mathbf{p}_0}^{\mathbf{p}_1} \mathbf{R}(t, \mathbf{x}; \mathbf{p}) W_{klm}(t, \mathbf{x}; \mathbf{p}) dt d\mathbf{x} d\mathbf{p} = 0 \quad (4.5)$$

is then solved for all  $0 \leq k \leq K$ ,  $0 \leq l \leq L$ ,  $0 \leq m \leq M$ , with  $W_{klm}(t, \mathbf{x}; \mathbf{p})$  a set of  $(K+1)(L+1)(M+1)$  properly chosen weight functions.

### 4.1.2 Weighted residual method in Chebyshev spectral domain

For simplicity, in the following of this section we consider a one dimensional function  $u = u(t, x; p)$  depending on one spatial dimension  $x$  and one input parameter  $p$ . The generalization to equations with more dimensions and parameters does not present conceptual difficulties. We choose the Chebyshev polynomials of the first kind  $T_n(x) = \cos(n \cos^{-1} x)$ , defined for  $x \in [-1, 1]$ , as basis functions. Hereafter  $T_n$  are simply named Chebyshev polynomials. We note that  $n \in \mathbb{N}$  constitutes the degree of the Chebyshev polynomial. Chebyshev polynomials are orthogonal over the weight function  $w(x) = (1 - x^2)^{-1/2}$  in the interval  $x \in [-1, 1]$ , i.e.

$$\int_{-1}^1 \frac{T_m(x) T_n(x)}{\sqrt{1-x^2}} dx = \frac{\pi}{2} (\delta_{mn} + \delta_{m0} \delta_{n0}), \quad (4.6)$$

where  $\delta_{mn}$  is the Kronecker delta ( $\delta_{mn} = 1$  if  $m = n$  and  $\delta_{mn} = 0$  otherwise), and are characterized by the *minmax* property, i.e. the expansion of a continuous function  $f(x) \simeq \sum_{l=0}^L a_l T_l(x)$ , with

$$a_l = \frac{2}{\pi B_x} \int_{x_0}^{x_1} f(x) \frac{T_l(x)}{\sqrt{1-x^2}} dx, \quad (4.7)$$

provides the most accurate approximation of  $f$  under the maximum norm,  $\| - \|_\infty$ , for a polynomial of degree  $L$  [105]. The *minmax* property motivates the choice of using the  $T_n$  polynomials as basis functions, since it implies that the best approximation of  $f$  at order  $L$  in Chebyshev space is simply the series truncated at  $l = L$ . Consequently, we approximate

$$u(t, x; p) \simeq \sum_{k=0}^K \sum_{l=0}^L \sum_{m=0}^M a_{klm} T_k(\tau) T_l(\chi) T_m(\sigma), \quad (4.8)$$

where

$$\tau = \frac{t - A_t}{B_t}, \quad \chi = \frac{x - A_x}{B_x}, \quad \sigma = \frac{p - A_p}{B_p}, \quad (4.9)$$

with  $A_t = (t_1 + t_0)/2$  and  $B_t = (t_1 - t_0)/2$  (similar definitions apply to the other quantities), such that  $\tau, \chi, \sigma \in [-1, 1]$ . Primes on summation signs indicate that the



0-th term of each sum is multiplied by a factor 1/2, i.e.

$$\sum_{n=0}^N 'a_n T_n(\chi) = \frac{a_0}{2} T_0(\chi) + a_1 T_1(\chi) + a_2 T_2(\chi) + \dots + a_N T_N(\chi). \quad (4.10)$$

The residual  $R$  is thus given by

$$R = u(t, x; p) - \left[ u_0(x; p) + \int_{t_0}^t (D\{u\} + S) dt' \right], \quad (4.11)$$

with  $u$  approximated according to Eq. (4.8). The equations that provide the coefficients  $a_{klm}$ , Eq. (4.5), take the form

$$\int_{t_0}^{t_1} \int_{x_0}^{x_1} \int_{p_0}^{p_1} RW_{klm}(t, x; p) dt dx dp = 0, \quad (4.12)$$

where we choose

$$W_{klm}(t, x; p) = \frac{T_k(\tau)}{\sqrt{1-\tau^2}} \frac{T_l(\chi)}{\sqrt{1-\chi^2}} \frac{T_m(\sigma)}{\sqrt{1-\sigma^2}} \quad (4.13)$$

to take advantage of the orthogonality property of Chebyshev polynomials. We now express Eq. (4.12) in a form useful for further progress. Using the orthogonality property, Eq. (4.6), we have

$$\int_{t_0}^{t_1} \int_{x_0}^{x_1} \int_{p_0}^{p_1} u(t, x, p) W_{klm}(t, x; p) dt dx dp = B_t B_x B_p \left(\frac{\pi}{2}\right)^3 a_{klm}. \quad (4.14)$$

Moreover, we approximate

$$\int_{t_0}^t S(t', x; p) dt' \simeq \sum_{k=0}^K ' \sum_{l=0}^L ' \sum_{m=0}^M ' S_{klm} T_k(\tau) T_l(\chi) T_m(\sigma) \quad (4.15)$$

and

$$u_0(x; p) \simeq \sum_{l=0}^L ' \sum_{m=0}^M ' b_{lm} T_l(\chi) T_m(\sigma). \quad (4.16)$$

Finally, we write

$$\int_{t_0}^t D\{u(t', x; p)\} dt' \simeq \sum_{k=0}^K ' \sum_{l=0}^L ' \sum_{m=0}^M ' A_{klm} T_k(\tau) T_l(\chi) T_m(\sigma), \quad (4.17)$$

where  $A_{klm}$  are assumed to be known coefficients. Using again the orthogonality property of Chebyshev polynomials, Eq. (4.12) yields

$$a_{klm} = 2\delta_{k0} b_{lm} + A_{klm} + S_{klm}, \quad (4.18)$$

which is a set of  $(K + 1)(L + 1)(M + 1)$  coupled algebraic equations. Equation (4.18) approximates

$$u(t, x; p) = u_0(x; p) + \int_{t_0}^t (D\{u\} + S) dt' \quad (4.19)$$

in Chebyshev space. The term  $2\delta_{k0}b_{lm}$  represents the initial condition, and  $A_{klm}$  and  $S_{klm}$  the projection along the Chebyshev basis of the operator  $D$  applied to  $u$  and of the source term, respectively.

In order to solve Eq. (4.18) for  $a_{klm}$ , one has to deduce first an explicit relation between  $A_{klm}$  and the Chebyshev expansion of  $D\{u\}$ . Approximating

$$D\{u\} \simeq \sum_{k=0}^K \sum_{l=0}^L \sum_{m=0}^M c_{klm} T_k(\tau) T_l(\chi) T_m(\sigma), \quad (4.20)$$

we have

$$\begin{aligned} A_{klm} &= \frac{B_t}{2k} (c_{k-1,l,m} - c_{k+1,l,m}) & 0 < k \leq K - 1 \\ A_{klm} &= \frac{B_t}{2k} c_{k-1,l,m} & k = K. \end{aligned} \quad (4.21)$$

Since

$$\int_{t_0}^{t_0} D\{u(t', x; p)\} dt' = 0, \quad (4.22)$$

the coefficients  $A_{0lm}$  are found by imposing

$$A_{0lm} = -2 \sum_{k=1}^K A_{klm} (-1)^k. \quad (4.23)$$

Finally, in order to solve Eq. (4.18) for the coefficients  $a_{klm}$ , one has to express the operator  $D$  in the Chebyshev spectral domain and determine the coefficients  $c_{klm}$  as a function of  $a_{klm}$ . This is detailed in Section 4.1.3.

To conclude our discussion on the use of the WRM with Chebyshev polynomials, we would like to make two remarks. First, while Eq. (4.1) constitutes a single differential equation, Eq. (4.18) represents a set of coupled algebraic equations that can be solved, either analytically or numerically, to compute the coefficients  $a_{klm}$  and obtain an approximated semi-analytical solution of Eq. (4.19) with explicit dependence on  $p$ . Second, in general, Eq. (4.18) cannot be solved analytically if  $D$  is a nonlinear operator. In such a case, Eq. (4.18) must be solved numerically, usually employing an iterative solver. Since the latter might need a good initial guess to converge, it is suitable to decompose the time interval into sub-domains. More precisely, the time coordinate  $t \in [t_0, t_1]$  is decomposed in  $N$  sub-domains of length  $\Delta t^i > 0$ , with  $i = 1, \dots, N$  and  $t_1 - t_0 = \sum_{i=1}^N \Delta t^i$ . Equation (4.18) is then solved for  $t \in [t_0, t_0 + \Delta t^1]$  using the initial guess  $a_{klm}^1 = 2\delta_{0k}b_{lm}^1$  ( $a_{klm}^i$  and  $b_{lm}^i$  denote respectively the coefficients  $a_{klm}$  and  $b_{lm}$  corresponding to the sub-domain  $\Delta t^i$ ). Finally, for the subsequent sub-domains we impose  $b_{lm}^{i+1} = \sum_{k=0}^K a_{klm}^i$  and

we use the initial guess  $a_{klm}^{i+1} = a_{klm}^i$  for the solution of Eq. (4.18) in the sub-domain  $i + 1$ .

### 4.1.3 Operators in the Chebyshev spectral domain

In order to relate the coefficients  $a_{klm}$  in Eq. (4.8), used to represent  $u$  in the Chebyshev spectral domain, with the coefficients  $c_{klm}$  in Eq. (4.20), one has to express differential and nonlinear operators in Chebyshev space. This is performed by exploiting some useful properties of Chebyshev polynomials, as summarized in the following of this section (see Ref. [105] for a more detailed discussion). For simplicity, we consider functions of one variable  $x$ . The generalization to multiple variables does not present any conceptual difficulty.

#### Differential operators

First-order spatial derivatives of a function can be easily obtained by exploiting the property

$$\frac{dT_n}{dx}(\chi) = \frac{2n}{B_x} \sum_{\substack{l \geq 0 \\ n-l \text{ odd}}}^{n-1} 'T_l(\chi). \quad (4.24)$$

Therefore, the first-order derivative of a function  $f(x) = \sum_{l=0}^L 'a_l T_l(\chi)$  is computed as

$$\frac{df}{dx}(x) = \frac{d}{dx} \sum_{l=0}^L 'a_l T_l(\chi) = \frac{2}{B_x} \sum_{l=0}^{L-1} ' \sum_{\substack{\lambda=l+1 \\ \lambda-l \text{ odd}}}^L \lambda a_\lambda T_l(\chi). \quad (4.25)$$

Similarly, second order derivatives of  $f(x)$  are expressed as

$$\frac{d^2 f}{dx^2}(x) = \frac{d^2}{dx^2} \sum_{l=0}^L 'a_l T_l(\chi) = \frac{1}{B_x^2} \sum_{l=0}^{L-2} ' \sum_{\substack{\lambda=l+2 \\ \lambda-l \text{ even}}}^L \lambda (\lambda^2 - l^2) a_\lambda T_l(\chi). \quad (4.26)$$

Higher order derivatives are obtained iterating Eq. (4.24). We remark that, when differentiating with respect to  $x$  a function  $f(x)$  represented in terms of Chebyshev polynomials of order  $L$ , the function  $df/dx$  only includes polynomials up to order  $L - 1$ .

#### Nonlinear terms

Linear operators involving the addition or subtraction of two functions are easily handled in the Chebyshev spectral domain. On the other hand, care must be taken in computing nonlinear operators related, for example, to the multiplication of two functions. Therefore, we focus on the product operator, that is the basis of all nonlinear operators.

## Chapter 4. Uncertainty propagation

---

Considering that the product of two Chebyshev polynomials is given by

$$T_m(x)T_n(x) = \frac{1}{2} [T_{m+n}(x) + T_{|m-n|}(x)], \quad (4.27)$$

it is possible to write the product  $h(x) = f(x)g(x)$  between the two functions  $f(x) = \sum_{l=0}^L a_l T_l(\chi)$  and  $g(x) = \sum_{l=0}^L b_l T_l(\chi)$  as

$$\begin{aligned} h(x) &= \sum_{l=0}^L a_l \sum_{i=0}^L b_i \frac{1}{2} [T_{l+i}(\chi) + T_{|l-i|}(\chi)] \\ &= \sum_{l=0}^L a_l \left[ \sum_{i=l}^{L+l} \frac{a_i b_{i-l}}{2} T_i(\chi) + \sum_{i=0}^{l-1} \frac{a_l b_{l-1-i}}{2} T_i(\chi) + \sum_{i=1}^{L-l} \frac{a_l b_{i+l}}{2} T_i(\chi) \right]. \end{aligned} \quad (4.28)$$

We now approximate

$$h(x) \simeq \sum_{l=0}^L c_l T_l(\chi), \quad (4.29)$$

i.e. we truncate the expansion of  $h(x)$  at order  $L$  (because of the *minmax* property, this truncated series is the most accurate polynomial representation of  $h(x)$  to order  $L$ ). To express the coefficients  $c_l$  in terms of  $a_l$  and  $b_l$ , we impose

$$\sum_{l=0}^{\infty} c_l T_l(\chi) = \sum_{l=0}^L \left[ \sum_{i=l}^{L+l} \frac{a_i b_{i-l}}{2} T_i(\chi) + \sum_{i=0}^{l-1} \frac{a_l b_{l-1-i}}{2} T_i(\chi) + \sum_{i=1}^{L-l} \frac{a_l b_{i+l}}{2} T_i(\chi) \right] \quad (4.30)$$

and we multiply both sides of Eq. (4.30) by  $T_p(\chi)/\sqrt{1-\chi^2}$ , with  $0 \leq p \leq L$ . Then, by applying Eq. (4.6), we obtain

$$c_l = \sum_{i=0}^l \frac{a_i b_{l-i}}{2} + \sum_{i=1}^{L-l} \frac{a_i b_{i+l} + a_{i+l} b_i}{2}, \quad (4.31)$$

with  $0 \leq l \leq L$ .

### 4.1.4 Initial condition

The coefficients  $b_{lm}$  in Eq. (4.16), used to express the initial condition in Chebyshev space, are given by

$$b_{lm} = \frac{1}{B_x B_p} \left( \frac{2}{\pi} \right)^2 \int_{x_0}^{x_1} \int_{p_0}^{p_1} u_0(x; p) \frac{T_l(\chi)}{\sqrt{1-\chi^2}} \frac{T_m(\sigma)}{\sqrt{1-\sigma^2}} dx dp. \quad (4.32)$$

Since, in general, the integrals in Eq. (4.32) cannot be computed analytically, we introduce an approximation of  $b_{lm}$  that can be easily evaluated. In fact [105],

$$b_{lm} \simeq \frac{2}{L+1} \frac{2}{M+1} \sum_{r=1}^{L+1} \sum_{s=1}^{M+1} u(t_0, x_r; p_s) T_l(\chi_r) T_m(\sigma_s), \quad (4.33)$$

where

$$\chi_r = \cos \left[ \frac{\pi}{L+1} \left( r - \frac{1}{2} \right) \right], \quad \sigma_s = \cos \left[ \frac{\pi}{M+1} \left( s - \frac{1}{2} \right) \right], \quad (4.34)$$

$x_r = B_x \chi_r + A_x$ , and  $p_s = B_p \sigma_s + A_p$ . The expression in Eq. (4.33) approximates  $b_{lm}$  in the limit  $L, M \rightarrow \infty$  and it is easily generalized to continuous functions with more variables. We remark that  $u_0(x; p)$  should satisfy the spatial boundary conditions.

### 4.1.5 Boundary conditions

We now discuss how to set the boundary conditions in the Chebyshev spectral domain. To simplify the discussion, we consider differential equations where the operator  $D$  involves second order derivatives of  $u$  with respect to  $x$ . The generalization to higher order derivatives does not present any conceptual difficulty. Moreover, we consider that the same kind of boundary conditions is applied at the two boundaries of the spatial domain and we focus on Dirichlet, Neumann, and periodic boundary conditions. Combinations of different kinds of boundary conditions are easily obtained following the procedure described hereafter. We finally remark that, when a multi-dimensional spatial domain is considered, only one condition can be applied at the corners. It is generally good practice to ensure that the boundary conditions applied at the different edges are compatible at the corner points.

#### Dirichlet boundary conditions

Dirichlet boundary conditions are applied as follows. Since the expansion of  $d^2u/dx^2$  leads to polynomials up to order  $L - 2$ , the boundary conditions are set by imposing the  $L - 1$  and  $L$  coefficients of the Chebyshev expansion. Considering the two boundary conditions

$$u(t, x_0; p) = \alpha(t; p), \quad u(t, x_1; p) = \beta(t; p) \quad (4.35)$$

and approximating

$$\begin{aligned} \alpha(t; p) &\simeq \sum_{k=0}^K ' \sum_{m=0}^M ' \alpha_{km} T_k(\tau) T_m(\sigma), \\ \beta(t; p) &\simeq \sum_{k=0}^K ' \sum_{m=0}^M ' \beta_{km} T_k(\tau) T_m(\sigma), \end{aligned} \quad (4.36)$$

the orthogonality property of Chebyshev polynomials implies that, to satisfy Eq. (4.35),

$$\begin{aligned}\alpha_{km} &= \sum_{l=0}^L ' a_{klm} T_l(-1) = \sum_{l=0}^L ' a_{klm} (-1)^l, \\ \beta_{km} &= \sum_{l=0}^L ' a_{klm} T_l(1) = \sum_{l=0}^L ' a_{klm},\end{aligned}\tag{4.37}$$

where we use  $T_l(-1) = (-1)^l$  and  $T_l(1) = 1$ . Taking the sum and the difference of these two expressions and rearranging we obtain

$$\begin{aligned}a_{k,L-1,m} &= \frac{\beta_{km} - \alpha_{km}}{2} - \sum_{\substack{l=1 \\ l \text{ odd}}}^{L-3} a_{k,l,m}, \\ a_{kLm} &= \frac{\alpha_{km} + \beta_{km}}{2} - \sum_{\substack{l=0 \\ l \text{ even}}}^{L-2} ' a_{k,l,m}\end{aligned}\tag{4.38}$$

if  $L$  is even, whereas we have

$$\begin{aligned}a_{k,L-1,m} &= \frac{\alpha_{km} + \beta_{km}}{2} - \sum_{\substack{l=0 \\ l \text{ even}}}^{L-3} ' a_{k,l,m}, \\ a_{kLm} &= \frac{\beta_{km} - \alpha_{km}}{2} - \sum_{\substack{l=1 \\ l \text{ odd}}}^{L-2} a_{k,l,m}\end{aligned}\tag{4.39}$$

if  $L$  is odd.

### Neumann boundary conditions

Neumann boundary conditions can be imposed in a similar way. Let us consider two boundary conditions of the form

$$\frac{du}{dx}(t, x_0; p) = \alpha(t; p), \quad \frac{du}{dx}(t, x_1; p) = \beta(t; p),\tag{4.40}$$

with

$$\begin{aligned}\alpha(t; p) &\simeq \sum_{k=0}^K ' \sum_{m=0}^M ' \alpha_{km} T_k(\tau) T_m(\sigma), \\ \beta(t; p) &\simeq \sum_{k=0}^K ' \sum_{m=0}^M ' \beta_{km} T_k(\tau) T_m(\sigma).\end{aligned}\tag{4.41}$$

Using again the orthogonality property of Chebyshev polynomials, we obtain

$$\begin{aligned}\alpha_{km} &= \frac{2}{B_x} \sum_{l=0}^{L-1} \sum_{\substack{\lambda=l+1 \\ \lambda-l \text{ odd}}}^L \lambda a_{k\lambda m} (-1)^l, \\ \beta_{km} &= \frac{2}{B_x} \sum_{l=0}^{L-1} \sum_{\substack{\lambda=l+1 \\ \lambda-l \text{ odd}}}^L \lambda a_{k\lambda m}.\end{aligned}\tag{4.42}$$

Now, by taking the sum and the difference of these two expressions and rearranging, we obtain

$$\begin{aligned}\alpha_{km} + \beta_{km} &= \frac{2}{B_x} \sum_{\substack{l=1 \\ l \text{ odd}}}^L l^2 a_{klm}, \\ \beta_{km} - \alpha_{km} &= \frac{2}{B_x} \sum_{\substack{l=2 \\ l \text{ even}}}^L l^2 a_{klm}.\end{aligned}\tag{4.43}$$

An explicit expression for the coefficients  $a_{k,L-1,m}$  and  $a_{kLm}$  is easily computed by rearranging Eq. (4.43), that is

$$\begin{aligned}a_{k,L-1,m} &= B_x \frac{\alpha_{km} + \beta_{km}}{2(L-1)^2} - \sum_{\substack{l=1 \\ l \text{ odd}}}^{L-3} \frac{l^2}{(L-1)^2} a_{klm}, \\ a_{kLm} &= B_x \frac{\beta_{km} - \alpha_{km}}{2L^2} - \sum_{\substack{l=2 \\ l \text{ even}}}^{L-2} \frac{l^2}{L^2} a_{klm}\end{aligned}\tag{4.44}$$

if  $L$  is even, whereas we have

$$\begin{aligned}a_{k,L-1,m} &= B_x \frac{\beta_{km} - \alpha_{km}}{2(L-1)^2} - \sum_{\substack{l=2 \\ l \text{ even}}}^{L-3} \frac{l^2}{(L-1)^2} a_{klm}, \\ a_{kLm} &= B_x \frac{\alpha_{km} + \beta_{km}}{2L^2} - \sum_{\substack{l=1 \\ l \text{ odd}}}^{L-2} \frac{l^2}{L^2} a_{klm}\end{aligned}\tag{4.45}$$

if  $L$  is odd.

### Periodic boundary conditions

In order to apply the periodic boundary conditions

$$u(t, x_0; p) = u(t, x_1; p), \quad \frac{du}{dx}(t, x_0; p) = \frac{du}{dx}(t, x_1; p),\tag{4.46}$$

we impose in the Chebyshev spectral domain

$$\begin{aligned} \sum_{l=0}^L ' a_{klm} (-1)^l &= \sum_{l=0}^L ' a_{klm}, \\ \sum_{l=0}^{L-1} ' (-1)^l \sum_{\substack{\lambda=l+1 \\ \lambda-l \text{ odd}}}^L \lambda a_{k\lambda m} &= \sum_{l=0}^{L-1} ' \sum_{\substack{\lambda=l+1 \\ \lambda-l \text{ odd}}}^L \lambda a_{k\lambda m}. \end{aligned} \quad (4.47)$$

These expressions are rewritten as

$$\sum_{\substack{l=1 \\ l \text{ odd}}}^L a_{klm} = 0, \quad \sum_{\substack{l=2 \\ l \text{ even}}}^L l^2 a_{klm} = 0. \quad (4.48)$$

The coefficients  $a_{k,L-1,m}$  and  $a_{kLm}$  are thus given as

$$a_{k,L-1,m} = - \sum_{\substack{l=1 \\ l \text{ odd}}}^{L-3} a_{klm}, \quad a_{kLm} = - \sum_{\substack{l=2 \\ l \text{ even}}}^{L-2} \frac{l^2}{L^2} a_{klm} \quad (4.49)$$

if  $L$  is even, whereas we have

$$a_{k,L-1,m} = - \sum_{\substack{l=2 \\ l \text{ even}}}^{L-3} \frac{l^2}{(L-1)^2} a_{klm}, \quad a_{kLm} = - \sum_{\substack{l=1 \\ l \text{ odd}}}^{L-2} a_{klm} \quad (4.50)$$

if  $L$  is odd.

## 4.2 Uncertainty propagation through a plasma turbulence model

To illustrate an application of the Chebyshev spectral method to the study of uncertainty propagation, we present here the simulation of plasma turbulence in a simple magnetized torus (SMT), carried out with a two-dimensional drift-reduced Braginskii model (see Ref. [104] for a detailed discussion of this model). The implementation of the model in a numerical code using the WRM approach and the decomposition in Chebyshev polynomials described above allows us to study the propagation of the uncertainty affecting the input parameter that characterizes the plasma losses at the vessel.

### 4.2.1 Two-dimensional drift-reduced Braginskii equations

We consider an SMT configuration, where the magnetic field,  $\mathbf{B}$ , is obtained by superimposing a toroidal magnetic field on a vertical magnetic field. This results in helical field lines that wind around the toroidal vacuum vessel from the bottom to the top of the de-



---

## 4.2. Uncertainty propagation through a plasma turbulence model

---

vice. This configuration, implemented in a number of basic plasma physics experiments such as TORPEX [48, 49] and Texas Helimak [111], is of interest to the plasma physics community because it offers a simple and well diagnosed scenario to study the turbulent transport related to instabilities such as interchange modes and drift waves, which are present also in the tokamak SOL. In fact, models similar to the one used to investigate SMT plasma dynamics have been used also for the tokamak SOL (see e.g. Ref. [106]). To describe the SMT configuration, we indicate with  $x$  the radial coordinate, with  $z$  the coordinate along  $B$  (approximately the toroidal direction), and with  $y$  the coordinate perpendicular to  $x$  and  $z$  (approximately the vertical direction). Because of the high collisionality typical of SMT experiments, it is justified to model the plasma dynamics with a set of drift-reduced Braginskii equations. Within the hypothesis  $k_{\parallel} = 0$  and under the assumption of cold ions, it is possible to integrate the drift-reduced Braginskii equations detailed in Appendix A in the parallel direction, in order to evolve the line-averaged density  $n(y, x) = \int n(y, x, z) dz / L_{\parallel}$ , electrostatic potential  $\phi(y, x) = \int \phi(y, x, z) dz / L_{\parallel}$ , and electron temperature  $T_e(y, x) = \int T_e(y, x, z) dz / L_{\parallel}$ , with  $L_{\parallel} = 2\pi N R_0$  the magnetic field line length,  $R_0$  the major radius, and  $N$  the number of turns of the magnetic field line in the device. Neglecting the parallel electron thermal conductivity term, using the Boussinesq approximation to simplify the vorticity equation, assuming an infinite aspect ratio, and applying Bohm's boundary conditions at the sheath edge,  $v_{\parallel i} = \pm\sqrt{T_e}$  and  $v_{\parallel e} = \pm\sqrt{T_e} \exp(\Lambda - \phi/T_e)$ , the resulting system of equations is

$$\frac{\partial n}{\partial t} = -R_0\{\phi, n\} + 2\left(\frac{\partial n}{\partial y} - n\frac{\partial\phi}{\partial y}\right) - \sigma n\sqrt{T_e} \exp\left(\Lambda - \frac{\phi}{T_e}\right) + \mathcal{D}_n(n) + S_n, \quad (4.51)$$

$$\frac{\partial\omega}{\partial t} = -R_0\{\phi, \omega\} + \frac{2}{n}\frac{\partial p_e}{\partial y} + \sigma\sqrt{T_e}\left[1 - \exp\left(\Lambda - \frac{\phi}{T_e}\right)\right] + \mathcal{D}_\omega(\omega), \quad (4.52)$$

$$\begin{aligned} \frac{\partial T_e}{\partial t} = & -R_0\{\phi, T_e\} + \frac{4T_e}{3}\left(\frac{1}{n}\frac{\partial p_e}{\partial y} + \frac{5}{2}\frac{\partial T_e}{\partial y} - \frac{\partial\phi}{\partial y}\right) \\ & - \frac{2\sigma T_e\sqrt{T_e}}{3}\left[1.71 \exp\left(\Lambda - \frac{\phi}{T_e}\right) - 0.71\right] + \mathcal{D}_{T_e}(T_e) + S_{T_e}, \end{aligned} \quad (4.53)$$

where  $\Lambda \simeq 3$  for hydrogen plasmas,  $\omega = \nabla_{\perp}^2\phi$ , and the operators  $\nabla_{\perp}^2$  and  $\{\phi, -\}$  are detailed in Section A.3.4 (in this chapter we use the normalization described in Appendix A). The parameter  $\sigma$  is used to model the parallel losses at the magnetic pre-sheath (MP) entrance. Assuming that the plasma density at the MP entrance is  $n_{MP}(y, x) = n(y, x)/2$ , we obtain  $\sigma = 1/(2\pi N)$ . However, the value of  $n_{MP}$  is not precisely known, and  $\sigma$  is affected by a large uncertainty.

To further simplify the problem, we write  $\exp(\Lambda - \phi) \simeq 1 + \Lambda - \phi$  and  $n \ln(n) \simeq n - 1$  assuming  $\phi$  close to  $\Lambda$  and  $n$  close to 1, respectively. Moreover, we use the isothermal

plasma approximation, i.e.  $T_e = 1$ . Equations (4.51)-(4.53) are thus rewritten as

$$\begin{aligned} \partial_t \theta = & -R_0\{\phi', \theta\} + 2(\partial_y \theta - \partial_y \phi') - \sigma(1 - \phi') + \mathcal{D}_n(\theta) \\ & + D_n [(\partial_x \theta)^2 + (\partial_y \theta)^2] + S_n(1 - \theta), \end{aligned} \quad (4.54)$$

$$\partial_t \omega = -R_0\{\phi', \omega\} + 2\partial_y \theta + \sigma\phi' + \mathcal{D}_\omega(\omega), \quad (4.55)$$

where  $\omega = \partial_x^2 \phi' + \partial_y^2 \phi'$ ,  $\phi' = \phi - \Lambda$ , and  $\theta = \log(n)$ . The model in Eqs. (4.54)-(4.55) is an ideal test bed for the application of the WRM discussed in Section 4.1, since it contains both first and second order derivatives and nonlinear convective terms.

We note that the scalar fields  $\theta$ ,  $\omega$ , and  $\phi'$  depend on time,  $t \in [t_0, t_1]$ , on the spatial coordinates  $x \in [x_0, x_1]$  and  $y \in [y_0, y_1]$ , and on the parameter  $\sigma$ , which is estimated from experimental measurements with large uncertainty. Our goal is to estimate the impact of variations of  $\sigma$  in the interval  $[\sigma_0, \sigma_1]$  on the solution of Eqs. (4.54)-(4.55).

## 4.2.2 Numerical implementation

In order to investigate the dependence of  $n$ ,  $\phi$  and  $\omega$  on  $\sigma$ , we developed a simulation code that solves Eqs. (4.54)-(4.55), together with  $\omega = \partial_x^2 \phi' + \partial_y^2 \phi'$ , implementing the WRM with Chebyshev decomposition described in Section 4.1. More precisely, we write

$$\theta(t, x, y; \sigma) \simeq \sum_{k=0}^K \sum_{l=0}^L \sum_{m=0}^M \sum_{n=0}^N a_{klmn}^\theta T_k(\tau) T_l(\chi) T_m(v) T_n(\mu), \quad (4.56)$$

$$\omega(t, x, y; \sigma) \simeq \sum_{k=0}^K \sum_{l=0}^L \sum_{m=0}^M \sum_{n=0}^N a_{klmn}^\omega T_k(\tau) T_l(\chi) T_m(v) T_n(\mu), \quad (4.57)$$

$$\phi'(t, x, y; \sigma) \simeq \sum_{k=0}^K \sum_{l=0}^L \sum_{m=0}^M \sum_{n=0}^N a_{klmn}^\phi T_k(\tau) T_l(\chi) T_m(v) T_n(\mu), \quad (4.58)$$

where  $K$ ,  $L$ ,  $M$ , and  $N$  are the highest order Chebyshev polynomials used for the decomposition along the temporal, radial, vertical, and parameter coordinates, respectively, and

$$\tau = \frac{t - A_t}{B_t}, \quad \chi = \frac{x - A_x}{B_x}, \quad v = \frac{y - A_y}{B_y}, \quad \mu = \frac{\sigma - A_\sigma}{B_\sigma}, \quad (4.59)$$

with  $A_t = (t_1 + t_0)/2$  and  $B_t = (t_1 - t_0)/2$ . Similar definitions apply to the other quantities.

Following the procedure described in Section 4.1.3, we write the operators  $\{\phi, A\}$ ,  $\partial_x A$ ,  $\partial_y A$ ,  $\mathcal{D}_A(A)$ , with  $A = \theta, \omega, \phi'$ , as well as the terms  $\sigma(1 - \phi')$ ,  $\sigma\phi'$ , and  $S_n(1 - \theta)$ , in the Chebyshev spectral domain, obtaining a set of  $3K(L - 1)(M - 1)(N + 1)$  algebraic nonlinear equations for the coefficients  $a_{klmn}^\theta$ ,  $a_{klmn}^\omega$ , and  $a_{klmn}^\phi$ , with  $0 < k \leq K$ ,  $0 \leq l \leq L - 2$ ,  $0 \leq m \leq M - 2$ ,  $0 \leq n \leq N$ . The  $k = 0$  coefficients are obtained as described in Section 4.1.4, while the  $l = L - 1$ ,  $l = L$ ,  $m = M - 1$ , and  $m = M$  coefficients are computed by applying the boundary conditions as discussed in Section 4.1.5. To solve

---

## 4.2. Uncertainty propagation through a plasma turbulence model

---

numerically the resulting system of equations, we write the equation  $\omega = \partial_x^2 \phi' + \partial_y^2 \phi'$  in the Chebyshev spectral domain as a linear system  $\underline{\underline{A}} \mathbf{a}^\phi = \mathbf{a}^\omega$ , with  $\underline{\underline{A}}$  a square matrix with inverse  $\underline{\underline{A}}^{-1}$ , whereas  $\mathbf{a}^\omega$  and  $\mathbf{a}^\phi$  are vectors containing respectively the coefficients  $a_{klmn}^\omega$  and  $a_{klmn}^\phi$ . This allows us to express  $\mathbf{a}^\phi = \underline{\underline{A}}^{-1} \mathbf{a}^\omega$ . Using the expression of  $\mathbf{a}^\phi$  as a function of  $\mathbf{a}^\omega$  in Eqs. (4.54)-(4.55), we obtain a set of  $2(K+1)(L+1)(M+1)(N+1)$  algebraic nonlinear equations for the coefficients  $a_{klmn}^\theta$  and  $a_{klmn}^\omega$  in the form

$$\begin{bmatrix} \mathbf{a}^\theta \\ \mathbf{a}^\omega \end{bmatrix} = \mathbf{f} \left( \begin{bmatrix} \mathbf{a}^\theta \\ \mathbf{a}^\omega \end{bmatrix} \right), \quad (4.60)$$

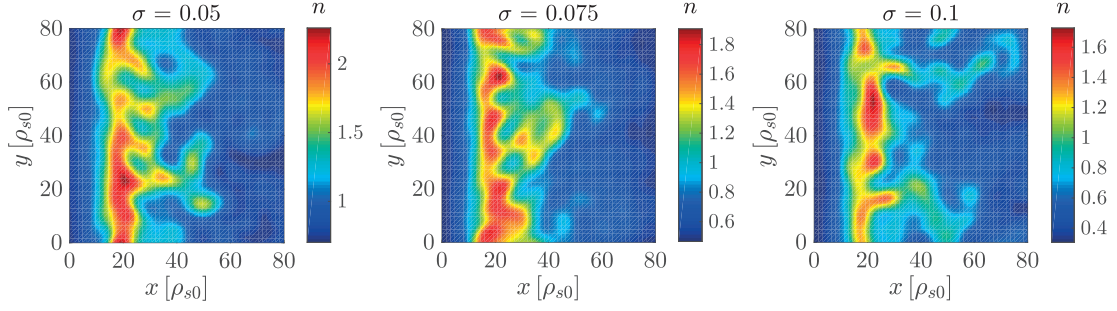
with  $\mathbf{a}^\theta$  a vector containing the coefficients  $a_{klmn}^\theta$  and  $\mathbf{f}$  a vector function reflecting Eqs. (4.54)-(4.55) in Chebyshev spectral domain.

The nonlinear system of equations, Eq. (4.60), is implemented in a numerical code written in Fortran90 and interfaced with the *MATLAB* environment [112] and it is solved with the *MATLAB fsolve* nonlinear system solver using a trust-region algorithm. To facilitate the solver convergence, we separate the time coordinate in sub-domains as described in Section 4.1.2 (the results we show consider  $\Delta t = \Delta t^i = 0.2$  and we verified numerically that they are converged with respect to  $\Delta t$ ).

### 4.2.3 Simulation results

For our simulations we consider a spatial domain extending radially from  $x_0 = 0$  to  $x_1 = 80$  and vertically from  $y_0 = 0$  to  $y_1 = 80$ . Moreover, we consider a parameter  $\sigma$  varying between  $\sigma_0 = 0.05$  and  $\sigma_1 = 0.1$ . At  $x_0$  and  $x_1$  we impose Dirichlet boundary conditions for  $\phi'$  and  $\omega$ , while we use Neumann boundary conditions for  $\theta$ . Because of the assumption  $k_{\parallel} = 0$ , we impose periodic boundary conditions at  $y_0$  and  $y_1$  and we set the length  $y_1 - y_0$  as the distance between the returns of a field line in the poloidal plane. Moreover, we use a source term of the form  $S_n(x) \propto \exp[-(x - x_S)^2/16]$ , with  $x_S = 20$ , to mimic the plasma source at the electron cyclotron resonance layer present in TORPEX, and we set  $D_n = D_\omega = 5$  and  $R_0 = 200$ . We note that similar parameters were used for TORPEX simulations in Ref. [104].

Figure 4.1 shows typical snapshots of the plasma density for  $\sigma = 0.05, 0.075, 0.1$ , resulting from a simulation performed with  $(K, L, M, N) = (1, 22, 22, 3)$ . The turbulent character of the plasma dynamics observed in previous finite difference simulations is retrieved also by our WRM simulation, with eddies extending radially outward from the source location and that detach from it, creating blobs that propagate towards the low-field side part of the domain. We remark that the WRM approach makes it possible to simulate the TORPEX plasma dynamics for any value of  $\sigma$  between 0.05 and 0.1 solving once Eq. (4.60).



**Figure 4.1** – Typical snapshot of plasma density for  $\sigma = 0.05, 0.075, 0.1$  (left, middle, and right panels, respectively), obtained by evaluating  $\theta$  according to Eq. (4.56) with  $(K, L, M, N) = (1, 22, 22, 3)$ .

#### 4.2.4 Uncertainty propagation

Numerical simulations of plasma turbulence in basic plasma physics experiments, as well as in the tokamak SOL, are often employed to evaluate time-averaged quantities (e.g. the time-averaged pressure gradient length  $L_p = -\langle p_e \rangle_t / \nabla \langle p_e \rangle_t$ , with  $\langle p_e \rangle_t$  the time-averaged plasma pressure [113]). Therefore, as an example of uncertainty propagation study, in the following we focus on the radial profile of the vertical- and time-averaged plasma density,

$$\langle n \rangle_{y,t}(x; \sigma) = \frac{1}{(y_1 - y_0)(t_1 - t_0)} \int_{y_0}^{y_1} \int_{t_0}^{t_1} n(t, x, y; \sigma) dy dt. \quad (4.61)$$

We verified numerically that  $\langle n \rangle_{y,t} = \langle \exp(\theta) \rangle_{y,t} \simeq \exp(\langle \theta \rangle_{y,t})$ . This simplifies the evaluation of  $\langle n \rangle_{y,t}$ , since it allows to analytically integrate the Chebyshev expansion of  $\theta$  over  $t$  and  $y$  as

$$\begin{aligned} & \int_{y_0}^{y_1} \int_{t_0}^{t_1} \sum_{k=0}^K \sum_{l=0}^L \sum_{m=0}^M \sum_{n=0}^N a_{klmn}^\theta T_k(\tau) T_l(\chi) T_m(v) T_n(\mu) dy dt \\ &= 4B_t B_y \sum_{l=0}^L \sum_{n=0}^N T_l(\chi) T_n(\mu) \sum_{\substack{k=0 \\ k \text{ even}}}^{K_1} \sum_{\substack{m=0 \\ m \text{ even}}}^{M_1} \frac{a_{klmn}^\theta}{(1-k^2)(1-m^2)}, \end{aligned} \quad (4.62)$$

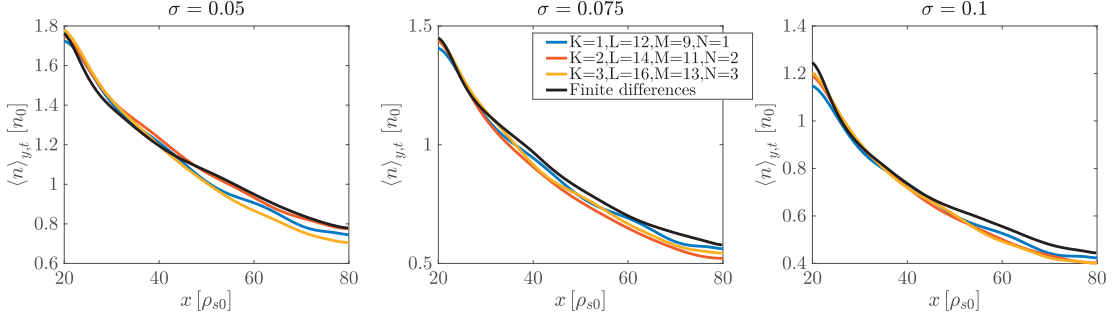
and approximate

$$\langle n \rangle_{y,t}(x; \sigma) \simeq \exp \left( \sum_{l=0}^L \sum_{n=0}^N \bar{a}_{ln} T_l(\chi) T_n(\mu) \right), \quad (4.63)$$

where

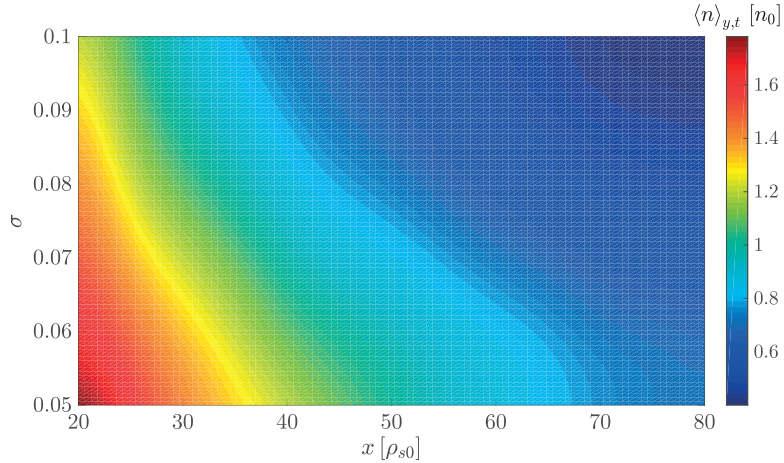
$$\bar{a}_{ln} = 4 \frac{B_t}{t_1 - t_0} \frac{B_y}{y_1 - y_0} \sum_{\substack{k=0 \\ k \text{ even}}}^K \sum_{\substack{m=0 \\ m \text{ even}}}^M \frac{a_{klmn}^\theta}{(1-k^2)(1-m^2)}. \quad (4.64)$$

## 4.2. Uncertainty propagation through a plasma turbulence model



**Figure 4.2** – Radial profiles of  $\langle n \rangle_{y,t}(x; \sigma)$  computed according to Eq. (4.63) for  $\sigma = 0.5, 0.075, 0.1$  (left, middle, and right panels, respectively). The simulation results are obtained by using a Chebyshev decomposition with  $(K, L, M, N) = (1, 12, 9, 1), (2, 14, 11, 2), (3, 16, 13, 3)$  (blue, red, and yellow lines, respectively), and by using the finite difference code that solves Eqs. (4.54)-(4.55) [104] (black lines).

We note that the results presented herein consider time averages performed over intervals of approximately 40 time units, with simulations in turbulent quasi-steady state. Figure 4.2 shows the profiles of  $\langle n \rangle_{y,t}(x; \sigma)$  for  $\sigma = 0.5, 0.075, 0.1$  (left, middle, and right panels, respectively) obtained from three WRM simulations with  $(K, L, M, N) = (1, 12, 9, 1), (2, 14, 11, 2), (3, 16, 13, 3)$  (blue, red, and yellow lines, respectively). The profiles are compared to the results of simulations performed with a finite difference code [104] that solves Eqs. (4.54)-(4.55) (black lines). We observe that the results obtained with the WRM are consistent with simulations carried out with the finite difference approach, the differences probably due to the relatively small number of spectral terms used in the decomposition of  $\theta$ ,  $\phi'$ , and  $\omega$  (a large number of spectral terms is needed to accurately represent the density profile obtained with finite difference simulations in Chebyshev space). In Fig. 4.3 we also show the two-dimensional profile of



**Figure 4.3** – Profile of  $\langle n \rangle_{y,t}(x; \sigma)$  as function of  $x$  and  $\sigma$  obtained by using the WRM in the Chebyshev spectral domain with  $(K, L, M, N) = (3, 16, 13, 3)$ .

$\langle n \rangle_{y,t}(x; \sigma)$  as a function of  $x$  and  $\sigma$ . We observe that increasing the parallel loss term, i.e.  $\sigma$ , the value of  $\langle n \rangle_{y,t}(x; \sigma)$  decreases, as expected.

To investigate quantitatively the impact of  $\sigma$  on the simulation results, we focus on

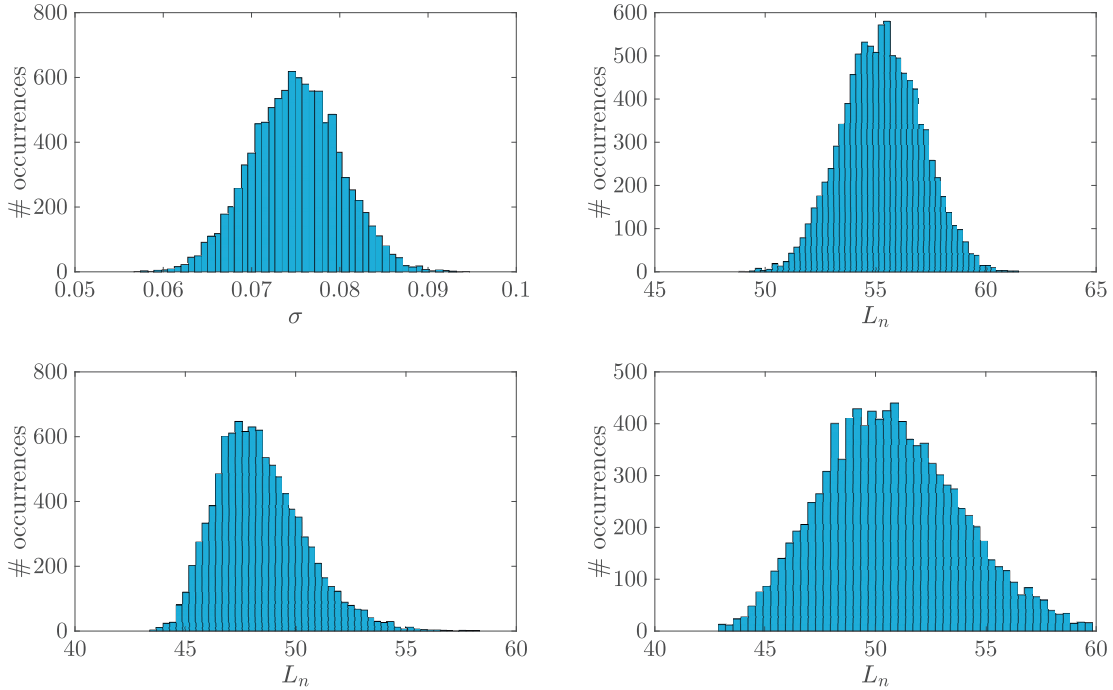
the time-averaged density gradient length  $L_n = -\langle n \rangle_{y,t} / \partial_x \langle n \rangle_{y,t}$ . More precisely, we assume that the density decreases exponentially for  $x > x_S$ , i.e.  $\langle n \rangle_{y,t}(x; \sigma) \propto \exp[-(x - x_S)/L_n(\sigma)]$  and we compute  $L_n(\sigma)$  by evaluating

$$L_n(\sigma) = [x_{1/2}(\sigma) - x_S] / \ln(2), \quad (4.65)$$

with  $x_{1/2}(\sigma)$  satisfying  $\langle n \rangle_{y,t}(x_{1/2}; \sigma) = \langle n \rangle_{y,t}(x_S; \sigma)/2$ . Assuming that  $\sigma$  is characterized by a probability density function

$$f_\sigma(\sigma) \propto \exp[-(\sigma - A_\sigma)^2 / 0.005^2], \quad (4.66)$$

we randomly generate a number of samples  $\sigma^i$ ,  $i = 1, 2, \dots$ , distributed according to  $f_\sigma$  (see Fig. 4.4, top left panel), and for each element  $\sigma^i$  we compute the corresponding  $L_n(\sigma^i)$ . The results thus obtained are presented in Fig. 4.4 for  $(K, L, M, N) = (1, 12, 9, 1), (2, 14, 11, 2), (3, 16, 13, 3)$  (top right, bottom left, and bottom right panels, respectively). We observe that the three averaged values  $L_n \simeq 55, 48, 51$ , obtained



**Figure 4.4** – Top left panel: Values of  $\sigma^i$  randomly distributed according to  $f_\sigma$ . The corresponding distributions of  $L_n(\sigma^i)$ , computed according to Eq. (4.65), are displayed for  $(K, L, M, N) = (1, 12, 9, 1), (2, 14, 11, 2), (3, 16, 13, 3)$  (top right, bottom left, and bottom right panels, respectively).

for  $(K, L, M, N) = (1, 12, 9, 1), (2, 14, 11, 2), (3, 16, 13, 3)$ , respectively, depend slightly on  $K, L, M$ , and  $N$ . Moreover,  $L_n$  does not depend linearly on  $\sigma$ , since the three  $L_n$  distributions do not exhibit the same Gaussian distribution as  $\sigma$ . While the averaged value of  $L_n$  is only slightly affected by the number of spectral terms used in the simulation, the distribution of  $L_n$  shows a more significant dependence. However, when evaluating the standard deviations  $\text{SD}(L_n)$  corresponding to each distribution,



we obtain  $\text{SD}(L_n) = 1.8, 2.0, 3.1$ . The difference between these values is small. This means that it is possible to obtain a rough estimate of the spread of  $L_n$ , due to variations of  $\sigma$ , by considering a small number of spectral terms, which is exactly the target of our methodology. This is particularly remarkable, since the simulation performed with  $(K, L, M, N) = (1, 12, 9, 1)$  is less demanding, in terms of computational resources, by approximately a factor 350 with respect to the simulation carried out with  $(K, L, M, N) = (3, 16, 13, 3)$ .

### 4.3 Conclusions

In the present chapter we propose a rigorous methodology to assess the uncertainty affecting a simulation result due to the propagation of input parameter uncertainties. The methodology is general, simple to apply, and allows to approximate the model equation solution with a semi-analytic expression that depends explicitly on time, spatial coordinates, and input parameters.

In order to study the impact of input parameter variations on the results of a plasma turbulence model, we propose to use a WRM with decomposition in Chebyshev polynomials. This choice is motivated by the *minmax* property characterizing a Chebyshev decomposition. By applying the WRM, a system of nonlinear algebraic equations is derived for the coefficients of the Chebyshev expansion. The solution of these equations provides directly the information on the dependence of the simulation result on the input parameters.

We apply the proposed methodology to a two-dimensional drift-reduced Braginskii model used to investigate the plasma dynamics in basic plasma physics experiments and in the tokamak SOL. These equations are decomposed in the Chebyshev spectral domain and the resulting system of equations is implemented in a numerical code. The plasma turbulent dynamics is retrieved by our simulations and an explicit dependence of the time-averaged density profiles on the parameter describing the parallel losses is obtained. To our knowledge, this is the first time that a fully spectral approach is used to successfully simulate plasma turbulence and study uncertainty propagation.

Assuming that the input parameter under consideration is distributed according to a Gaussian probability distribution function, we compute the standard deviation that characterizes the corresponding (in principle, non-Gaussian) distribution of time-averaged density gradient lengths. We find that a reasonable value of the dispersion of the density scale length due to the uncertainty affecting the input parameter that describes the parallel losses can be obtained using a small number of Chebyshev polynomials, i.e. by carrying out reduced-cost simulations.





# CHAPTER 5

## A multi-code validation: blob dynamics in TORPEX

Blobs, also known as filaments, are structures with an excess of density and temperature relative to the surrounding plasma, elongated in the direction parallel to the magnetic field. Blobs detach from the main plasma and move outwards in the SOL due to a self-generated  $\mathbf{E} \times \mathbf{B}$  field. Experimental evidences point out that the transport associated with these structures could reach half of the total perpendicular transport observed in a tokamak SOL [114], leading to significant particle and heat fluxes to the walls. In the recent past, a large effort has been carried out to improve the knowledge of the blob dynamics, both experimentally and theoretically, achieving significant progress [114, 115]. While illustrating the first validation exercise of the present thesis, the goal of this chapter is to further improve our understanding of the blob dynamics, and therefore increase the reliability of the numerical tools employed for their simulation, performing a validation that involves several plasma turbulence codes used to model the SOL region. We remark that the work presented herein represents the first multi-code validation of blob dynamics, despite the very large use of simulations in analyzing this phenomenon (see e.g. Refs. [21, 116–123]). Two-dimensional and three-dimensional simulations of seeded blobs, based on five different models implemented in four turbulence codes (BOUT++ [18], GBS [20], HESEL [124, 125], and TOKAM3X [126]), are validated against experimental blob measurements. We assess the consistency of the numerical results with experimental measurements and, at the same time, we investigate the differences between the simulation results of the five models through a benchmark study. Thanks to the differences among the models, we identify and assess the key physics elements that determine the blob motion.

The experimental measurements are taken from the TORPEX experiment [48, 49], an ideal device for the validation of plasma turbulence codes. In fact, the TORPEX configuration mimics the main features of the tokamak SOL, while remaining relatively simple, and it is equipped with a complete set of diagnostics. Conditionally-averaged

measurements taken on TORPEX provide the two-dimensional profiles of plasma density, electron temperature, and floating potential for the blob, which are needed to accurately initialize seeded blob simulations. At the same time, it also provides the measurement of the blob velocity used to validate the numerical results. A parameter scan is performed by detecting blobs with different density peak values. This allows for a comparison between experimental measurements and simulations of blobs propagating at different velocities and having different internal stability properties.

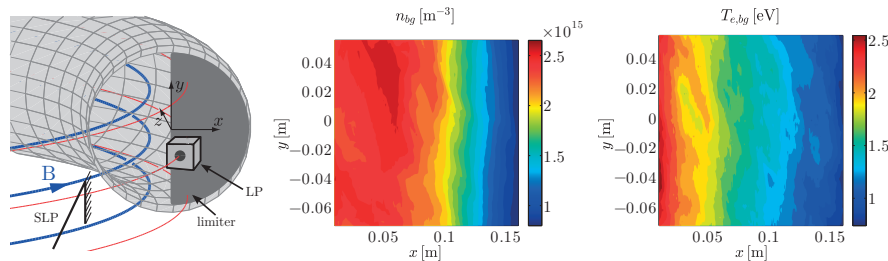
Because of the relatively high collisionality of TORPEX plasmas, all models we consider are based on the drift-reduced Braginskii equations. However, they differ in the assumptions used to simplify the equations, such as the hypothesis of cold ions, isothermal electrons, or negligible electron inertia. Some of the models make use of the infinite aspect ratio approximation. We also consider two-dimensional models, based on different closures of the parallel currents on the vessel wall. The influence of all these assumptions on the blob dynamics is analysed through a benchmark study, where the same scenario is considered for all the models, and the differences observed in the simulation results are investigated.

This chapter is structured as follows. In Section 5.1 we illustrate the TORPEX device and the experimental setup used in the present work. Section 5.2 introduces the five models used to simulate the blob dynamics and discusses their main differences. Then, we illustrate our simulations, focusing on their initialization, in Section 5.3. Section 5.4 present a sensitivity study performed to investigate the influence of the input parameter uncertainties on the numerical results. The comparison of the experimental measurements and the simulations are the subject of Section 5.5. The Conclusions follow in Section 5.6. The results discussed in the present chapter are published in Ref. [127].

### 5.1 Experimental scenario

The experimental data shown in the present chapter are obtained on the TORPEX experiment, a toroidal device with major radius  $R = 1$  m and minor radius  $a = 0.2$  m that features the simple magnetized toroidal (SMT) configuration. A toroidal magnetic field ( $B_\varphi = 76$  mT on axis) superposed on a vertical magnetic field ( $B_v = 1.6$  mT) results in helical field lines that wind around the device. The field lines intercept the top and bottom walls of the device in the inner half part of the cross section (high field side), while in the outer half of the cross section (low field size) a poloidal steel limiter provides a region that has a nearly constant connection length  $L_\parallel \simeq 2\pi R$ , and near-perpendicular incidence of the magnetic field lines on the target [128]. This configuration is schematically shown in Fig. 5.1 (left panel). The coordinate system  $(y, x, z)$  used in this chapter is also represented in Fig. 5.1 (left panel):  $x$  is the radial direction,  $z$  is the direction parallel to  $\mathbf{B}$  (and coincides approximatively with the toroidal direction), and  $y$  is perpendicular to  $x$  and  $z$  (and coincides approximatively with the vertical direction).

A hydrogen plasma is produced and sustained by microwaves in the electron cyclotron



**Figure 5.1** – Schematic representation of the TORPEX experiment (left panel). The limiter located in the low-field side region is shown together with the probes used to perform blob conditional sampling. Note that SLP is not represented at  $180^\circ$  from the limiter as it is in the experiment for drawing convenience. Background profiles of  $n$  (middle panel) and  $T_e$  (right panel) in the low-field side region, where blobs propagate, are also presented. The profiles are measured in a poloidal plane 3 cm away from the limiter.

range of frequencies. On the high-field side of the device, turbulence driven by ideal interchange modes [129, 130] results in blobs, which dominate transport on the low field side. Typical plasma parameters are  $n \simeq 10^{16} \text{ m}^{-3}$  and  $T_e \simeq 5 \text{ eV}$  in the source region, and are slightly smaller in the blob region. The ions are typically much colder than the electrons. An example of the time-averaged profiles measured in the blob region is presented in Fig. 5.1 (middle and right panels).

The experimental results used in this chapter are obtained using two diagnostics: (i) a vertically oriented linear array of Langmuir probes (LPs) with 1.8 cm distance between tips, located at  $x = 7 \text{ cm}$  and toroidally separated approximately by  $180^\circ$  from the limiter, referred to as SLP, and (ii) a single-sided LP, positioned approximately 3 cm away from the limiter and with the collecting plate oriented perpendicularly to the magnetic field lines. Time-resolved two-dimensional measurements associated with blobs are obtained using conditional sampling over many blob events, allowing the reconstruction of the  $I - V$  Langmuir characteristic. This technique is explained in details in Ref. [129], and can be summarized as follows. The probes of the SLP array, biased at  $-40 \text{ V}$  and operated in ion saturation current mode, are located at fixed positions in the blob region and are used as reference probes, while the single-sided probe, placed close to the limiter, is operated in swept mode. Positive bursts in the SLP reference signals are interpreted as blobs moving in front of the reference probe. When a blob is detected, the voltage  $V$  applied to the swept probe and the corresponding measured current value  $I$  are retained. The whole set of voltage and current values is interpreted as the  $I - V$  characteristic associated with blobs, which is evaluated as a function of time with respect to the detection. To reconstruct the two-dimensional profiles, the single-sided LP is moved radially in between discharges. This experimental setup has been used to investigate the parallel current structure associated with blobs, as presented in Ref. [122].

The TORPEX experiment is an ideal device for the validation of plasma turbulence codes for two reasons. First, a wide range of observables can be provided with high spatial resolution, such as the plasma density, the electron temperature, the floating potential, and the parallel current. This is crucial to perform accurate seeded blob simulations, which require the profiles of all evolved fields at a certain time to set the proper

initial conditions. Second, the SMT configuration mimics the main features of the tokamak SOL, such as open field lines, curvature and gradients of the magnetic field, and plasma pressure gradients, but in a simpler configuration. This facilitates considerably the analysis and the interpretation of the experimental and simulation results.

## 5.2 The models and the simulation codes

Because of the relatively high collisionality of TORPEX plasmas, we use a fluid approach based on the Braginskii equations [13] to model the blob dynamics. Moreover, since the time scale of the blob dynamics is such that  $d/dt \ll \omega_{ci}$  (where  $\omega_{ci} = eB/m_i$  is the ion gyrofrequency), we can consider these equations in the drift limit (see e.g. Ref. [14]). Finally, we note that, since in the present scenario magnetic perturbations are negligible, only electrostatic models are considered.

Although the drift-reduced Braginskii model is now well established (see Appendix A for a detailed discussion of this model), for practical purposes several approximations are introduced to simplify the equations. Those approximations vary from code to code and, in general, their effect on the blob motion is not well known. In order to evaluate their impact on the blob dynamics, while identifying the physical processes that play the most important role in setting the blob motion, we perform several seeded blob simulations by using five different nonlinear models, implemented in four different simulation codes, each of which is used to simulate the plasma dynamics in the tokamak SOL. The five models are: two isothermal models, one three-dimensional and the other one two-dimensional, written in the STORM module [123] within the BOUT++ framework [18] and named in the following BOUT++3D and BOUT++2D, a three-dimensional cold ion model implemented in the GBS code [20], a two-dimensional model implemented in the HESEL code [124, 125], and a three-dimensional isothermal model implemented in the TOKAM3X code [126]. In the remainder of this section, each of the simulation models and codes is described and a discussion of the differences between the models is provided.

All the equations presented in the following of this section are normalized as illustrated in Appendix A, except for the time scales, which are normalized to  $\omega_{ci}^{-1}$ . Additionally, the magnetic field amplitude in normalized units is defined as  $B = 1/(1+x)$  and the plasma vorticity as  $\omega = \nabla_{\perp}^2 \phi$ . The differential operators  $\nabla_{\perp}^2$  and  $\{\phi, A\}$  are detailed in Section A.3.4, while  $\nabla_{\parallel} = \partial_z$ .

### 5.2.1 BOUT++3D

Assuming cold ions and isothermal electrons, neglecting plasma-neutral interactions, and considering the infinite aspect ratio limit (in particular  $\nabla \cdot \mathbf{b} = 0$ ), the BOUT++3D

drift-reduced Braginskii equations in normalized units can be written as

$$\frac{d^0 n}{dt} = C(n) - nC(\phi) - \nabla_{\parallel} (nv_{\parallel e}) + \mathcal{D}_n(n) + S_n, \quad (5.1)$$

$$\frac{d^0 \omega}{dt} = \frac{C(n)}{n} - v_{\parallel i} \nabla_{\parallel} \omega + \frac{\nabla_{\parallel} j_{\parallel}}{n} + \mathcal{D}_{\omega}(\omega), \quad (5.2)$$

$$\frac{d^0 v_{\parallel i}}{dt} = -v_{\parallel i} \nabla_{\parallel} v_{\parallel i} - \nabla_{\parallel} \phi - \eta_{\parallel} j_{\parallel} - \frac{S_n v_{\parallel i}}{n}, \quad (5.3)$$

$$\frac{d^0 v_{\parallel e}}{dt} = -v_{\parallel e} \nabla_{\parallel} v_{\parallel e} + \frac{m_i}{m_e} \left( \nabla_{\parallel} \phi - \frac{\nabla_{\parallel} n}{n} + \eta_{\parallel} j_{\parallel} \right) - \frac{S_n v_{\parallel e}}{n}. \quad (5.4)$$

Here  $d^0 f/dt = \partial_t f + \{ \phi, f \}$  is the convective derivative that takes into account the  $\mathbf{E} \times \mathbf{B}$  drift,  $C(f) = -g \partial_y f$  is the curvature operator, where  $g = 2/R_0$  represents the strength of the  $\nabla B$  and curvature drifts,  $S_n$  is a particle source defined in Section 5.3,  $\eta_{\parallel} = m_e \nu_{ei} / (1.96 m_i \omega_{ci})$  is the normalized parallel resistivity,  $\nu_{ei} = n_0 e^4 \ln \Lambda / [3 m_e^{1/2} \epsilon_0^2 (2\pi T_{e0})^{3/2}]$  is the electron-ion collision frequency,  $\mathcal{D}_n(n) = D_n \nabla_{\perp}^2 n$  and  $\mathcal{D}_{\omega}(\omega) = \mu_i \nabla_{\perp}^2 \omega$  are perpendicular diffusion operators, where  $D_n = 2 m_e \nu_{ei} / (m_i \omega_{ci})$  and  $\mu_i = 3 \nu_{ei} / (4 \omega_{ci}) \sqrt{m_e / m_i}$  are the normalized particle perpendicular diffusivity and the normalized ion perpendicular viscosity, respectively. We note that in all the models considered in the present chapter the Boussinesq approximation is used to simplify the evaluation of the divergence of the polarization current (the validity of this assumption in modelling the SOL plasma dynamics is discussed in Refs. [91–93]).

Equations (5.1)-(5.4), supplemented by standard sheath boundary conditions [131] [i.e.  $v_{\parallel i} = \pm 1$  and  $v_{\parallel e} = \pm \exp(-\phi)$  at the target], constitutes the BOUT++3D model, which is implemented within the BOUT++ framework. A first order upwinding scheme is employed to evaluate the parallel advection derivatives, while the Arakawa scheme [132] is used for the perpendicular  $\mathbf{E} \times \mathbf{B}$  advective derivatives. Other derivatives are computed using second order central difference schemes. Time integration is carried out with a variable time-step, variable order, fully implicit Newton-Krylov backwards difference formula solver from the PVODE library [133]. We note that only half of the physical domain is evolved, assuming a symmetric evolution of the blobs with respect to the plane perpendicular to the magnetic field that is midway between the two limiter surfaces. A more detailed discussion of the model is presented in Ref. [123].

### 5.2.2 BOUT++2D

Assuming  $k_{\parallel} = 0$  and linearising the sheath boundary conditions, such that  $v_{\parallel e} = \pm \exp(-\phi) \simeq \pm(1 - \phi)$ , Eqs. (5.1)-(5.4) are integrated in the parallel direction, in order to evolve line-averaged quantities. Consequently, the three-dimensional system of

equations reduces to the following two-dimensional fluid equations

$$\frac{d^0 n}{dt} = C(n) - nC(\phi) - 2\frac{n(1-\phi)}{L_{\parallel}} + \mathcal{D}_n(n) + S_n, \quad (5.5)$$

$$\frac{d^0 \omega}{dt} = \frac{C(n)}{n} + 2\frac{\phi}{L_{\parallel}} + \mathcal{D}_\omega(\omega). \quad (5.6)$$

Here  $L_{\parallel} = 2\pi R_0$  is the connection length in normalized units and  $S_n = 2n_{bg}/L_{\parallel}$  is a particle source defined in Section 5.3. The quantities  $n$ ,  $\omega$ , and  $\phi$  are the line-averaged plasma density, vorticity, and electrostatic potential respectively, and, taking  $n$  as an example, are defined as  $n(x, y) = \int_0^{L_{\parallel}} n(x, y, z) dz / L_{\parallel}$ . We note that a similar model is used in Chapter 4 to investigate the propagation of input uncertainties through the model equations.

Eqs. (5.5)-(5.6) constitute the BOUT++2D model, which is implemented within the BOUT++ framework. For its solution, the same numerical scheme employed in BOUT++3D is used. For a more detailed discussion of this model see Ref. [123].

### 5.2.3 GBS

Assuming cold ions, neglecting the electron thermal conductivity, and considering the infinite aspect ratio limit, the drift-reduced Braginskii equations implemented in the GBS code, Eqs. (A.65)-(A.70), reduce to

$$\frac{d^0 n}{dt} = C(p_e) - nC(\phi) - \nabla_{\parallel} (n v_{\parallel e}) + \mathcal{D}_n(n) + S_n, \quad (5.7)$$

$$\frac{d^0 \omega}{dt} = \frac{C(p_e)}{n} - v_{\parallel i} \nabla_{\parallel} \omega + \frac{\nabla_{\parallel} j_{\parallel}}{n} + \mathcal{D}_\omega(\omega) + \frac{C(G_i)}{3n}, \quad (5.8)$$

$$\frac{d^0 v_{\parallel i}}{dt} = -v_{\parallel i} \nabla_{\parallel} v_{\parallel i} - \frac{\nabla_{\parallel} p_e}{n} - \nu_{in} v_{\parallel i} + \mathcal{D}_{v_{\parallel i}}(v_{\parallel i}) - \frac{2\nabla_{\parallel} G_i}{3n}, \quad (5.9)$$

$$\begin{aligned} \frac{d^0 v_{\parallel e}}{dt} = & -v_{\parallel e} \nabla_{\parallel} v_{\parallel e} - \nu_{en} v_{\parallel e} + \mathcal{D}_{v_{\parallel e}}(v_{\parallel e}) \\ & + \frac{m_i}{m_e} \left( \nabla_{\parallel} \phi - \frac{\nabla_{\parallel} p_e}{n} - 0.71 n \nabla_{\parallel} T_e + \eta_{\parallel} j_{\parallel} - \frac{2\nabla_{\parallel} G_e}{3n} \right), \end{aligned} \quad (5.10)$$

$$\begin{aligned} \frac{d^0 T_e}{dt} = & \frac{2}{3} T_e \left[ \frac{C(p_e)}{n} + \frac{5}{2} C(T_e) - C(\phi) \right] - v_{\parallel e} \nabla_{\parallel} T_e \\ & + \frac{2}{3} T_e \left( 0.71 \frac{\nabla_{\parallel} j_{\parallel}}{n} - \nabla_{\parallel} v_{\parallel e} \right) + \mathcal{D}_{T_e}(T_e) + S_{T_e}. \end{aligned} \quad (5.11)$$

The normalized ion-neutral and electron-neutral collision frequencies,  $\nu_{in}$  and  $\nu_{en}$ , evaluated as described in Ref. [134], are introduced here to mimic collisions with the neutral particles present in a weakly ionized plasma, such as that found in TORPEX, whereas other plasma-neutral interactions are neglected for simplicity. The terms  $S_n$  and  $S_{T_e}$  are the particle and electron temperature sources, respectively. Small perpendicular diffu-

sion terms, written in the form  $\mathcal{D}_a(a) = D_a \nabla_{\perp}^2 a$ , where  $D_a$  are constant coefficients, are introduced mainly for numerical reasons. The two terms representing gyroviscous effects are given by  $G_i = -\eta_{0i} [2\nabla_{\parallel} v_{\parallel i} + C(\phi)]$  and  $G_e = -\eta_{0e} [2\nabla_{\parallel} v_{\parallel e} - C(p_e)/n + C(\phi)]$ . Equations (5.7)-(5.11), supplemented by Bohm's boundary conditions [i.e.  $v_{\parallel i} = \pm c_s$  and  $v_{\parallel e} = \pm c_s \exp(\Lambda - \phi/T_e)$  at the sheath entrance, where  $c_s = \sqrt{T_e}$  and  $\Lambda = 3$  for hydrogen plasmas], constitute the GBS model used in the present chapter. Spatial derivatives are discretized using a second-order finite difference scheme, except for the  $\mathbf{E} \times \mathbf{B}$  advective terms, which are discretized with a second-order Arakawa scheme. Time is advanced using a standard fourth-order Runge-Kutta scheme. For a more detailed discussion of the GBS code, see Appendix A.

In the present chapter, all GBS fields have been separated into background and seeded blob components, and only the latter component is evolved. This allows us to use arbitrary background profiles, with no need to find and implement the appropriate plasma source. To clarify this procedure, we rewrite Eq. (5.7) in the form that is actually solved by GBS

$$\begin{aligned} \frac{\partial n_{bl}}{\partial t} = & - \{ \phi_{bg}, n_{bl} \} - \{ \phi_{bl}, n_{bg} \} - \{ \phi_{bl}, n_{bl} \} + \mathcal{D}_n(n_{bl}) \\ & + (T_{e,bg} + T_{e,bl})C(n_{bl}) + (n_{bg} + n_{bl}) [C(T_{e,bl}) - C(\phi_{bl})] \\ & - (n_{bg} + n_{bl}) \nabla_{\parallel} v_{\parallel e,bl} - n_{bl} \nabla_{\parallel} v_{\parallel e,bg} - (v_{\parallel e,bg} + v_{\parallel e,bl}) \nabla_{\parallel} n_{bl} - v_{\parallel e,bl} \nabla_{\parallel} n_{bg}, \end{aligned} \quad (5.12)$$

where the indexes *bg* and *bl* refer to the background and blob components, respectively. In Eq. (5.12) it has been assumed that the background profiles are constant in time and independent of  $y$ . Equations (5.8)-(5.11) and Bohm's boundary conditions are treated with the same procedure.

It has been verified with a two-dimensional version of the GBS model that there are no significant differences between seeded blob simulations carried out by separating the background and blob quantities with respect to the ones where they are both evolved simultaneously.

### 5.2.4 HESEL

In the HESEL model, the drift-reduced Braginskii equations are reduced to a set of two-dimensional fluid equations by neglecting the instantaneous parallel currents, while retaining the equilibrium one, and estimating the parallel advection terms under the hypothesis  $v_{\parallel i} \nabla_{\parallel} = v_{\parallel e} \nabla_{\parallel} \approx c_s/L_{\parallel}$ . The resulting model, which is implemented in the HESEL code, is presented in Refs. [124,125]. Since the ion temperature dynamics shows a very small impact on the present seeded blob simulation results (see Section 5.2.6), we choose to not show here the ion temperature equation and to present only the cold ion model to simplify the discussion of the differences between the models considered in the present chapter.

Neglecting electron-ion collisions and, therefore, assuming cold ions, the system of equa-



tion presented in Refs. [124, 125] reduces to

$$\frac{dn}{dt} = \hat{C}(p_e) - n\hat{C}(\phi) - \frac{n}{\tau_n} + \mathcal{D}_n(n + \phi), \quad (5.13)$$

$$\nabla \cdot \left( \frac{d^0}{dt} \nabla_{\perp} \phi \right) = \hat{C}(p_e) + D_{\omega}(\omega) - \nu_{in}\omega + \frac{2c_s}{L_{\parallel}} \left[ 1 - \exp \left( \Lambda - \frac{\langle \phi \rangle_y}{\langle T_e \rangle_y} \right) \right], \quad (5.14)$$

$$\frac{3}{2} \frac{d}{dt} p_e = \frac{5}{2} \hat{C} \left( \frac{p_e^2}{n} \right) - \frac{5}{2} p_e \hat{C}(\phi) - \frac{p_e}{\tau_{\parallel, p_e} n} + \frac{3}{2} \mathcal{D}_n(p_e - \phi). \quad (5.15)$$

Here  $df/dt = \partial_t f + \{\phi, f\}/B$  is the convective derivative and  $\hat{C}(f) = -\hat{g}\partial_y f$  is the HESEL curvature operator, with  $\hat{g} = 1/R_0$ . Equilibrium currents to the limiter are approximated by the sheath dissipation term entering in Eq. (5.14), where  $\langle - \rangle_y$  represents the average along  $y$ , with  $\Lambda = 2.8$  in this case. The perpendicular diffusion terms  $\mathcal{D}_n(a) = D_{en}\nabla_{\perp}^2 a$  and  $\mathcal{D}_{\omega}(a) = D_{\omega}\nabla_{\perp}^2 a$  are introduced to describe electron-neutral and ion-ion collisions, where  $D_{en} = \rho_e^2\nu_{en}$  and  $D_{\omega} = \rho_i^2\nu_{ii}$ ,  $\rho_e$  and  $\rho_i$  are respectively the electron and ion Larmor radius in normalized units, and  $\nu_{en}$  and  $\nu_{ii}$  are respectively the electron-neutral and the ion-ion collision frequencies in normalized units. The loss of the plasma density due to the parallel flow is parametrized by the characteristic time  $\tau_n = L_{\parallel}/(2c_s)$ , while the electron pressure parallel dynamics by  $\tau_{\parallel, p_e} = 15L_{\parallel}^2\nu_{en}(1 + 4/\nu_{es})/(128v_e^2)$ , where  $\nu_{es} = L_{\parallel}\nu_{en}/(2v_e)$  and  $v_e$  is the thermal electron velocity in normalized units. The parallel advection of the vorticity is neglected here, because of its small amplitude with respect to the ion-neutral collision drag term  $\nu_{in}\omega$ .

The HESEL model is implemented in the code using the Arakawa scheme to discretize the  $\mathbf{E} \times \mathbf{B}$  advective terms, a finite difference scheme to discretize the  $x$  and  $y$  derivatives, and a stiffly-stable third-order scheme [135] for time integration. A more complete discussion of this code is presented in Refs. [124, 125].

### 5.2.5 TOKAM3X

Assuming cold ions and isothermal electrons, the version of the drift-reduced Braginskii equations evolved by TOKAM3X can be written as

$$\frac{dn}{dt} = C(n) - nC(\phi) - \nabla \cdot \left[ (\Gamma - j_{\parallel}) \mathbf{b} \right] + \mathcal{D}_n(n) + S_n, \quad (5.16)$$

$$\frac{d\Omega}{dt} = 2C(n) - \Omega C(\phi) + \nabla \cdot \left[ \left( j_{\parallel} - \Omega \frac{\Gamma}{n} \right) \mathbf{b} \right] + \mathcal{D}_{\Omega}(\Omega), \quad (5.17)$$

$$\frac{d\Gamma}{dt} = -\Gamma C(\phi) - C(\Gamma) - \nabla \cdot \left( \Gamma \frac{\Gamma}{n} \mathbf{b} \right) - 2\nabla_{\parallel} n + \mathcal{D}_{\Gamma}(\Gamma), \quad (5.18)$$

$$0 = n\nabla_{\parallel} \phi - \nabla_{\parallel} n + \eta_{\parallel} n j_{\parallel}, \quad (5.19)$$

where  $\Omega = \nabla \cdot (\nabla_{\perp} \phi / B^2)$  is the plasma vorticity that takes into account magnetic field variations,  $\Gamma = nv_{\parallel i}$  is the ion parallel momentum and  $S_n$  is a particle source [see Section 5.3, Eq. (5.22) for its definition]. Small perpendicular diffusion terms of the form



$\mathcal{D}_a(a) = D_a \nabla_{\perp}^2 a$ , where  $D_a$  are constant coefficients, are introduced to dissipate turbulent structures of size comparable to the grid spacing.

Equations (5.16)-(5.19), completed by the linearized Bohm's boundary conditions [i.e.  $\Gamma = \pm n$  and  $j_{\parallel} = \pm n [1 - \exp(\Lambda - \phi)] \simeq \pm n (\phi - \Lambda)$  at the target], are solved by the TOKAM3X code with a first order operator splitting. Advection terms and source terms are first advanced explicitly, using a shock-capturing algorithm (the Roe-Marquina scheme based on the WENO interpolation [136]). Parallel current terms are advanced using a fully implicit three-dimensional solver in order to capture the associated fast dynamics without considerably constraining the time step. Finally, perpendicular diffusion terms are advanced implicitly. The spatial discretization is done based on conservative finite differences evaluated on a structured flux-surface aligned mesh. A more detailed discussion of the TOKAM3X code is presented in Ref. [126].

### 5.2.6 Summary of analogies and differences among the physical models

Besides the differences related to the numerical schemes used to evolve the five models, which are neglected here as we consider simulations that are numerically converged, the five models differ because of several assumptions made to simplify the drift-reduced Braginskii equations. The remainder of this section is dedicated to a discussion of these differences. To examine the differences between the models, we note that Eqs. (5.16)-(5.19) can be recast in the GBS and BOUT++ form by expressing  $\Omega$ ,  $\Gamma$  and  $j_{\parallel}$  in terms of the quantities evolved by these codes. Moreover, we note that it has been verified with the TOKAM3X code that, in the considered blob scenarios, the Boussinesq approximation has a negligible influence on the numerical results. The analogies and differences among the codes can be summarized as follows.

(i) Two-dimensional closures. In order to reduce the three-dimensional model to a two-dimensional set of equations, in BOUT++2D we impose the sheath dissipation closure, that is  $k_{\parallel} = 0$  [123], while in HESEL we impose the vorticity advection closure by approximating  $v_{\parallel i} \nabla_{\parallel} = v_{\parallel e} \nabla_{\parallel} \approx c_s / L_{\parallel}$  [123]. By imposing the sheath dissipation closure, one assumes that the parallel gradients are negligible, and that the filaments extend from target to target. On the other hand, by applying the vorticity advection closure, instantaneous sheath currents are neglected. As discussed in Section 5.5 and shown previously in Ref. [123], these approximations may have a strong impact on the simulation results.

(ii) Boundary conditions. The BOUT++3D and GBS models are supplemented by full Bohm's boundary conditions, TOKAM3X and BOUT++2D employ linearised Bohm's boundary conditions, while the HESEL model makes use of the weak sheath formulation, and therefore the equilibrium currents to the limiter are described by the sheath dissipation term entering in Eq. (5.14). It has been verified with BOUT++3D that the linearisation of the boundary conditions has negligible impact on the simulation results presented herein.

(iii) Temperature effects. In the present work the BOUT++2D, BOUT++3D, and TOKAM3X models assume isothermal electrons and cold ions, GBS assumes cold ions, while HESEL evolves the ion dynamics, assuming ions initially at room temperature ( $T_i = 0.025$  eV at  $t = 0$ ). A detailed investigation of the  $T_e$  effects on the simulation results is presented in Section 5.4. The  $T_i$  influence on the simulation results was investigated with HESEL, showing negligible impact on the blob dynamics, and will not be further discussed.

(iv) Magnetic field equilibrium and background profiles. First, regarding the magnetic geometry, we note that BOUT++2D, BOUT++3D, and GBS models are here written in the infinite aspect ratio limit, while TOKAM3X retains the  $\nabla \cdot \mathbf{b}$  terms. Moreover, in TOKAM3X and HESEL the variation of the magnetic field is retained in the  $\mathbf{E} \times \mathbf{B}$  advective terms, while it is neglected in BOUT++2D, BOUT++3D, and GBS. Due to the TORPEX large aspect ratio, these approximations have a negligible influence on the results. We also note that in BOUT++2D, BOUT++3D, GBS, and TOKAM3X models, the curvature operator,  $C(-)$ , is twice as large as the curvature operator  $\hat{C}(-)$  appearing in HESEL (i.e.  $g = 2\hat{g}$ ). Therefore, the ballooning instability drive is halved in HESEL with respect to the one present in the other four models (a reduced ballooning drive leads to slower blob radial motion [128]). Finally, the background profiles are time-independent and can be arbitrarily imposed in the version of GBS used here because of the separation between background and blob components. On the other hand, in BOUT++2D, BOUT++3D, and TOKAM3X the full quantities are evolved, and the equilibrium profiles are sustained by appropriate source terms, while in HESEL the full quantities are evolved, but, assuming slow variation of the plasma background with respect to the time-scale evolution of blobs, no source terms are introduced to sustain the background profiles (this is justified a posteriori by the simulation results).

(v) Electron inertia. In BOUT++3D the electron inertia is retained in both the parallel ion and electron momentum equations, Eqs. (5.3)-(5.4), in GBS it is neglected in the ion parallel momentum equation, Eq. (5.9), while in TOKAM3X it is neglected in both the Eqs. (5.18)-(5.19). As a matter of fact, the simulations presented in Section 5.5 show that the electron inertia has negligible influence on the blob motion.

(vi) Dissipative terms. In BOUT++2D and BOUT++ 3D the perpendicular diffusion coefficients are set using the physical values of the electron-ion and ion-ion collision frequencies, while in HESEL electron-neutral and ion-neutral collisions are also taken into account. These classical diffusion coefficients are computed according to Ref. [137]. In contrast, in GBS and TOKAM3X arbitrary perpendicular dissipative terms are introduced ( $D_a \simeq 5 \cdot 10^{-4} - 10^{-3}$ ), and we ensured that they have a negligible influence on the simulation results using GBS to perform a sensitivity scan of the diffusion coefficients over two orders of magnitude, i.e. decreasing and increasing the value of the diffusion coefficients used for the simulation by a factor of ten. Moreover, in GBS the  $\nu_{en}$  and  $\nu_{in}$  terms are introduced to mimic the electron-neutral and ion-neutral collisions, while the  $G_e$  and  $G_i$  terms model the plasma viscosity. A sensitivity scan of these dissipative coefficients over two orders of magnitude show that they have a negligible impact on the simulation results.

In Section 5.4 and 5.5 we discuss the influence of the approximations listed above on the blob dynamics, comparing the simulations performed with the five models among themselves and against experimental data. In particular, we identify the modelling of the electron temperature and the parallel current closure used to derive the two-dimensional models as the most important differences among the models, and therefore, we focus our attention on those.

### 5.3 Seeded blob simulations

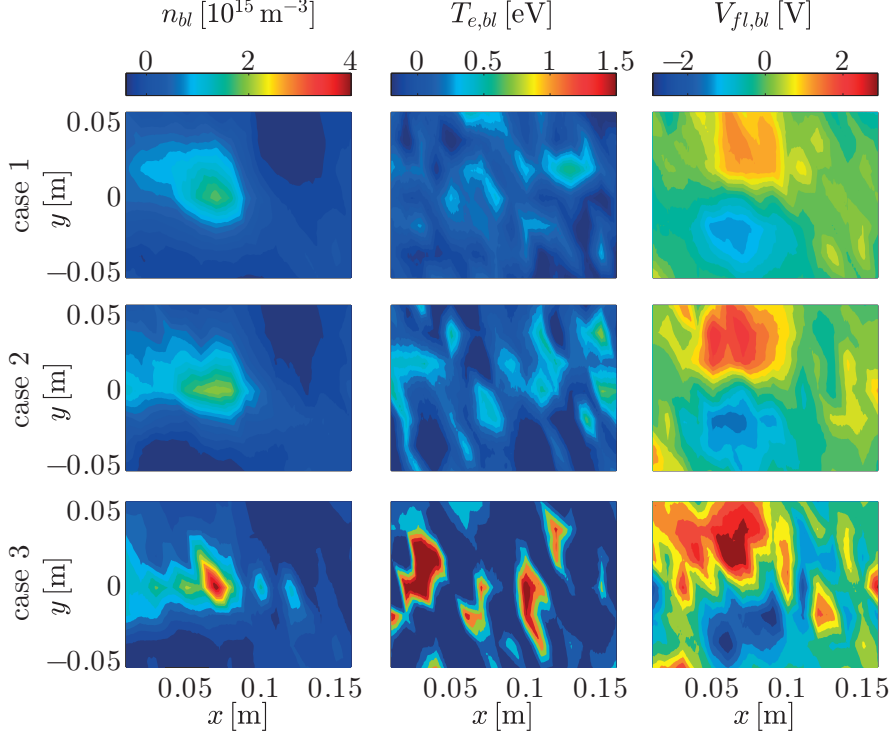
Using the five models presented in Section 5.2, we perform simulations of seeded blob motion with three different sets of initial conditions. This allows us to compare the simulations among themselves and validate them against experimental observations over a set of different conditions. The three different cases correspond to considering three different amplitude windows for the blob detection in the  $I_{sat}$  reference signal provided by the SLP tips. More precisely, we consider trigger events for which the  $I_{sat}$  peaks of the reference signal fall in (i) the interval  $2.0\sigma - 2.75\sigma$ , where  $\sigma$  is the standard deviation of the reference signal ( $\sigma/\langle I_{sat} \rangle_t \approx 0.5$ , where  $\langle I_{sat} \rangle_t$  is the time-average of the  $I_{sat}$  signal), (ii) the interval  $2.75\sigma - 3.5\sigma$ , and (iii) the interval  $3.5\sigma - 4.25\sigma$ . In the following, these three scenarios are dubbed “case 1”, “case 2”, and “case 3”, respectively. The three trigger windows result in blobs with different density peak values  $n_0$ , with  $n_0/n_{bg} \approx 0.85, 1.0, 1.9$  for the three cases, where  $n_{bg}$  is the time-averaged plasma density at the reference probe position. These blobs are found to have different velocities and internal stability properties. For the three cases, the blob profiles at the detection time  $t = 0$  are shown in Fig. 5.2.

The simulations are initialized according to the experimental measurements. The background profiles of  $n$  and  $T_e$  are evaluated as the median value of the time-dependent signal reconstructed from the fit of the  $I - V$  curves, and are shown in Fig. 5.1 (middle and right panels). As the dependence of the background profiles on the  $y$  coordinates is weak, they are fitted with expressions that depend only on  $x$ ;

$$n_{bg}(x) = \alpha(\beta x)^\gamma + \delta, \quad (5.20)$$

$$T_{e,bg}(x) = \epsilon \cdot \exp(\zeta x), \quad (5.21)$$

where  $\alpha = -4.2 \cdot 10^{17} \text{ m}^{-3}$ ,  $\beta = 1 \text{ m}^{-1}$ ,  $\gamma = 2.9$ ,  $\delta = 2.5 \cdot 10^{15} \text{ m}^{-3}$ ,  $\epsilon = 2.8 \text{ eV}$ , and  $\zeta = -5.9 \text{ m}^{-1}$ . We note that the measurements of the plasma quantities are taken on a poloidal plane at a distance of approximately 3 cm from the limiter and no experimental information is available on the parallel dependence of the profiles. This introduces an uncertainty in setting the  $z$  dependence of the equilibrium profiles (and blob initial conditions) in the three-dimensional codes. However, in previous TORPEX experiments it has been observed that the plasma density background profile is approximately flat along  $z$ , except for a drop in the proximity of the limiter [138]. Therefore, as suggested



**Figure 5.2** – Plasma density (first column), electron temperature (second column), and floating potential (third column) at detection time  $t = 0$ , for “case 1” (first row), “case 2” (second row), and “case 3” (third row) conditionally averaged blobs, from which have been subtracted the backgrounds.

in Ref. [123], a density source

$$S(x, z) = n_{bg}(x) \cdot \frac{10 \exp(10|z - \pi|/\pi)}{\pi [\exp(10) - 1]} \quad (5.22)$$

is introduced in BOUT++3D and TOKAM3X. It follows that the source term to be used in BOUT++2D is  $S_n = n_{bg}(x)/\pi$ . In GBS we linearise the model equations and, therefore, there is no need to introduce plasma sources. On the other hand, we have to impose the density background, and we choose to impose the one that is produced by using the source of Eq. (5.22) in BOUT++3D. In HESEL there are no plasma sources, and  $n_{bg}$  is imposed at  $t = 0$  accordingly to Eq. (5.20). For the electron background temperature profile, we note that in BOUT++2D, BOUT++3D, and TOKAM3X, the electron dynamics is assumed to be isothermal. Therefore, in these models, a uniform background temperature is imposed, and a sensitivity study of  $T_{e,bg}$  is performed (see Section 5.4). On the other hand, in GBS and HESEL,  $T_{e,bg}$  is expressed according to Eq. (5.21) and it is assumed constant along  $z$ . Moreover, we note that HESEL describes finite ion temperature effects. Assuming the ion temperature as the ambient temperature (which approximatively corresponds to the neutral temperature), an uniform  $T_i = 0.025$  eV is imposed at  $t = 0$ .

Finally, the background profiles of  $\phi$ ,  $\omega$ ,  $v_{\parallel i}$  and  $v_{\parallel e}$  are obtained by imposing Bohm’s

boundary conditions at the limiter plate and assuming no net background current flowing to the limiter, as discussed in Ref. [123]. The blob initial conditions are set by using the conditionally averaged profiles at the detection time  $t = 0$ . The experimental profiles of  $n_{bl}$ ,  $T_{e,bl}$ , and  $V_{fl,bl}$  have been fitted, imposing a monopolar structure in the poloidal plane for  $n_{bl}$  and  $T_{e,bl}$ , and a dipolar structure for  $V_{fl,bl}$ , according to the expressions

$$n_{bl}(x, y) = n_0 \cdot \exp \left[ - \left( \frac{(x - x_0)}{\sigma_{n,x}} \right)^2 - \left( \frac{y}{\sigma_{n,y}} \right)^2 \right], \quad (5.23)$$

$$T_{e,bl}(x, y) = T_0 \cdot \exp \left[ - \left( \frac{(x - x_0)}{\sigma_{T,x}} \right)^2 - \left( \frac{y}{\sigma_{T,y}} \right)^2 \right], \quad (5.24)$$

$$V_{fl,bl}(x, y) = V_1 \cdot \exp \left[ - \left( \frac{(x - x_0)}{\sigma_{V,x,1}} \right)^2 - \left( \frac{y - y_1}{\sigma_{V,y,1}} \right)^2 \right] \\ + V_2 \cdot \exp \left[ - \left( \frac{(x - x_0)}{\sigma_{V,x,2}} \right)^2 - \left( \frac{y - y_2}{\sigma_{V,y,2}} \right)^2 \right], \quad (5.25)$$

where  $x_0 = 0.07$  m and the value of the other parameters appearing in Eqs. (5.23)-(5.25) are summarized in Table 5.1 for the three cases.

**Table 5.1** – Parameters used to initialize the seeded blob simulations, deduced by fitting the experimental measurements using Eqs. (5.23)-(5.25).

	case 1	case 2	case 3
$n_0$ [ $10^{15}$ m $^{-3}$ ]	$1.975 \pm 0.135$	$2.335 \pm 0.325$	$4.395 \pm 0.855$
$\sigma_{n,x}$ [cm]	$2.20 \pm 0.20$	$2.40 \pm 0.30$	$1.65 \pm 0.45$
$\sigma_{n,y}$ [cm]	$2.40 \pm 0.20$	$2.10 \pm 0.20$	$1.75 \pm 0.25$
$T_0$ [eV]	$0.345 \pm 0.065$	$0.960 \pm 0.250$	$1.730 \pm 0.280$
$\sigma_{T,x}$ [cm]	$1.05 \pm 0.15$	$1.05 \pm 0.25$	$0.80 \pm 0.20$
$\sigma_{T,y}$ [cm]	$3.65 \pm 1.05$	$1.45 \pm 0.25$	$2.85 \pm 0.95$
$V_1$ [V]	$2.330 \pm 0.170$	$4.600 \pm 0.740$	$4.715 \pm 0.405$
$\sigma_{V,x,1}$ [cm]	$3.55 \pm 0.25$	$3.25 \pm 0.25$	$4.95 \pm 0.35$
$y_1$ [cm]	$2.55 \pm 0.25$	$2.60 \pm 0.20$	$1.15 \pm 0.35$
$\sigma_{V,y,1}$ [cm]	$2.95 \pm 0.05$	$3.10 \pm 0.20$	$4.90 \pm 0.60$
$V_2$ [V]	$-1.540 \pm 0.140$	$-2.350 \pm 0.550$	$-6.155 \pm 0.965$
$\sigma_{V,x,2}$ [cm]	$3.10 \pm 0.20$	$2.75 \pm 0.35$	$2.95 \pm 0.45$
$y_2$ [cm]	$-2.10 \pm 0.40$	$-0.50 \pm 0.80$	$-2.45 \pm 0.15$
$\sigma_{V,y,2}$ [cm]	$4.00 \pm 0.30$	$4.75 \pm 0.45$	$2.50 \pm 0.30$

The expressions of  $n_{bl}$  and  $V_{fl,bl}$  in Eqs. (5.23) and (5.25) are relatively well supported by the experimental measurements. On the other hand, the fitting of  $T_{e,bl}$  using Eq. (5.24) is only partially justified, due to the high uncertainties affecting the measurements (see Fig. 5.2). Because of these uncertainties, we also impose  $\phi_{bl} \approx V_{fl,bl}$ , neglecting the  $\Delta T_e$  term (previous studies show that the  $\Delta T_e$  term increases the blob spinning [139]). The three-dimensional initial profiles are obtained by using Bohm's boundary conditions at

the limiter, and assuming that  $n_{bl}$ ,  $T_{e,bl}$ , and  $\phi_{bl}$  are constant along the parallel direction, while  $v_{\parallel e,bl}$  and  $v_{\parallel i,bl}$  are a linear function of  $z$ . We note that in the isothermal models  $T_{e,bl} = 0$  is imposed. Finally, we enforce  $\omega_{bl} = \nabla_{\perp}^2 \phi_{bl}$ . The influence of the approximations introduced to initialize the seeded blob simulations on the numerical results is discussed in Section 5.4.

The seeded blob motion is simulated on a time interval that is equal to the experimental blob correlation time, i.e. approximatively  $50 \mu s$ . Longer simulations are not useful for comparison with the experiments, because the coherence of the conditionally averaged blob is completely lost on longer timescales.

To compare the numerical simulations against experimental measurements, we focus our attention on the blob radial and vertical motion. The position of the blob is computed as follows. First, for the simulation results, the blob ion saturation density current profile is computed as

$$j_{bl}(x, y, t) = \frac{1}{2}[n_{bg}(x) + n_{bl}(x, y, t)]\sqrt{T_{e,bg}(x) + T_{e,bl}(x, y, t)} - \frac{1}{2}n_{bg}(x)\sqrt{T_{e,bg}(x)}, \quad (5.26)$$

while for the experimental results it is simply given by

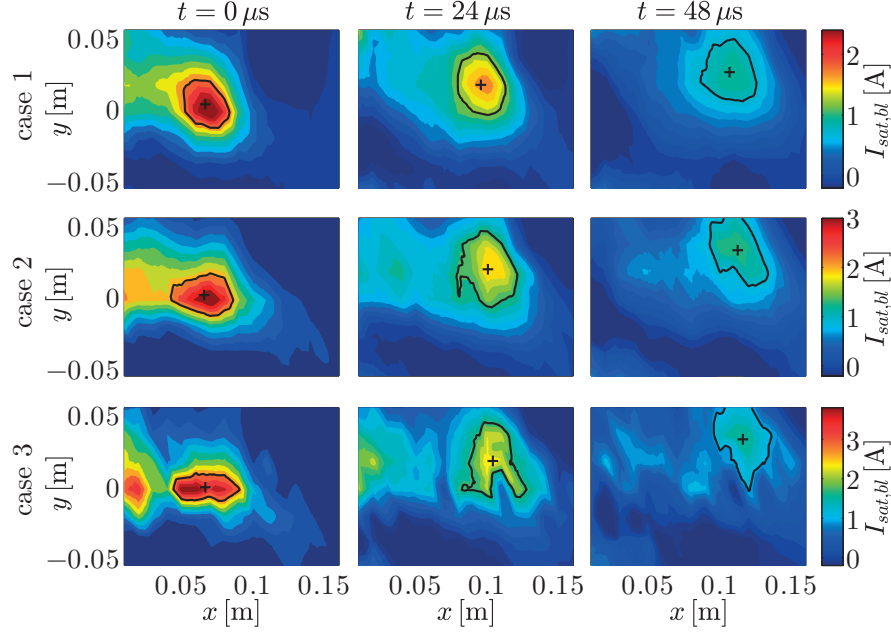
$$j_{bl}(x, y, t) = \frac{I_{sat}(x, y, t) - \langle I_{sat}(x, y, t) \rangle_t}{A}, \quad (5.27)$$

where  $I_{sat}(x, y, t)$  is the measured ion saturation current,  $\langle - \rangle_t$  denotes the median value in time, and  $A$  is the projected area of the single-sided LP probe. Second, the  $j_{bl}$  signal is averaged in space,  $j_{bl}(t) = \langle j_{bl}(x, y, t) \rangle_{x,y}$ , where  $\langle - \rangle_{x,y}$  denotes averaging along the  $x$  and  $y$  coordinates on the entire physical domain. Third, we identify the surface  $S(t)$  which satisfies  $\langle j_{bl}(x, y, t) \rangle_{S(t)} = 0.2 \cdot j_{bl}(t)$ , where  $\langle - \rangle_{S(t)}$  denotes the average carried out on the domain defined by  $S(t)$ . Finally, the position of the blob is identified as the geometric center of the surface  $S$  according to

$$\begin{aligned} x_{bl}(t) &= \frac{\iint_{S(t)} x dx dy}{\iint_{S(t)} dx dy}, \\ y_{bl}(t) &= \frac{\iint_{S(t)} y dx dy}{\iint_{S(t)} dx dy}. \end{aligned} \quad (5.28)$$

The use of this procedure allows us to reduce the sensitivity of the results to the noise present in the profiles. The radial and vertical velocities of the blob are simply defined as  $v_x(t) = dx_{bl}(t)/dt$  and  $v_y(t) = dy_{bl}(t)/dt$ . To exemplify the use of this procedure, in Fig. 5.3 we consider the experimental measurements associated with blobs at the three times  $t = 0$ ,  $t = 24 \mu s$ , and  $t = 48 \mu s$ . The black contours represent the boundaries of the surfaces  $S$  and the black crosses denote the blob positions,  $x_{bl}$  and  $y_{bl}$ .

We note that, while it is possible to perform quasi-steady state turbulence simulations and compare the numerical results with the experimental measurements, in the present chapter we choose to consider seeded blob simulations in order to decrease the computational cost of the simulations and simplify the comparison of the numerical results. As a matter of fact, we ensure that the velocity of conditionally-averaged, turbulence-



**Figure 5.3** – Experimentally measured  $I_{sat}$  profiles, from which have been subtracted the backgrounds, at  $t = 0$  (first column),  $t = 24 \mu s$  (second column), and  $t = 48 \mu s$  (third column), for the “case 1” (first row), “case 2” (second row), and “case 3” (third row) blobs. The black contours represent the boundaries of the surfaces which satisfy  $\langle j_{bl}(x, y, t) \rangle_{S(t)} = 0.2 \cdot j_{bl}(t)$  and the black crosses denote the blob positions  $x_{bl}$  and  $y_{bl}$ .

generated blobs does not significantly differ from the velocity of a seeded blob by proceeding as follows. Applying the conditional average technique described in Section 5.1 to a sufficiently long two-dimensional fully turbulent GBS simulation, we obtain the conditionally-averaged profiles associated with blobs, and we use these profiles to initialize a two-dimensional seeded blob simulation. The comparison of the velocities obtained from the seeded blob simulation and the conditionally-averaged blob, which is not displayed here, shows that the difference between the two velocities is to within an error of 10%.

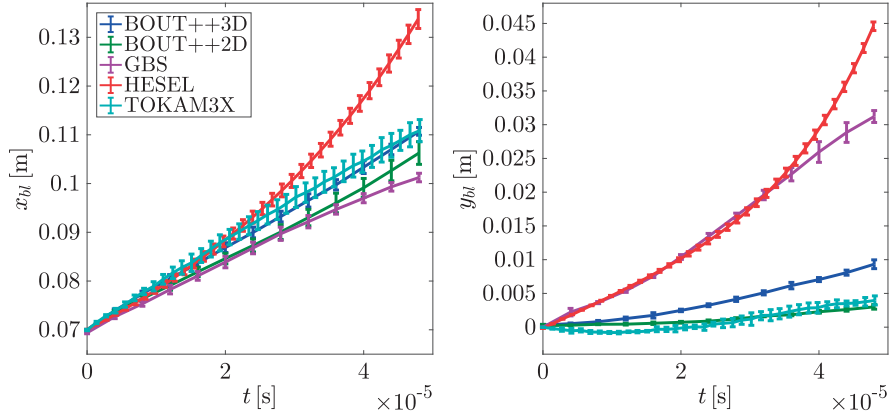
## 5.4 Sensitivity studies

In order to compare the simulation results among each other and against the experimental measurements, four sensitivity scans are performed. We first investigate the sensitivity of the simulation results to the input parameters and initial conditions. Second, we focus our attention on the influence of the equilibrium electron temperature profiles on the numerical results. Third, we analyse the impact of the electron temperature dynamics on the blob motion. Finally, we study the sensitivity of the simulation results to the numerical parameters, such as the diffusion coefficients introduced in GBS and TOKAM3X.

To estimate the effect of the uncertainties found in setting the initial conditions (dis-



cussed in Section 5.3) on the simulation results, we first estimate the confidence intervals of the fitting parameters (Table 5.1). Second, we perform a sensitivity scan of the blob size ( $\sigma_x$  and  $\sigma_y$  coefficients) and of the peak-to-peak value of its dipolar potential, as they are expected to be the parameters that affect the blob velocity the most. More precisely, we perform five simulations for each of the three cases: one simulation, dubbed standard simulation, initialized with the reference fitting parameters, two simulations setting the size of the blob using the minimum and maximum values within the confidence interval of the fitting parameters, and two other simulations with the minimum and maximum peak-to-peak values of the dipolar profile of the plasma potential. Third, we compute the maximum of the difference between the standard simulation and the other four simulations, and we use this as the measure of the uncertainty affecting the numerical results.



**Figure 5.4** – Radial (left panel) and vertical (right panel) position of the blob as function of time for “case 1”, with error bars representing the uncertainties affecting the numerical results due to the uncertainties on the initial conditions.

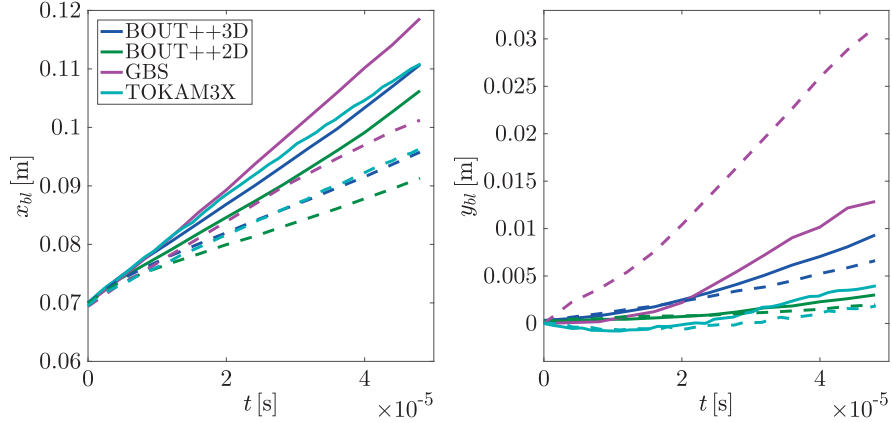
The results of the sensitivity scan for the “case 1” blob are shown in Fig. 5.4, where the error bars represent the evaluated uncertainties. It is evident that the uncertainties affecting the radial and vertical position of the blobs are relatively small, which ensures that the uncertainties on the blob initial conditions do not strongly affect the simulation results. Similar results (not shown) are obtained for “case 2” and “case 3”.

Next, we perform a sensitivity scan of the electron temperature background value. This is motivated by the fact that, while BOUT++3D, BOUT++2D, and TOKAM3X assume a uniform  $T_{e,bg}$ , the experimental temperature background profile shows a strong radial variation.

The results of this sensitivity study are presented in Fig. 5.5. BOUT++3D, BOUT++2D, and TOKAM3X are used to carry out two simulations each, one with  $T_{e,bg} = 2.8$  eV ( $T_{e,bg}$  value at  $x = 0.0$  cm, corresponding to the maximum value of  $T_{e,bg}$  over the considered domain) and one with  $T_{e,bg} = 1.85$  eV ( $T_{e,bg}$  value at  $x = 0.07$  cm, corresponding to the  $T_{e,bg}$  value at the position where the blob is initialized). Moreover, two simulations are performed with GBS, one imposing a uniform  $T_{e,bg} = 2.8$  eV, and one with  $T_{e,bg}$  set according to Eq. (5.21).

Figure 5.5 shows that the radial velocity of the blob is strongly affected by  $T_{e,bg}$  (a larger



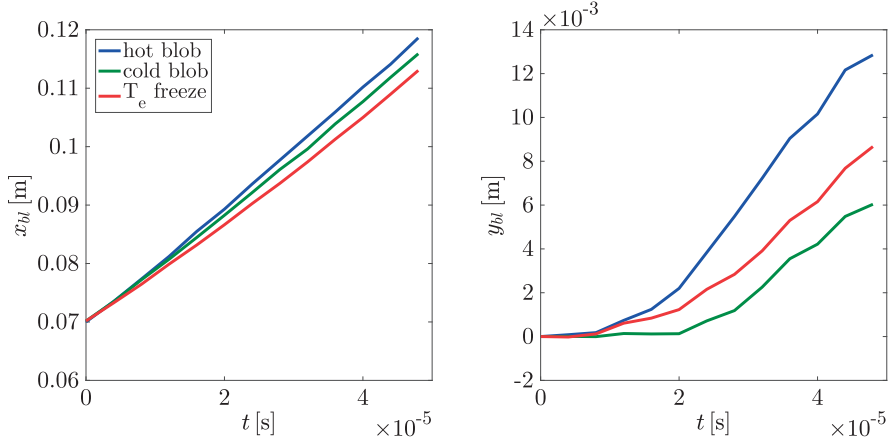


**Figure 5.5** – Radial (left panel) and vertical (right panel) position of the blob as function of time for “case 1”. Solid curves represent the simulation results with  $T_{e,bg} = 2.8$  eV, dashed curves consider  $T_{e,bg} = 1.85$  eV for BOUT++3D, BOUT++2D, and TOKAM3X and  $T_{e,bg}$  given by Eq. (5.21) for GBS.

$T_{e,bg}$  leads to a larger radial velocity). Furthermore, GBS simulations point out that the blob radial motion is faster for  $T_{e,bg} = 2.8$  eV than for the experimental  $T_{e,bg}$  profile. Regarding the vertical motion of the blob, we observe that the radial variation of  $T_{e,bg}$  strongly impacts the blob dynamics, while varying a uniform  $T_{e,bg}$  value has a minor impact. In fact, when a radially dependent profile of  $T_{e,bg}$  is considered, by imposing Bohm’s boundary conditions at the limiter and no net parallel background current flowing to the target, we obtain a radially dependent electrostatic background potential, which leads to a positive  $v_{E \times B}$  in the vertical direction. Similar results are obtained for the “case 2” and “case 3” blobs. Consequently, the  $T_{e,bg}$  profile considerably affects the simulation results. In Section 5.5 this has to be taken into account in the comparison of the simulation results among each other and with experimental measurements.

Then, we simulate with the GBS code an isothermal blob (i.e. we impose  $T_{e,bl} = 0$  at all times), an initially thermalized blob (i.e. we impose  $T_{e,bl} = 0$  at  $t = 0$  and then let the blob temperature evolve), and an initially hot blob [i.e. we impose  $T_{e,bl} = T_{e,bl}(x, y)$  at  $t = 0$ , accordingly to Eq. (5.24), and then let the blob temperature evolve]. A uniform  $T_{e,bg} = 2.8$  eV is imposed. This is motivated by two things. First, as discussed in Section 5.3, high uncertainties are affecting the experimental measurements of the electron temperature. Second, in BOUT++3D, BOUT++2D, and TOKAM3X the blob is assumed isothermal, while in GBS and HESEL the electron temperature is evolved. In Fig. 5.6 we present the results of this study. Considering the radial motion, we observe that the isothermal blob is the slowest one, while the hot blob is the fastest. However, the motion of the blob is only slightly affected by temperature effects. Moreover, Fig. 5.6 shows that the impact of the isothermal blob assumption on the blob vertical velocity is very small. This indicates that the presence of a radially-varying  $T_{e,bg}$  profile is the main drive of the vertical motion, as discussed above.

Finally, we would like to make a few remarks. First, we note that all the simulations used in this chapter are converged with respect to the temporal and spatial discretization and, therefore, the numerical uncertainty can be neglected. Second, we note that,



**Figure 5.6** – Radial (left panel) and vertical (right panel) position of the blob as function of time, obtained from GBS simulations of “case 1” with  $T_{e,bg} = 2.8$  eV. The blue curves correspond to an initially hot blob [i.e.  $T_{e,bl} = T_{e,bl}(x, y)$  at  $t = 0$ , as in Eq. (5.24)], the green curves correspond to an initially thermalized blob (i.e.  $T_{e,bl} = 0$  at  $t = 0$  and then letting the blob temperature evolve), and the red curves correspond to an isothermal blob (i.e. imposing  $T_{e,bl} = 0$  at all times).

performing several sensitivity scans, it has been verified that the values of the numerical parameters, such as the diffusion coefficients introduced in GBS and TOKAM3X, do not significantly affect the simulation results. Third, we remark that in Chapter 4 we discuss a rigorous methodology for estimating the impact of uncertainty propagation related to not precisely known input parameters on the simulation results. However, since we do not have developed yet a fully spectral code for investigating the blob dynamics, we cannot apply here such methodology.

## 5.5 Analysis and validation of the simulation results

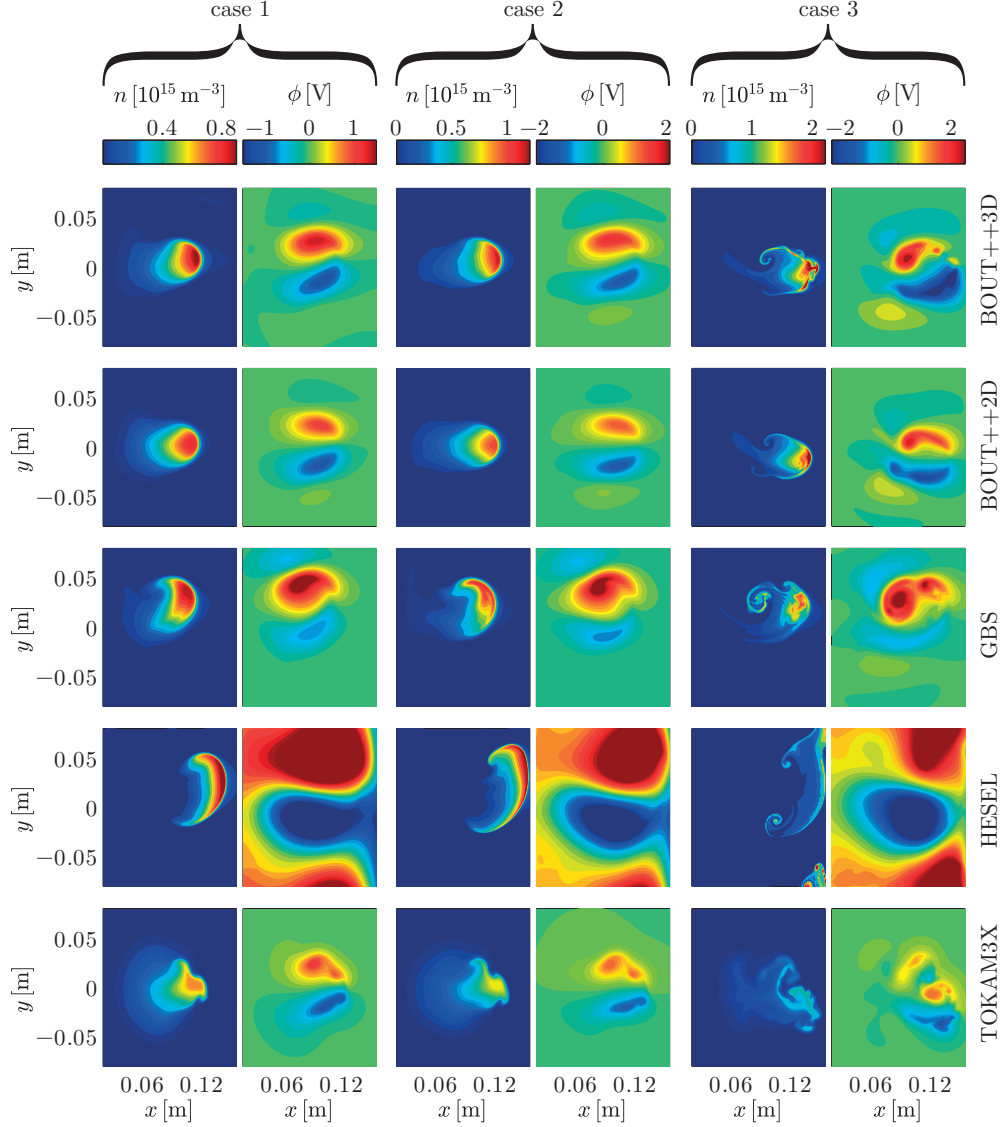
First, focusing our attention on the qualitative analysis of the blob simulation results, we present in Fig. 5.7 the two-dimensional poloidal profiles of plasma density and electrostatic potential associated with “case 1”, “case 2”, and “case 3” blobs at  $t = 48 \mu\text{s}$ , for the five simulation models. We consider  $T_{e,bg} = 2.8$  eV in the isothermal and Eq. (5.21) in the non-isothermal models. Several observations can be made from these results.

(i) Noticeable differences exist between the “case 1” and “case 2” blobs, and the “case 3”. In particular, the size of the “case 3” blob is significantly smaller than in the two other cases. This leads to steeper gradients and stronger secondary instabilities, consistent with the numerical results.

(ii) The BOUT++2D results are qualitatively similar to the ones from BOUT++3D, the main difference being the amplitude of the density profiles. This is due to the fact that in BOUT++2D line-averaged quantities are evolved and plotted, while for BOUT++3D snapshots on a poloidal plane close to the limiter are shown, where the blob density is smaller than at the center of the device.

(iii) Comparing the results of BOUT++3D, GBS, and TOKAM3X, although we

## 5.5. Analysis and validation of the simulation results



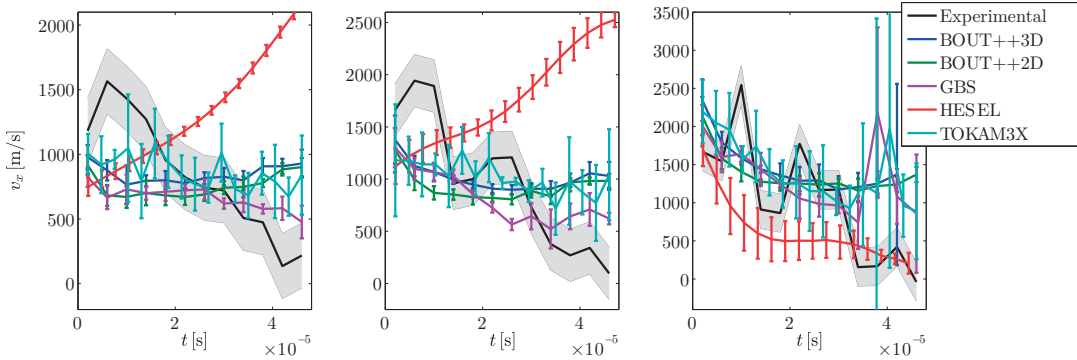
**Figure 5.7** – Plasma density and electrostatic potential, from which have been subtracted the backgrounds, for the “case 1”, “case 2”, and “case 3” blobs, simulated with BOUT++2D, BOUT++3D, GBS, HESEL, and TOKAM3X, at  $t = 48 \mu\text{s}$ . We note that for BOUT++2D and for HESEL the line-averaged quantities  $n(x, y)/2$  and  $\phi(x, y)$  are represented, while the profiles of  $n(x, y, z)$  and  $\phi(x, y, z)$  are displayed for the three-dimensional models at the simulated poloidal plane closest to the limiter.

observe a similar global evolution of the blobs, some differences in the details of the structures are apparent. In TOKAM3X the blobs are subject to fingering effects, not visible in the other simulations. The shape of the BOUT++3D blobs is rounder than in GBS and the blob tails are less pronounced in GBS (tests show that this is related to the plasma-neutral collisions, not taken into account by the other three-dimensional models). Moreover, we note that the blobs simulated with GBS show an upward motion, and are spinning counterclockwise. The upward motion is related to the  $\mathbf{E} \times \mathbf{B}$  vertical motion due to the radial dependence of the electron temperature background, as discussed in

Section 5.4. The spinning occurs because of effects of the evolving electron temperature on the blob plasma potential, which is consistent with the observations in Ref. [139].

(iv) Focusing on the HESEL results, we note that the blobs are more “mushroom-like” and show a completely different evolution than in the other four models. This is related to the HESEL assumption that diamagnetic currents are predominately closed through polarization currents.

The analysis of the differences among the five models helps us understand the results of the validation of the simulation results against the experimental measurements. The radial and vertical blob velocities produced by the simulations and as measured from the experiment are plotted versus time in Figs. 5.8 and 5.9.



**Figure 5.8** – Radial velocity of the blob as function of time for “case 1” (left panel), “case 2” (middle panel), and “case 3” (right panel), obtained from numerical simulations and experimental measurements (the gray shaded region represents the experimental uncertainty due to the finite spatial resolution of the probes).

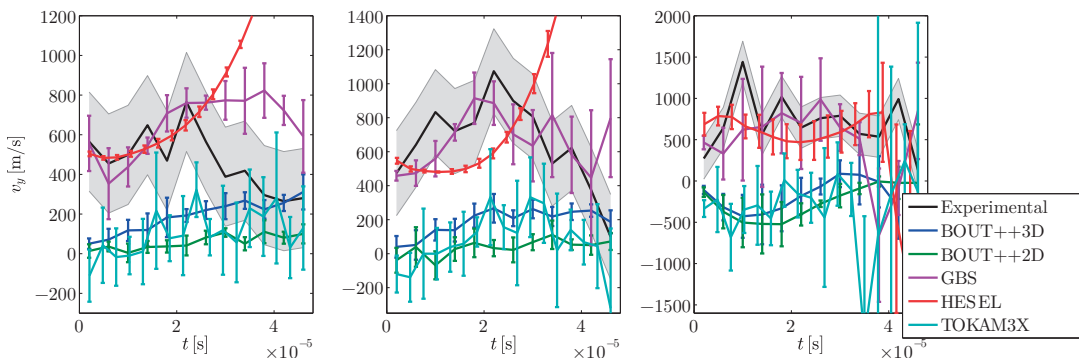
Regarding the experimental measurements of the radial velocity, despite some fluctuations mainly due to experimental uncertainties, it is visible that blobs decelerate as they move outwards, for each of the three cases. The radial velocity of the blobs simulated with BOUT++3D, BOUT++2D, GBS, and TOKAM3X also decreases in time. However, particularly in “case 1” and “case 2”, the blob deceleration is weaker in the simulation results than in experiments, and the initial velocity peak is not well captured. On the other hand, the radial blob velocity simulated with HESEL shows a completely different evolution. In fact, while moving outwards, the HESEL “case 1” and “case 2” blobs accelerate significantly, while the “case 3” blob decelerates.

To perform a quantitative comparison between experiments and numerical results, and to gain a deeper insight on the blob dynamics, we average the experimental and the simulated radial blob velocities between  $t = 0$  and  $t = 48 \mu\text{s}$ , for the three cases, and we theoretically predict the radial blob velocity according to Ref. [128] as

$$v_x = \frac{\sqrt{\frac{2a}{R}} C_s}{1 + \frac{1}{\rho_s^2 L_{\parallel}} \sqrt{\frac{R}{2}} a^{5/2} + \frac{\nu_{in} \sqrt{Ra}}{\sqrt{2} c_s} n_{bg} + n_0} \frac{n_0}{n_0}, \quad (5.29)$$

where  $a = \sqrt{\ln 2} \sigma_{n,y}$  is the vertical size of the blob,  $\rho_s \approx \sqrt{T_{e,bg} m_i / (eB)}$  is the ion

## 5.5. Analysis and validation of the simulation results



**Figure 5.9** – Vertical velocity of the blob as function of time for “case 1” (left panel), “case 2” (middle panel), and “case 3” (right panel), obtained from numerical simulations and experimental measurements (the gray shaded region represents the experimental uncertainty due to the finite spatial resolution of the probes).

**Table 5.2** – Blob radial velocity for the three cases, computed according to Eq. (5.29) and time-averaging simulations and experimental results.

	case 1 [m/s]			case 2 [m/s]			case 3 [m/s]		
	1.85 eV	2.8 eV	Eq. (5.21)	1.85 eV	2.8 eV	Eq. (5.21)	1.85 eV	2.8 eV	Eq. (5.21)
$T_e$ Background									
Analytical scaling	420±40	670± 50	-	530±50	810±60	-	800±80	1170±90	-
Polarization closure	910±40	1100±50	-	990±70	1180±90	-	1280±100	1480±110	-
BOUT++2D	440	760±50	-	570	930±30	-	1000	1370±30	-
BOUT++3D	540	850±20	-	670	1020±20	-	1100	1460±130	-
TOKAM3X	560	870±60	-	710	1020±40	-	1030	1210±210	-
GBS isothermal	-	890	-	-	1070	-	-	1520	-
GBS	-	-	660±10	-	-	790±40	-	-	1370±150
HESEL	-	-	1470±40	-	-	1850±70	-	-	650±200
Experimental	-	-	840±20	-	-	970±20	-	-	1030±20

Larmor radius,  $c_s \approx \sqrt{T_{e,bg}/m_i}$  is the ion sound speed, and  $n_0/(n_{bg} + n_0)$  is the ratio between the peak density value of the blob,  $n_0$ , and the total density,  $n_{bg} + n_0$ . The three terms in the denominator represent possible closures of the diamagnetic current due to, respectively, the ion polarization current, the parallel current to the sheath, and the ion-neutral collisions (the latter is neglected in the following due to the low value of  $\nu_{in}$ ). The  $n_0/(n_{bg} + n_0)$  term represents the slowing down of the blob due to a finite background density. We note that Eq. (5.29) is derived under the assumption of isothermal evolution.

In Table 5.2 we summarize the results of our analysis. First, considering the two background electron temperatures  $T_{e,bg} = 1.85$  eV and  $T_{e,bg} = 2.8$  eV, we compute the velocities theoretically predicted by using Eq. (5.29). We compute both the expected velocity from the full scaling in Eq. (5.29) (“Analytical scaling”), and the expected velocity from the ion-polarization closure scaling when an halved ballooning instability drive is considered, which corresponds to  $v_x = \sqrt{a/Rc_s n_0/(n_{bg} + n_0)}$  (“Polarization closure”). Second, averaging in time the radial velocities, we list the BOUT++2D, BOUT++3D, and TOKAM3X results corresponding to  $T_{e,bg} = 1.85$  eV and  $T_{e,bg} = 2.8$  eV. For GBS, simulations with a uniform  $T_{e,bg} = 2.8$  eV background and considering an isothermal blob (i.e.  $T_{e,bl} = 0$  at all times) are listed. Moreover, we present the GBS and HESEL results when the experimental background temperature profile is used and the blob temperature

is evolved. Finally, the experimental radial velocity measurements are averaged in time, to obtain the values presented in the last row of Table 5.2. Several observations can be made from these results.

(i) The velocities obtained for  $T_{e,bg} = 2.8$  eV from BOUT++3D, GBS with isothermal electrons, and TOKAM3X are very similar. It follows that the three models are equally able to predict the radial velocity of the blobs. Within the uncertainties affecting the measured quantities used as input parameters, they are consistent with experimental observations for “case 1” and “case 2”, while the “case 3” experimental measurements do show a smaller velocity with respect to the simulations. This difference is due to the blob motion in late part of the considered time interval, when the blob loses its coherence and the difference between experimental measurements and simulation results increases, as shown in Fig. 5.8.

(ii) BOUT++2D gives a radial velocity that is slightly smaller than the one computed with the three-dimensional isothermal models. Previous comparisons between two-dimensional and three-dimensional simulation results showed that the density drop in the three-dimensional simulations is larger than the one estimated to derive the two-dimensional model [21]. This leads to smaller parallel sheath currents and, therefore, faster blob dynamics in the three-dimensional simulations. However, the difference between BOUT++2D and BOUT++3D results are relatively small, indicating that the sheath dissipation closure represents well the considered experiments. This is consistent with previous experimental investigations, from which it has been concluded that for similar TORPEX experimental scenarios the parallel currents play an important role in setting the blob motion [128, 140, 141].

(iii) The “case 1” and “case 2” blobs simulated with HESEL produce radial velocities that are much larger than the ones observed experimentally. This leads us to conclude that, for the present experimental scenario, the weak sheath formulation combined with the vorticity advection closure is not a good representation of the plasma dynamics, consistently with point (ii). It is emphasized that this result concerns only the considered experimental scenario, and it is not generally true.

(iv) The analysis of “case 3” HESEL results shows a velocity that is smaller than both the experimental velocity and the velocity resulting from the other models. This may be due to the fact that the blob completely loses its coherence through the simulation, as can be observed in Fig. 5.7.

(v) The simulation results obtained with GBS considering the experimental temperature background show a radial velocity that is slightly smaller than the measured one for “case 1” and “case 2”, while it is slightly higher for “case 3”. It could appear surprising at first sight that the experiments agree better with the isothermal models than with the results of the non-isothermal GBS simulations. However, Fig. 5.8 shows that this is due to a fortuitous event: the differences in the radial velocity between experimental measurements and isothermal simulations in the first and second halves of the simulations are cancelling out, giving an apparently better agreement of the averaged radial velocities.

(vi) Comparing our isothermal simulation results with the analytical scaling derived



from the isothermal estimate presented in Ref. [128], we observe that Eq. (5.29) underestimates the blob radial velocity. In the appendix of Ref. [140] and in Ref. [142] other possible blob velocity scalings are discussed, for which it is assumed that the blob is subject to a Kelvin-Helmholtz instability. These scalings produce results which differ from Eq. (5.29) when low values of  $n_0/(n_{bg} + n_0)$  are considered. In particular, they give larger velocities than the ones predicted by using Eq. (5.29), which are closer to the numerical results of the isothermal simulations. Additionally, assuming that the analytical scaling of Eq. (5.29) underestimates the blob radial velocity because of the small value of  $n_0/(n_{bg} + n_0)$ , and comparing the HESEL results with the ‘‘Polarization closure’’ row in Table 5.2, we conclude that the HESEL model should be able to properly describe the blob dynamics in cases where diamagnetic currents are predominantly closed through polarization currents.

Considering the experimental vertical motion, we observe that the blobs move in TORPEX with positive, almost constant, velocity. The dynamics of the blobs simulated with GBS is consistent with the experimental measurements. The HESEL model, which assumes a non-isothermal background according to Eq. (5.21), presents a vertical velocity that is initially consistent with experimental measurements and diverges later from the experimental results. On the other hand, the other models are not in agreement with the experimental measurements. This reflects the fact that the vertical blob motion is mostly driven by a background  $v_{E \times B}$  flow, as discussed in Section 5.4.

## 5.6 Conclusions

In this chapter we present numerical results obtained from seeded blob simulations carried out with five different models, which are validated against the experimental data obtained from the TORPEX device. The models differ because of a number of assumptions used to simplify the drift-reduced Braginskii equations, such as the hypothesis of cold ions, isothermal electrons, or negligible electron inertia. Moreover, some of the models make use of the infinite aspect ratio approximation. In addition to three-dimensional models, we also consider two-dimensional models, based on different closures of the parallel currents on the vessel walls.

The comparison between the results of the different models and the experiments allow us to identify the most important physics elements that play a role in setting the blob velocity. For the present experimental scenario, we show that the vorticity advection closure, such as the one implemented in HESEL, is not able to correctly reproduce the plasma dynamics associated with the blobs, while the sheath dissipation closure, such as the one implemented in BOUT++2D, is in agreement with the three-dimensional simulations and experimental results. This is consistent with previous experimental analysis, whereby it was shown that parallel currents are important in setting the radial velocity of blobs in typical TORPEX hydrogen plasmas [128, 140, 141]. To properly validate the HESEL model, one would need to consider plasmas with higher ion mass or blobs with

a smaller size, for which it has been shown that the parallel current contribution is negligible [140].

We also observe that the value of the background electron temperature is important in setting the radial velocity of blobs, meaning that an accurate measurement of this quantity is necessary to perform reliable simulations. Moreover, comparing the results of the five codes between each other, we conclude that the radial dependence of the electron temperature background plays a role in determining the correct vertical motion of the blobs, while the evolution of the electron temperature is only necessary to describe their spinning. We also showed that the electron inertia, the Boussinesq approximation, and the infinite aspect ratio limit have a minor importance in determining the blob velocity. The results presented in this chapter provide us with a better understanding of the blob dynamics, and increase the reliability of the models used to carry out the seeded blob simulations. The experimental scenario presented in this chapter constitutes an ideal test bed also for future benchmarks and the validation of seeded blob simulations, thanks to the measurements available, which allow accurate initialization of the simulations and detailed comparisons with the numerical results. The magnetic configuration that we consider facilitates considerably the analysis and the interpretation of the experimental and simulation results.

This work represents a fundamental step towards the validation of full turbulence simulations against experimental measurements in more complex geometries, as it is in the tokamak SOL. Such validation exercises are described in Chapters 6 and 7.



# CHAPTER 6

## A validation in a tokamak: turbulence in the RFX-mod SOL

Due to the complex nonlinear phenomena taking place in the tokamak SOL region, the plasma dynamics is usually investigated numerically, thanks to state-of-the-art simulation codes (see e.g. Refs. [18, 20, 125, 126]). In the present chapter we focus on the tokamak limited SOL configuration. Besides being of interest as a stepping stone towards the simulation of more complex experimental scenarios, this configuration has recently attracted large attention since the ITER start-up and ramp-down phases will be performed using the high-field side part of the vacuum vessel as the limiting surface [143, 144].

In the past, extensive theoretical and numerical studies of the instabilities driving the SOL dynamics were performed (see e.g. Ref. [145]). It was found that, in the limiter configuration, SOL turbulence is generally driven by drift-waves (DWs) and ballooning modes (BMs) [145, 146]. It was also demonstrated that these linear instabilities typically saturate due to a nonlinear local flattening of the plasma gradient and the resulting removal of the instability drive [100]. These theoretical findings were subsequently validated against experimental measurements taken on a number of tokamaks around the world, such as TCV, Alcator C-Mod, and ISTTOK, showing good agreement between simulations and experimental measurements of plasma turbulence [147–149]. Moreover, using these observations and assuming that resistive BMs drive the SOL turbulence dynamics and that the parallel losses at the vessel are balanced by the turbulent transport, an analytical scaling for the equilibrium pressure gradient length was derived [24, 113]. It was found that this scaling is consistent with measurements taken on a number of experimental devices [150].

The goal of the present chapter is to investigate a SOL parameter regime that was not explored earlier and, in general, difficult to access experimentally. More precisely, we investigate the SOL plasma dynamics in a circular limiter configuration with a low safety factor at the last close flux surface (LCFS),  $q_{LCFS} \lesssim 3$ , for which the SOL turbulence

is expected to be clearly in the inertial DW (InDW) regime [145]. Our study is based on performing SOL turbulence simulations considering two tokamak circular plasma discharges carried out in the RFX-mod experiment [50] with  $q_{LCFS} \approx 2, 3$ . The RFX-mod device can access such low safety factors thanks to an advanced feedback magnetic boundary control system, which allows stabilizing resistive wall modes and to carry out plasma discharges with  $q_{LCFS} \approx 2$  without disruptions [151]. We then analyse the nature of the turbulence in the SOL of RFX-mod and we carefully compare our simulation results against RFX-mod measurements.

The present chapter is organized as follows. We first discuss the RFX-mod experimental setup in Section 6.1. In Section 6.2 we describe the simulations of the RFX-mod plasma discharges. The simulations are then used to uncover the instability that drive the SOL plasma dynamics in Section 6.3. Finally, the numerical results are validated against experimental measurements in Section 6.4. The conclusions follow.

## 6.1 The experimental setup

The RFX-mod experiment is a flexible toroidal device with major radius  $R = 2$  m and minor radius  $a = 0.459$  m, equipped with a set of 192 actively controlled coils that cover the whole vacuum vessel [50]. While RFX-mod plasma discharges have been performed mainly in the reversed field pinch (RFP) configuration, recent developments now allow operating the device also with magnetic geometries that feature inner-wall limited and diverted ohmic tokamaks [151–153]. Using a toroidal magnetic field on axis  $B_\varphi \simeq 0.6$  T and a plasma current up to  $I_p \simeq 150$  kA, it is possible to perform plasma pulses longer than 1 s with integrated plasma densities  $n_e \geq 10^{19} \text{ m}^{-3}$  and core electron temperatures  $T_e \geq 500$  eV.

In the following we consider two circular inboard-limited ohmic L-mode deuterium plasma discharges (#38373 and #38413), carried out in the RFX-mod device with a toroidal magnetic field on axis  $B_\varphi = 0.54$  T and plasma currents  $I_p = 150$  kA and  $I_p = 100$  kA. These two plasma currents correspond to the safety factors  $q_{LCFS} = 2$  and  $q_{LCFS} = 3$ , respectively. The plasma densities and electron temperatures at the LCFS for the two discharges are  $n_{e0} = 7.7 \times 10^{17}, 2.0 \times 10^{17} \text{ m}^{-3}$  and  $T_{e0} = 16, 19$  eV, respectively, and correspond to the two normalised plasma collisionalities  $\nu^* = L_{\parallel}/\lambda^{mfp} = 6.9, 1.3$ , where  $L_{\parallel} = 2\pi q_{LCFS} R$  is the parallel connection length and  $\lambda^{mfp}$  the electron mean free path.

The experimental measurements illustrated in the following of the present chapter are obtained using the U-probe installed in RFX-mod. This probe consists of two boron nitride arms, each of them equipped with 25 electrostatic pins [154, 155]. Some of the pins are used as a five-pin triple probe [156], allowing simultaneous measurements of ion saturation,  $I_{sat}$ , plasma density,  $n$ , electron temperature,  $T_e$ , and floating potential,  $V_{fl}$ , with time resolution of  $0.2 \mu\text{s}$ . The U-probe is located at a fixed radial position at the outward equatorial midplane, with its arms in the horizontal direction. In or-

der to obtain measurements at different radial locations, the plasma column is slowly shifted towards the inner wall of the device during the discharge, while keeping a constant edge safety factor. We note that the experimental measurements related to the #38373 plasma discharge we use for the present validation are taken only in between sawtooth crashes. This leads to a reduced number of measurements for the considered time traces (20'000 measurements) available for the analysis of the #38373 discharge (approximately a factor ten less with respect to the #38413 discharge, for which we have 175'000 measurements).

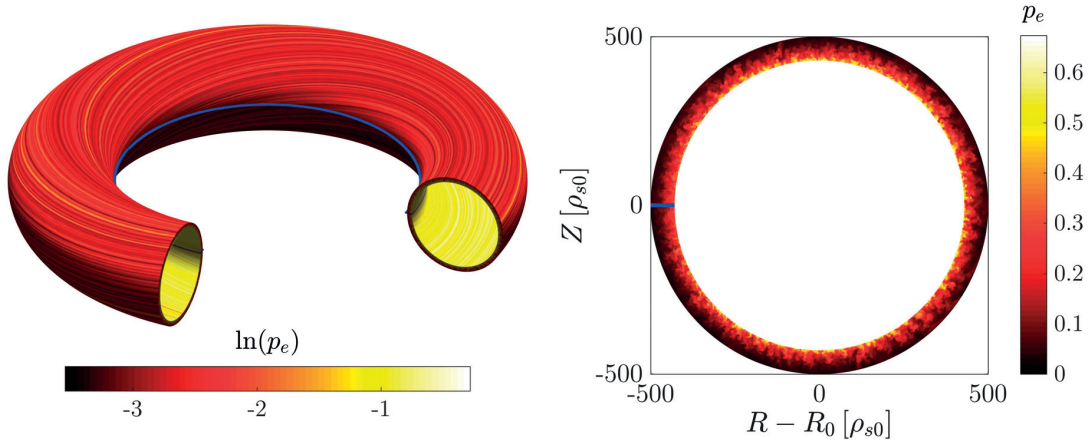
## 6.2 GBS simulations of the RFX-mod SOL

The tokamak SOL region is generally studied by employing a plasma fluid description, such as the Braginskii fluid model [13]. Moreover, since the SOL turbulent time scales are much slower than the ion cyclotron time, and the perpendicular (to  $\mathbf{B}$ ) scale lengths are longer than the ion Larmor radius, the drift approximation can be applied to simplify the fluid model, thus obtaining a set of drift-reduced Braginskii equations useful to describe the SOL plasma dynamics (see Appendix A for a discussion of this model). We consider this model also for the present study, although the conditions for the applicability of the fluid model are marginally satisfied for the RFX-mod #38413 plasma discharge. Neglecting electromagnetic effects as suggested in Ref. [157], since  $\beta_e R/L_p \leq 10^{-3}$  in the RFX-mod SOL ( $\beta_e$  is the plasma to magnetic pressure ratio and  $L_p$  the equilibrium pressure gradient length), assuming cold ions (no ion temperature measurements are available on RFX-mod for these discharges, the impact of ion temperature effects on SOL turbulence is investigated in Ref. [158]), and employing the Boussinesq approximation [91–93] to simplify the vorticity equation, the drift-reduced Braginskii equations, Eqs. (A.65)-(A.70), reduce to Eqs. (2.28)-(2.32). The drift-reduced Braginskii system is closed by the Poisson's equation  $\nabla_{\perp}^2 \phi = \omega$  and by the set of boundary conditions in Eqs. (2.33)-(2.38), which describe the plasma dynamics at the limiter magnetic presheath entrance. The resulting model is solved thanks to the GBS code, as detailed in Appendix A. Note that, unless specified otherwise, in the present chapter we make use of the normalization and of the toric coordinate system  $(y, x, z)$  detailed in Appendix A. Focusing on a circular plasma with a toroidal limiter located at the high-field side, and assuming a large aspect ratio geometry and no magnetic shear to simplify the equations (a discussion of the impact of these assumptions on DWs and BMs is presented in Refs. [145, 146]), we perform two nonlinear GBS simulations based on the RFX-mod experimental parameters  $R$ ,  $q_{LCFS}$ ,  $n_{e0}$ , and  $T_{e0}$ . For the two plasma discharges #38373 and #38413 these parameters lead to the normalized plasma resistivities  $\nu = 0.005, 0.001$ , the normalized major radii  $R_0 = 1872, 1716$ , and the poloidal domain sizes  $L_y = 2700, 2470$ . In addition, we consider  $\Lambda = 3$ , a reduced ion to electron mass ratio  $m_i/m_e = 800$ , a reduced normalized parallel electron thermal conductivity  $\chi_{\parallel e} = 2$ , and the normalized perpendicular diffusion coefficients  $D_A = 5$ , where  $A = n, \omega, v_{\parallel e}, v_{\parallel i}, T_e$ . The particle

and temperature sources, used to mimic the plasma outflow from the core, are assumed poloidally and toroidally constant, with radial dependence  $S_{n,T_e}(x) \propto \exp[-(x-a)^2/\sigma^2]$ , being  $\sigma = 2.5$ . The radial domain extends from the inner radius  $x_i = a - 30$  to the outer radius  $x_o = a + 70$  in both simulations. Since a set of first-principle boundary conditions describing the plasma interaction with the outer wall and the interface between the SOL and the core does not exist yet, ad hoc boundary conditions are applied at  $x_i$  and  $x_o$ , with Neumann's boundary conditions used for  $n$ ,  $v_{\parallel e}$ ,  $v_{\parallel i}$ , and  $T_e$ , and Dirichlet's boundary conditions for  $\omega$  and  $\phi$ . To mitigate the impact of these boundary conditions on the simulation results, the two regions extending from  $x_i$  to  $x = a$ , and from  $x = a + 55$  to  $x_o$  are considered as buffers and are not included in the analysis of the results.

We note that, because of the necessary rather large numerical grids  $(N_x, N_y, N_z) = (128, 1279, 320)$ ,  $(128, 1279, 212)$ , with  $N_x$ ,  $N_y$ , and  $N_z$  the number of points in the radial, poloidal, and toroidal directions, the two simulations discussed herein are extremely expensive. The reduced mass ratio and parallel electron thermal conductivity are consequently used to considerably decrease the computational cost of the simulations. We also note that, while in the RFX-mod discharges the plasma current and the toroidal magnetic field are in the same direction, we use a current that is in the opposite direction to the magnetic field in the GBS simulations.

A typical GBS snapshot for the RFX-mod plasma discharge #38373 is shown in Fig. 6.1. On the left panel we present a three-dimensional view of  $\ln(p_e)$ . We clearly observe



**Figure 6.1** – Snapshot of a three-dimensional nonlinear GBS simulation based on the RFX-mod plasma discharge #38373. Left panel: Three-dimensional visualization of  $\ln(p_e)$ , showing a section of the domain simulated by GBS with the toroidal limiter at the high-field side (in blue). Right panel: Poloidal cross section of the electron pressure plasma profile, with the limiter represented in blue at the high-field side.

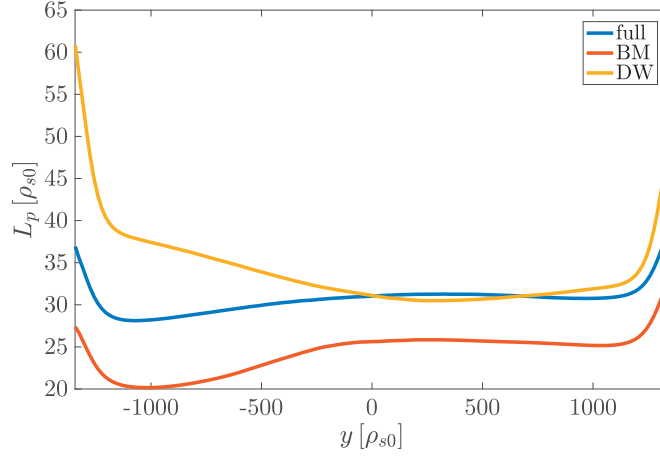
plasma turbulent eddies that are aligned to the magnetic field. On the right panel we show a poloidal cross section of the simulated  $p_e$  profile, with  $R$  and  $Z$  the radial and vertical coordinates of a cylindrical coordinate system located at the center of the tokamak. The plasma dynamics described by our simulations results from the interplay of the plasma outflow from the core, mimicked by the particle and temperature sources, the radial outwards transport due to turbulence, the parallel plasma flows, and the losses at

the limiter plates. No separation between background and turbulent quantities is made. The equilibrium pressure gradient length  $L_p = -p_e/\nabla p_e$  is directly evaluated from nonlinear simulations by computing the averaged radial  $p_e$  profile at the outer midplane,  $p_e(x) = \langle p_e(0, x, z, t) \rangle_{z,t}$ , where  $\langle - \rangle_{z,t}$  denotes averaging over  $z$  (the toroidal coordinate) and  $t$ , and fitting  $p_e(x)$  between  $x = a$  and  $x = a+55$  assuming  $p_e(x) \propto \exp[-(x-a)/L_p]$ . For the two plasma discharges #38373 and #38413 we find  $L_p = 31$  and  $L_p = 37$ , respectively. Computing the power spectral density of the  $p_e$  fluctuations in the nonlinear simulations (not shown here), it is also possible to estimate the poloidal wave number of the mode that drives most of the turbulent transport. For the two plasma discharges considered herein we find  $k_y \approx 0.1 - 0.2$ .

## 6.3 Identification of the instability driving the SOL transport

Previous investigations of the SOL plasma dynamics indicate DWs and BMs as the main instabilities driving SOL turbulent transport [17, 145, 157]. BMs are interchange-like modes, driven unstable by magnetic curvature and pressure gradients pointing in the same direction. The mechanism leading to BMs can be summarized as follows. Because of the presence of a pressure gradient, electrons and ions drift in opposite direction, perpendicular both to  $\nabla p_e$  and  $\mathbf{B}$ . In presence of a small density perturbation, a charge separation occurs, which gives rise to an electric field. When the magnetic curvature and the pressure gradient point in the same direction, the generated electric field leads to an  $\mathbf{E} \times \mathbf{B}$  flow that amplifies the initial perturbation, resulting in an instability with a phase shift between the pressure and potential perturbations close to  $\pi/2$ . The parallel component of the electric field can be balanced by plasma resistivity, in this case the instability is known as resistive BMs (RBM), by electron inertial effects, giving rise to inertial BMs (InBM), or by electromagnetic effects, which cause the growth of ideal BMs (IBM). On the other hand, DWs are due to an  $\mathbf{E} \times \mathbf{B}$  convection of the plasma pressure when electron adiabaticity is broken by resistivity or finite electron mass, leading respectively to resistive DWs (RDW) and InDWs. This is understood as follows. If we consider a plasma with approximately adiabatic electrons, regions with high plasma pressure correspond to regions with high plasma potential, and vice versa. Consequently,  $p_e$  perturbations are associated with an  $\mathbf{E} \times \mathbf{B}$  flow. In the presence of plasma pressure gradients, this flow might result in an instability, the DW instability.

Past works show that  $q_{LCFS}$  and  $\nu$  strongly affect the SOL turbulent regime. In particular, it is demonstrated that it exists a threshold value of  $\nu$  below which a transition from RBMs to InDWs is observed, and this threshold value increases with the decrease of the safety factor [145]. While in typical tokamak conditions the SOL is expected to be in the RBM regime or marginally in the DW regime, for the parameters considered herein turbulence is expected to be clearly in the InDW regime [145]. In the following of the present section we investigate the nature of the instability that drives most of the



**Figure 6.2** – Profiles of  $L_p$  as function of  $y$  based on the RFX-mod discharge #38373, solving the “full” GBS model, Eqs. (2.28)-(2.32) (blue line), the “BM” model (red line), and the “DW” model (yellow line).

SOL turbulent transport in the plasma discharges #38373 and #38413.

### 6.3.1 Nonlinear simulations

In order to identify the instability that drives most of the RFX-mod SOL turbulent transport, we proceed as follows. Considering the plasma discharge #38373, we perform three nonlinear simulations solving (i) the “full” GBS model, Eqs. (2.28)-(2.32), (ii) the “BM” model, considering Eqs. (2.28)-(2.32) where we neglect the diamagnetic term in the Ohm’s equation, i.e. we simplify Eq. (2.30) as

$$\partial_t v_{\parallel e} = -\frac{R}{\rho_{s0}} \{ \phi, v_{\parallel e} \} + \frac{m_i}{m_e} \left[ \nabla_{\parallel} \phi + \nu j_{\parallel} - \frac{2}{3n} \nabla_{\parallel} G_e \right] - v_{\parallel e} \nabla_{\parallel} v_{\parallel e} + D_{v_{\parallel e}} \nabla_{\perp}^2 v_{\parallel e}, \quad (6.1)$$

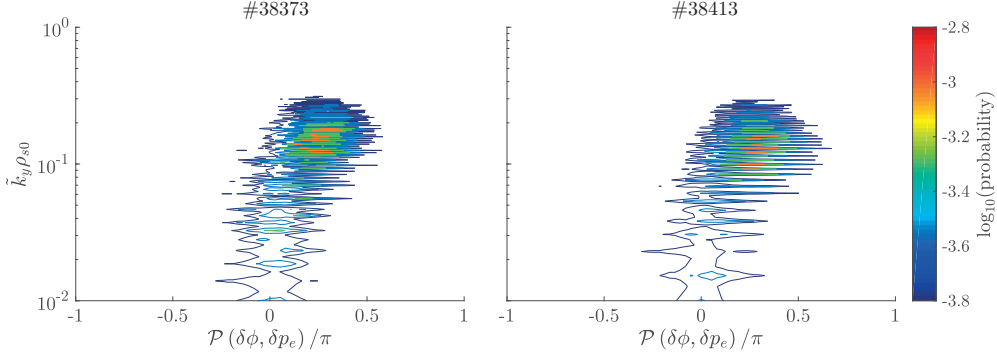
and (iii) the “DW” model, where we neglect the pressure curvature term in the vorticity equation of the “full” GBS model, which corresponds to rewriting Eq. (2.29) as

$$\partial_t \omega = -\frac{R}{\rho_{s0}} \{ \phi, \omega \} - v_{\parallel i} \nabla_{\parallel} \omega + \frac{1}{n} \nabla_{\parallel} j_{\parallel} + \frac{1}{3n} C(G_i) + D_{\omega} \nabla_{\perp}^2 \omega. \quad (6.2)$$

For each simulation we then compute  $p_e(y, x) = \langle p_e(y, x, z, t) \rangle_{z,t}$  and, at fixed  $y$ , we fit  $p_e(y, x)$  between  $x = a$  and  $x = a + 55$  assuming  $p_e(y, x) \propto \exp[-(x - a)/L_p(y)]$ . The values of  $L_p(y)$  thus obtained are shown in Fig. 6.2 for the three models. We observe that the “full” and the “DW” models lead to quite similar  $L_p$  for  $y > 0$ , while  $L_p$  is larger for  $y < 0$  in the “DW” simulations, particularly in the proximity of the limiter. This is probably due to the stabilizing effect of magnetic curvature on SOL turbulence at the tokamak high-field side. On the other hand, the value of  $L_p$  for the “BM” model is considerably smaller than for the original simulation. This suggests that DWs are driving most of the SOL turbulent transport, in agreement with the expectations in Ref. [145].

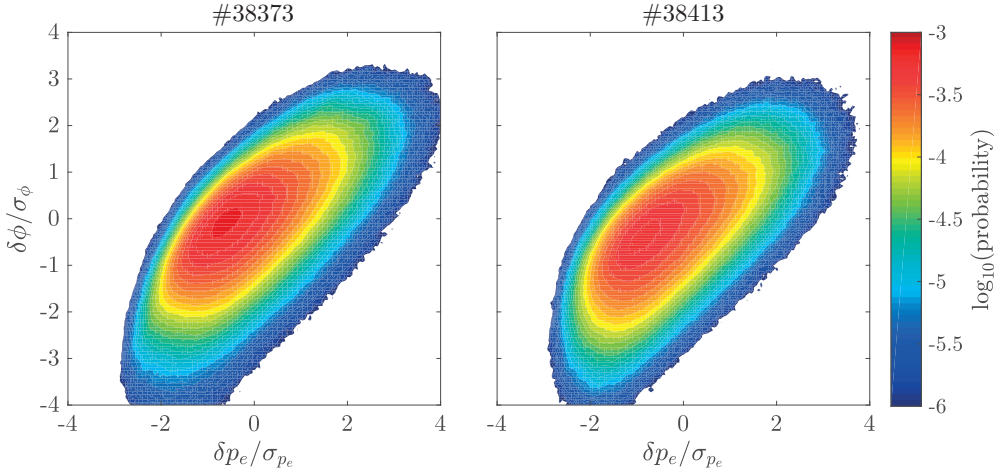


### 6.3. Identification of the instability driving the SOL transport



**Figure 6.3** – Probability distribution function of the phase shift between  $\delta p_e$  and  $\delta\phi$  for the non-linear simulations based on the two RFX-mod plasma discharges #38373 (left panel) and #38413 (right panel).

Another approach we explore to understand the nature of the dominant instability, based on investigating the phase shift and the correlation between electron pressure and plasma potential fluctuations,  $\delta p_e$  and  $\delta\phi$ , turns out not conclusive. Since DWs lead to electrons in conditions close to adiabaticity, we expect a small phase shift between  $\delta p_e$  and  $\delta\phi$  and a noticeable correlation between these two quantities. On the other hand, BMs are expected to show a phase shift close to  $\pi/2$  and no correlation between  $\delta p_e$  and  $\delta\phi$  [159]. Following the methodology illustrated in Ref. [146], we compute for the two GBS simulations of RFX-mod the probability distribution function of the phase shift and the joint probability between  $\delta p_e$  and  $\delta\phi$  at  $x = a + 15$  for all  $\theta_* \in [-\pi, \pi]$ . The results obtained are presented in Figs. 6.3 and 6.4. Concerning the phase shift, we observe that



**Figure 6.4** – Joint probability of  $\delta p_e$  and  $\delta\phi$  normalized to their standard deviation for the non-linear simulations based on the two RFX-mod plasma discharges #38373 (left panel) and #38413 (right panel).

$\mathcal{P}(\delta p_e, \delta\phi) \simeq \pi/4$  at the  $k_y$  of the largest amplitude mode. Moreover,  $\delta p_e$  and  $\delta\phi$  do not show a clear correlation, nor they are clearly not correlated (for both simulations the correlation coefficient is approximately 0.7). Therefore, while the results obtained from the analysis of the phase shift and of the joint probability are not incompatible with a

SOL dynamics mostly driven by DWs, they are not conclusive to indicate the instability that drives most of the SOL turbulent transport.

### 6.3.2 Linear instabilities

As a confirmation of the nature of the turbulent transport identified by using the nonlinear simulations, we consider the linear properties of the instability dominating the SOL plasma dynamics. This approach allows us also to disentangle more easily the role of resistivity and electron inertia and to study ion to electron mass ratios not accessible by the nonlinear simulations.

First, in order to deduce a linear model useful for investigating the SOL plasma dynamics, we introduce the flux-tube ( $X = r, Y = a\alpha/q, Z = qR\theta_*$ ) coordinate system, where  $r$  is a flux coordinate,  $\alpha = \varphi - q(r)\theta_*$  a field line label,  $\varphi$  the toroidal angle,  $\theta_*$  and  $q(r)$  are the straight field line angle and safety factor defined in Section A.3.4, and  $q = q(a) = q_{LCFS}$ . Equations (2.28)-(2.32) are then expressed in the  $(X, Y, Z)$  coordinate system and the resulting system of equations is linearised assuming that the equilibrium plasma profiles depend only on the radial coordinate  $X$ . Moreover, each quantity  $A = A(X, Y, Z, t)$  is split between an equilibrium part  $A_0(X)$  and the perturbation  $\delta A(Y, Z, t) = \delta A(Z) \exp[ik_Y Y + \gamma t]$ , with  $k_Y$  the poloidal wave number and  $\gamma$  the linear growth rate. Equilibrium gradients are defined as  $\partial_X A = -A_0/L_A$ , where  $L_A$  is a characteristic length associated with  $A_0$  at  $X = a$ . The  $X$  dependence of  $\delta A$  is neglected here because  $k_Y/k_X \sim \sqrt{k_Y L_n} > 1$  for both DWs and BMs [160, 161]. Assuming  $\phi_0 = v_{\parallel i,0} = v_{\parallel e,0} = 0$ , noting that  $n_0 = 1$  and  $T_{e0} = 1$  in normalized units, and neglecting gyroviscous and diffusion terms, the resulting linearised system of equations is written as

$$\gamma \delta n = -ik_Y \frac{R_0}{L_n} \delta \phi - 2ik_Y \cos(\theta_*) (\delta p_e - \delta \phi) + \partial_Z (\delta j_{\parallel} - \delta v_{\parallel i}), \quad (6.3)$$

$$\gamma \delta \omega = -2ik_Y \cos(\theta_*) \delta p_e + \partial_Z \delta j_{\parallel}, \quad (6.4)$$

$$\frac{m_e}{m_i} \gamma \delta v_{\parallel e} = \partial_Z (\delta \phi - \delta p_e - 0.71 \delta T_e) + \nu \delta j_{\parallel}, \quad (6.5)$$

$$\gamma \delta v_{\parallel i} = -\partial_Z \delta p_e, \quad (6.6)$$

$$\begin{aligned} \gamma \delta T_e = & -ik_Y \eta \frac{R_0}{L_n} \delta \phi - ik_Y \frac{4 \cos(\theta_*)}{3} \left( \delta p_e + \frac{5}{2} \delta T_e - \delta \phi \right) \\ & + \frac{2}{3} \partial_Z (1.71 \delta j_{\parallel} - \delta v_{\parallel i}), \end{aligned} \quad (6.7)$$

where  $\delta p_e = \delta n + \delta T_e$ ,  $\delta j_{\parallel} = \delta v_{\parallel i} - \delta v_{\parallel e}$ ,  $\delta \omega = -k_Y^2 \delta \phi$ , and  $\eta = L_n/L_{Te}$ . Equations (6.3)-(6.7) determine the linear growth rate of the SOL plasma instabilities.

To solve Eqs. (6.3)-(6.7), a numerical code was developed, which evaluates  $\gamma$  as a function of the parameters  $R_0/L_n$ ,  $m_e/m_i$ ,  $\eta$ ,  $\nu$ ,  $q$ , and  $k_Y$ . The numerical implementation of the code is detailed in Ref. [146], and its main features are summarized here. First, the  $Z$  coordinate is discretized using a fourth order finite difference scheme. Second, Dirichlet



### 6.3. Identification of the instability driving the SOL transport

boundary conditions are imposed at the end of the flux tube to  $\delta n$ ,  $\delta\phi$ , and  $\delta T_e$ , while no boundary conditions are applied to the ion and electron parallel velocities. We note that we extend the simulation domain along the  $Z$  coordinate to mitigate the impact of the boundary conditions on the obtained results. Finally, the discretized system of equations is integrated implicitly in time, starting from random noise. By studying the growth of the most unstable mode, we obtain  $\gamma$ .

As discussed in Section 6.3.1, it is possible to remove the BM instability from the system, Eqs. (6.3)-(6.7), by zeroing out the curvature term in the vorticity equation, i.e. neglecting the first term of the right-hand side of Eq. (6.4). The solution of the resulting reduced model is denoted in the following as  $\gamma_{DW}$ . On the other hand, DWs are removed from the model by neglecting the diamagnetic term in the Ohm's equation, i.e. zeroing out the  $\delta p_e$  and  $\delta T_e$  terms of Eq. (6.5). The solution of this reduced model is denoted in the following as  $\gamma_{BM}$ .

Considering  $m_i/m_e = 800$ , the parameters  $\nu$ ,  $R_0$  and  $q$ , inputs of the linear code, provided by experimental measurements of the plasma equilibrium, setting  $\eta \simeq 0.7$  according to typical simulation results (see e.g. Ref. [161], and also in agreement with the nonlinear results obtained with the two GBS simulations of RFX-mod), and imposing  $L_p$  and  $k_Y$  as evaluated in Section 6.2 from the nonlinear simulations, we solve Eqs. (6.3)-(6.7) for  $\gamma$ ,  $\gamma_{DW}$ , and  $\gamma_{BM}$ . For the two discharges #38373 and #38413 we obtain  $\gamma = 5.1, 4.4$ ,  $\gamma_{DW} = 5.2, 4.5$ , and  $\gamma_{BM} = 0.3, 0.1$ , respectively. While the values of  $\gamma_{DW}$  are similar to the growth rates obtained by solving the original Eqs. (6.3)-(6.7), removing the DWs from the system leads to a growth rate close to zero. This means that the DW is the instability that drives most of the SOL turbulent transport in the two plasma discharges considered herein, in agreement with the nonlinear results.

In order to disentangle the impact of resistivity and electron inertia on DWs, we simplify Eqs. (6.3)-(6.7) as follows. We first neglect the curvature terms to avoid coupling with BMs, together with the compressibility terms in the continuity and temperature equations. Then, assuming  $\gamma \gg k_Z$ , we remove the sound wave coupling from the model. The resulting system of equations is written as

$$\gamma \delta n = -ik_Y \frac{R_0}{L_n} \delta\phi - \partial_Z \delta v_{\parallel e}, \quad (6.8)$$

$$\gamma \delta \omega = -\partial_Z \delta v_{\parallel e}, \quad (6.9)$$

$$\frac{m_e}{m_i} \gamma \delta v_{\parallel e} = \partial_Z (\delta\phi - \delta p_e - 0.71 \delta T_e) - \nu \delta v_{\parallel e}, \quad (6.10)$$

$$\gamma \delta T_e = -ik_Y \eta \frac{R_0}{L_n} \delta\phi - 1.71 \frac{2}{3} \partial_Z \delta v_{\parallel e}. \quad (6.11)$$

Equations (6.8)-(6.11) constitute the minimal model necessary to describe the linear dynamics of RDWs and InDWs. RDWs and InDWs are removed from the model, Eqs. (6.8)-(6.11), by setting  $\nu = 0$  and  $m_e/m_i = 0$ , respectively. Solving Eqs. (6.8)-(6.11) with the linear code discussed above, we obtain  $\gamma = 6.1, 4.7$  for  $\nu = 0$  and  $\gamma = 3.1, 1.9$  for  $m_e/m_i = 0$ . Since the growth rates are approximately a factor two smaller for  $m_e/m_i = 0$  with respect to  $\nu = 0$ , we conclude that InDWs are driving most of the SOL

turbulent transport in the two plasma discharges considered here, in agreement with the conclusions in Ref. [145].

We note that, while  $k_Y$  and  $L_n$ , input of the linear code, can be obtained from the nonlinear simulation results, they can also be estimated semi-analytically. In fact, in the limit of a negligible  $\mathbf{E} \times \mathbf{B}$  flow, the saturation of the growth of BMs and DWs is usually determined by the gradient removal mechanism, i.e. the saturation of the mode is due to the nonlinear local flattening of the plasma pressure profile, thus removing the drive of the instability [100]. The main aspects of this theory are detailed in Appendix B and the main result is

$$L_p = \frac{L_n}{1 + \eta} = \frac{q}{c_s} \left( \frac{\gamma}{k_Y} \right)_{\max}, \quad (6.12)$$

with  $c_s = 1$  because of the normalization employed. Equation (6.12) is an implicit equation for  $L_n$ , that is solved by scanning  $\gamma$ , solution of Eqs. (6.3)-(6.7), over the parameter space  $(k_Y, L_n)$  and searching for the value of  $L_n$  and  $k_Y$  that satisfies Eq. (6.12). This procedure is applied to determine the equilibrium pressure gradient length of the two plasma discharges #38373 and #38413 for  $m_i/m_e = 800$ , obtaining  $L_p = 44, 56$ ,  $k_Y = 0.17, 0.17$ , and  $\gamma = 3.8, 3.1$ , respectively. The  $L_p$  values computed according to Eq. (6.12) are in qualitative agreement with the results obtained from the nonlinear simulations discussed in Section 6.2. Moreover, the wave number associated with the instability that drives most of the SOL turbulent transport is in good quantitative agreement with nonlinear results.

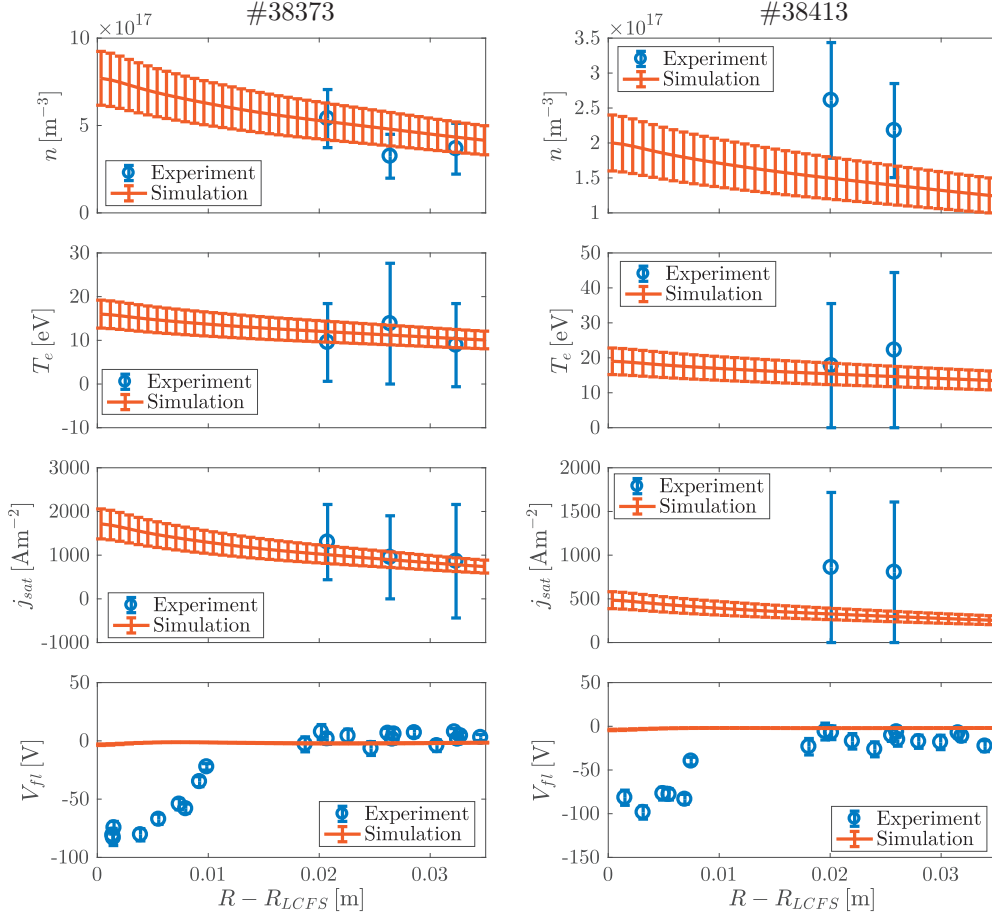
Equation (6.12) allows us to investigate the impact of the reduced ion to electron mass ratio on our results. This is necessary since performing nonlinear simulations with  $m_i/m_e = 3600$  is too demanding in terms of computational resources. Imposing a realistic ion to electron mass ratio  $m_i/m_e = 3600$ , we obtain  $L_p = 39, 52$  and  $k_Y = 0.14, 0.16$ . We see that  $L_p$  and  $k_Y$  are only slightly affected by increasing the ion to electron mass ratio to a realistic value. Moreover, we note that the same turbulent regime obtained with  $m_i/m_e = 800$  is found also for the experimental value  $m_i/m_e = 3600$ , i.e. the SOL turbulent transport is mainly driven by InDWs.

## 6.4 Validation of the GBS simulations against experimental measurements

In order to assess the reliability of the drift-reduced Braginskii model and of the GBS simulations, we now compare the nonlinear numerical results with RFX-mod experimental measurements. We remark that the plasma dynamics inside the LCFS is neglected in the simulations considered herein. Therefore, we expect a better agreement between simulation results and experimental measurements in the far SOL than in the near SOL. Note that the results illustrated in the present section are in SI units.

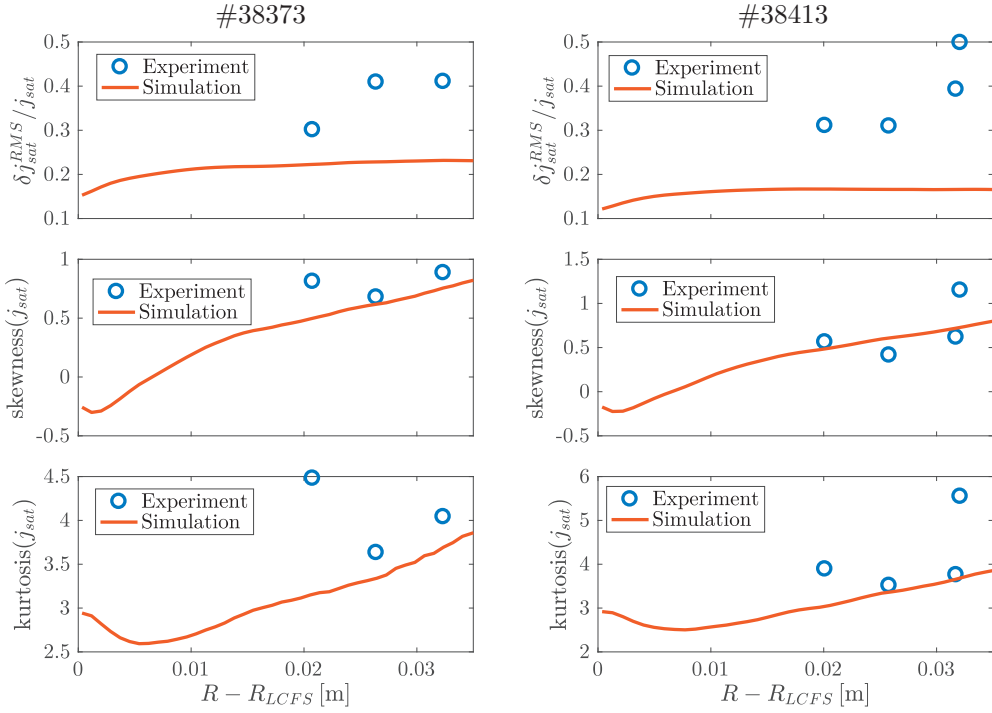
First, we present the experimental and simulation radial equilibrium profiles of  $n$ ,  $T_e$ ,

## 6.4. Validation of the GBS simulations against experimental measurements

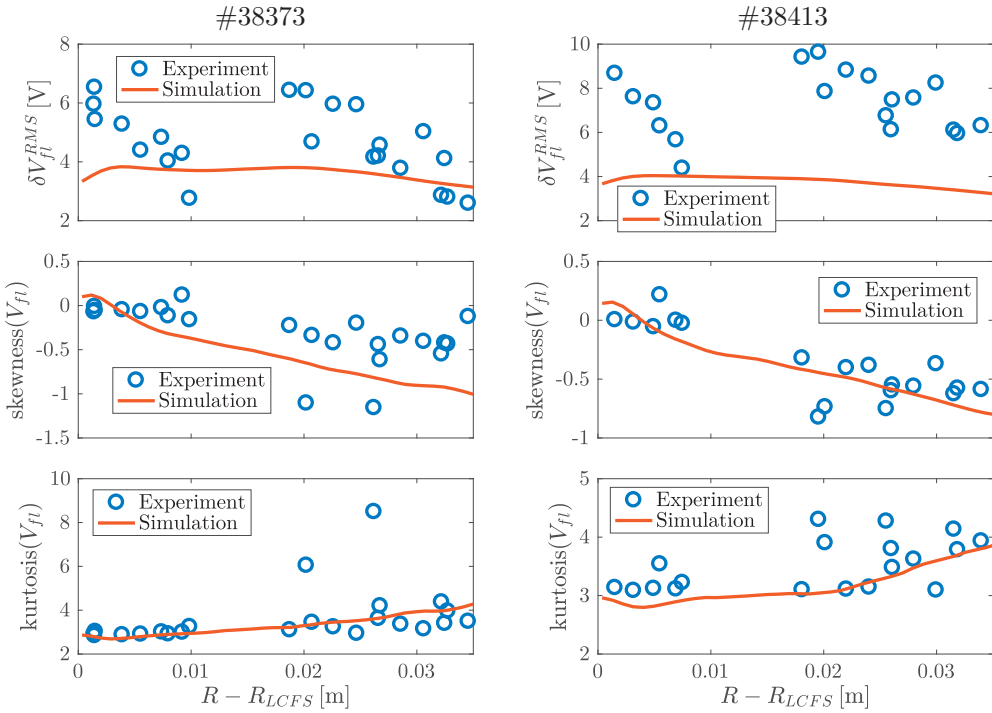


**Figure 6.5** – Experimental (blue circles) and simulation (red lines) radial equilibrium profiles of plasma density (first row), electron temperature (second row), ion saturation current density (third row), and floating potential (fourth row), for the two RFX-mod plasma discharges #38373 (left column) and #38413 (right column).

$j_{sat}$ , and  $V_{fl}$  for the two RFX-mod discharges discussed above in Fig. 6.5 (we evaluate  $j_{sat} = enc_s/2$  and  $V_{fl} = \phi - \Lambda T_e/e$  in the simulations). According to the results presented in Chapter 3, we assume a 20% relative numerical error affecting the simulation equilibrium profiles and we neglect other sources of uncertainties. We observe that the equilibrium radial profiles of  $n$ ,  $T_e$ , and  $j_{sat}$  obtained from the nonlinear simulations of both discharges are consistent with the experimental results within the estimated uncertainties. However, since the experimental uncertainties are rather large, it is not possible to reliably estimate the  $n$  and  $T_e$  equilibrium gradient lengths. Concerning  $V_{fl}$ , the simulation results do not agree with the experimental measurements, in particular in the proximity of the LCFS. We note that the uncertainty on the LCFS position is approximately 5 mm. However, this uncertainty is not sufficiently large to explain the discrepancy in the  $V_{fl}$  radial equilibrium profile. This discrepancy is probably related to simulating only the open field line region of RFX-mod, since the plasma dynamics close to the LCFS plays an important role in setting  $V_{fl}$  in the near SOL [162], and to neglecting ion temperature effects.

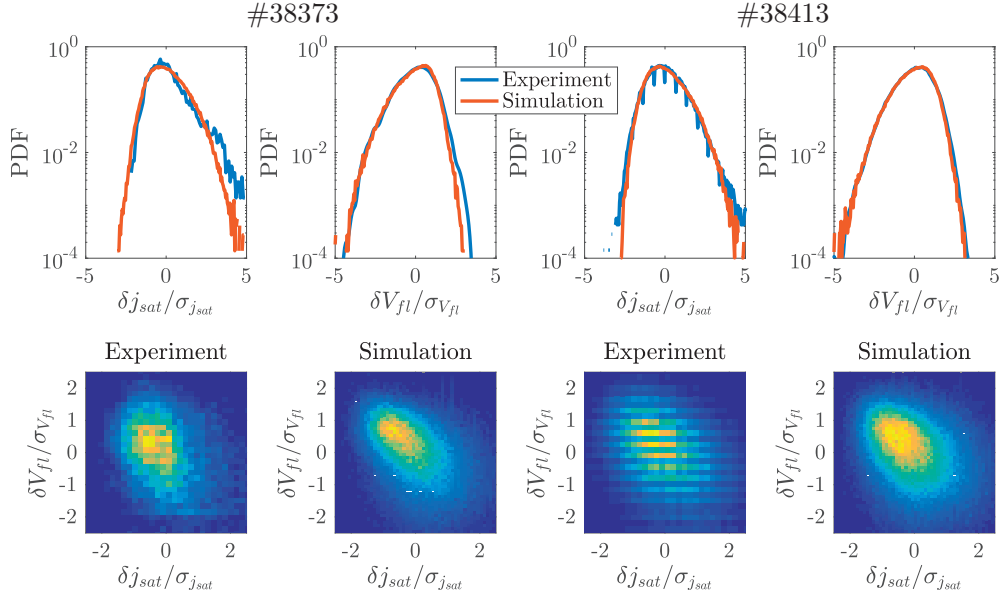


**Figure 6.6** – Experimental (blue circles) and simulation (red lines) radial profiles of  $\delta j_{sat}^{RMS} / j_{sat}$  (first row),  $j_{sat}$  skewness (second row), and  $j_{sat}$  kurtosis (third row), for the two RFX-mod plasma discharges #38373 (left column) and #38413 (right column).



**Figure 6.7** – Experimental (blue circles) and simulation (red lines) radial profiles of  $\delta V_{fl}^{RMS}$  (first row),  $V_{fl}$  skewness (second row), and  $V_{fl}$  kurtosis (third row), for the two RFX-mod plasma discharges #38373 (left column) and #38413 (right column).

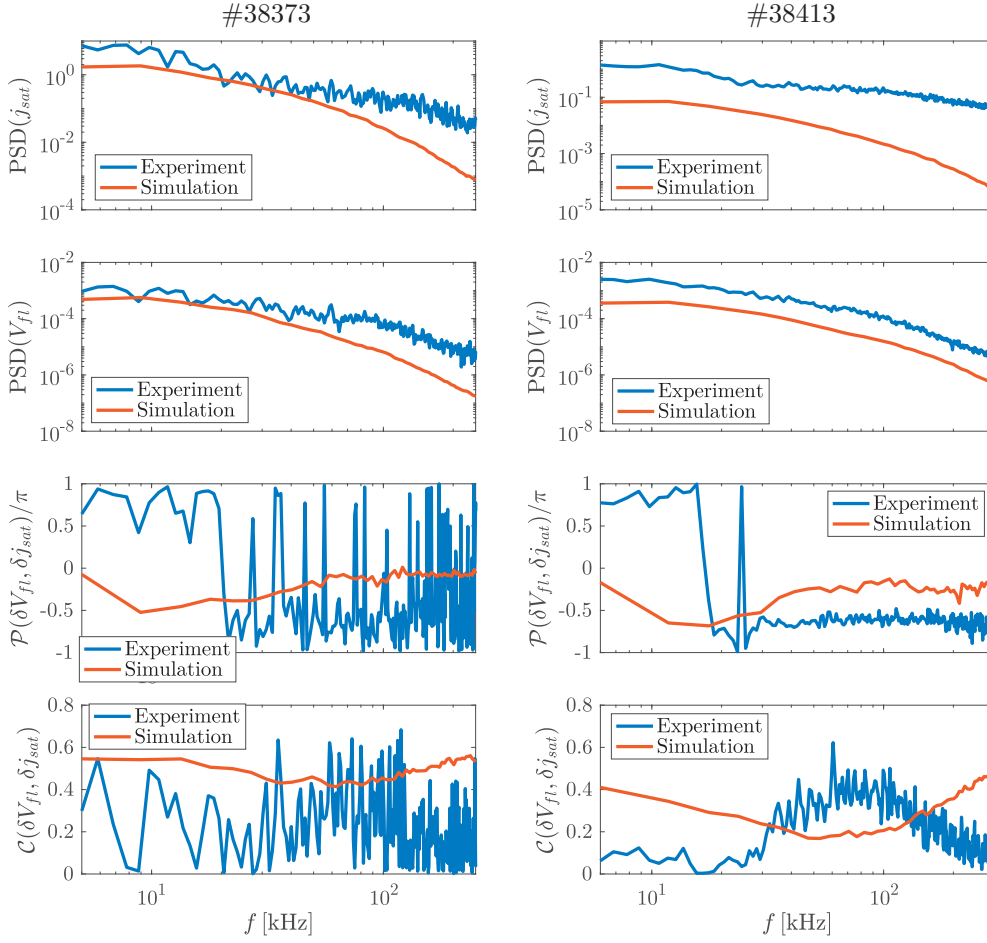
## 6.4. Validation of the GBS simulations against experimental measurements



**Figure 6.8** – First row: Experimental (blue lines) and simulation (red lines) probability density function of  $\delta j_{sat}$  (first and third column) and  $\delta V_{fl}$  (second and fourth column) normalized to their standard deviation. Second row: Experimental (first and third panels) and simulation (second and fourth panels) joint probability of  $\delta j_{sat}$  and  $\delta V_{fl}$  normalized to their standard deviation. The results are evaluated approximately at 2 cm from the LCFS and are displayed for the two RFX-mod plasma discharges #38373 (first and second columns) and #38413 (third and fourth columns).

In Fig. 6.6 we compare the experimental root-mean-square (RMS) values of  $\delta j_{sat}$ ,  $\delta j_{sat}^{RMS}$ , normalized to the equilibrium  $j_{sat}$ , with the simulation results. We observe that the simulations underestimate approximately by a factor of two the  $j_{sat}$  fluctuations for both the considered discharges. In Fig. 6.6 we also display the skewness and the kurtosis related to the experimental and numerical  $j_{sat}$  time traces. For these quantities the simulation results show a better agreement with the experimental measurements than for  $\delta j_{sat}^{RMS}$ . In particular, the simulation results display a  $j_{sat}$  skewness close to zero in the proximity of the LCFS and monotonically increasing in the SOL, in agreement with previous experimental SOL investigations [163–165].

In Fig. 6.7 we present the radial profiles of the  $\delta V_{fl}$  RMS values,  $\delta V_{fl}^{RMS}$ , and of the  $V_{fl}$  skewness and kurtosis. Concerning the RMS values, we observe an almost radial constant level of fluctuations both in the simulations and in the experiment. However, while the numerical results display a quite good agreement with RFX-mod experimental measurements for the #38373 discharge, the agreement worsen considering the discharge with lower plasma collisionality. The  $V_{fl}$  skewness monotonically decreases in the SOL both for the simulations and for the RFX-mod experimental measurements, with good quantitative agreements between the two quantities. Finally, concerning the  $V_{fl}$  kurtosis, we observe good qualitative agreement for both discharges, with an almost constant value close to three, except for  $R - R_{LCFS} > 2.5$  cm, where the kurtosis is larger. We note that a comparison of the  $V_{fl}$  moments between simulations and experimental measurements is also discussed in Ref. [166], considering TORPEX plasma discharges. Considerably larger discrepancies between numerical results and experimental measurements

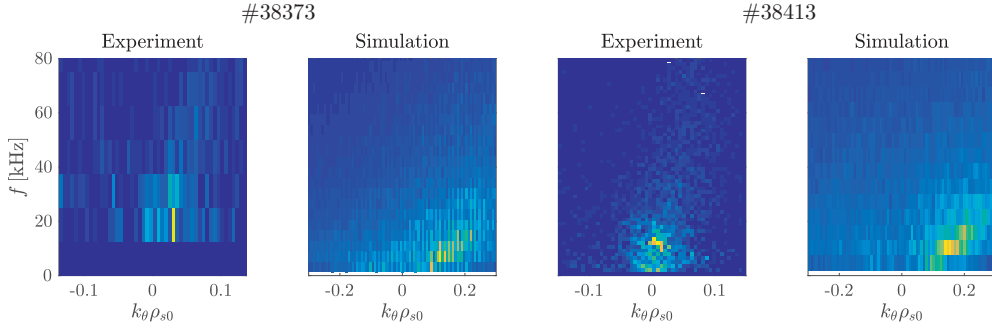


**Figure 6.9** – Experimental (blue lines) and simulation (red lines) profiles of the  $j_{sat}$  (first row) and  $V_{fl}$  (second row) power spectral densities, and of the phase shift (third row) and the coherence (fourth row) between  $\delta j_{sat}$  and  $\delta V_{fl}$ , for the two RFX-mod plasma discharges #38373 (left column) and #38413 (right column).

were found in that case, probably due to the presence of fast electrons, resulting from the source operating at the electron cyclotron resonance.

Our observations on the agreement of  $j_{sat}$  and  $V_{fl}$  skewness and kurtosis are confirmed by comparing the numerical and experimental probability density functions (PDF) corresponding to  $j_{sat}$  and  $V_{fl}$  fluctuations normalized to their standard deviation in the far SOL, at approximately 2 cm from the LCFS, as shown in the first row of Fig. 6.8. We observe that the simulation results are in quite good agreement with experimental measurements for both physical quantities and for both discharges. The  $j_{sat}$  PDF displays a positive skewness, while the  $V_{fl}$  PDF is negatively skewed. We note that small differences are observed between experimental measurements and simulation results for the distribution tails, particularly for the plasma discharge #38373. This might be due to intermittent events occurring inside the LCFS, which are not taken into account in the simulations. However, these differences are too small to explain the different level of  $j_{sat}$  fluctuations, and allow us to conclude that the different levels of fluctuations between

## 6.4. Validation of the GBS simulations against experimental measurements



**Figure 6.10** – Experimental (first and third panels) and simulation (second and fourth panels)  $S(k_\theta, f)$  spectra obtained from  $V_{fl}$  time-traces for the two RFX-mod plasma discharges #38373 (first and second panels) and #38413 (third and fourth panels).

simulations and experimental measurements are not related to coherent intermittent events, which would strongly affect the PDF tails. In Fig. 6.8 we also compare the experimental joint probabilities between  $\delta j_{sat}$  and  $\delta V_{fl}$  at approximately 2 cm from the LCFS, normalized to their standard deviation, with the simulation results. A good qualitative agreement between experimental measurements and simulation results is found, with  $\delta j_{sat}/\sigma_{j_{sat}}$  and  $\delta V_{fl}/\sigma_{V_{fl}}$  showing moderate anticorrelation.

For the analysis of the equilibrium profiles and fluctuation properties, it emerges that the major difference between experimental measurements and simulations lies in the level of  $j_{sat}$  fluctuations. We explore the reason of the discrepancy in Fig. 6.9, where we display the numerical and experimental power spectral densities (PSD) related to  $j_{sat}$  and  $V_{fl}$ . We observe that for both discharges and for both quantities the PSD monotonically decreases for  $f \gtrsim 10$  kHz, in agreement with previous observations [149]. However, the simulation PSD is smaller than the experimental one, particularly for the #38413 discharge, whose plasma collisionality is smaller. We also see that the discrepancy between simulations and experimental measurements increases for  $f \gtrsim 100$  kHz. In Fig. 6.9 we also display the phase shift and the coherence between  $j_{sat}$  and  $V_{fl}$  fluctuations. First, we note that the experimental measurements are noisier for the #38373 discharge because of the presence of sawtooth instabilities and of the resulting lower temporal statistics used for the analysis. We also observe that the phase shift resulting from the nonlinear simulations is in better agreement with experimental measurements for  $f \gtrsim 20$  kHz than it is at low frequencies, where  $\mathcal{P}(\delta V_{fl}, \delta j_{sat}) > 0$  for the experimental measurements. This discrepancy at low frequencies seems related to incoherent experimental fluctuations, as shown in the last row of Fig. 6.9. In fact, the simulation results display a quite strong coherence between  $j_{sat}$  and  $V_{fl}$  at all frequencies, while the experimental measurements show a lower coherence, particularly at low frequencies. Overall, the results presented in Fig. 6.9 indicate a better agreement between simulations and experimental measurements in the frequency range  $10 \text{ kHz} \lesssim f \lesssim 100 \text{ kHz}$ , where the coherence between  $\delta j_{sat}$  and  $\delta V_{fl}$  is higher, while the agreement worsens at low and high frequencies, with the RFX-mod measurements dominated by incoherent structures.

To further investigate the discrepancies observed between simulations and experimental measurements, in Fig. 6.10 we display the  $S(k_\theta, f)$  spectra obtained from the  $V_{fl}$  time



traces at  $R - R_{LCFS} \simeq 2$  cm and related to the two plasma discharges #38373 and #38413. We note that, while the spectra obtained from the  $V_{fl}$  experimental measurements are evaluated according the two-point correlation technique describe in Ref. [167], the simulation results are obtained computing the Fourier transform of the  $V_{fl}$  time signals along  $t$  and  $y$ . In Fig. 6.10 we observe that the modes are mainly rotating in the ion diamagnetic direction, both for the experiment and the simulations. However, while in the experiment the dominant modes have  $k_{\theta}\rho_{s0} \lesssim 0.03$ , for the simulations  $0.1 \leq k_{\theta}\rho_{s0} \leq 0.2$ , as previously discussed. We note that, assuming a linear relation between  $k$  and  $f$ , in the simulations we obtain  $k_{\theta} \gtrsim 1/\rho_{s0}$  for  $f \gtrsim 100$  kHz. Since the drift approximation is not justified for  $k_{\theta}\rho_{s0} \gtrsim 1$ , and  $k_{\theta}\rho_{s0} = 1$  corresponds approximately to the maximum wave number resolved by the grids used for the present simulations, we infer that the increasing discrepancy observed for  $f \gtrsim 100$  kHz in the PSD may be related to the limits of the drift-reduced Braginskii model and to the simulation finite grid resolution.

In summary, the GBS model is able to qualitatively reproduce most of the RFX-mod experimental measurements, with the noteworthy exception of  $\delta j_{sat}$  and, in general, a better agreement for the RFX-mod plasma discharge #38373, whose plasma collisionality is higher than in the #38413 discharge. Since the  $\delta j_{sat}$  and  $\delta V_{fl}$  phase shift and joint probability agree between simulation results and experimental measurements at the frequencies where the fluctuations are more coherent, we conclude that the turbulent transport should be driven by InDWs both in simulations and in the experiment, with coherent structures having similar statistical properties.

The differences observed in the  $V_{fl}$  radial profile and in the level of  $j_{sat}$  fluctuations may be, at least in part, related to simulating only the tokamak SOL region, neglecting the plasma dynamics inside the LCFS. As a matter of fact, we note that previous tests performed considering GBS simulations of ISSTOK [168] indicate an increase of  $\delta j_{sat}^{RMS}/j_{sat}$  up to 30% when the plasma dynamics inside the LCFS is included in the simulations. In addition, sensitivity tests pointed out that  $\delta j_{sat}^{RMS}$  depends on the plasma resistivity, with  $\delta j_{sat}^{RMS}$  increasing by approximately 15% when increasing  $\nu$  by a factor ten.

## 6.5 Conclusions

In the present chapter GBS simulations based on two RFX-mod plasma discharges with low edge safety factors are discussed. The SOL turbulent regime in the two discharges is identified. Moreover, the GBS simulations are compared with experimental measurements, showing good qualitative and quantitative agreement for most of the considered quantities.

The nonlinear simulations, carried out with GBS, are based on the two RFX-mod plasma discharges #38373 and #38413. They point out that, for the two considered discharges, the turbulent transport is mostly driven by DWs. To disentangle the effect of resistivity and electron inertia on the RFX-mod SOL dynamics, a linear model is introduced. It is



found that plasma adiabaticity is mostly broken by electron inertia, resulting in InDWs. Moreover, assuming that the linear growth of BMs and DWs saturates because of the nonlinear local flattening of the plasma pressure profile, the equilibrium pressure gradient length and the wave number associated with the instability that drives most of the turbulence transport are estimated with a quasi-linear theory, showing good agreement with the nonlinear results. This theory is then employed to investigate the impact of the reduced ion to electron mass that is used in the nonlinear simulations. It is found that InDWs are expected to drive the SOL turbulence also for realistic  $m_i/m_e = 3600$ .

In order to expand the GBS validation parameter regime and assess the reliability of the GBS model at low safety factor values, the simulation results are carefully compared with RFX-mod experimental measurements. It is found that the numerical results are in good agreement with experimental radial equilibrium profiles, fluctuation measurements, and higher order moments of  $j_{sat}$  and  $V_{fl}$ , except for the equilibrium profile of  $V_{fl}$  and the level of fluctuations of  $j_{sat}$ . We infer that the observed discrepancies between simulations and experimental measurements are, at least in part, related to simulating only the tokamak SOL region, without including the plasma dynamics inside the LCFS, and to the limits of applicability of the drift reduced approximation.



# CHAPTER 7

## A rigorous validation: plasma shaping effects in TCV

Despite the fact that the effects of plasma shaping on core turbulence have been extensively investigated, both experimentally and numerically (see e.g. Refs. [169–173]), the attempts to study the effects of plasma shaping on scrape-off layer (SOL) turbulence in limited geometry are only experimental (see e.g. Ref. [174]). This motivates the work discussed in the present chapter, where the impact of tokamak inverse aspect ratio ( $\epsilon$ ), Shafranov's shift ( $\Delta$ ), elongation ( $\kappa$ ), and triangularity ( $\delta$ ) on the SOL dynamics is studied. The goal of our analysis is threefold. First, the main linear instability driving the SOL dynamics is identified, depending on the shape of the magnetic geometry. Second, an analytical model for the characteristic equilibrium gradient pressure length,  $L_p = -p_e/\nabla p_e$ , that features shaping effects is deduced. Finally, the model employed for investigating the impact of plasma shaping on the SOL plasma dynamics is rigorously validated against TCV [175] experimental measurements.

Our study is based on the drift-reduced Braginskii equations, which we express in arbitrary magnetic geometry in Section 7.1. Focusing on a limited SOL, this being the simplest configuration retaining the relevant effects of plasma shaping, we then use a simple analytical equilibrium model [176] to express the dependence of the magnetic field on  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta$ . The impact of the magnetic geometry on the growth rate of ballooning modes (BMs) and drift waves (DWs) is analysed using a linearised model, as discussed in Section 7.2. Assuming that the linear instabilities saturate due to a nonlinear local flattening of the plasma gradient and the resulting removal of the instability drive [100], we determine the main instability driving SOL turbulence and express  $L_p$  as a function of the shaping parameters, as illustrated in Section 7.3. Our theoretical findings are compared in Section 7.4 with the results of three-dimensional, global, flux-driven, nonlinear simulations of SOL turbulence carried out with GBS [20, 177]. Then, in Section 7.5 we present a possible physical interpretation of our results. Finally, the methodology used to perform a rigorous validation of a model against experimental measurements is dis-

cussed and applied to validate the GBS model against TCV measurements in Section 7.6. Our conclusions follow in Section 7.7. Part of the results presented in this chapter are published in Ref. [178].

## 7.1 The model

The high plasma collisionality in the tokamak SOL allows neglecting kinetic effects and using a fluid approach to describe plasma dynamics. In this section we present the model we consider in the present chapter, used to analyse the impact of shaping effects on SOL turbulence and to investigate the TCV SOL plasma dynamics.

### 7.1.1 Fluid moment equations

The analysis of the SOL dynamics presented in this chapter is based on the use of the drift-reduced Braginskii equations discussed in Appendix A. Within the electrostatic and Boussinesq approximations, the GBS model, Eqs. (A.65)-(A.70), reduces to

$$\frac{\partial n}{\partial t} = -\frac{R_0}{B} \{\phi, n\} + \frac{2}{B} [C(p_e) - nC(\phi)] - \nabla \cdot (nv_{\parallel e} \mathbf{b}_0) + \mathcal{D}_n(n) + S_n, \quad (7.1)$$

$$\begin{aligned} \frac{\partial \omega}{\partial t} = & -\frac{R_0}{B} \{\phi, \omega\} + \frac{2B}{n} C(p_e + \tau p_i) - v_{\parallel i} \nabla_{\parallel} \omega + \frac{B^2}{n} \nabla \cdot (j_{\parallel} \mathbf{b}_0) \\ & + \frac{B}{3n} C(G_i) + \mathcal{D}_{\omega}(\omega), \end{aligned} \quad (7.2)$$

$$\begin{aligned} \frac{\partial v_{\parallel e}}{\partial t} = & -\frac{R_0}{B} \{\phi, v_{\parallel e}\} + \frac{m_i}{m_e} \left[ \nabla_{\parallel} \phi - \frac{\nabla_{\parallel} p_e}{n} - 0.71 \nabla_{\parallel} T_e + \frac{\nu j_{\parallel}}{n} - \frac{2}{3n} \nabla_{\parallel} G_e \right] \\ & - v_{\parallel e} \nabla_{\parallel} v_{\parallel e} + \mathcal{D}_{v_{\parallel e}}(v_{\parallel e}), \end{aligned} \quad (7.3)$$

$$\frac{\partial v_{\parallel i}}{\partial t} = -\frac{R_0}{B} \{\phi, v_{\parallel i}\} - v_{\parallel i} \nabla_{\parallel} v_{\parallel i} - \frac{1}{n} \nabla_{\parallel} (p_e + \tau p_i) - \frac{2}{3n} \nabla_{\parallel} G_i + \mathcal{D}_{v_{\parallel i}}(v_{\parallel i}), \quad (7.4)$$

$$\begin{aligned} \frac{\partial T_e}{\partial t} = & -\frac{R_0}{B} \{\phi, T_e\} + \frac{4T_e}{3B} \left[ \frac{C(p_e)}{n} + \frac{5}{2} C(T_e) - C(\phi) \right] - v_{\parallel e} \nabla_{\parallel} T_e + S_{T_e} \\ & + \frac{2T_e}{3} \left[ 0.71 \frac{\nabla \cdot (j_{\parallel} \mathbf{b}_0)}{n} - \nabla \cdot (v_{\parallel e} \mathbf{b}_0) \right] + \nabla_{\parallel} (\chi_{\parallel e} \nabla_{\parallel} T_e) + \mathcal{D}_{T_e}(T_e), \end{aligned} \quad (7.5)$$

$$\begin{aligned} \frac{\partial T_i}{\partial t} = & -\frac{R_0}{B} \{\phi, T_i\} + \frac{4T_i}{3B} \left[ \frac{C(p_e)}{n} - \frac{5\tau}{2} C(T_i) - C(\phi) \right] - v_{\parallel i} \nabla_{\parallel} T_i + S_{T_i} \\ & + \frac{2T_i}{3} \left[ \frac{\nabla \cdot (j_{\parallel} \mathbf{b}_0)}{n} - \nabla \cdot (v_{\parallel i} \mathbf{b}_0) \right] + \nabla_{\parallel} (\chi_{\parallel i} \nabla_{\parallel} T_i) + \mathcal{D}_{T_i}(T_i), \end{aligned} \quad (7.6)$$

where  $\omega = \nabla_{\perp}^2 \phi + \tau \nabla_{\perp}^2 T_i$  is the plasma vorticity and we use the normalization detailed in Appendix A. As usual, the Poisson brackets are defined as  $\{\phi, A\} = \mathbf{b}_0 \cdot (\nabla \phi \times \nabla A)$ , the curvature operator as  $C(A) = B/2 [\nabla \times (\mathbf{b}_0/B)] \cdot \nabla A$ , the perpendicular Laplacian

as  $\nabla_{\perp}^2 A = -\nabla \cdot [\mathbf{b}_0 \times (\mathbf{b}_0 \times \nabla A)]$ , and the parallel gradient reduces to  $\nabla_{\parallel} A = \mathbf{b}_0 \cdot \nabla A$  in the electrostatic limit, with  $A = n, \omega, \phi, v_{\parallel i}, v_{\parallel e}, T_e, T_i$ . Equations (7.1)-(7.6) are completed by the set of boundary conditions describing the plasma dynamics at the magnetic pre-sheath entrance, Eqs. (A.75)-(A.81), detailed in Section A.3.3. With respect to the model used for the RFX-mod simulations, we remark that the present equations consider finite ion temperature effects, since typical charge exchange measurements indicate  $\tau \approx 1$  in the proximity of the last close flux surface (LCFS) in TCV.

### 7.1.2 Coordinate systems and differential operators

We express here Eqs. (7.1)-(7.6) for arbitrary magnetic geometries. We note that the dependence on the magnetic field geometry enters in the model through: (i) the norm of the magnetic field  $B$ , (ii) the direction of the unit vector  $\mathbf{b}_0$ , and (iii) the differential operators  $\{\phi, -\}$ ,  $C(-)$ ,  $\nabla_{\parallel}(-)$ , and  $\nabla_{\perp}^2(-)$ , which are computed, having defined a magnetic geometry, by expressing the covariant and contravariant components of the magnetic field and of the metric tensor in the chosen coordinate system.

In the present chapter, we make use of the toric  $(\theta_*, r, \varphi)$  and the flux-tube  $(r, \alpha, \theta_*)$  coordinate systems, where  $r$  is a flux coordinate,  $\varphi$  is the toroidal angle,  $\alpha = \varphi - q(r)\theta_*$  is a field line label, and  $\theta_*$  is the straight field line angle defined as

$$\theta_*(r, \theta) = \frac{1}{q(r)} \int_0^{\theta} \frac{\mathbf{B} \cdot \nabla \varphi}{\mathbf{B} \cdot \nabla \theta'} d\theta', \quad (7.7)$$

with  $\theta$  and  $\theta'$  the poloidal angle and

$$q(r) = \frac{1}{2\pi} \int_0^{2\pi} \frac{\mathbf{B} \cdot \nabla \varphi}{\mathbf{B} \cdot \nabla \theta} d\theta \quad (7.8)$$

the safety factor. We remark that, in the remainder of this chapter, we rescale the toric coordinate system as  $y = a\theta_*$ ,  $x = r$ , and  $z = R_0\varphi$ , where  $a$  is the tokamak minor radius in  $\tilde{\rho}_{s0}$  units. We also rescale the flux-tube coordinate system as  $X = r$ ,  $Y = (a/q)\alpha$ , and  $Z = qR_0\theta_*$ , with  $q = q(a)$  the safety factor at the LCFS.

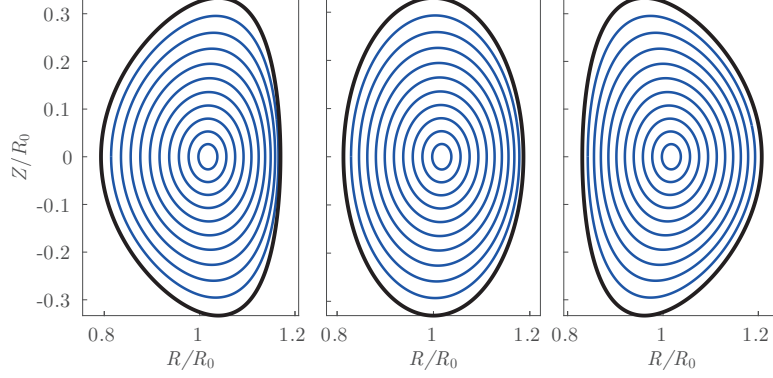
In the  $(y, x, z)$  coordinate system, the differential operators can be written in their advection form, obtaining

$$\{\phi, A\} = \mathcal{P}^{yx}\{\phi, A\}_{yx} + \mathcal{P}^{xz}\{\phi, A\}_{xz} + \mathcal{P}^{zy}\{\phi, A\}_{zy}, \quad (7.9)$$

$$\nabla_{\parallel} A = \mathcal{D}^y \frac{\partial A}{\partial y} + \mathcal{D}^x \frac{\partial A}{\partial x} + \mathcal{D}^z \frac{\partial A}{\partial z}, \quad (7.10)$$

$$C(A) = \mathcal{C}^y \frac{\partial A}{\partial y} + \mathcal{C}^x \frac{\partial A}{\partial x} + \mathcal{C}^z \frac{\partial A}{\partial z}, \quad (7.11)$$

$$\begin{aligned} \nabla_{\perp}^2 A &= \mathcal{N}^{yy} \frac{\partial^2 A}{\partial y^2} + \mathcal{N}^{yx} \frac{\partial^2 A}{\partial y \partial x} + \mathcal{N}^{xx} \frac{\partial^2 A}{\partial x^2} + \mathcal{N}^y \frac{\partial A}{\partial y} + \mathcal{N}^x \frac{\partial A}{\partial x} \\ &+ \mathcal{N}^{zz} \frac{\partial^2 A}{\partial z^2} + \mathcal{N}^{xz} \frac{\partial^2 A}{\partial x \partial z} + \mathcal{N}^{yz} \frac{\partial^2 A}{\partial y \partial z} + \mathcal{N}^z \frac{\partial A}{\partial z}, \end{aligned} \quad (7.12)$$



**Figure 7.1** – Poloidal cuts of the magnetic surfaces generated assuming a magnetic equilibrium given by Eqs. (7.20)-(7.24), for  $\epsilon = 0.25$ ,  $\Delta(0) \simeq 7$  (in  $\tilde{\rho}_{s0}$  units and for  $\beta_p = 0$ ),  $\kappa = 1.8$ , and  $\delta = -0.3$  (left panel),  $\delta = 0$  (middle panel), and  $\delta = 0.3$  (right panel). Black thick lines represent the LCFS, while blue lines represent magnetic surfaces in the core.

with  $\{\phi, A\}_{yx} = \partial_x A \partial_y \phi - \partial_y A \partial_x \phi$ , while equivalent expressions are found considering the  $(X, Y, Z)$  coordinate system. Assuming an axisymmetric magnetic field and that turbulence is characterized by  $\nabla_{\perp} A / \nabla_{\parallel} A \gg 1$ , one can compute the coefficients in Eqs. (7.9)-(7.12) (the details of the derivation are presented in Appendix C). The resulting nonzero coefficients are

$$\mathcal{P}^{yx} = \frac{a}{\mathcal{J}b^{\varphi}}, \quad \mathcal{C}^x = -\frac{R_0 B}{2\mathcal{J}} \frac{\partial c_{\varphi}}{\partial \theta_*}, \quad \mathcal{C}^y = \frac{aR_0 B}{2\mathcal{J}} \left[ \frac{\partial c_{\varphi}}{\partial r} + \frac{1}{q} \left( \frac{\partial c_{\theta_*}}{\partial r} - \frac{\partial c_r}{\partial \theta_*} \right) \right], \quad (7.13)$$

$$\mathcal{N}^{xx} = g^{rr}, \quad \mathcal{N}^{xy} = 2ag^{r\theta_*}, \quad \mathcal{N}^{yy} = a^2 \left( g^{\theta_*\theta_*} + \frac{g^{\varphi\varphi}}{q^2} \right), \quad (7.14)$$

$$\mathcal{N}^x = \nabla^2 r, \quad \mathcal{N}^y = a\nabla^2 \theta_* \quad (7.15)$$

for the  $(y, x, z)$  coordinate system and

$$\mathcal{P}^{XY} = -\frac{b_{\theta_*} a}{\mathcal{J}q}, \quad \mathcal{D}^Z = qR_0 b^{\theta_*}, \quad (7.16)$$

$$\mathcal{C}^X = -\frac{R_0 B}{2\mathcal{J}} \frac{\partial c_{\alpha}}{\partial \theta_*}, \quad \mathcal{C}^Y = \frac{aR_0 B}{2\mathcal{J}q} \left( \frac{\partial c_r}{\partial \theta_*} - \frac{\partial c_{\theta_*}}{\partial r} \right), \quad (7.17)$$

$$\mathcal{N}^{XX} = g^{rr}, \quad \mathcal{N}^{XY} = \frac{2g^{\alpha r} a}{q}, \quad \mathcal{N}^{YY} = \frac{a^2 g^{\alpha\alpha}}{q^2}, \quad (7.18)$$

$$\mathcal{N}^X = \nabla^2 r, \quad \mathcal{N}^Y = \frac{a}{q} \nabla^2 \alpha \quad (7.19)$$

for the  $(X, Y, Z)$  coordinate system.

### 7.1.3 Magnetic equilibrium

In the present chapter we consider the magnetic equilibrium that is obtained by solving the Grad-Shafranov equation in the  $r/R_0 \rightarrow 0$  limit, taking into account elongation and

nonzero triangularity [176], neglecting the plasma pressure contribution, and extrapolating the equilibrium to  $r = a$ . The solution of the Grad-Shafranov equation, presented in Appendix D, leads to

$$R_c(r, \theta) = R_0 \left\{ 1 + \frac{r}{R_0} \cos \theta + \frac{\Delta(r)}{R_0} + \sum_{m=2}^3 \frac{S_m(r)}{R_0} \cos[(m-1)\theta] - \frac{P(r)}{R_0} \cos \theta \right\}, \quad (7.20)$$

$$Z_c(r, \theta) = R_0 \left\{ \frac{r}{R_0} \sin \theta - \sum_{m=2}^3 \frac{S_m(r)}{R_0} \sin[(m-1)\theta] - \frac{P(r)}{R_0} \sin \theta \right\}, \quad (7.21)$$

$$F(r) = R_0 \left\{ 1 - \frac{r^2}{R_0^2} \frac{1}{q(r)^2} \left[ 2 + \frac{q - q_0}{q_0} \left( \frac{r}{a} \right)^2 \right] \right\}, \quad (7.22)$$

$$\psi'(r) = \frac{F(r)}{2\pi q(r)} \int_0^{2\pi} \frac{\mathcal{J}_{\theta r \varphi}}{R_c(r, \theta)^2} d\theta, \quad (7.23)$$

$$q(r) = q_0 + (q - q_0) \left( \frac{r}{a} \right)^2, \quad (7.24)$$

where  $\Delta(r)$  is the Shafranov's shift in the  $\beta_p \rightarrow 0$  limit (being  $\beta_p$  the ratio of the plasma pressure to the poloidal magnetic pressure),  $S_2(r)$  and  $S_3(r)$  are shaping functions related to  $\kappa$  and  $\delta$  by  $\kappa = [a - S_2(a)]/[a + S_2(a)]$  and  $\delta = 4S_3(a)/a$ ,  $q_0$  is the safety factor at the magnetic axis, and  $\mathcal{J}_{\theta r \varphi} = R_c(\partial_r R_c \partial_\theta Z_c - \partial_\theta R_c \partial_r Z_c)$  is the Jacobian associated with the  $(\theta, r, \varphi)$  coordinate system. The analytical expressions of the functions  $S_2(r)$ ,  $S_3(r)$ ,  $\Delta'(r)$ , and  $P(r)$  are given in Appendix D. Examples of magnetic surfaces resulting from this model are presented in Fig. 7.1.

Combining Eq. (7.7) with the expression of the axisymmetric field  $\mathbf{B}$ , one obtains

$$\theta_*(r, \theta) = \frac{F(r)}{q(r)\psi'(r)} \int_0^\theta \frac{\mathcal{J}_{\theta' r \varphi}}{R_c(r, \theta')^2} d\theta'. \quad (7.25)$$

We note that  $\theta_* = \theta = 0$  and  $\theta_* = \theta = 2\pi$  correspond to the equatorial low-field side midplane and that, in general, Eq. (7.25) is not analytically integrable, but it is possible to find numerically the two functions  $\theta_* = \theta_*(r, \theta)$  and the inverse  $\theta = \theta(r, \theta_*)$ . From these, all the coefficients of Eqs. (7.13)-(7.19) can be computed, and Eqs. (7.1)-(7.6) are completely defined. Finally, we note that in the limit  $\Delta(0) \rightarrow 0$ ,  $\kappa \rightarrow 1$ ,  $\delta \rightarrow 0$ , and neglecting  $P(r)$  and the  $r^2/R_0^2$  term of Eq. (7.22), we obtain the circular magnetic equilibrium discussed in Ref. [179].

## 7.2 Linear instabilities

The turbulent transport observed in the tokamak SOL is due to the nonlinear development of linear modes that are destabilized by plasma gradients and unfavorable magnetic curvature in the presence of resistivity and electron inertia [180–184]. For typical SOL parameters, BMs and DWs are found to be the main instabilities driving plasma turbu-

lence [17, 145, 157].

In the remainder of this section, focusing on the resistive branch of BMs and DWs, we discuss the properties of SOL linear instabilities and, in particular, the derivation of resistive BM (RBM) and resistive DW (RDW) dispersion relations in non-circular magnetic geometries. Following the procedure applied in Section 6.3.2, we linearise Eqs. (7.1)-(7.6) assuming that the equilibrium plasma profiles depend only on the radial coordinate  $X$  and we write the perturbation as  $\delta A(Y, Z, t) = \delta A(Z) \exp[ik_Y Y + \gamma t]$ , with  $k_Y$  the poloidal wave number and  $\gamma$  the linear growth rate. In the cold ion limit (for a discussion of the impact of ion temperature on SOL instabilities we refer to Ref. [158]), i.e. for  $\tau = 0$ , the resulting system of equations writes

$$\gamma \delta n = \frac{R_0}{L_n} \frac{1}{B} \mathcal{P}^L(\delta \phi) + \frac{2}{B} \mathcal{C}^L(\delta p_e - \delta \phi) + (\nabla_{\parallel} + \nabla \cdot \mathbf{b}_0) (\delta j_{\parallel} - \delta v_{\parallel i}), \quad (7.26)$$

$$\frac{1}{B^2} \gamma \delta \omega = \frac{2}{B} \mathcal{C}^L(\delta p_e) + (\nabla_{\parallel} + \nabla \cdot \mathbf{b}_0) \delta j_{\parallel}, \quad (7.27)$$

$$\frac{m_e}{m_i} \gamma \delta v_{\parallel e} = \nabla_{\parallel} (\delta \phi - \delta p_e - 0.71 \delta T_e) + \nu \delta j_{\parallel}, \quad (7.28)$$

$$\gamma \delta v_{\parallel i} = -\nabla_{\parallel} \delta p_e, \quad (7.29)$$

$$\begin{aligned} \gamma \delta T_e = & \frac{R_0}{L_n} \frac{\eta}{B} \mathcal{P}^L(\delta \phi) + \frac{4}{3B} \mathcal{C}^L \left( \delta p_e + \frac{5}{2} \delta T_e - \delta \phi \right) \\ & + \frac{2}{3} (\nabla_{\parallel} + \nabla \cdot \mathbf{b}_0) (1.71 \delta j_{\parallel} - \delta v_{\parallel i}), \end{aligned} \quad (7.30)$$

where  $\delta p_e = \delta n + \delta T_e$ ,  $\delta j_{\parallel} = \delta v_{\parallel i} - \delta v_{\parallel e}$ ,  $\delta \omega = (\nabla_{\perp}^2)^L \delta \phi$ ,  $\eta = L_n/L_{T_e}$  and

$$\mathcal{P}^L(A) = i \mathcal{P}^{XY} k_Y A, \quad \mathcal{C}^L(A) = i \mathcal{C}^Y k_Y A, \quad (\nabla_{\perp}^2)^L A = -\mathcal{N}^{YY} k_Y^2 A. \quad (7.31)$$

Equations (7.26)-(7.30) determine the linear growth rate of SOL plasma instabilities in arbitrary magnetic geometry. Plasma shaping affects the growth rate through four terms:  $\nabla_{\parallel}(-) + \nabla \cdot \mathbf{b}_0$  that results from the plasma advection along magnetic field lines,  $\mathcal{P}^L(-)/B$  that represents the  $\mathbf{E} \times \mathbf{B}$  convection,  $\mathcal{C}^L(-)/B$  that introduces curvature drifts and plasma compressibility, and  $(\nabla_{\perp}^2)^L(-)/B^2$ , the vorticity operator.

To solve Eqs. (7.26)-(7.30) in arbitrary magnetic geometry, the numerical code detailed in Section 6.3.2 is generalized to evaluate  $\gamma$  as a function of the parameters  $R_0/L_n$ ,  $m_i/m_e$ ,  $\eta$ ,  $\nu$ ,  $q$ ,  $k_Y$ ,  $\epsilon$ ,  $\kappa$ , and  $\delta$ , with  $\epsilon = a/R_0$ . The coefficients of the differential operators are implemented according to Eqs. (7.16)-(7.19) at  $X = a$  as follows. First, Eq. (7.25) is solved numerically, using the trapezoidal rule to approximate the integral. Then, the relation  $\theta_* = \theta_*(r, \theta)$  is inverted using a linear interpolation scheme, to obtain the function  $\theta = \theta(r, \theta_*)$ . The derivatives of the magnetic field components and of the metric coefficients appearing in Eqs. (7.16)-(7.19) are finally computed using second order finite difference schemes. We verified that the evaluation of the geometric coefficients is numerically converged, and that, in the limit of circular magnetic surfaces, we recover the analytical expressions given in Ref. [179].

In the following we focus our attention on the high plasma resistivity regime by setting  $\nu = 0.1$  and describe the impact of plasma shaping on RBMs and RDWs. This value,



larger than in usual tokamaks SOL plasma conditions, is chosen to completely decouple the resistive and the inertial branches of the BMs and DWs. As shown in the remainder of this chapter, plasma shaping mainly influences the RBMs growth rate, while the value of  $\nu$  has only a minor impact in determining the linear growth rate and the wave number of the mode driving the turbulent transport. We note that we consider  $q = 4$ ,  $R_0 = 500$ , and  $\eta = 0.66$  for our linear studies ( $\eta$  is obtained from nonlinear simulations, see Section 7.4, in agreement with Ref. [161] results). The small positive value of the magnetic shear in limited discharges is expected to stabilize DWs [146, 160] and should weakly affect BMs.

### 7.2.1 Resistive ballooning modes

Resistive BMs are interchange-like modes, driven unstable in the presence of finite resistivity when the plasma pressure gradient and the magnetic field line curvature point in the same direction [185–189]. Setting  $\partial_Z \rightarrow ik_Z$  and neglecting the compressibility terms with respect to the advection terms, the parallel dynamics in the continuity and temperature equations, the diamagnetic term in the Ohm's law to avoid coupling with DWs, as well as the coupling with sound waves (valid for  $\gamma \gg k_Z$ ,  $k_Z \sim 1/q$  for RBMs), Eqs. (7.26)-(7.30) can be simplified to obtain

$$\gamma \delta p_e = \frac{R_0}{L_p} \frac{1}{B} \mathcal{P}^L(\delta \phi), \quad (7.32)$$

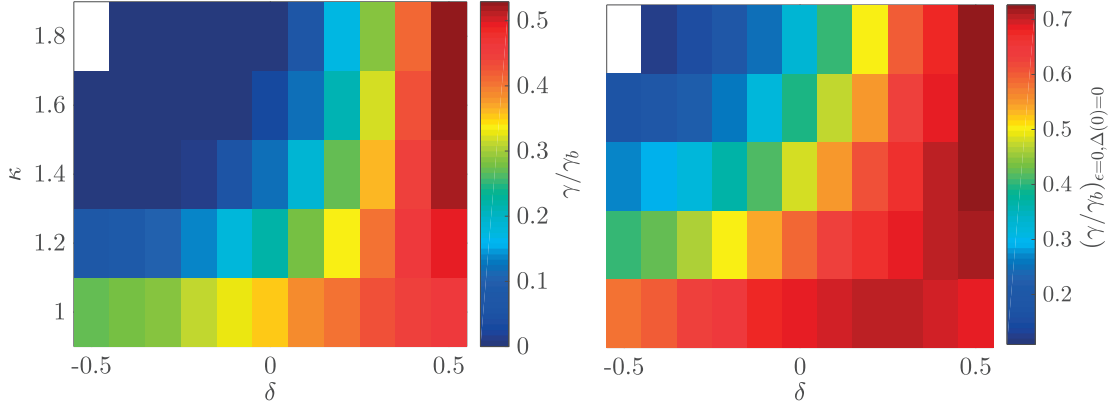
$$\frac{1}{B^2} \gamma (\nabla_{\perp}^2)^L \delta \phi = \frac{2}{B} \mathcal{C}^L(\delta p_e) + (i\mathcal{D}^Z k_Z + \nabla \cdot \mathbf{b}_0) \delta j_{\parallel}, \quad (7.33)$$

$$0 = \nu \delta j_{\parallel} + i\mathcal{D}^Z k_Z \delta \phi, \quad (7.34)$$

where  $L_p = L_n/(1 + \eta)$ . Equations (7.32)-(7.34) constitute the minimal model describing the linear properties of RBMs.

Considering Eqs. (7.32)-(7.34) in a circular magnetic geometry in the  $\epsilon \rightarrow 0$  limit, it was shown [146] that the peak growth rate,  $\gamma \sim \gamma_b$ , with  $\gamma_b = \sqrt{2R_0/L_p}$  being the ideal BM growth rate, occurs for  $k_Y \sim k_b$ , where  $k_b = 1/\sqrt{q^2 \nu \gamma_b}$ . Imposing  $k_Y = k_b$ , we compute the ratio  $\gamma/\gamma_b$  as a function of  $\kappa$  and  $\delta$  for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ . The results are presented in Fig. 7.2 (left panel). We observe that RBMs are stabilized for  $\kappa > 1$  and for  $\delta < 0$ , while their growth rate is enhanced for  $\delta > 0.2$ . To isolate the different shaping effects on the linear growth rate, we perform the same analysis for  $\epsilon = 0$  and  $\Delta(0) = 0$ . The results are presented in Fig. 7.2 (right panel). While the same trends are recovered, the ratio  $\gamma/\gamma_b$  is larger for  $\epsilon = 0$  and  $\Delta(0) = 0$  than for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ . This suggests that aspect ratio and Shafranov's shift effects stabilize RBMs, in agreement with the observations in Ref. [179]. We note that these results are independent of  $L_p$ .

A detailed analysis of the linear growth rate shows that the curvature operator affects the RBM growth rate the most. As a matter of fact, assuming  $\mathcal{P}^{XY} = -1$ ,  $\mathcal{N}^{YY} = 1$ ,  $\mathcal{D}^Z = 1$ ,  $\nabla \cdot \mathbf{b}_0 = 0$ ,  $B = 1$ , and  $\mathcal{C}^Y = -\partial_r R(r, \theta)|_{\theta_*}$ , where  $|\theta_*$  indicates that the  $r$  derivative is computed at fixed  $\theta_*$ , the results of Fig. 7.2 are recovered, within an error



**Figure 7.2** – Value of  $\gamma/\gamma_b$  as a function of  $\kappa$  and  $\delta$ , obtained considering RBMs and solving Eqs. (7.32)-(7.34) for  $k_Y = k_b$ . The left panel is obtained for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ , while the right panel is computed for  $\epsilon = 0$  and  $\Delta(0) = 0$ .

on  $\gamma/\gamma_b$  typically less than 20%.

This observation allows to simplify Eqs. (7.32)-(7.34) assuming a strongly ballooned mode around  $\theta_* = 0$ ,  $k_Z = 1/q$ ,  $\mathcal{P}^{XY} = -1$ ,  $\mathcal{N}^{YY} = 1$ ,  $\mathcal{D}^Z = 1$ ,  $\nabla \cdot \mathbf{b}_0 = 0$ ,  $B = 1$ , and  $\mathcal{C}^Y = -\partial_r R(r, \theta)|_{\theta_* = 0}$ . In fact, one can write the RBMs dispersion relation as

$$\bar{\gamma}^2 + 2\gamma_D \bar{\gamma} - \gamma_I^2 = 0, \quad (7.35)$$

where  $\gamma_D = -1/(2\nu q^2 k_Y^2 \gamma_b)$ ,  $\gamma_I = \sqrt{\partial_r R(r, \theta)|_{\theta_* = 0}}$ , and  $\bar{\gamma} = \gamma/\gamma_b$ . Equation (7.35) shows that the RBM has a growth rate  $\gamma = \gamma_I \gamma_b$  and it is stabilized by finite  $k_Z$  effects through the  $\gamma_D$  term.

## 7.2.2 Resistive drift waves

Resistive DWs are instabilities driven by the  $\mathbf{E} \times \mathbf{B}$  plasma convection and destabilized when electron adiabaticity is broken by finite resistivity [190–194]. Assuming  $\gamma \gg k_Z$  as for the RBMs, we can neglect sound wave coupling in Eqs. (7.26)-(7.30). Moreover, the curvature term in the vorticity equation can be neglected (to avoid coupling with BMs), together with the compressibility terms in the continuity and temperature equations. Consequently, we obtain a reduced system of equations describing the RDWs dynamics, which writes

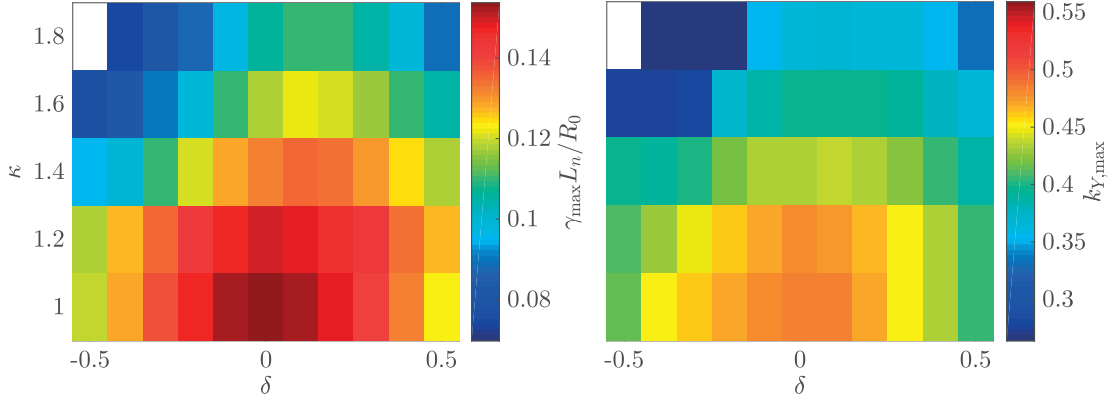
$$\gamma \delta n = \frac{R_0}{L_n} \frac{1}{B} \mathcal{P}^L(\delta \phi) + (i\mathcal{D}^Z k_Z + \nabla \cdot \mathbf{b}_0) j_{\parallel}, \quad (7.36)$$

$$\frac{1}{B^2} \gamma \delta \omega = (i\mathcal{D}^Z k_Z + \nabla \cdot \mathbf{b}_0) \delta j_{\parallel}, \quad (7.37)$$

$$0 = i\mathcal{D}^Z k_Z (\delta \phi - \delta p_e - 0.71 \delta T_e) + \nu \delta j_{\parallel}, \quad (7.38)$$

$$\gamma \delta T_e = \frac{R_0}{L_n} \frac{\eta}{B} \mathcal{P}^L(\delta \phi) + 1.71 \frac{2}{3} (i\mathcal{D}^Z k_Z + \nabla \cdot \mathbf{b}_0) \delta j_{\parallel}. \quad (7.39)$$

### 7.3. Estimate of the pressure gradient length



**Figure 7.3** – Values of  $\gamma_{\max} L_n / R_0$  (left panel) and  $k_{Y,\max}$  (right panel) as a function of  $\kappa$  and  $\delta$ , obtained considering RDWs and solving Eqs. (7.36)-(7.39) for  $L_p = 10$ ,  $\epsilon \simeq 0.25$ , and  $\Delta(0) \simeq 7$ .

Equations (7.36)-(7.39) are solved as a function of  $k_Y$ ,  $\kappa$ , and  $\delta$ , by using the linear solver previously described. For each magnetic shape, the growth rate is maximized over all possible  $k_Y$  values, and the maximum value  $\gamma_{\max}$  and the corresponding  $k_Y$ , denoted as  $k_{Y,\max}$ , are presented in Fig. 7.3 for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ . We note that  $\gamma_{\max}$  normalized to  $R_0/L_n$ , i.e.  $\gamma_{\max} L_n / R_0$ , is independent of  $L_n$ . We see that  $\gamma_{\max}$  associated with the RDWs decreases both by increasing  $\kappa$  and  $|\delta|$ , suggesting that RDWs are most unstable in a circular magnetic geometry. However, the effect of plasma shaping on the RDW growth rate is considerably weaker than on RBMs. We also note that  $k_{Y,\max}$  decreases with  $\kappa$  and  $|\delta|$ . The analysis of the impact of aspect ratio and Shafranov's shift effects on RDWs growth rate shows a small influence on  $\gamma_{\max}$  and  $k_{Y,\max}$ , in agreement with the observations in Ref. [179].

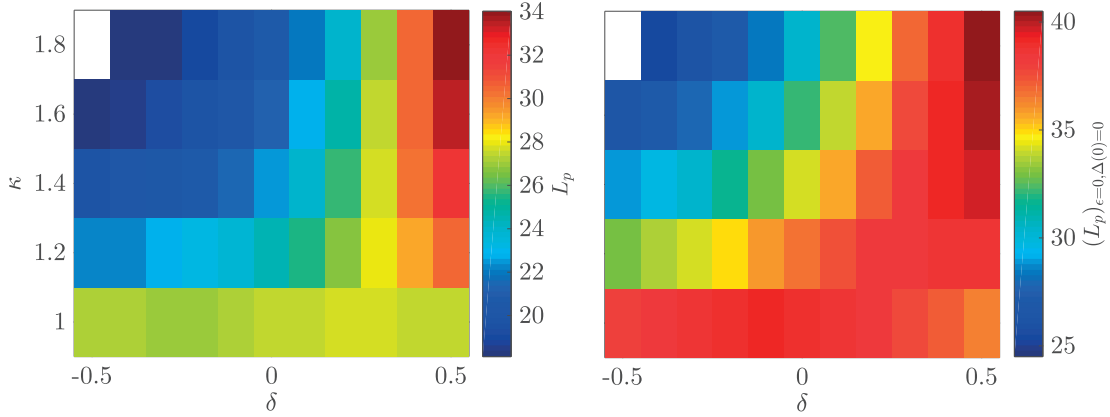
Finally, we note that, considering a circular magnetic geometry in the infinite aspect ratio limit and setting  $\partial_Z \rightarrow ik_Z$ , we simplify Eqs. (7.36)-(7.39) to write the dispersion relation of RDWs as

$$\frac{1}{2} \hat{\gamma}^2 + ab\hat{\gamma} + i\alpha = 0, \quad (7.40)$$

where  $\alpha = k_Z^2 L_n / [2k_Y R_0 \nu (1 + 1.71\eta)]$ ,  $b = 1/k_Y + k_Y (1 + 1.71^2 \cdot 2/3)$ , and  $\hat{\gamma} = \gamma L_n / [R_0 (1 + 1.71)\eta]$ . We remark that Eq. (7.40) has to be solved over all the possible  $k_Y$  and  $k_Z$  values. In fact,  $k_Z$  is a priori unknown for RDWs and depends on  $\nu$ ,  $R_0/L_n$ , and  $\eta$ . Solving Eq. (7.40) numerically, it results that  $\text{Re}(\hat{\gamma}_{\max}) \simeq 0.0874$  for  $k_Y \simeq 0.582$  and  $\alpha \simeq 0.0412$ , where  $\text{Re}(A)$  indicates the real part of  $A$ .

### 7.3 Estimate of the pressure gradient length

The time-averaged plasma pressure gradient length  $L_p$  in the tokamak SOL originates from a balance between the turbulent perpendicular transport of particles and heat, resulting from the nonlinear development of the unstable modes, and the parallel losses



**Figure 7.4** – Value of  $L_p$  as a function of  $\kappa$  and  $\delta$  that are solution of Eq. (7.41). The left panel is obtained for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ , while the right panel is computed for  $\epsilon = 0$  and  $\Delta(0) = 0$ .

at the end of the magnetic field lines. In the limit of a negligible  $\mathbf{E} \times \mathbf{B}$  shear flow and for typical SOL parameters, we assume that the gradient removal turbulence saturation mechanism, i.e. the local nonlinear flattening of the plasma pressure profile and the resulting removal of the instability drive, is the mechanism that regulates the amplitude of SOL turbulence [100]. Our estimates of  $L_p$  based on the gradient removal theory in circular magnetic flux surface geometries show agreement with nonlinear simulations and experimental observations [24, 113].

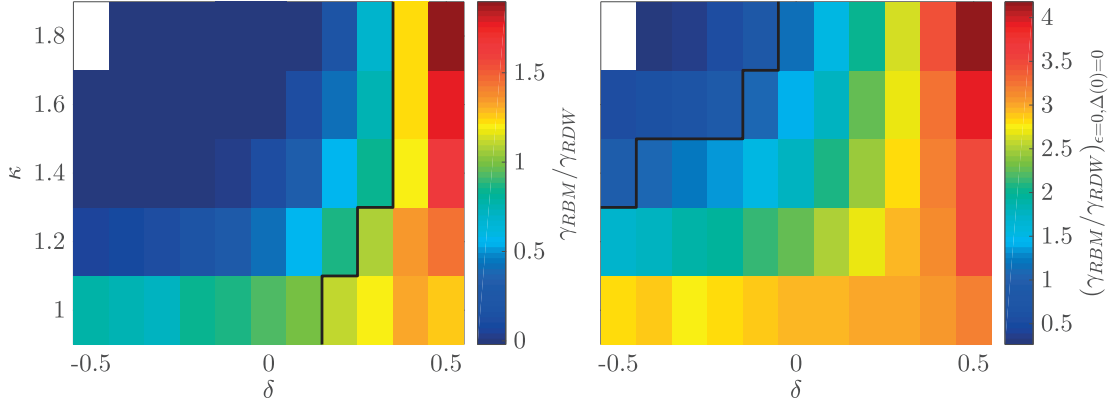
The main features of the theory are summarized in Appendix B, where the equality

$$L_p = \frac{q}{c_s} \left( \frac{\gamma}{k_Y} \right)_{\max} \quad (7.41)$$

is deduced. Since Eq. (7.41) is an implicit equation in  $L_p$ , it must be approached numerically. We follow the same procedure detailed in Section 6.3.2. In particular, we solve Eqs. (7.26)-(7.30) for  $\gamma$  over the parameter space  $(k_Y, L_p)$  in search for the value of  $L_p$  that satisfies Eq. (7.41). We note that in the remainder of this section we assume  $c_s = 1$  (i.e. that the reference temperature corresponds to the one at the LCFS) and we consider  $m_i/m_e = 2000$ ,  $\eta = 0.66$ ,  $q = 4$ , and  $R_0 = 500$ .

The values of  $L_p$  that satisfy Eq. (7.41) for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$  are presented in Fig. 7.4 (left panel) as a function of  $\kappa$  and  $\delta$ . Several observations can be made based on these results: (i) triangularity has a weak impact on  $L_p$  when  $\kappa = 1$ ; (ii) for  $\delta = 0$ ,  $L_p$  is reduced by increasing  $\kappa$ , suggesting that turbulence is suppressed by elongation; (iii) for  $\kappa > 1$ ,  $L_p$  is reduced for  $\delta < 0$ , indicating that turbulence is suppressed by negative triangularity; and (iv) for  $\kappa > 1$ ,  $L_p$  is increased for  $\delta \gtrsim 0$ , the effect becoming more relevant at  $\delta \gtrsim 0.3$ , meaning that turbulence is enhanced by large positive triangularity. To isolate the different shaping effects on the SOL width, we perform a  $L_p$  scan on  $\kappa$  and  $\delta$  with  $\Delta(0) = 0$  and  $\epsilon = 0$ . The results are presented in Fig. 7.4 (right panel). While the same trends previously observed for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$  are recovered,  $L_p$  is larger, suggesting that Shafranov's shift and  $\epsilon$  effects stabilize plasma turbulence. This is in agreement with the observations presented in Section 7.2.

### 7.3. Estimate of the pressure gradient length



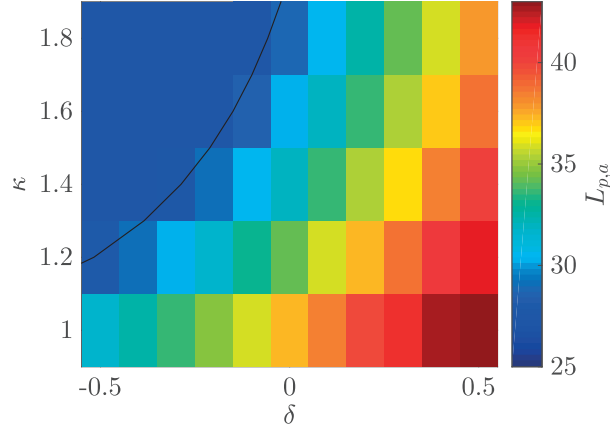
**Figure 7.5** – Ratio between  $\gamma_{RBM}$  and  $\gamma_{RDW}$  as a function of  $\kappa$  and  $\delta$ , obtained solving Eqs. (7.32)-(7.34) and (7.36)-(7.39) with the linear code imposing the  $L_p$  and  $k_Y$  values that are solution of Eq. (7.41). The left panel is obtained for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ , while the right panel is computed for  $\epsilon = 0$  and  $\Delta(0) = 0$ . The black lines indicate the transition between  $\gamma_{RDW} > \gamma_{RBM}$  and  $\gamma_{RDW} < \gamma_{RBM}$ .

We note that the solution of Eq. (7.41) provides also the  $k_Y$  value of the mode that leads to most of the transport. An analysis of  $k_Y$  as a function of  $\kappa$  and  $\delta$  has been performed, showing that plasma shaping has a small impact on its value.

To investigate the impact of the plasma shaping on the turbulent regimes, and therefore to gain a deeper insight on the turbulence properties, we compute the growth rate associated with RBMs ( $\gamma_{RBM}$ ) and RDWs ( $\gamma_{RDW}$ ) at the  $L_p$  and  $k_Y$  solutions of Eq. (7.41), as a function of  $\kappa$  and  $\delta$ . This is done by solving Eqs. (7.32)-(7.34) and (7.36)-(7.39) with the linear solver presented in Section 7.2. The ratio between  $\gamma_{RBM}$  and  $\gamma_{RDW}$  as a function of  $\kappa$  and  $\delta$  for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$  is presented in Fig. 7.5 (left panel). We observe that: (i) the RBM is the dominant instability for positive values of the triangularity; (ii) the RDW dominates when negative  $\delta$  values are considered; (iii) a combination of the two instabilities characterizes the plasma dynamics for  $\kappa \simeq 1$  and for  $\delta \simeq 0.2 - 0.3$ . This is in agreement with the observations in Section 7.2, where it is shown that RDWs are only slightly affected by plasma shaping, while RBMs are strongly stabilized by elongation and negative triangularity.

We also study the impact of Shafranov's shift and  $\epsilon$  effects on RBMs and RDWs by performing the same analysis for  $\epsilon \simeq 0$  and  $\Delta(0) = 0$ . The results (see Fig. 7.5, right panel) show that Shafranov's shift and  $\epsilon$  effects stabilize the RBMs, leading to a decrease of the equilibrium pressure gradient length, as discussed earlier.

We now deduce an analytical scaling of  $L_p$  as a function of  $\kappa$  and  $\delta$ . Since RBMs are strongly affected by plasma shaping, while the  $(\gamma/k_Y)_{\max}$  ratio associated with RDWs depends weakly on the magnetic shape [variation of  $(\gamma/k_Y)_{\max}$  less than 30% for  $1 \leq \kappa \leq 1.8$  and  $-0.5 \leq \delta \leq 0.5$ ], we derive the analytical scaling of  $L_p$  considering the RBM dispersion relation only, and we assume that the resulting scaling breaks down if  $L_{p,a} < L_{p,RDW}$ , where  $L_{p,a}$  is the result of the RBM analytical scaling and  $L_{p,RDW}$  is the equilibrium pressure gradient length associated with RDWs deduced from Eqs. (7.40) and (7.41) for a circular magnetic geometry ( $L_{p,RDW} = 27.7$  for  $\nu = 0.1$ ,  $q = 4$ ,  $R_0 = 500$ , and  $\eta = 0.66$ ).



**Figure 7.6** – Value of  $L_p$  as a function of  $\kappa$  and  $\delta$ , obtained from Eq. (7.44) for  $q = 4$ ,  $\nu = 0.1$ , and  $R_0 = 500$ . The black line indicates  $L_p(\kappa, \delta) = L_{p,RDW}$ . We note that in the top-left corner we impose  $L_p = L_{p,RDW}$ .

Having observed that magnetic shape affects the RBM growth rate mostly through the curvature operator, we assume a strongly ballooned mode around  $\theta_* = 0$ ,  $k_Z \sim 1/q$ , and  $\mathcal{C}^Y \simeq -\partial_r R(r, \theta)|_{\theta_*=0}$  to simplify Eqs. (7.32)-(7.34). Moreover, imposing  $\epsilon = 0$  and  $\Delta(0) = 0$  for the sake of simplicity, we evaluate  $\partial_{k_Y}(\gamma/k_Y) = 0$  to identify the largest  $\gamma/k_Y$  ratio, obtaining

$$\gamma^2 = \gamma_b^2 \frac{C(\kappa, \delta, q)}{3}, \quad k_Y^2 = \sqrt{3} \frac{k_b^2}{2} C(\kappa, \delta, q)^{-1/2}, \quad (7.42)$$

where

$$C(\kappa, \delta, q) = \partial_r R_c(r, \theta)|_{\theta_*=0} = 1 + \frac{\delta q}{1+q} + \frac{\delta^2(7q-1)}{16(1+q)} - \frac{\kappa-1}{2(\kappa+1)} - \frac{(\kappa-1)(5q-2)}{(\kappa+1)^2(2+q)}, \quad (7.43)$$

as deduced in Appendix E. Plugging these values into Eq. (7.41), we derive

$$L_{p,a} = \frac{2^{5/7}}{3^{3/7}} C(\kappa, \delta, q)^{3/7} \nu^{2/7} q^{8/7} R_0^{3/7}. \quad (7.44)$$

Equation (7.44) is a generalization of the scaling deduced in Refs. [24,100,113] to include non-circular magnetic geometries and allows us to predict the SOL width of inner-wall limited discharges from first-principle arguments when elongation and nonzero triangularity are considered. In Fig. 7.6 we present the result of this scaling. We recover the same trends observed in Fig. 7.4 and the transition between RDWs and RBMs, taking place at  $L_{p,a} = L_{p,RDW}$  and indicated by the black line, in good agreement with the condition  $\gamma_{RBM}/\gamma_{RDW} = 1$  of Fig. 7.5 (right panel). However, we notice two main differences between the analytical scaling and the numerical results: (i) for  $\kappa = 1$  the analytical scaling shows that  $L_p$  depends on  $\delta$  while  $L_p$  is almost independent of  $\delta$  in Fig. 7.4; and (ii) for  $\delta > 0.3$  the value of  $L_p$  decreases by increasing the plasma elongation, while the opposite behavior is observed in the results of Fig. 7.4. By solving Eq. (7.41) within each assumption made in the derivation of Eq. (7.44), we observe that

the differences between the analytical scaling and the numerical results are mainly due to the approximation  $\mathcal{C}^Y \approx \mathcal{C}^Y|_{\theta_*=0}$ . In fact, the global magnetic geometry has an effect on RBMs that cannot be correctly captured by the modification of the curvature at the outer midplane.

## 7.4 Nonlinear simulations

In this section we carry out a set of nonlinear simulations of the SOL plasma dynamics using the GBS code [20] and compare these with the results presented in Sections 7.2 and 7.3. We first describe the numerical scheme implemented in GBS, focusing on the modifications introduced to generalize the magnetic geometry of the code. We then present the nonlinear simulation results and their comparison with our theoretical findings.

### 7.4.1 Implementation and numerics

The model presented in Section 7.1 is now implemented in the GBS code [20]. A detailed description of the code can be found in Appendix A.

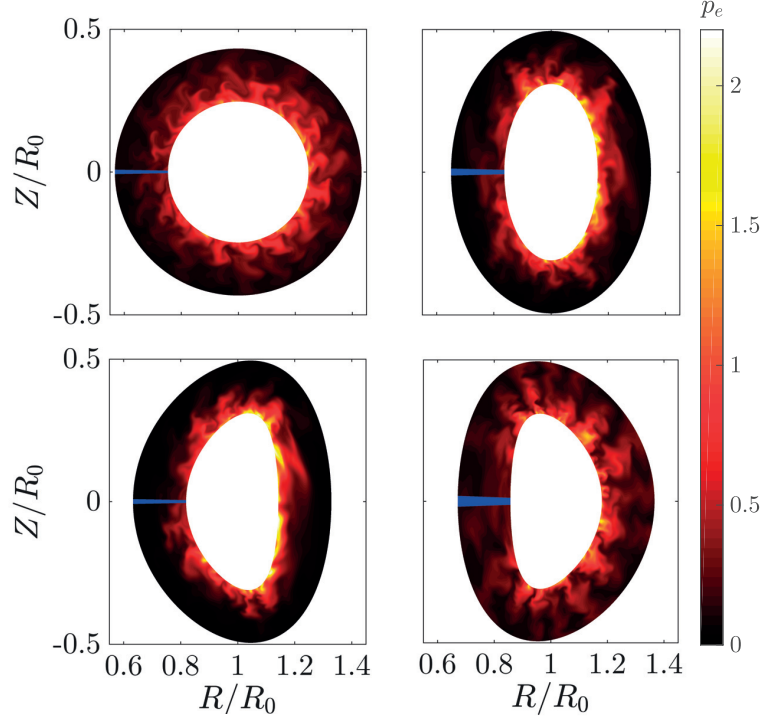
While the implementation of the differential operators  $\{\phi, A\}$ ,  $C(A)$ , and  $\nabla_{\parallel}A$  does not present any conceptual difficulty, the computation of the  $\nabla_{\perp}^2$  operator in arbitrary geometry is not straightforward. In fact, the discretization of the operator  $\nabla_{\perp}^2$  can introduce numerical instabilities with positive growth rate if  $\nabla_{\perp}^2$  is represented with a non-symmetric real matrix  $\mathbf{D}$ . To ensure the self-adjointness of  $\mathbf{D}$ , we write

$$\nabla_{\perp}^2 A = \frac{\partial}{\partial x} \left( \mathcal{N}^{xx} \frac{\partial A}{\partial x} + \frac{\mathcal{N}^{xy}}{2} \frac{\partial A}{\partial y} \right) + \frac{\partial}{\partial y} \left( \mathcal{N}^{yy} \frac{\partial A}{\partial y} + \frac{\mathcal{N}^{xy}}{2} \frac{\partial A}{\partial x} \right) + \mathcal{N}'^x \frac{\partial A}{\partial x} + \mathcal{N}'^y \frac{\partial A}{\partial y}, \quad (7.45)$$

with  $\mathcal{N}'^x = \mathcal{N}^x - \partial_x \mathcal{N}^{xx} - \partial_y \mathcal{N}^{xy}/2$  and  $\mathcal{N}'^y = \mathcal{N}^y - \partial_y \mathcal{N}^{yy} - \partial_x \mathcal{N}^{xy}/2$ , and we neglect the  $\mathcal{N}'^x$  and  $\mathcal{N}'^y$  terms with respect to the  $\mathcal{N}^{xx}$ ,  $\mathcal{N}^{xy}$ , and  $\mathcal{N}^{yy}$  terms, since the first two terms are usually  $a$  times smaller than the last three, and they are therefore expected to have a negligible impact on the simulation results. Finally, we note that the geometric coefficients in Eqs. (7.13)-(7.15) are computed with the same numerical scheme presented in Section 7.2 and used for the coefficients in Eqs. (7.16)-(7.19).

### 7.4.2 Simulation results

Two sets of four nonlinear simulations with  $(\kappa, \delta) = (1, 0), (1.8, 0), (1.8, -0.3), (1.8, 0.3)$  are carried out with GBS, considering in one case  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ , and in the other  $\epsilon = 0$  and  $\Delta(0) = 0$ . Other relevant physical parameters are  $q = 4$ ,  $\nu = 0.1$ ,  $R_0 = 500$ ,  $m_i/m_e = 200$ ,  $\tau = 0$ ,  $L_y = 800$ , and  $L_x = 150$ , being  $L_x$  the radial size of the domain, extending from  $x_i = a - 50$  to  $x_o = a + 100$ . The particle and electron

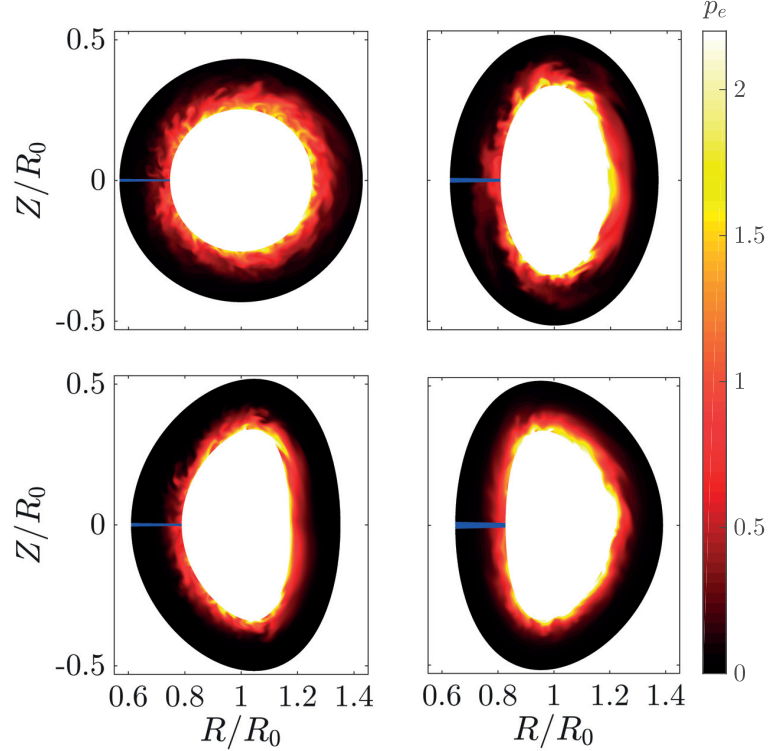


**Figure 7.7** – Poloidal cross sections of the electron pressure plasma profile resulting from nonlinear simulations carried out for  $\epsilon = 0$  and  $\Delta(0) = 0$ , extending from  $x = a$  to  $x = a + 90$ . The four magnetic geometries considered are characterized by  $\kappa = 1$  and  $\delta = 0$  (top left panel),  $\kappa = 1.8$  and  $\delta = 0$  (top right panel),  $\kappa = 1.8$  and  $\delta = -0.3$  (bottom left panel), and  $\kappa = 1.8$  and  $\delta = 0.3$  (bottom right panel). The limiter is indicated by a blue line at the inner midplane.

temperature sources, used to mimic the plasma outflow from the core, are modelled as  $S_{n,T_e} \propto \exp[-(x-a)^2/\sigma^2]$ , with  $\sigma = 2.5$ . Since most of the particles coming from the core are lost at the limiter plates, preventing them from reaching the vessel wall, the conditions applied at  $x = x_o$ , the outer edge of the simulation domain, should not significantly impact turbulence. Therefore, a buffer region is located between  $x = a + 90$  and  $x = x_o$ , and ad hoc boundary conditions (Dirichlet for  $\phi$  and  $\omega$ , and Neumann for  $n$ ,  $v_{\parallel i}$ ,  $v_{\parallel e}$ , and  $T_e$ ) are applied at  $x = x_o$ . On the other hand, at the LCFS, the plasma outflow from the core is mimicked by the source terms. These sources are located at a distance of 50 units from the inner boundary of the computational domain, and the domain between  $x = x_i$  and  $x = a$  is used as buffer region and it is not taken into account for turbulence analysis. Consequently, also at  $x = x_i$ , ad hoc boundary conditions (the same applied at  $x_o$ ) are used, and we verified that their impact on turbulence properties is not significant.

As  $\gamma m_e/m_i < \nu$ , we expect that the unphysical value of the mass ratio does not influence the results [146]. The simulations are carried out with the following numerical parameters:  $N_x = 192$ ,  $N_y = 512$ ,  $N_z = 64$ ,  $\eta_i \approx \eta_e \approx 2$ ,  $\chi_{\parallel, T_e} = 0.2$ , and all the perpendicular diffusion coefficients are in the range 5–10. We remark that the value of the dissipative coefficients does not affect significantly the simulation results (simulations carried out with  $\chi_{\parallel, T_e} = 100$  show a  $\sim 20\%$  steeper  $L_p$ .) In Fig. 7.7 we present typical poloidal



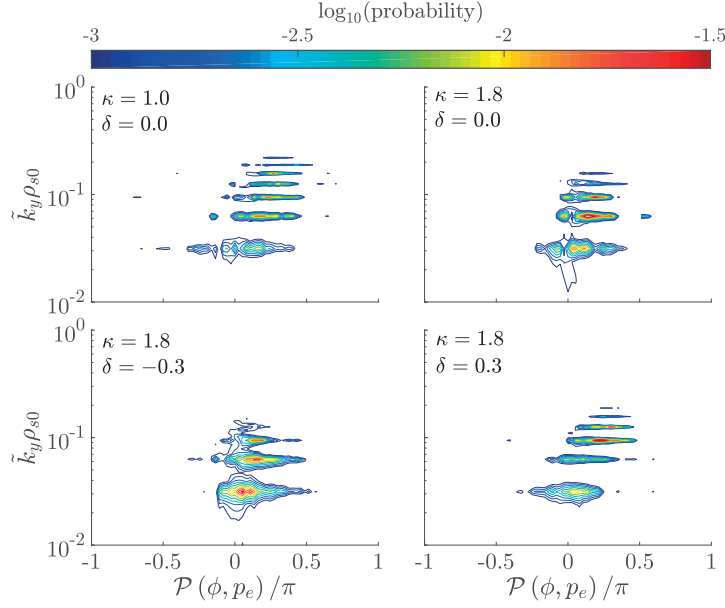


**Figure 7.8** – Poloidal cross sections of the electron pressure plasma profile resulting from nonlinear simulations carried out for  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ , extending from  $x = a$  to  $x = a + 90$ . The four magnetic geometries considered are characterized by  $\kappa = 1$  and  $\delta = 0$  (top left panel),  $\kappa = 1.8$  and  $\delta = 0$  (top right panel),  $\kappa = 1.8$  and  $\delta = -0.3$  (bottom left panel), and  $\kappa = 1.8$  and  $\delta = 0.3$  (bottom right panel). The limiter is indicated by a blue line at the inner midplane.

snapshots of the plasma pressure for  $\epsilon = 0$  and  $\Delta(0) = 0$ . One observes the presence of turbulent eddies that transport plasma radially outward. In the two simulations with  $(\kappa, \delta) = (1.8, 0), (1.8, -0.3)$  plasma turbulence penetrates considerably less in the SOL with respect to a circular magnetic geometry. This is consistent with Section 7.3 results, which show that elongation and negative triangularity decrease  $L_p$ . In these two simulations, turbulence results to be suppressed in particular at the outer midplane, in agreement with the findings in Section 7.2, which shows that RBMs are stabilized by elongation and negative triangularity. On the other hand, plasma turbulence appears to have similar amplitude in a circular magnetic geometry and for  $\kappa = 1.8$  and  $\delta = 0.3$ , in agreement with Section 7.3 results.

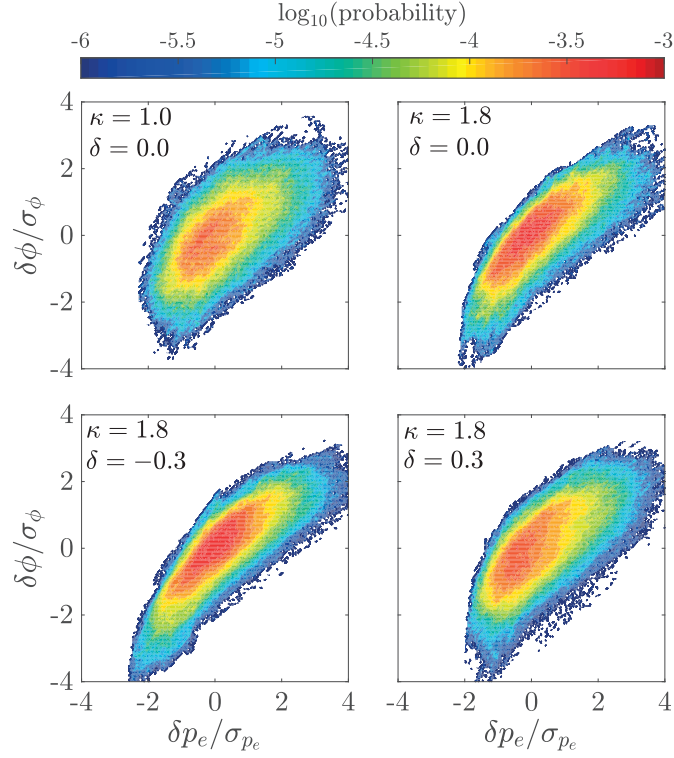
We note that the four simulations carried out with  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$  (see Fig. 7.8) display the same trends, but their differences are less pronounced than in the  $\epsilon = 0$  and  $\Delta(0) = 0$  case. Therefore, in the following we focus our attention on the four simulations with  $\epsilon = 0$  and  $\Delta(0) = 0$ , where the differences are larger, and we briefly discuss the results of the simulations with  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ .

To investigate the nature of the turbulence present in the nonlinear simulations, we apply the same procedure described in Section 6.3.1 to compute the probability distribution function of the phase shift between the electron pressure and the potential fluctuations



**Figure 7.9** – Probability distribution function of the phase shift between the electron pressure and the potential fluctuations resulting from nonlinear simulations carried out for  $\epsilon = 0$  and  $\Delta(0) = 0$ . The four magnetic geometries considered are characterized by  $\kappa = 1$  and  $\delta = 0$  (top left panel),  $\kappa = 1.8$  and  $\delta = 0$  (top right panel),  $\kappa = 1.8$  and  $\delta = -0.3$  (bottom left panel), and  $\kappa = 1.8$  and  $\delta = 0.3$  (bottom right panel).

and the joint probability of  $\delta p_e$  and  $\delta\phi$  normalized to their standard deviation. This is done for  $\theta_* \in [-\pi, \pi]$  and at  $x - a = L_p \ln 2$ , over a time interval of approximately 30 units, for each of the four simulations with  $\epsilon = 0$  and  $\Delta(0) = 0$ . The results thus obtained are presented in Figs. 7.9 and 7.10. In the simulations with  $(\kappa, \delta) = (1, 0), (1.8, 0.3)$  the phase shift is closer to  $\pi/2$  and electron pressure and potential fluctuations are less correlated, with a correlation coefficient of 0.66 and 0.73, respectively, indicating that RBMs contribute to the SOL dynamics. On the other hand, for  $\kappa = 1.8$  and  $\delta = -0.3$ , the phase shift is close to 0 and the electron pressure and potential fluctuations are more correlated, with a correlation coefficient of 0.84, suggesting that RDWs are the turbulence drive. For  $\kappa = 1.8$  and  $\delta = 0$  it is not possible to clearly discriminate between RBMs and RDWs, since the electron pressure and potential fluctuations have a correlation coefficient of 0.82, similar to the simulation performed for  $\kappa = 1.8$  and  $\delta = -0.3$ , but the phase shift is larger than 0. We note that, performing the same analysis for the four simulations with  $\epsilon \simeq 0.25$  and  $\Delta(0) \simeq 7$ , we find that RDWs drive turbulence except for  $\kappa = 1$  and  $\delta = 0$ , where a combination of RBMs and RDWs is responsible of the turbulent dynamics. These results are in agreement with Sections 7.2 and 7.3 findings. Finally, to perform a quantitative comparison between the estimate of  $L_p$  given by Eq. (7.41), the analytical scaling in Eq. (7.44), and the nonlinear simulation results, we time-average the pressure profiles provided by GBS over an interval of approximately 30 time units, and over four time subdomains, each of about 7.5 time units. The results are then averaged over the toroidal angle and the averages fitted at the outer midplane assuming  $p_e(x) \propto \exp[-(x - a)/L_p]$ . The fit over the interval of 30 time units provides

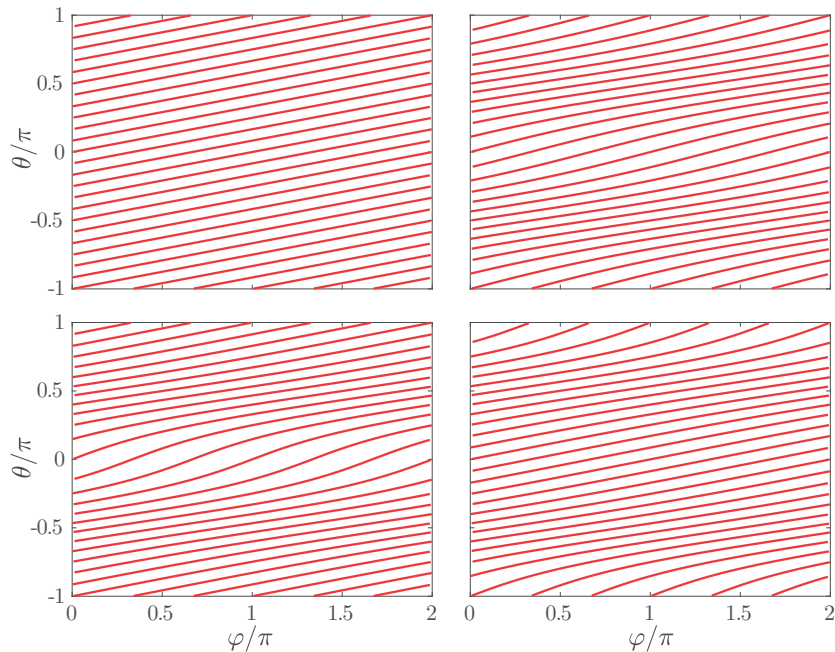


**Figure 7.10** – Joint probability of the electron pressure and the potential fluctuations normalized to their standard deviation resulting from nonlinear simulations carried out for  $\epsilon = 0$  and  $\Delta(0) = 0$ . The four magnetic geometries considered are characterized by  $\kappa = 1$  and  $\delta = 0$  (top left panel),  $\kappa = 1.8$  and  $\delta = 0$  (top right panel),  $\kappa = 1.8$  and  $\delta = -0.3$  (bottom left panel), and  $\kappa = 1.8$  and  $\delta = 0.3$  (bottom right panel).

$L_p$ , while the difference between the  $L_p$ s obtained over the four subdomains gives an estimate of its uncertainty. Applying this methodology, and computing the  $L_p$  given by Eq. (7.41) and Eq. (7.44), we obtain the results listed in Table 7.1. We note that, since Eq. (7.44) gives  $L_p = 23.4$  for  $\kappa = 1.8$  and  $\delta = -0.3$ , we replace this value with  $L_{p,RDW}$  in Table 7.1. Several observations can be made based on these results: (i) Eq. (7.41) is in fairly good agreement with the results obtained from the nonlinear simulations, indicating that the gradient removal theory is able to predict  $L_p$  even when non-circular magnetic geometries are considered; and (ii) if  $\epsilon = 0$  and  $\Delta(0) = 0$ , Eq. (7.44) provides an estimate of  $L_p$  that is in good agreement with Eq. (7.41), meaning that the analytical scaling correctly describes the dependence of  $L_p$  on  $\kappa$  and  $\delta$ . We remark that the results

**Table 7.1** – Values of  $L_p$  obtained from the nonlinear simulations, from Eq. (7.41), and from Eq. (7.44).

$(\kappa, \delta)$	Nonlinear sim.	Eq. (7.41)	Nonlinear sim.	Eq. (7.41)	Eq. (7.44)
	$\epsilon \simeq 0.25, \Delta(0) \simeq 7$	$\epsilon \simeq 0.25, \Delta(0) \simeq 7$	$\epsilon = 0, \Delta(0) = 0$	$\epsilon = 0, \Delta(0) = 0$	
(1.0, 0.0)	$25 \pm 1$	27.4	$37 \pm 2$	38.9	37.1
(1.8, 0.0)	$20 \pm 1$	20.7	$26 \pm 3$	30.3	28.8
(1.8, -0.3)	$15 \pm 1$	18.1	$20 \pm 1$	26.2	27.7
(1.8, 0.3)	$23 \pm 1$	26.8	$43 \pm 3$	36.8	34.1



**Figure 7.11** – Magnetic field lines in the  $\varphi - \theta$  plane, at  $x = a$ , for  $\epsilon = 0$  and  $\Delta(0) = 0$ . The four magnetic geometries considered are characterized by  $\kappa = 1$  and  $\delta = 0$  (top left panel),  $\kappa = 1.8$  and  $\delta = 0$  (top right panel),  $\kappa = 1.8$  and  $\delta = -0.3$  (bottom left panel), and  $\kappa = 1.8$  and  $\delta = 0.3$  (bottom right panel).

for the nonlinear simulations in Table 7.1 are obtained at the outer midplane. However, we note that similar results are obtained also when considering the poloidally averaged plasma pressure profile.

## 7.5 Discussion of the theoretical results

To intuitively explain the impact of plasma shaping on SOL turbulence, we represent in Fig. 7.11 the magnetic field lines that characterize the four simulations with  $\epsilon = 0$  and  $\Delta(0) = 0$  in the  $\varphi - \theta$  plane and at fixed radial position  $x = a$ . We observe that for the two simulations with  $(\kappa, \delta) = (1, 0), (1.8, 0.3)$ , magnetic field lines are almost straight and their slope is close to  $1/q = 0.25$  at the outer midplane. On the other hand, the two simulations with  $(\kappa, \delta) = (1.8, 0), (1.8, -0.3)$  are characterized by magnetic field lines that are stretched in the poloidal direction and their slope strongly increases close to  $\theta = 0$ . A detailed investigation of the impact of  $\kappa$  and  $\delta$  on the magnetic field lines indicates that values of  $\kappa > 1$  and  $\delta < 0$  stretch the magnetic field lines in the poloidal direction near  $\theta = 0$ , while they are compressed for large positive triangularities. This suggests that particle trajectories lie longer in the bad curvature region in a circular magnetic geometry, or for  $\delta > 0$ , with respect to the case of  $\kappa > 1$  or  $\delta < 0$ . Since BMs are strongly destabilized at the outer midplane, i.e. in the proximity of  $\theta = 0$ , elongation and negative triangularity result in a stabilization of BMs, as observed in the

simulation results previously discussed. The same argument can be used to explain the impact of  $\Delta$  and  $\epsilon$ . As a matter of fact, magnetic field lines are even more stretched in the proximity of  $\theta = 0$  when Shafranov's shift and finite aspect ratio effects are included. We remark that in Sections 7.1.3 we computed the Shafranov's shift neglecting the plasma pressure contribution. Since  $\Delta(0)$  usually increases almost linearly with  $\beta_p$ , we expect that including the plasma pressure contribution in the magnetic equilibrium stabilizes the RBMs and decreases  $L_p$ .

We would like to point out that plasma shaping seems to have a similar impact both on core and SOL turbulence. In fact, Refs. [169–172] pointed out that core turbulence is (i) usually stabilised by elongation, except for large positive triangularities, for which it is enhanced, (ii) stabilised by negative triangularity, and (iii) destabilised by large positive triangularity.

## 7.6 Rigorous validation against TCV measurements

In Section 7.1 we discuss an analytical equilibrium model to account for plasma shaping effects, which we use to investigate theoretically the impact of  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta$  on the SOL turbulence dynamics. While our theoretical results are in qualitative agreement with previous experimental investigations (Ref. [174] shows that  $L_p$  decreases with  $\kappa$ ), an accurate quantitative comparison between simulations and experiments is needed in order to assess the reliability of our theoretical results and confirm that the implementation of a non-circular magnetic geometry in Eqs. (7.1)-(7.6) actually improves our modeling capabilities of the SOL plasma dynamics. An experimental campaign was carried out on TCV [175] for this purpose, providing experimental measurements that can be compared with GBS simulation results by employing a rigorous validation methodology. This comparison allows us to rigorously quantify the improvement of the agreement between GBS simulations and non-circular TCV discharges when plasma shaping effects are taken into account according to Eqs. (7.1)-(7.6).

In the present section we first discuss how to perform a rigorous validation of a physical model against experimental measurements. Then, considering three TCV plasma discharges with different equilibrium magnetic shapes, we validate against experimental measurements (i) the three-dimensional GBS model in Eqs. (7.1)-(7.6), (ii) the three-dimensional GBS model in Eqs. (7.1)-(7.6) in circular magnetic geometry and in the infinite aspect ratio limit, and (iii) the two-dimensional version of the GBS model illustrated in Chapter 4. Note that the results discussed in the following of the present section are in SI units.

### 7.6.1 Validation methodology

The methodology to perform a rigorous validation of the simulation results is discussed in Refs. [39–41] and was applied for the first time to TORPEX simulations [104]. Herein we briefly present the key elements of the methodology.

Simulations and experiments have to be compared considering a number of physical quantities, common to the experimental measurements and simulation results, and analysed using the same techniques. These physical quantities are denoted as validation observables. In order for an observable to be considered for the validation, it should satisfy the following criteria. First, the observable should be physically relevant. This means that focus should be put on observables containing the most important theoretical predictions and being more sensitive to the model assumptions. Second, each observable should be independent of the other observables. Third, if an observable depends on space or time, its resolution should be sufficient to describe its variation along all dimensions. Once the observables are defined and evaluated, the agreement between experiments and simulations relative to each observable has to be quantified. We denote with  $e_j$  and  $s_j$  the values of the  $j$ -th observable used in the comparison, as coming from the experimental measurement and the simulation results, respectively. Most of the observables depend on space and time, and are typically given on a discrete number of points, denoted as  $N_j$ . We denote with  $e_{j,i}$  and  $s_{j,i}$  the values of the  $j$ -th observable at points  $i = 1, 2, \dots, N_j$  (the present notation can therefore be used for zero-, one-, two-, etc., dimensional observables). For the  $j$ -th observable, we normalize the distance  $d_j$  between experiments and simulations with respect to the uncertainty related to these quantities as

$$d_j = \sqrt{\frac{1}{N_j} \sum_{i=1}^{N_j} \frac{(e_{j,i} - s_{j,i})^2}{\Delta e_{j,i}^2 + \Delta s_{j,i}^2}}, \quad (7.46)$$

where  $\Delta e_{j,i}$  and  $\Delta s_{j,i}$  are the uncertainties affecting  $e_{j,i}$  and  $s_{j,i}$ . Since simulations and experiments can be considered to agree if their difference is smaller than their uncertainties, we define the level of agreement between experiments and simulations with respect to the observable  $j$  as

$$R_j = \frac{\tanh[(d_j - 1/d_j - d_0)/\lambda] + 1}{2} \quad (7.47)$$

with  $R_j \lesssim 0.5$  corresponding to agreement (experiment and simulation results agree within uncertainties), while  $R_j \gtrsim 0.5$  denoting disagreement (experiment and simulation results are outside the uncertainties). Here, we choose  $d_0 = 1$  and  $\lambda = 0.5$ . Our tests show that the conclusions of a validation exercise are not affected by the specific choices of the parameters  $d_0$  and  $\lambda$  if these parameters are within a reasonable range. Some authors prefer to normalize the distance between experimental and simulation results to the actual value of the observables, rather than to their uncertainty [26]. We believe that the normalization to the uncertainty is the most appropriate choice in the present case, as we are interested in understanding if the basic physics mechanisms at play in

---

## 7.6. Rigorous validation against TCV measurements

the system are well captured by the model under consideration. The normalization to the actual value of the observable is instead preferable in the case that the predictive capabilities of the code are tested.

Since the distance between experiments and simulations is normalized to their uncertainties, particular attention should be paid in evaluating  $\Delta e_{j,i}$  and  $\Delta s_{j,i}$ . In the case of the experiments, we can identify three main uncertainty sources. First, a measuring device is typically modelled to evaluate the physical quantities of interest (e.g. from the  $I - V$  curve of a Langmuir probe one can deduce  $n$  and  $T_e$ ). Since experimental measurements typically do not follow perfectly the model predictions, a fit has to be made in order to evaluate the relevant physical parameters, introducing an uncertainty that we denote with  $\Delta e_{j,i}^{fit}$ . Second, a source of uncertainty is due to properties of the measuring device that are often difficult to evaluate accurately (e.g. geometry and surface condition of a Langmuir probe). Thus, measurements should be performed with different tools (e.g. Langmuir probes that differ in dimension, surface condition, and electronics). The quantity  $\Delta e_{j,i}^{prb}$  denotes the uncertainty related to the probe properties. Finally, the plasmas are not perfectly reproducible due to control parameters difficult to set or know precisely (e.g. the vacuum pressure). Experiments should be repeated in order to check the reproducibility of the plasma, while measurements are taken with different measurement devices. The quantity  $\Delta e_{j,i}^{rep}$  is the uncertainty due to the plasma reproducibility, averaged over the different measuring devices. The total experimental uncertainty is given by  $\Delta e_{j,i}^2 = (\Delta e_{j,i}^{fit})^2 + (\Delta e_{j,i}^{prb})^2 + (\Delta e_{j,i}^{rep})^2$ .

As discussed in Chapters 3 and 4, simulations are also affected by uncertainties. These result from two sources: (i) errors due to the numerics  $\Delta s_{j,i}^{num}$  and (ii) uncertainties due to unknown or imprecise input parameters  $\Delta s_{j,i}^{inp}$ . As in the case of the experimental error bars, the two sources of error should be added, such that  $\Delta s_{j,i}^2 = (\Delta s_{j,i}^{num})^2 + (\Delta s_{j,i}^{inp})^2$ . We note that the error bars should not take into account the uncertainties related to model assumptions and/or to combinations of measurements, which are often needed to deduce the validation observables from the simulation results and the raw experimental data [166]. Evaluating rigorously those uncertainties is usually very challenging. The idea is to take them into account approximately by organizing the observables into a hierarchy, which is based on the number of model assumptions and combinations of measurements necessary to obtain each observable and indicates how stringent each observable is for comparison purposes. More specifically, the higher the hierarchy level of an observable is, the lower the importance of the observable in the comparison metric. The overall level of agreement between simulations and experiments can be measured by considering a composite metric, which should take into account the level of agreement of each observable,  $R_j$ , and weight it according to how constraining each observable is for comparison purposes. This means that the hierarchy level of each observable and the level of confidence characterizing the measurement or the simulation of each observable have to be considered. The higher the level in the primacy hierarchy and the larger the error affecting the observable measurement, the smaller the weight of the observable



should be. We thus define the metric  $\chi$  as

$$\chi = \frac{\sum_j R_j H_j S_j}{\sum_j H_j S_j}, \quad (7.48)$$

where  $H_j$  and  $S_j$  are functions defining the weight of each observable according to its hierarchy level and the precision of the measurement, respectively. Thanks to the definition of  $R_j$ ,  $\chi$  is normalized in such a way that perfect agreement is observed for  $\chi = 0$ , while simulation and experiment disagree completely for  $\chi = 1$ .

The definition of  $H_j$  and  $S_j$  is somewhat arbitrary.  $H_j$  should be a decreasing function of the hierarchy level. The definition we adopt is  $H_j = 1/h_j$ , where  $h_j$  is the combined experimental/simulation primacy hierarchy level, which takes into account the number of assumptions or combinations of measurements used in evaluating the observable both from the experiments and from the simulations. In practice, if no assumptions or combinations of measurements are used for obtaining an observable,  $h_j = 1$ , any assumption or combination of measurement leads  $h_j$  to increase of a unity (see Ref. [166] for examples of observables and related  $h_j$  values). The quantity  $S_j$  should be a decreasing function of the experimental and simulation uncertainty, and we define it as

$$S_j = \exp\left(-\frac{\sum_i \Delta e_{j,i} + \sum_i \Delta s_{j,i}}{\sum_i |e_{j,i}| + \sum_i |s_{j,i}|}\right), \quad (7.49)$$

such that  $S_j = 1$  in the case of zero uncertainty.

The validation metric should be complemented by an index,  $Q$ , that assesses the “quality” of the comparison. The idea is that a validation is more reliable with a larger number of independent observables, particularly if they occupy a low level in the primacy hierarchy and the measurement and simulation uncertainties are low. The quality of the comparison  $Q$  can thus be defined as

$$Q = \sum_j H_j S_j. \quad (7.50)$$

## 7.6.2 Experimental setup

The TCV experiment is a tokamak with major and minor radii  $R = 0.88$  m and  $a = 0.25$  m [175]. By being equipped with sixteen independent set of shaping and positioning coils, TCV has unique capabilities of exploring a wide range of magnetic geometries, including negative triangularities, second-order X points, and more exotic plasma shapes [195]. For this reason TCV is an ideal test bed for validating the plasma shaping model discussed in the present chapter against experimental measurements. An experimental campaign was carried out on TCV for this purpose.

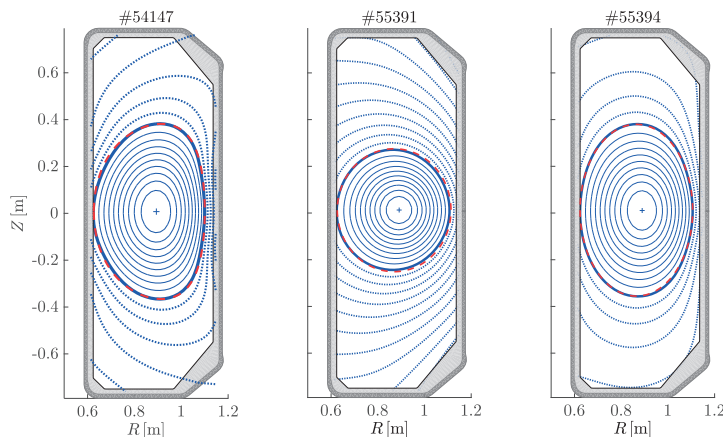
In the following we consider three TCV inner-wall limited plasma discharges: #54147, #55391, and #55394. The experimental parameters measured for the three discharges are summarized in Table 7.2 and the poloidal cross sections of their magnetic surfaces



## 7.6. Rigorous validation against TCV measurements

**Table 7.2** – Parameters for the three TCV plasma discharges #54147, #55391, and #55394 used for the present validation.

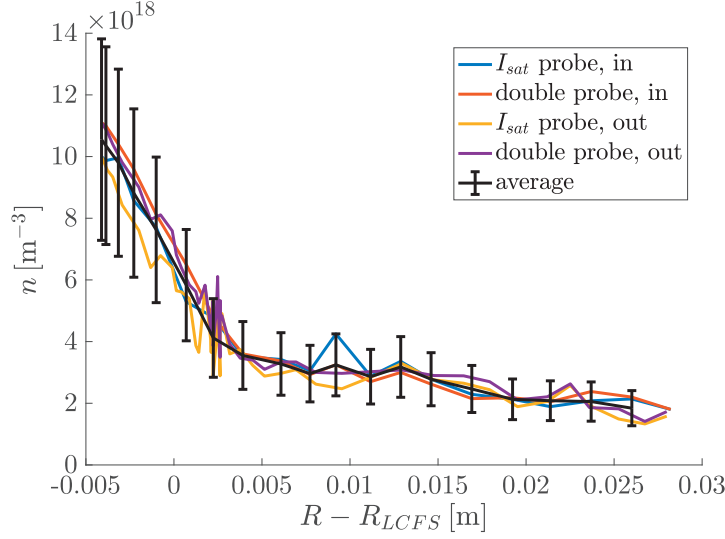
	$q$	$\kappa$	$\delta$	$\epsilon$	$n_0 [10^{18} \text{ m}^{-3}]$	$T_{e0} [\text{eV}]$	$\nu$
#54147	3.3	1.57	-0.19	0.34	$6 \pm 2$	$37 \pm 15$	0.0036
#55391	3.4	1.07	0.00	0.30	$4 \pm 1$	$36 \pm 16$	0.0025
#55394	3.2	1.53	0.00	0.30	$5 \pm 1$	$41 \pm 14$	0.0023



**Figure 7.12** – Poloidal cross sections of the magnetic surfaces for the three TCV plasma discharges #54147 (left panel), #55391 (middle panel), and #55394 (right panel). The blue lines indicate the LIUQE reconstruction, with thin continuous lines denoting the core flux surfaces, thick continuous lines denoting the LCFS, and dashed lines denoting the open field line region. The red dashed lines represent the reconstruction of the LCFS with the model discussed in Section 7.1.

are shown in Fig. 7.12. The plasma has an almost circular magnetic equilibrium in the discharge #55391, an elongated equilibrium in the discharge #55394, and an elongated equilibrium with negative triangularity in the discharge #54147. Comparing the thick blue continuous lines, representing the LIUQE [196] reconstruction of the LCFS, with the red dashed lines, indicating the LCFS obtained by best fitting the LIUQE reconstruction with the model discussed in Section 7.1, we observe that the results are in good agreement. We also see that the three discharges have similar safety factors,  $q$ , densities,  $n_0$ , and temperatures,  $T_{e0}$ , at the LCFS. It results that they have also similar normalized plasma resistivities. This allows to decouple the influence of plasma shaping on the SOL dynamics from other effects, as shown, for example, in Eq. (7.44).

The experimental measurements discussed in the following of the present chapter are obtained with a horizontal reciprocating manipulator located at the tokamak outer mid-plane. This diagnostic provides high spatial resolution measurements of equilibrium and fluctuating physical quantities. It is equipped with a probe head having ten electrostatic pins (two are used as a swept double probe and provide measurements of  $I_{sat}$ ,  $n$ , and  $T_e$ , one is used to obtain direct measurements of  $I_{sat}$ , five are used as floating Langmuir probes to measure  $V_{fl}$ , and the last two are used to measure the poloidal Mach number). Moreover, taking the difference of two of the  $V_{fl}$  pins, it is possible to obtain measurements related to the poloidal electric field  $E_\theta$ . Radial profiles related to these quantities



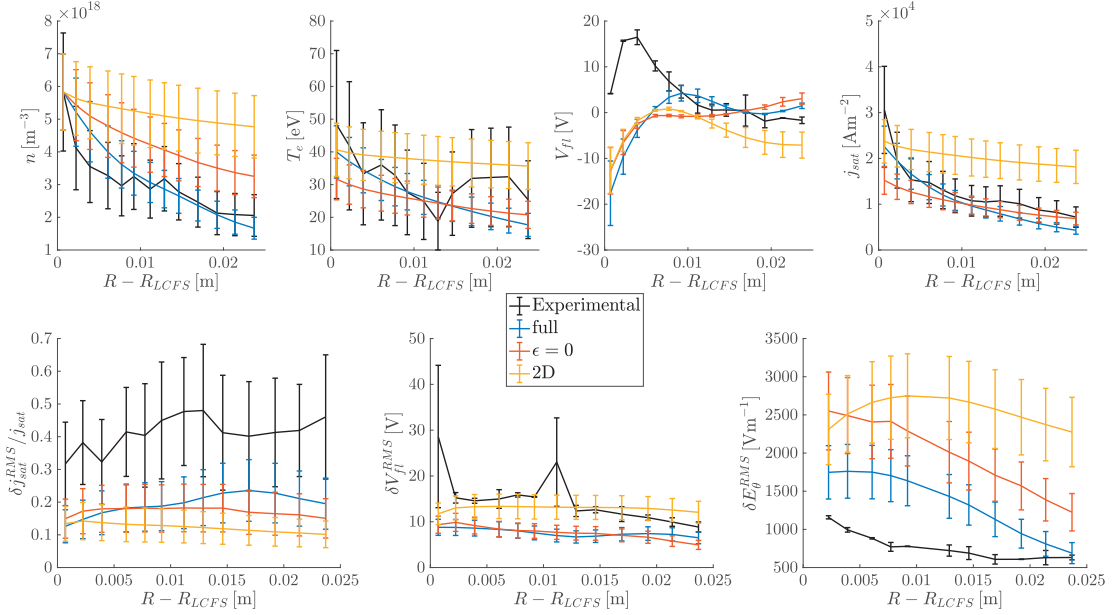
**Figure 7.13** – Plasma density profiles for the TCV discharge #54147. The measurements obtained from the  $I_{sat}$  (blue and yellow lines) and the double (red and purple lines) probes are shown, both for the horizontal manipulator entering (blue and red lines) and exiting (yellow and purple lines) the plasma. The averaged profile and the corresponding uncertainties are represented in black.

are obtained sampling the measured time traces in intervals of 1 ms. A detailed description of the reciprocating probe is presented in Ref. [197].

As an example, the measurements of the time-averaged density profile for the TCV plasma discharge #54147 are shown in Fig. 7.13, where we present the results obtained by using the  $I_{sat}$  (blue and yellow lines) and the double probes (red and purple lines), both for the horizontal manipulator entering (blue and red lines) and exiting (yellow and purple lines) the plasma. Assuming that the plasma equilibrium and fluctuating properties are approximately constant throughout the two probe reciprocations, we take the average between the four profiles as the measure of the plasma density (black line). Moreover, we obtain an estimate of the experimental uncertainty  $\Delta e_{n,i}$  by computing the maximum difference between this averaged profile and the four measured profiles. To ensure a safe estimation of  $\Delta e_{n,i}$ , we take the maximum of the relative uncertainty over the radial position,  $\max_i(\Delta e_{n,i}/e_{n,i})$ , as the relative uncertainty affecting the experimental results. This is represented by the black error bars in Fig. 7.13. Since we use two different probes to evaluate  $n$ , and we repeat twice the measurements, the two uncertainty contributions  $\Delta e_{n,i}^{prb}$  and  $\Delta e_{n,i}^{rep}$  are taken into account by  $\Delta e_{n,i}$ . We note that we neglect  $\Delta e_{n,i}^{fit}$ .

The procedure illustrated to evaluate  $n$  and  $\Delta e_{n,i}$  is also applied to compute the time-averaged  $I_{sat}$  profile and its uncertainty. On the other hand, since the five  $V_{fl}$  pins are not simultaneously located on the same flux surface, it is not justified to average the experimental measurements of the different  $V_{fl}$  probes. Therefore, the time-averaged  $V_{fl}$  profile is evaluated by averaging a single  $V_{fl}$  pin measurement over the data provided by the reciprocating probe entering and exiting the plasma. The uncertainty affecting the  $V_{fl}$  profile are then obtained computing the maximum difference between this averaged profile and the two measurements, which corresponds to neglecting the  $\Delta e_{V_{fl},i}^{prb}$  and

## 7.6. Rigorous validation against TCV measurements

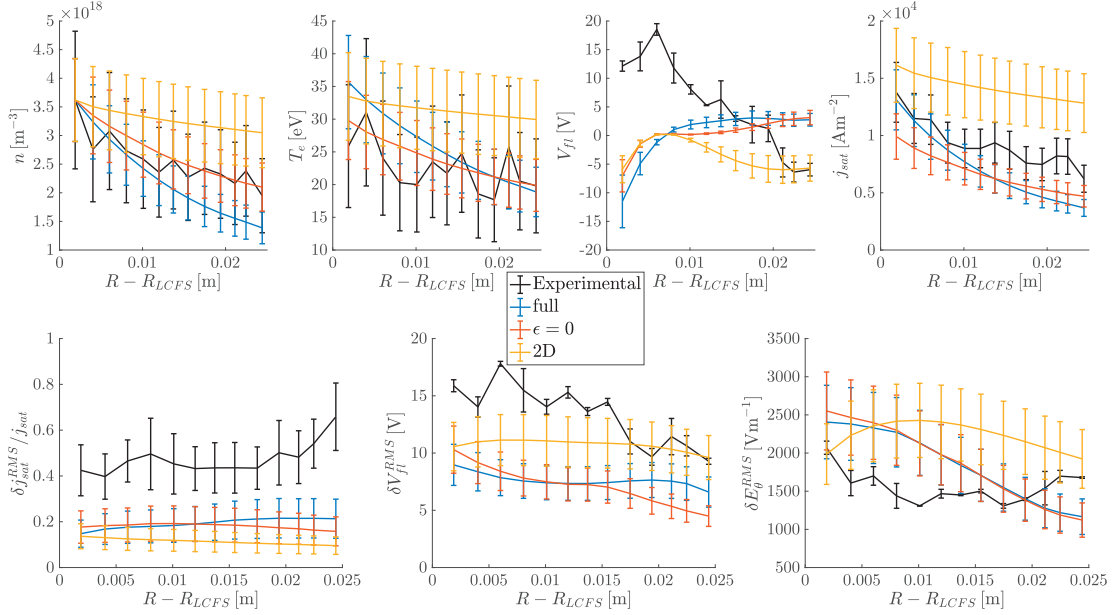


**Figure 7.14** – Radial profiles of time-averaged (first row) and fluctuating quantities (second row) for the TCV discharge #54147. The results displayed are obtained from experimental measurements (black lines), from the two-dimensional model (yellow), from the three-dimensional model and considering circular magnetic surfaces in the infinite aspect ratio limit (red lines), and from the full GBS model (blue lines).

$\Delta e_{V_{r1},i}^{fit}$  contributions.

### 7.6.3 GBS simulations and validation results

We assess the reliability of the model illustrated in Section 7.1 by rigorously validating nonlinear GBS simulations based on the three TCV plasma discharges discussed above against experimental TCV measurements. For each TCV plasma discharge under consideration we perform: (i) a three-dimensional GBS simulation that includes shaping effects, (ii) a three-dimensional GBS simulation considering a circular magnetic geometry in the infinite aspect ratio limit, and (iii) a two-dimensional GBS simulation solving Eqs. (4.51)-(4.53). To carry out the simulations we consider  $\Lambda = 3.2$ , the normalized perpendicular diffusion coefficients  $D_A = 5$ , and we adjust the value of  $n_0$  within the experimental uncertainties reported in Table 7.2 such that the simulated plasma densities agree with the experimental measurements at the LCFS. For the three-dimensional simulations we also consider an ion to electron temperature ratio  $\tau = 1$ . The radial domain extends from the inner radius  $x_i = a - 30$  to the outer radius  $x_o = 70$ . We remark that, since the three-dimensional simulations require three rather large numerical grids  $(N_x, N_y, N_z) = (128, 1280, 196), (128, 1280, 188), (128, 1280, 200)$ , a reduced mass ratio  $m_i/m_e = 800$  and a reduced parallel electron thermal conductivity  $\chi_{\parallel e} = 10$  are used to considerably decrease the computational cost. For the two-dimensional simulations we consider  $\sigma = R/L_{\parallel} \approx 1/(2\pi q)$  and we use the numerical grid  $(N_x, N_y) = (128, 512)$ , with



**Figure 7.15** – Radial profiles of time-averaged (first row) and fluctuating quantities (second row) for the TCV discharge #55391. The results displayed are obtained from experimental measurements (black lines), from the two-dimensional model (yellow), from the three-dimensional model and considering circular magnetic surfaces in the infinite aspect ratio limit (red lines), and from the full GBS model (blue lines).

radial and vertical domains extending from  $x_i = a - 30$  to  $x_o = 70$  and from  $y_0 = 0$  to  $y_1 = 2\pi a/q$ , respectively.

For the three plasma discharges considered here we compute  $L_p$  according to Eq. (7.41), thus obtaining  $L_p \simeq 40, 49, 42 \rho_{s0} \simeq 2.4, 2.9, 2.8$  cm. Moreover, fitting the three-dimensional simulation results at the outer midplane by assuming  $p_e(x) \propto p_{e0} \exp[-(x - a)/L_p]$ , we obtain  $L_p \simeq 18, 23, 19 \rho_{s0} \simeq 1.1, 1.4, 1.2$  cm. The  $L_p$  estimate given by Eq. (7.41), approximately a factor two larger than for the three-dimensional simulations, retrieves the same qualitative behavior given by the nonlinear results, with  $\kappa$  and  $\delta$  stabilizing the SOL plasma turbulence. Performing the same analysis, but neglecting plasma shaping effects, we obtain  $L_p \simeq 49, 51, 47 \rho_{s0} \simeq 3.0, 3.0, 3.0$  cm and  $L_p \simeq 41, 39, 45 \rho_{s0} \simeq 2.5, 2.3, 2.8$  cm, for the nonlinear simulations and the solution of Eq. (7.41), respectively. As expected, the value of  $L_p$  increases when  $\epsilon$  effects are neglected. However, plasma shaping effects in nonlinear simulations have a stronger impact on  $L_p$  than for the quasi-linear theory. Following the procedure described in Sections 6.3.2 and 7.3, we also investigate linearly the regime of the instability that drives most of the SOL turbulent transport in TCV. It results that TCV SOL turbulence is driven by InDWs. This is in agreement with the results in Ref. [145].

To validate the GBS model against experimental measurements, we consider the following observables: (i) the radial time-averaged profiles of  $n$ ,  $T_e$ ,  $V_{fl}$ , and  $j_{sat}$ , (ii) the radial profile of  $\delta j_{sat}^{RMS}/j_{sat}$ , (iii) the radial profiles of  $\delta V_{fl}^{RMS}$  and  $\delta E_{\theta}^{RMS}$ , and (iv) the equilibrium density gradient length  $L_n = -\langle n \rangle_t / \partial_x \langle n \rangle_t$ . The time-averaged and fluctuation radial profiles are used as validation observables since they are at the lowest available

## 7.6. Rigorous validation against TCV measurements

**Table 7.3** – Values of  $L_n$  [cm] obtained from the experimental measurements, from the three-dimensional GBS simulations with and without shaping effects (Full and  $\epsilon = 0$  columns, respectively), and from the two-dimensional simulations.

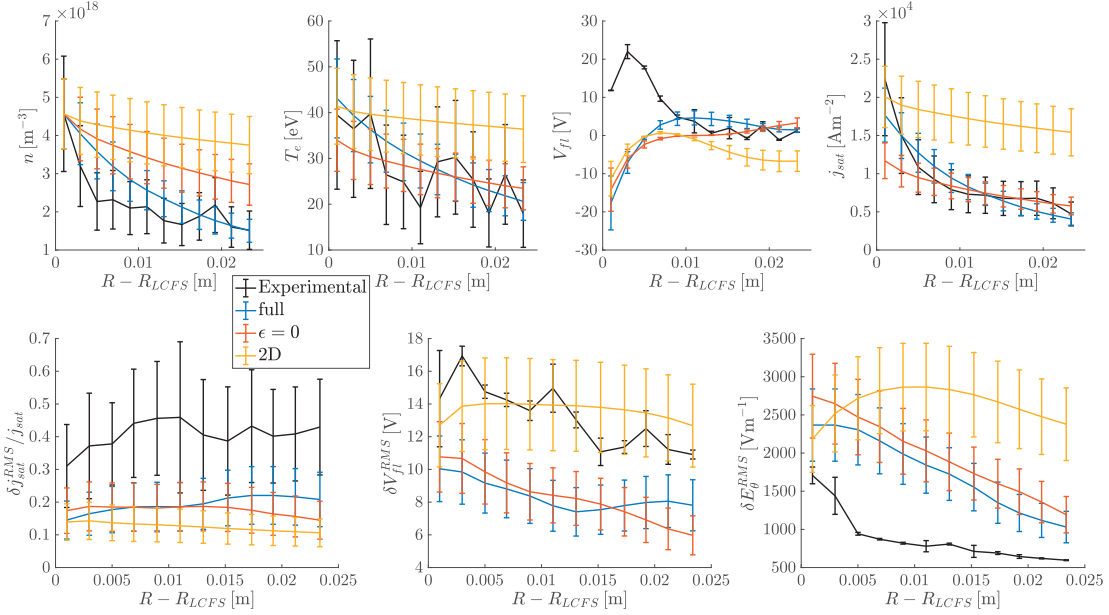
	Experimental	Full	$\epsilon = 0$	Two-dimensional
54147	$1.8 \pm 0.6$	$1.8 \pm 0.1$	$3.7 \pm 0.3$	$10 \pm 2$
55391	$4.0 \pm 1.0$	$2.5 \pm 0.3$	$4.3 \pm 0.3$	$13 \pm 3$
55394	$1.5 \pm 0.6$	$2.0 \pm 0.1$	$4.1 \pm 0.4$	$11 \pm 1$

combined primacy hierarchy level, while  $L_n$  is used because of its importance in SOL turbulence analysis (here we use  $L_n$  instead than  $L_p$  because of the large experimental uncertainties on  $T_e$ ).

Concerning the experimental equilibrium profiles and their uncertainties,  $n$ ,  $I_{sat}$ , and  $V_{fl}$  results are obtained as discussed in Section 7.6.2. The same procedure used for  $V_{fl}$  measurements is also employed to obtain experimental quantities related to  $\delta j_{sat}^{RMS}$ ,  $\delta V_{fl}^{RMS}$ , and  $\delta E_{\theta}^{RMS}$ . Finally,  $L_n$  is estimated by fitting the time-averaged experimental density profile assuming  $n(R) \propto n_0 \exp[-(R - R_{LCFS})/L_n]$ , with its uncertainty provided by the 95% confidence interval of the exponential fit. This corresponds to neglecting the two contributions  $\Delta e_{L_n}^{prb}$  and  $\Delta e_{L_n}^{rep}$ , since they are expected to be considerably smaller than  $\Delta e_{L_n}^{fit}$ .

Concerning the GBS simulations, the  $j_{sat}$  and  $V_{fl}$  profiles are obtained assuming  $j_{sat} = enc_s/2$  and  $V_{fl} = \phi - [\Lambda - \ln(1 + T_i/T_e)/2]T_e/e$ . To obtain the radial profiles, we toroidally and time-average the numerical results at the outer midplane for the three-dimensional simulations, while we perform an average of the numerical results along  $y$  and  $t$  for the two-dimensional simulations. Moreover, we assume  $\Delta s_{j,i}^{num} = 0.2s_{j,i}$ , as discussed in Chapter 3, for all the time-averaged quantities, except for  $V_{fl}$  that is typically affected by a larger numerical error (we assume  $\Delta s_{V_{fl},i} = 0.4s_{V_{fl},i}$ ). For the input parameter uncertainty propagation, we note that typical GBS simulations indicate variations of the time-averaged radial profiles smaller than 10% due to a tenfold increase of  $\nu$ . Therefore, for the time-averaged radial profiles we neglect the  $\Delta s_{j,i}^{imp}$  contribution. Moreover, typical GBS simulations also indicate that the numerical error affecting  $\delta j_{sat}^{RMS}$ ,  $\delta V_{fl}^{RMS}$ , and  $\delta E_{\theta}^{RMS}$  is approximately 10% – 15%, and that the fluctuation level increases approximately by 15% when increasing  $\nu$  by a factor of ten. Therefore, for  $\delta j_{sat}^{RMS}$ ,  $\delta V_{fl}^{RMS}$ , and  $\delta E_{\theta}^{RMS}$  we assume  $\Delta s_{j,i} = [(\Delta s_{j,i}^{num})^2 + (\Delta s_{j,i}^{imp})^2]^{1/2} \approx 0.2s_{j,i}$ . Finally, we note that the same procedure used to evaluate  $L_n$  from the time-averaged experimental density profile is employed for the simulations.

The resulting radial profiles, together with the corresponding uncertainties, are displayed for the three TCV discharges #54147, #55391, and #55394 in Figs. 7.14, 7.15, and 7.16, respectively, while the equilibrium density gradient lengths are presented in Table 7.3. The plasma shaping has a stronger impact on the simulation results for the two plasma discharges #54147 and #55394. In these cases, we see that the radial time-averaged profiles of  $n$ ,  $T_e$  and  $j_{sat}$ , simulated with the full GBS model, present a better agreement with the experimental measurements with respect to those obtained with the circular and two-dimensional models. The same conclusions are obtained for the results presented



**Figure 7.16** – Radial profiles of time-averaged (first row) and fluctuating quantities (second row) for the TCV discharge #55394. The results displayed are obtained from experimental measurements (black lines), from the two-dimensional model (yellow), from the three-dimensional model and considering circular magnetic surfaces in the infinite aspect ratio limit (red lines), and from the full GBS model (blue lines).

in Table 7.3. For the TCV discharge #55391, the radial time-averaged profiles of  $n$ ,  $T_e$  and  $j_{sat}$  simulated with the three-dimensional models are quite similar, and in better agreement with the experimental measurements than the results obtained with the two-dimensional model. Concerning the radial profiles of  $V_{fl}$ ,  $\delta j_{sat}^{RMS}/j_{sat}$ , and  $\delta E_{\theta}^{RMS}$ , we see that none of the models is able to correctly predict the experimental measurements. This might be due to simulating only the open field line region, as discussed in Chapter 6. Finally, we note that the two-dimensional model is in better agreement with the  $\delta V_{fl}^{RMS}$  experimental measurements than the three-dimensional models.

To assess quantitatively the agreement between the three models with experimental measurements, we apply the methodology discussed in Section 7.6.1. This is done by assuming  $h_j = 2$  for all the observables considered herein [166], except for  $\delta E_{\theta}^{RMS}$  and  $L_n$ , for which we assume  $h_j = 3$ . The resulting validation parameters  $\chi$  and  $Q$  are illustrated in Table 7.4. Concerning the full GBS model, we see that  $\chi \leq 0.4$  for all the considered TCV discharges, while the agreement between experimental measurements and simulations decreases by assuming a circular magnetic geometry in the infinite aspect ratio limit for the two TCV discharges #54147 and #55394. This indicates that the shaping model implemented in GBS significantly improves the description of SOL plasma turbulence taking into account the impact of elongation and triangularity. Moreover, considering the two-dimensional model, we observe that  $\chi \geq 0.6$  for all the three plasma discharges. This indicates that the hypothesis  $k_{\parallel} = 0$  is not justified in the present experimental scenario. This result is consistent with the fact that, for the three considered discharges, the SOL turbulent transport is mostly driven by DWs. Finally, we note that  $Q \approx 2.5$



**Table 7.4** – Validation metric,  $\chi$ , and validation quality,  $Q$ , computed according to Eqs. (7.48) and (7.50), respectively, for the validation of the three-dimensional GBS simulations with and without shaping effects (Full and  $\epsilon = 0$  columns, respectively), and the two-dimensional simulations, against experimental measurements.

	Full		$\epsilon = 0$		Two-dimensional	
	$\chi$	$Q$	$\chi$	$Q$	$\chi$	$Q$
54147	0.34	2.46	0.46	2.46	0.76	2.54
55391	0.40	2.51	0.39	2.50	0.62	2.60
55394	0.34	2.46	0.51	2.49	0.85	2.56

for all the comparisons considered herein.

## 7.7 Conclusions

In the present chapter, the effects of plasma shaping on SOL turbulence are discussed. Depending on the magnetic geometry, the SOL turbulence regime is identified, and the impact of Shafranov’s shift, finite aspect ratio, elongation, and triangularity on  $L_p$  is investigated. The results obtained from the linear theory are compared with nonlinear simulations, which are rigorously validated against TCV experimental measurements.

The drift-reduced Braginskii equations are written in an arbitrary magnetic geometry and, using the flux-tube and the toric coordinate systems, the coefficients characterizing the differential operators entering the model equations are computed. An analytical equilibrium model is used to express the dependence of the magnetic equilibrium on  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta$ . This model allows us to isolate the different shaping effects that affect SOL turbulence and to investigate their impact on SOL instabilities.

First, the influence of the plasma shaping on RBMs and RDWs and the effect of  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta$  on their growth rate are discussed. It turns out that plasma shaping strongly impacts the RBM growth rate, while RDWs are considerably less affected. In particular, it is observed that  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta < 0$  effects stabilize RBMs, while these are enhanced for  $\delta > 0$ .

Second, a nonlinear saturation theory of the growth of the unstable linear modes is presented, and it is used to estimate the SOL width from a linear analysis of the main instability driving the SOL turbulence. It is found that  $L_p$  decreases for  $\kappa > 1$  and  $\delta < 0$ , while it increases for  $\delta > 0.2$ . This result is used to identify the turbulent regime depending on the shaping parameters. It turns out that, for  $\kappa > 1$  and  $\delta < 0$ , RDWs drive the SOL dynamics, while RBMs dominate for large positive  $\delta$  values, or if  $\kappa = 1$  and aspect ratio effects are neglected. Moreover, assuming that RBMs are mainly affected by plasma shaping through the curvature operator, an analytical scaling for  $L_p$  is deduced, taking into account  $\kappa$  and  $\delta$  effects. This analytical scaling generalizes the scaling presented in Refs. [24, 100, 113] by including non-circular magnetic geometries.

Third, the results obtained from the linear theory are compared with nonlinear simu-

lations carried out with the flux-driven fluid code GBS, showing good qualitative and quantitative agreement. An intuitive explanation for the strong impact of  $\kappa$  and  $\delta$  on the RBMs growth rate is also given, showing that elongation and negative triangularity result in a stabilization of RBMs.

Finally, in order to assess the reliability of our shaping model, we compare nonlinear simulations of SOL plasma turbulence with TCV experimental measurements. We consider: (i) three-dimensional GBS simulations carried out with  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta$  effects, (ii) the three-dimensional GBS model with circular magnetic geometry in the infinite aspect ratio limit, and (iii) the two-dimensional GBS model. A rigorous validation methodology is applied to assess the global agreement between the three simulation models and the experimental measurements. It is found that the model used to express the dependence of the magnetic equilibrium on  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta$  is able to significantly improve the agreement of simulation results with the experimental measurements. Moreover, confirming the DW nature of the instability driving the SOL transport in the considered TCV discharges, we find that the hypothesis  $k_{\parallel} = 0$  is not justified in the present experimental scenario, also in agreement with previous theoretical investigations [145].



# CHAPTER 8

## Conclusions and outlook

The present thesis is devoted to the development and the application of V&V procedures to numerical simulations that are used to investigate the plasma dynamics in basic plasma physics experiments and in the tokamak SOL. While these procedures allow us to increase the reliability of numerical simulations, ultimately they also lead to an improvement of our understanding of the phenomena at play in the SOL region. Developing and applying rigorous V&V methodologies is now crucial in the fusion community. In fact, with the design and the construction of increasingly larger experiments, errors affecting simulation codes that are used to predict the performances of future fusion devices can have far reaching consequences.

The general methodology to perform a rigorous plasma simulation code verification is presented in Chapter 2. First, focusing on grid-based simulation codes, we propose to use the method of manufactured solutions to assess the correct implementation of a physical model in a simulation code. The procedure requires choosing a manufactured solution that satisfies some reasonable conditions, adding the resulting source terms to the model equations, and performing a number of simulations to verify that the numerical solution converges to the manufactured one at the rate expected for the considered algorithm. If it is the case, the code is verified. The methodology is then generalized to PIC codes, accounting for the intrinsic statistical nature of PIC simulations and providing a measure between continuous analytical distribution functions and finite samples of computational particles. Since the proposed methodology might become very demanding in terms of computational resources, several norms of the numerical error are investigated, providing a procedure that is easily generalized to a six-dimensional phase-space with a limited computational cost. The proposed code verification methodology is then successfully applied to verify GBS and a one-dimensional, electrostatic, collisionless PIC code, considerably increasing the reliability of the numerical results obtained with these two simulation codes.

In Chapter 3 we present a solution verification procedure to rigorously estimate the statistical and discretization errors affecting plasma simulations. The statistical uncertainty

is quantified by repeating the simulation with different pseudorandom number generator seeds, while the discretization error is evaluated by approximating the exact solution with the Richardson extrapolation. Since simulations belonging to the asymptotic regime are required to compute the Richardson extrapolation, condition which might require extremely expensive simulations, the GCI is also introduced for estimating the discretization uncertainty. The total numerical error affecting the simulation results is then computed by summing up the different error contributions. The proposed methodology is applied to quantify the numerical error affecting GBS SOL turbulence simulation results and the two-stream instability growth rate evaluated from a PIC simulation. We remark that estimating the numerical error affecting a simulation is important not only to ensure their reliability, but also to perform a rigorous validation of the considered physical models against experimental data.

The impact of input parameter variations on the results of a plasma turbulence model is investigated in Chapter 4, providing a framework for studying uncertainty propagation and avoiding the use of a large number of expensive simulations. To reduce the computational cost of the uncertainty propagation analysis, a general and simple to apply methodology to approximate the model equation solution with a semi-analytic expression that depends explicitly on time, spatial coordinates, and input parameters is proposed. The procedure is based on decomposing the model equation solution in terms of Chebyshev polynomials. By employing a WRM, a set of nonlinear algebraic equations for the coefficients appearing in the Chebyshev decomposition is then obtained. The methodology is applied to a two-dimensional Braginskii model to study the impact of the input parameter that describes the parallel losses on the simulation results. Assuming that this input parameter is distributed according to a Gaussian probability density function, the standard deviation that characterizes the corresponding distribution of equilibrium density gradient lengths is evaluated. It results that a reasonable estimate of the standard deviation can be obtained with a small number of Chebyshev polynomials, i.e. with reduced-cost simulations.

The blob dynamics in the TORPEX device is investigated in Chapter 5. Five different models are used to carry out seeded blob simulations, which are compared with each other and validated against experimental measurements. The models differ because of a number of assumptions used to simplify the drift-reduced Braginskii equations, such as the hypothesis of cold ions, isothermal electrons, or negligible electron inertia. Both two-dimensional and three-dimensional models are considered. The comparison between numerical and experimental results allows us to identify the most important physics elements that play a role in setting the blob velocity. It results that parallel currents are important in setting the radial velocity of blobs in typical TORPEX hydrogen plasmas, in agreement with previous experimental observations, while their vertical motion is mostly governed by the vertical background  $\mathbf{E} \times \mathbf{B}$  flow.

In Chapter 6 we investigate the RFX-mod SOL plasma dynamics considering low edge safety factors. First, we identify the inertial DW as the instability driving most of the turbulence transport in the considered RFX-mod plasma discharges, both from nonlinear results and from a linear investigation of SOL instabilities. Then, we compare the

---

nonlinear simulations with RFX-mod experimental measurements, showing qualitatively and quantitatively good agreement, except for the  $V_{fi}$  equilibrium radial profile and the level of  $j_{sat}$  fluctuations. The observed discrepancies are attributed to events occurring inside the LCFS, which are not correctly described by GBS simulations.

Finally, in Chapter 7 we investigate the impact of shaping effects on the SOL plasma dynamics. The drift-reduced Braginskii equations are written in arbitrary geometry, and an analytical model is employed to express the dependence of the magnetic equilibrium on aspect ratio ( $\epsilon$ ), Shafranov's shift ( $\Delta$ ), elongation ( $\kappa$ ), and triangularity ( $\delta$ ). The influence of plasma shaping effects on the growth rate of RBMs and RDWs is analysed. It results that  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta < 0$  strongly stabilizes RBMs, while RDWs are considerably less affected. Assuming that the linear growth rate of SOL instabilities saturates because of the gradient removal mechanism, the impact of shaping effects on the equilibrium pressure scale length is investigated. It is found that  $L_p$  decreases for  $\kappa > 1$  and  $\delta < 0$ , while it increases for  $\delta > 0$ . The nature of the SOL turbulent regime, depending on the considered magnetic geometry, is also identified. It turns out that RDWs drive the SOL plasma dynamics for  $\kappa > 1$  and  $\delta < 0$ , while RBMs dominate for large positive values of  $\delta$ . An analytical scaling for  $L_p$ , which depends on  $\kappa$  and  $\delta$ , is also deduced. These linear results are then compared with nonlinear simulations, finding good agreement. Finally, the simulation results are validated against TCV experimental measurements, considering three plasma discharges with different magnetic geometries. For this purpose, three-dimensional GBS simulations with shaping effects, three-dimensional GBS simulations in circular geometry and in the infinite aspect ratio limit, and two-dimensional GBS simulations are considered. It is found that the model used to express the dependence of the equilibrium magnetic geometry on  $\epsilon$ ,  $\Delta$ ,  $\kappa$ , and  $\delta$  significantly improves the agreement of the simulation results with the experimental measurements. The DW nature of the instability driving the TCV SOL turbulence dynamics is also confirmed.

While in the present thesis rigorous methodologies to effectively increase the reliability of the numerical tools used to investigate the tokamak plasma dynamics are proposed, a number of issues remains open. First, since the manufactured solutions should be smooth, we note that the proposed code verification methodology cannot be applied to numerical codes that are used to simulate shocks or discontinuities. Techniques to rigorously verify these codes are still under debate. Using the MMS to verify finite element and fully spectral codes constitutes also an open issue, since adding the source terms to the model equations modifies the behaviour of the numerical error affecting the simulations. In addition, the requirement that the degree of mesh refinement is solely represented by the parameter  $h$  prevents the use of the proposed methodology to the verification of simulation codes involving local mesh refinement or mesh adaptation. Moreover, we note that the verification of numerical code packages, for which different simulation codes are coupled together, also represents an open issue. In fact, while the methodology presented in the present thesis can be used to verify independently each block of the package, it is not straightforward to apply it to rigorously verify the interfaces among the different codes. In general, additional work is needed to develop a

rigorous methodology for the verification of these simulation packages.

Concerning the solution verification procedure presented in this thesis, we note that the proposed methodology based on the Richardson extrapolation requires simulations in the asymptotic regime, which might be very expensive. This can represent a serious problem for the application of such a methodology. At the same time, while the GCI allows to evaluate the discretization error also for less expensive simulations, a discussion of its generality and its reliability is still ongoing. Therefore, progress is still necessary to develop a reliable solution verification methodology based on less expensive simulations. The study of uncertainty propagation through a physical model is a general and extremely interesting problem, which is being investigated in a number of scientific domains. In Chapter 4 we propose a rigorous methodology to carry out uncertainty propagation studies. However, the proposed approach requires the development of new fully spectral codes. In general, the evaluation of uncertainties affecting simulation results due to uncertainties on input parameters remains an outstanding scientific problem and a number of techniques might be explored and applied to plasma physics simulations (see e.g. Ref. [198] for a more complete discussion of this subject).

In the present thesis we discuss the validation of plasma simulations against experimental measurements, reporting on three validation exercises. More precisely, we investigate SOL phenomena in a basic plasma physics experiments and in two tokamaks, improving our understanding of the SOL plasma dynamics. Our work shows that, to improve the reliability of the physical models used to investigate the plasma dynamics, it is important to design and execute dedicated validation experiments that allow testing individually the different phenomena at play in the system (an example is the blob motion in TORPEX). Consequently, it would be useful to develop a database of experimental measurements, accessible by the whole plasma physics community, containing the information necessary to carry out rigorous validation exercises. Since the physical quantities obtained from simulations typically do not coincide with the physical quantities measured in experiments, additional work in developing synthetic diagnostics is required. At the same time, an effort should be made to provide accurate measurements of observables. This will decrease the uncertainties, thus increasing the quality factor of the validation.

The final target of V&V procedures is to increase the reliability of simulations in describing the tokamak plasma dynamics. This ultimately leads to an improvement of our understanding of plasma physics phenomena and, at the same time, allows more reliable predictions of the performances of future fusion devices. For this purpose, a rigorous methodology to assess the predictive capabilities of plasma simulation codes should be developed. This requires, for example, estimating the uncertainties on the model results introduced by the model assumptions.

Based on the work in the present thesis it is our hope that all numerical tools used to investigate the tokamak plasma dynamics and design fusion devices will be rigorously verified, and the simulation results rigorously validated against experimental measurements.

# APPENDIX A

## The SOL model and the GBS code

In the tokamak SOL the plasma is relatively cold, with temperatures usually lower than 100 eV, and the mean free path of electrons and ions between two Coulomb collisions is typically shorter than the parallel connection length. It follows that the SOL plasma is characterized by a relatively high collisionality, which allows the plasma to reach a local thermodynamic equilibrium. Consequently, it is usually justified to model the SOL plasma dynamics with only a few fluid moments.

The three lowest-order fluid moments of the  $\alpha$  species distribution function  $f_\alpha$  are the plasma density  $n_\alpha$ , fluid velocity  $\mathbf{v}_\alpha$ , and temperature  $T_\alpha$ , which are defined as

$$n_\alpha(\mathbf{x}, t) = \int_{\mathbb{R}^3} f_\alpha(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}, \quad (\text{A.1})$$

$$\mathbf{v}_\alpha(\mathbf{x}, t) = \frac{1}{n_\alpha} \int_{\mathbb{R}^3} \mathbf{v} f_\alpha(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}, \quad (\text{A.2})$$

$$T_\alpha(\mathbf{x}, t) = \frac{1}{n_\alpha} \int_{\mathbb{R}^3} \frac{m_\alpha}{3} (\mathbf{v} - \mathbf{v}_\alpha)^2 f_\alpha(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}. \quad (\text{A.3})$$

In order to express the evolution of the fluid moments, one has to start from the Boltzmann equation, which describes the time evolution of  $f_\alpha$  as

$$\frac{\partial f_\alpha}{\partial t} + \mathbf{v} \cdot \frac{\partial f_\alpha}{\partial \mathbf{x}} + \frac{q_\alpha}{m_\alpha} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f_\alpha}{\partial \mathbf{v}} = S_\alpha + \sum_{\beta} C(f_\alpha, f_\beta), \quad (\text{A.4})$$

where  $S_\alpha$  is a source/sink of particles and  $C(f_\alpha, f_\beta)$  the collisional operator describing Coulomb collisions between the  $\alpha$  and  $\beta$  plasma species. Equation (A.4) can be properly integrated over the velocity space, thus obtaining a set of equations able to describe the evolution of the fluid moments. It turns out that the time evolution of each moment depends on a moment that is one order higher. To avoid solving for an infinite number of moments, a closure has to be introduced, i.e. an assumption on the moment one order higher than the highest order evolved moments.

Among the different two-fluid models derived to describe the plasma dynamics, we con-

sider here the one that was introduced by Braginskii in 1965 [13]. This model was deduced assuming that the electron and ion distribution functions are close to a Maxwellian, which is justified for strongly collisional plasmas, thus obtaining a closed set of equations for  $n_\alpha$ ,  $\mathbf{v}_\alpha$ , and  $T_\alpha$ .

Because of the wide range of spatial and temporal scales covered by these equations, which makes their numerical simulation extremely challenging, several approximations were introduced to obtain more suitable models, which have been implemented in a number of simulation codes. In particular, the drift approximation of the Braginskii model [14] was implemented into GBS, a simulation code developed in the last few years to simulate plasma turbulence in the open field region of basic plasma physics experiments and magnetic confinement devices, evolving the full plasma profiles without any separation between equilibrium and perturbation quantities [20, 177].

In this appendix we present the physical model that is implemented in GBS and the details of this code. The Braginskii equations are illustrated in Section A.1, and their drift-reduced limit is deduced in Section A.2. The main features of the GBS code are then summarized in Section A.3.

## A.1 Braginskii equations

It is a general result of statistical mechanics that the particles of any gas in thermal equilibrium have a Maxwellian distribution [13]

$$f_\alpha^0 = n \left( \frac{m_\alpha}{2\pi T_\alpha} \right)^{3/2} e^{-\frac{m_\alpha}{2T_\alpha} (\mathbf{v} - \mathbf{v}_\alpha)^2} \quad (\text{A.5})$$

and, as stated by the H-theorem, if a distribution function evolves only by virtue of collisions, it will approach a Maxwellian regardless of the initial condition. This process, called relaxation, occurs on a time-scale of the order of the collision time. In a plasma the electron and ion collision times can be written as

$$\tau_e = \frac{3\sqrt{m_e} T_e^{3/2}}{4\sqrt{\pi} \lambda e^4 Z^2 n_i}, \quad \tau_i = \frac{3\sqrt{m_i} T_i^{3/2}}{4\sqrt{\pi} \lambda e^4 Z^4 n_i}, \quad (\text{A.6})$$

where  $\lambda$  is the Coulomb logarithm and we define  $q_i = Ze$  the ion electric charge.

We focus our attention on phenomena occurring on time scales larger than the collision times and on spatial scales longer than both the Larmor radius and the distance traversed by particles between two Coulomb collisions. More precisely, we consider a plasma for which

$$\tau \gg \tau_\alpha, \quad L_\perp \gg \rho_\alpha, \quad L_\parallel \gg \lambda_\alpha^{mfp}, \quad (\text{A.7})$$

where  $\lambda_\alpha^{mfp} = v_{th,\alpha} \tau_\alpha$  is the mean free path between two Coulomb collisions,  $\rho_\alpha = v_{th,\alpha} / \omega_{c\alpha}$  the Larmor radius,  $v_{th,\alpha} = \sqrt{T_\alpha / m_\alpha}$  the thermal velocity,  $\omega_{c\alpha} = q_\alpha B / m_\alpha$

the cyclotron frequency,  $\tau$ ,  $L_{\parallel}$  and  $L_{\perp}$  are the characteristic time and lengths for the variation of macroscopic quantities, and the symbols  $\parallel$  and  $\perp$  refer to the direction parallel and perpendicular to the magnetic field, respectively. In this case, it results that the solution of the Boltzmann equation, Eq. (A.4), is locally close to a Maxwellian. In addition, we focus on strongly magnetized plasmas, i.e.  $\omega_{c\alpha}\tau_{\alpha} \gg 1$ . In this scenario, the equations that describe the behaviour of the macroscopic quantities  $n_{\alpha}$ ,  $\mathbf{v}_{\alpha}$ , and  $T_{\alpha}$ , which are called transport equations, were obtained by Braginskii in 1965 as described in the following. First, the distribution functions  $f_{\alpha}$  are written as  $f_{\alpha} = f_{\alpha}^0 + f_{\alpha}^1$ , where it is assumed  $|f_{\alpha}^1|/f_{\alpha}^0 \ll 1$  and  $f_{\alpha}^1$  is treated as a small perturbation on the Maxwellian distribution function  $f_{\alpha}^0$ . Second, the Boltzmann equation, Eq. (A.4), is linearised around  $f_{\alpha}^0$ . Finally, multiplying the linearised Boltzmann equation by 1,  $\mathbf{v}$ , and  $\mathbf{v}^2$ , respectively, and integrating over velocities, one obtains

$$\frac{\partial n_{\alpha}}{\partial t} + \nabla \cdot (n_{\alpha} \mathbf{v}_{\alpha}) = S_{n,\alpha}, \quad (\text{A.8})$$

$$m_{\alpha} n_{\alpha} \frac{d_{\alpha} \mathbf{v}_{\alpha}}{dt} = -\nabla p_{\alpha} - \nabla \cdot \underline{\pi}_{\alpha} + q_{\alpha} n_{\alpha} (\mathbf{E} + \mathbf{v}_{\alpha} \times \mathbf{B}) + \mathbf{R}_{\alpha}, \quad (\text{A.9})$$

$$\frac{3}{2} n_{\alpha} \frac{d_{\alpha} T_{\alpha}}{dt} = -p_{\alpha} \nabla \cdot \mathbf{v}_{\alpha} - \nabla \cdot \mathbf{q}_{\alpha} - \underline{\pi}_{\alpha} : \underline{\nabla \mathbf{v}_{\alpha}} + Q_{\alpha} + S_{Q,\alpha}. \quad (\text{A.10})$$

Here  $p_{\alpha} = n_{\alpha} T_{\alpha}$  is the scalar pressure,  $\underline{\pi}_{\alpha}$  the stress tensor,  $\mathbf{q}_{\alpha}(\mathbf{x}, t) = m_{\alpha} \int v^2 \mathbf{v} f_{\alpha}(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}/2$  the heat flux,  $S_{n,\alpha}$  and  $S_{Q,\alpha}$  are sources and/or sinks of particles and heat for the  $\alpha$  species,  $\underline{A} : \underline{B}$  the Frobenius inner product between the two tensors  $\underline{A}$  and  $\underline{B}$ , and

$$\frac{d_{\alpha}}{dt} = \frac{\partial}{\partial t} + \mathbf{v}_{\alpha} \cdot \nabla \quad (\text{A.11})$$

the material derivative. Equation (A.8) is the continuity equation and it ensures mass conservation. Equation (A.9) states the conservation of momentum, where  $\mathbf{R}_e = -\mathbf{R}_i$  represents the transfer of momentum due to the collisional friction between electrons and ions. Finally, Eq. (A.10) is an energy conservation equation, where  $-\underline{\pi}_{\alpha} : \underline{\nabla \mathbf{v}_{\alpha}}$  and  $Q_{\alpha}$  represent the viscous heat losses and the heat generated because of collisions between particles of different species, respectively. The set of equations (A.8)-(A.10) is closed by providing an expression for  $\mathbf{R}_{\alpha}$ ,  $\underline{\pi}_{\alpha}$ ,  $\mathbf{q}_{\alpha}$ , and  $Q_{\alpha}$  as function of  $n_{\alpha}$ ,  $\mathbf{v}_{\alpha}$ , and  $T_{\alpha}$ , which Braginskii obtained by assuming that the plasma is made by electrons and one single ion species, and that the plasma is quasi-neutral, i.e.  $n_e = Z n_i$ . In the following, we illustrate the expressions of  $\mathbf{R}_{\alpha}$ ,  $\underline{\pi}_{\alpha}$ ,  $\mathbf{q}_{\alpha}$ , and  $Q_{\alpha}$ , and we briefly discuss the underlying physics. For a detailed discussion we refer to Ref. [13].

The term  $\mathbf{R}_e = \mathbf{R}_u + \mathbf{R}_T$  is the sum of the two contributions

$$\mathbf{R}_u = en_e \left( \frac{\mathbf{j}_{\parallel}}{\sigma_{\parallel}} + \frac{\mathbf{j}_{\perp}}{\sigma_{\perp}} \right), \quad (\text{A.12})$$

$$\mathbf{R}_T = -0.71 n_e \nabla_{\parallel} T_e \mathbf{b} - \frac{3}{2} \frac{n_e}{\omega_{ce} \tau_e} \mathbf{b} \times \nabla T_e, \quad (\text{A.13})$$

## Appendix A. The SOL model and the GBS code

---

where  $\mathbf{j}_{\parallel} = en_e (v_{\parallel i} - v_{\parallel e}) \mathbf{b}$  and  $\mathbf{j}_{\perp} = en_e (\mathbf{v}_{\perp i} - \mathbf{v}_{\perp e})$  are the parallel and perpendicular current densities,  $\sigma_{\parallel} = 1.96\sigma_{\perp}$  and  $\sigma_{\perp} = e^2 n_e \tau_e / m_e$  are the parallel and perpendicular electrical conductivities, and  $\mathbf{b}$  is the unit vector parallel to the magnetic field. Here  $\mathbf{R}_u$  and  $\mathbf{R}_T$  represent the force of friction due to a nonzero relative velocity  $\mathbf{u} = \mathbf{v}_e - \mathbf{v}_i$  and due to the presence of electron temperature gradients, respectively.

The electron heat flux is made of two analogous terms:  $\mathbf{q}_e = \mathbf{q}_{e,u} + \mathbf{q}_{e,T}$ , where

$$\mathbf{q}_{e,u} = 0.71 n_e T_e \mathbf{u}_{\parallel} + \frac{3 n_e T_e}{2 \omega_{ce} \tau_e} \mathbf{b} \times \mathbf{u}, \quad (\text{A.14})$$

$$\mathbf{q}_{e,T} = -\chi_{\parallel e} \nabla_{\parallel} T_e - \chi_{\perp e} \nabla_{\perp} T_e - \frac{5 n_e T_e}{2 e B} \mathbf{u} \times \nabla T_e, \quad (\text{A.15})$$

with  $\nabla_{\perp} = \nabla - \mathbf{b}(\mathbf{b} \cdot \nabla)$  and  $\nabla_{\parallel} = \mathbf{b} \cdot \nabla$  the perpendicular and parallel gradients. Here  $\chi_{\parallel e} = 3.16 n_e T_e \tau_e / m_e$  and  $\chi_{\perp e} = 4.66 n_e T_e / (m_e \omega_{ce}^2 \tau_e)$  are the parallel and perpendicular electron thermal conductivities, respectively. The first term,  $\mathbf{q}_{e,u}$ , is related to the friction force  $\mathbf{R}_e$ , whereas the second one,  $\mathbf{q}_{e,T}$ , is related to the presence of electron temperature gradients.

The ion heat flux is defined in a similar way. However, since the ion momentum loss characteristic time for collisions against electrons is very large, it is justified to neglect the term  $\mathbf{q}_{i,u}$ . Thus, we obtain

$$\mathbf{q}_i = \mathbf{q}_{i,T} = -\chi_{\parallel i} \nabla_{\parallel} T_i - \chi_{\perp i} \nabla_{\perp} T_i + \frac{5 n_i T_i}{2 Z e B} \mathbf{b} \times \nabla T_i, \quad (\text{A.16})$$

where  $\chi_{\parallel i} = 3.9 n_i T_i \tau_i / m_i$  and  $\chi_{\perp i} = 2 n_i T_i / (m_i \omega_{ci}^2 \tau_i)$  are the parallel and perpendicular ion thermal conductivities.

Choosing a coordinate system with the  $z$  axis along the magnetic field, the components of the stress tensor  $\underline{\pi}_{\alpha}$  for a given species write

$$\pi_{zz} = -\eta_0 W_{zz}, \quad (\text{A.17})$$

$$\pi_{xx} = -\frac{\eta_0}{2} (W_{xx} + W_{yy}) - \frac{\eta_1}{2} (W_{xx} - W_{yy}) - \eta_3 W_{xy}, \quad (\text{A.18})$$

$$\pi_{yy} = -\frac{\eta_0}{2} (W_{xx} + W_{yy}) - \frac{\eta_1}{2} (W_{xx} - W_{yy}) + \eta_3 W_{xy}, \quad (\text{A.19})$$

$$\pi_{xy} = \pi_{yx} = -\eta_1 W_{xy} + \frac{\eta_3}{2} (W_{xx} - W_{yy}), \quad (\text{A.20})$$

$$\pi_{xz} = \pi_{zx} = -\eta_2 W_{xz} - \eta_4 W_{yz}, \quad (\text{A.21})$$

$$\pi_{yz} = \pi_{zy} = -\eta_2 + \eta_4 W_{xz}, \quad (\text{A.22})$$

where

$$W_{jk} = \frac{\partial v_j}{\partial x_k} + \frac{\partial v_k}{\partial x_j} - \frac{2}{3} \delta_{jk} \nabla \cdot \mathbf{v} \quad (\text{A.23})$$



---

## A.2. Drift-reduced Braginskii equations

are the components of the rate-of-strain tensor,

$$\eta_{0e} = 0.73n_e T_e \tau_e, \tag{A.24}$$

$$\eta_{1e} = 0.51 \frac{n_e T_e}{\omega_{ce}^2 \tau_e}, \quad \eta_{2e} = 4\eta_{1e}, \tag{A.25}$$

$$\eta_{3e} = -0.5 \frac{n_e T_e}{\omega_{ce}}, \quad \eta_{4e} = 2\eta_{3e} \tag{A.26}$$

the electron viscosity coefficients, and

$$\eta_{0i} = 0.96n_i T_i \tau_i, \tag{A.27}$$

$$\eta_{1i} = 0.3 \frac{n_i T_i}{\omega_{ci}^2 \tau_i}, \quad \eta_{2i} = 4\eta_{1i}, \tag{A.28}$$

$$\eta_{3i} = 0.5 \frac{n_i T_i}{\omega_{ci}}, \quad \eta_{4i} = 2\eta_{3i} \tag{A.29}$$

the ion viscosity coefficients.

Finally, the heat exchanged through collisions is expressed as

$$Q_i = \frac{3m_e n_e}{m_i \tau_e} (T_e - T_i) \tag{A.30}$$

for ions and

$$Q_e = -\mathbf{R}_e \cdot \mathbf{u} - Q_i \tag{A.31}$$

for electrons, where the Joule heating is neglected for ions since it is about  $m_e/m_i$  times smaller than  $-\mathbf{R}_i \cdot \mathbf{u}$ .

## A.2 Drift-reduced Braginskii equations

Although much simpler than the kinetic equation, the Braginskii equations, Eqs. (A.8)-(A.10), still cover an extremely wide range of spatial and temporal scales. It is therefore necessary to simplify the Braginskii equations to obtain a model suitable for the numerical investigation of SOL plasma turbulence. This is achieved by adopting the drift ordering, according to which [14]

$$\frac{d}{dt} \ll \omega_{ci}. \tag{A.32}$$

This is equivalent to assume that  $\rho_s/L_\perp \ll 1$  and that  $\partial/\partial t \ll \omega_{ci}$ , since

$$\mathbf{v}_\alpha \cdot \nabla \sim \frac{v_{E \times B}}{L_\perp} \sim \frac{\rho_s^2}{L_\perp^2} \omega_{ci}, \tag{A.33}$$

## Appendix A. The SOL model and the GBS code

---

where  $\rho_s = c_s/\omega_{ci}$  is the ion sound Larmor radius,  $c_s = \sqrt{T_e/m_i}$  the sound speed, and  $\mathbf{v}_{E \times B}$  the equilibrium  $\mathbf{E} \times \mathbf{B}$  drift. These assumptions are justified for the study of the SOL plasma dynamics, since SOL turbulence occurs on timescales much slower than the fast gyromotion and it is usually characterized by spatial variations that occur on scale lengths longer than  $\rho_s$ .

The drift-reduced approximation allows us to split the particle dynamics into parallel and perpendicular directions with respect to the magnetic field, and avoids to evolve a momentum equation to evaluate the ion and electron perpendicular velocities. Formally, it expresses the electron and ion velocities as

$$\mathbf{v}_e = v_{\parallel e} \mathbf{b} + \mathbf{v}_{\perp e}, \quad (\text{A.34})$$

$$\mathbf{v}_i = v_{\parallel i} \mathbf{b} + \mathbf{v}_{\perp i}. \quad (\text{A.35})$$

The perpendicular (to  $\mathbf{B}$ ) electron and ion velocities,  $\mathbf{v}_{\perp e}$  and  $\mathbf{v}_{\perp i}$ , are obtained by crossing Eq. (A.9) with  $\mathbf{B}/(en_\alpha B^2)$ . Assuming that the viscous terms are small and applying the ordering in Eq. (A.32), it is found that the leading order contribution to  $\mathbf{v}_{\perp \alpha}$  is given by

$$\mathbf{v}_{\perp \alpha 0} = \mathbf{v}_{E \times B} + \mathbf{v}_{*\alpha}, \quad (\text{A.36})$$

with  $\mathbf{v}_{E \times B} = \mathbf{E} \times \mathbf{B}/B^2$  the  $\mathbf{E} \times \mathbf{B}$  velocity and  $\mathbf{v}_{*\alpha} = \mathbf{B} \times \nabla p_\alpha / (q_\alpha n_\alpha B^2)$  the diamagnetic drift velocity. The first-order correction on  $\mathbf{v}_{\perp \alpha 0}$ , of order  $\rho_\alpha^2/L_\perp^2$  or proportional to  $\eta_{0\alpha}$ , constitutes the polarization velocity,  $\mathbf{v}_{pol,\alpha}$ . To compute  $\mathbf{v}_{pol,\alpha}$ , it is useful to simplify Eq. (A.9) as follows. First, the perpendicular momentum transfer vector  $\mathbf{R}_{\perp \alpha}$  is neglected. Second, the stress tensor  $\underline{\pi}_\alpha = \underline{\pi}_\alpha^{FLR} + \underline{\pi}_\alpha^{vis}$  is split into a finite Larmor radius (FLR) part and a viscous part, such that

$$\begin{aligned} \nabla \cdot \underline{\pi}_\alpha^{FLR} = & -m_\alpha n_\alpha (\mathbf{v}_{*\alpha} \cdot \nabla) \mathbf{v}_\alpha + p_\alpha \left[ \left( \nabla \times \frac{\mathbf{b}}{\omega_{c\alpha}} \right) \cdot \nabla \right] \mathbf{v}_\alpha \\ & + \nabla_\perp \left[ \frac{p_\alpha}{2\omega_{c\alpha}} \nabla \cdot (\mathbf{b} \times \mathbf{v}_\alpha) \right] + \mathbf{b} \times \nabla \left( \frac{p_\alpha}{2\omega_{c\alpha}} \nabla_\perp \cdot \mathbf{v}_\alpha \right) \end{aligned} \quad (\text{A.37})$$

and

$$\nabla \cdot \underline{\pi}_\alpha^{vis} = G_\alpha \boldsymbol{\kappa} - \frac{1}{3} \nabla G_\alpha + \mathbf{B} (\mathbf{b} \cdot \nabla) \left( \frac{G_\alpha}{B} \right), \quad (\text{A.38})$$

where  $\boldsymbol{\kappa} = \mathbf{b} \cdot \nabla \mathbf{b}$  is the field line curvature and  $G_\alpha = -3\eta_{0\alpha} (\nabla_\parallel v_{\parallel \alpha} - \boldsymbol{\kappa} \cdot \mathbf{v}_\alpha - \nabla \cdot \mathbf{v}_\alpha / 3)$  is the stress function (a complete derivation of these terms is discussed in Ref. [182] and references therein). The polarization velocity is finally evaluated as the leading terms in

$\mathbf{v}_{\perp\alpha} - \mathbf{v}_{\perp\alpha 0}$ , i.e.

$$\begin{aligned} \mathbf{v}_{pol,\alpha} = & \frac{\mathbf{b}}{\omega_{c\alpha}} \times \frac{d_{\alpha 0}}{dt} \mathbf{v}_{\perp\alpha 0} + \frac{1}{n_{\alpha} m_{\alpha} \omega_{c\alpha}} \left\{ \mathbf{b} \times \left[ p_{\alpha} \left( \nabla \times \frac{\mathbf{b}}{\omega_{c\alpha}} \right) \cdot \nabla \mathbf{v}_{\perp\alpha 0} \right] \right. \\ & + \mathbf{b} \times \nabla_{\perp} \left[ \frac{p_{\alpha}}{2\omega_{c\alpha}} \nabla \cdot (\mathbf{b} \times \mathbf{v}_{\perp\alpha 0}) \right] - \nabla_{\perp} \left( \frac{p_{\alpha}}{2\omega_{c\alpha}} \nabla_{\perp} \cdot \mathbf{v}_{\perp\alpha 0} \right) \\ & \left. + \mathbf{b} \times \left( G_{\alpha} \boldsymbol{\kappa} - \frac{\nabla G_{\alpha}}{3} \right) \right\}, \end{aligned} \quad (\text{A.39})$$

where

$$\frac{d_{\alpha 0}}{dt} = \partial_t + (\mathbf{v}_{E \times B} + v_{\parallel\alpha} \mathbf{b}) \cdot \nabla \quad (\text{A.40})$$

[ $\mathbf{v}_{pol,\alpha}$  is dropped because of the drift ordering, while the diamagnetic drift contribution is cancelled out by the first term of Eq. (A.37)]. To further simplify the model, it is possible to neglect  $\mathbf{v}_{pol,e}$  with respect to  $\mathbf{v}_{pol,i}$ , since  $m_e/m_i \ll 1$ . Moreover, as detailed in Ref. [182], it is possible to neglect the second, the third, and the fourth terms of Eq. (A.39), thus obtaining the final expressions of the perpendicular electron and ion velocities,

$$\mathbf{v}_{\perp e} = \mathbf{v}_{E \times B} + \mathbf{v}_{*e}, \quad (\text{A.41})$$

$$\mathbf{v}_{\perp i} = \mathbf{v}_{E \times B} + \mathbf{v}_{*i} + \mathbf{v}_{pol,i}, \quad (\text{A.42})$$

with

$$\mathbf{v}_{pol,i} = \frac{\mathbf{b}}{\omega_{ci}} \times \frac{d_{i0}}{dt} \mathbf{v}_{\perp i 0} + \frac{1}{n_i m_i \omega_{ci}} \mathbf{b} \times \left( G_i \boldsymbol{\kappa} - \frac{\nabla G_i}{3} \right). \quad (\text{A.43})$$

As pointed out earlier, the transport coefficients used in the Braginskii equations are derived under the assumption that the plasma is quasi-neutral. This is justified since the typical characteristic scale lengths for the turbulence are much larger than the Debye length  $\lambda_D = \sqrt{\epsilon_0 T_e / (e^2 n_e)}$ . In the following of this appendix we consider singly charged ions, i.e. we set  $Z = 1$ , and we write  $n = n_e = n_i$  and  $q_i = -q_e = e$ .

### A.2.1 The semi-electrostatic limit

It is useful to remove the compressional Alfvén wave dynamics from a turbulence model, as the timescale of this mode is considerably faster than the turbulent timescales of interest. This is achieved by assuming the perpendicular component of the electric field as electrostatic, thus writing  $\mathbf{E}_{\perp} = -\nabla_{\perp} \phi$ , with  $\phi$  the electrostatic potential. In other terms, we assume that the perturbed potential vector is parallel to the background magnetic field,  $\delta \mathbf{A} = -\psi \mathbf{b}_0$ , where  $\mathbf{b}_0$  is the unit vector parallel to the unperturbed magnetic field. In fact, within the assumptions that the plasma to magnetic pressure ratio is small, i.e.  $\beta = 2\mu_0(p_e + p_i)/B^2 \ll 1$ , that  $L_{\perp} \ll L_{\parallel}$ , and neglecting the displacement

## Appendix A. The SOL model and the GBS code

---

current, the perturbed magnetic field is written as

$$\delta\mathbf{B} = -\nabla \times (\psi\mathbf{b}_0) \simeq \mathbf{b}_0 \times \nabla_{\perp}\psi. \quad (\text{A.44})$$

The electric field is consequently given by

$$\mathbf{E} = -\nabla\phi + \mathbf{b}_0 \frac{\partial\psi}{\partial t}, \quad (\text{A.45})$$

where the perturbed poloidal flux  $\psi$  satisfies Ampère's law,

$$\nabla_{\perp}^2\psi = \mu_0 j_{\parallel}. \quad (\text{A.46})$$

We note that, besides entering in Eq. (A.45), magnetic fluctuations play also a role in the evaluation of the parallel gradient

$$\nabla_{\parallel} = \mathbf{b} \cdot \nabla = \mathbf{b}_0 \cdot \nabla + \frac{\mathbf{b}_0}{B} \times \nabla_{\perp}\psi \cdot \nabla, \quad (\text{A.47})$$

where  $B$  denotes the norm of the equilibrium magnetic field. Finally, we also note that, within the semi-electrostatic approximation, the  $\mathbf{E} \times \mathbf{B}$  velocity can be approximated as

$$\mathbf{v}_{E \times B} = \frac{\mathbf{b}_0 \times \nabla\phi}{B} \quad (\text{A.48})$$

and the diamagnetic drift velocities as

$$\mathbf{v}_{*e} = -\frac{\mathbf{b}_0 \times \nabla p_e}{enB}, \quad \mathbf{v}_{*i} = \frac{\mathbf{b}_0 \times \nabla p_i}{enB}. \quad (\text{A.49})$$

### A.2.2 Continuity and vorticity equations

Within the drift-reduced approximation considered above, the continuity equation, Eq. (A.8), for the electron and ion species is written as

$$\frac{\partial n}{\partial t} + \nabla \cdot \left[ n \left( \mathbf{v}_{E \times B} + \mathbf{v}_{*e} + v_{\parallel e} \mathbf{b}_0 \right) \right] = S_n, \quad (\text{A.50})$$

$$\frac{\partial n}{\partial t} + \nabla \cdot \left[ n \left( \mathbf{v}_{E \times B} + \mathbf{v}_{*i} + \mathbf{v}_{pol,i} + v_{\parallel i} \mathbf{b}_0 \right) \right] = S_n, \quad (\text{A.51})$$

where we use  $n = n_e = n_i$  and we assume  $S_{n,e} = S_{n,i} = S_n$  because of quasi-neutrality. By subtracting Eq. (A.50) from Eq. (A.51), we obtain the so-called vorticity equation,

written as

$$\frac{1}{3m_i\omega_{ci}} (\mathbf{b}_0 \times \boldsymbol{\kappa}) \cdot \nabla G_i + \nabla \cdot [n(\mathbf{v}_{*i} - \mathbf{v}_{*e})] + \frac{\nabla \cdot (j_{\parallel} \mathbf{b}_0)}{e} - \frac{1}{B\omega_{ci}} \nabla_{\perp} \cdot \left[ n \frac{d_{i0}}{dt} \left( \nabla_{\perp} \phi + \frac{1}{en} \nabla_{\perp} p_i \right) \right] = 0, \quad (\text{A.52})$$

where we neglect the inhomogeneity of the magnetic field in computing the divergence of  $\mathbf{v}_{pol,i}$ , we assume  $j_{\parallel} \simeq \mathbf{j} \cdot \mathbf{b}_0$ , valid within the semi-electrostatic approximation, and we use

$$\nabla \cdot \left[ \frac{\mathbf{b}_0}{B} \times \left( G_i \boldsymbol{\kappa} - \frac{\nabla G_i}{3} \right) \right] \simeq \frac{\mathbf{b}_0 \times \boldsymbol{\kappa}}{3B} \cdot \nabla G_i, \quad (\text{A.53})$$

which is justified when local current densities are negligible. Within these assumptions, the gyroviscous terms reduce to

$$G_i = -\eta_{0i} \left[ 2\nabla_{\parallel} v_{\parallel i} - \boldsymbol{\kappa} \cdot \mathbf{v}_{\perp i0} \right], \quad (\text{A.54})$$

$$G_e = -\eta_{0e} \left[ 2\nabla_{\parallel} v_{\parallel e} - \boldsymbol{\kappa} \cdot \mathbf{v}_{\perp e} \right]. \quad (\text{A.55})$$

Equation (A.52) ensures charge conservation, and it is equivalent to a current continuity equation  $\nabla \cdot \mathbf{j} = 0$ , where we account for parallel, diamagnetic, and polarization currents. A common approximation typically employed to further simplify Eq. (A.52) is the Boussinesq approximation (the validity of this assumption in modelling the SOL plasma dynamics is discussed in Refs. [91–93]), which states

$$\nabla_{\perp} \cdot \left[ n \frac{d_{i0}}{dt} \left( \nabla_{\perp} \phi + \frac{1}{en} \nabla_{\perp} p_i \right) \right] \simeq n \frac{d_{i0}}{dt} \left( \nabla_{\perp}^2 \phi + \frac{1}{en} \nabla_{\perp}^2 p_i \right). \quad (\text{A.56})$$

In several cases this approximation allows us to considerably decrease the computational cost of solving the drift-reduced Braginskii equations. While sometimes useful for simplifying Eq. (A.52), and used in all the simulations described in the present thesis, in the following of the present appendix we do not make use of this approximation.

### A.2.3 Generalized Ohm's law and parallel momentum equation

The parallel component of the electron momentum equation, Eq. (A.9), yields the generalized Ohm's law

$$m_e n \frac{d_{e0} v_{\parallel e}}{dt} = -\nabla_{\parallel} p_e + en \nabla_{\parallel} \phi - 0.71 n \nabla_{\parallel} T_e + en \frac{j_{\parallel}}{\sigma_{\parallel}} - \frac{2}{3} \nabla_{\parallel} G_e - en \frac{\partial \psi}{\partial t}. \quad (\text{A.57})$$

Here we use  $\mathbf{b} \cdot (\nabla \cdot \underline{\pi}_e^{FLR}) \simeq -m_e n (\mathbf{v}_{*e} \cdot \nabla) v_{\parallel e}$  and  $\mathbf{b} \cdot (\nabla \cdot \underline{\pi}_e^{vis}) \simeq 2\nabla_{\parallel} G_e/3$ , assuming that the modulus of  $B$  varies slowly along  $\mathbf{b}$ . We remark that the evolution of  $\psi$ , appearing in Eq. (A.57), is described by Ampère's law, Eq. (A.46).

## Appendix A. The SOL model and the GBS code

---

To deduce the parallel momentum equation, we sum the parallel ion and electron momentum equations. Neglecting electron inertia and the FLR contribution of electrons, since they are lighter with respect to ions by a factor  $m_i/m_e$ , as well as the viscous contribution in the electron stress tensor, which is smaller by a factor  $\sqrt{m_i/m_e}$  with respect to the ion contribution, we obtain

$$m_i n \frac{d_{i0} v_{\parallel i}}{dt} = -\nabla_{\parallel} (p_e + p_i) - \frac{2}{3} \nabla_{\parallel} G_i, \quad (\text{A.58})$$

where again we assume that the modulus of  $B$  varies slowly along  $\mathbf{b}$  and we neglect the  $\mathbf{v}_{pol,i}$  contribution to the material derivative because of the drift-ordering.

### A.2.4 Temperature equations

The ion temperature equation is derived within the drift reduced approximation from Eq. (A.10) by assuming  $\omega_{ci}\tau_i \gg 1$  and neglecting the electron-ion collisional heating term  $Q_i$  and the viscous heat losses, thus obtaining

$$\frac{3}{2} n \frac{d_i T_i}{dt} + p_i \nabla \cdot \mathbf{v}_i - \nabla \cdot (\chi_{\parallel i} \nabla_{\parallel} T_i) + \frac{5}{2e} \nabla \cdot \left( \frac{p_i \mathbf{b}_0}{B} \times \nabla T_i \right) = S_{Q,i}. \quad (\text{A.59})$$

Because of the drift ordering, Eq. (A.59) is simplified neglecting the  $\mathbf{v}_{pol,i} \cdot \nabla T_i$  term. On the other hand, the term  $\nabla \cdot \mathbf{v}_{pol,i}$  should be retained in the model. Therefore, using the electron and ion continuity equations, we eliminate this term by writing  $p_i \nabla \cdot \mathbf{v}_i \simeq p_i \nabla \cdot \mathbf{v}_e + T_i (v_{\parallel e} - v_{\parallel i}) \cdot \nabla_{\parallel} n + T_i (\mathbf{v}_{*e} - \mathbf{v}_{*i}) \cdot \nabla n$ , thus obtaining

$$\begin{aligned} \frac{3}{2} n \frac{d_{i0} T_i}{dt} + T_i \nabla \cdot (n \mathbf{v}_{*e}) + p_i \nabla \cdot (\mathbf{v}_{E \times B} + v_{\parallel e} \mathbf{b}_0) - T_i \frac{j_{\parallel}}{en} \cdot \nabla_{\parallel} n \\ - \nabla \cdot (\chi_{\parallel i} \nabla_{\parallel} T_i) + \frac{5p_i}{2e} \left( \nabla \times \frac{\mathbf{b}_0}{B} \right) \cdot \nabla T_i = S_{Q,i}, \end{aligned} \quad (\text{A.60})$$

where we use  $\mathbf{v}_{*i} \cdot \nabla p_i = 0$  to express  $n \mathbf{v}_{*i} \cdot \nabla T_i = -T_i \mathbf{v}_{*i} \cdot \nabla n$ .

The electron temperature equation is obtained within the drift-reduced approximation in the limit  $\omega_{ce}\tau_e \gg 1$  by neglecting the viscous heat losses, the frictional heating related to  $\mathbf{R}_u$ , and the electron-ion heat transfer, and it is written as

$$\begin{aligned} \frac{3}{2} n \frac{d_{e0} T_e}{dt} + T_e \nabla \cdot (n \mathbf{v}_{*e}) + p_e \nabla \cdot (\mathbf{v}_{E \times B} + v_{\parallel e} \mathbf{b}_0) - 0.71 \frac{T_e}{e} \nabla \cdot (j_{\parallel} \mathbf{b}_0) \\ - \nabla \cdot (\chi_{\parallel e} \nabla_{\parallel} T_e) - \frac{5p_e}{2e} \left( \nabla \times \frac{\mathbf{b}_0}{B} \right) \cdot \nabla T_e = S_{Q,e}, \end{aligned} \quad (\text{A.61})$$

where, similarly to the ion temperature equation, we use  $\mathbf{v}_{*e} \cdot \nabla p_e = 0$  to express  $n \mathbf{v}_{*e} \cdot \nabla T_e = -T_e \mathbf{v}_{*e} \cdot \nabla n$ .

## A.3 The GBS code

The GBS code makes use of the set of drift-reduced Braginskii equations described above, which are valid for high collisional plasmas. To develop GBS, increasingly complex magnetic configurations were considered. First, the code was developed to describe the plasma dynamics in basic plasma physics experiments, in particular linear devices such as LAPD [199] and simple magnetized toroidal devices such as TORPEX [161, 200, 201]. GBS was then extended to the tokamak geometry, and it is now able to model the tokamak SOL region in limited plasmas [100, 113, 145, 157]. In this section we discuss the model implemented in GBS and the algorithm used to numerically solve the GBS equations.

### A.3.1 The GBS differential operators

In order to express the drift-reduced Braginskii equations discussed above in an easy-to-implement form, we introduce here two differential operators. The curvature operator  $C(-)$  is introduced to simplify the expressions of  $\nabla B$  and curvature drifts, and of plasma compressibility, and it is written as

$$C(A) = \frac{B}{2} \left( \nabla \times \frac{\mathbf{b}_0}{B} \right) \cdot \nabla A, \quad (\text{A.62})$$

with  $A$  a scalar field. On the other hand, the  $\mathbf{E} \times \mathbf{B}$  convection is introduced in the equation by defining the Poisson's brackets operator

$$\{\phi, A\} = \mathbf{b}_0 \cdot (\nabla \phi \times \nabla A). \quad (\text{A.63})$$

With these definitions it is possible to write

$$\nabla \cdot (n\mathbf{v}_{*\alpha}) = \frac{2}{q_\alpha B} C(p_\alpha), \quad \nabla \cdot (n\mathbf{v}_{E \times B}) = \frac{1}{B} \{\phi, n\} + \frac{2n}{B} C(\phi). \quad (\text{A.64})$$

### A.3.2 The GBS model

Making use of the curvature and Poisson's brackets operators, the drift-reduced Braginskii equations are written in normalized units as

$$\frac{\partial n}{\partial t} = -\frac{R_0}{B} \{\phi, n\} - \nabla \cdot (nv_{\parallel e} \mathbf{b}_0) + \frac{2}{B} [C(p_e) - nC(\phi)] + \mathcal{D}_n(n) + S_n, \quad (\text{A.65})$$

$$\begin{aligned} \frac{\partial \Omega}{\partial t} = & -\frac{R_0}{B} \nabla \cdot \{\phi, \omega\} - \nabla \cdot [\nabla_{\parallel} (v_{\parallel i} \omega)] + \frac{B}{3} C(G_i) \\ & + B^2 \nabla \cdot (j_{\parallel} \mathbf{b}_0) + 2BC(p_e + \tau p_i) + \mathcal{D}_{\Omega}(\Omega), \end{aligned} \quad (\text{A.66})$$

$$\begin{aligned} \frac{\partial U_{\parallel e}}{\partial t} = & -\frac{R_0}{B} \{\phi, v_{\parallel e}\} - v_{\parallel e} \nabla_{\parallel} v_{\parallel e} \\ & + \frac{m_i}{m_e} \left[ \frac{\nu j_{\parallel}}{n} + \nabla_{\parallel} \phi - \frac{\nabla_{\parallel} p_e}{n} - 0.71 \nabla_{\parallel} T_e - \frac{2}{3n} \nabla_{\parallel} G_e \right] + \mathcal{D}_{v_{\parallel e}}(v_{\parallel e}), \end{aligned} \quad (\text{A.67})$$

$$\frac{\partial v_{\parallel i}}{\partial t} = -\frac{R_0}{B} \{\phi, v_{\parallel i}\} - v_{\parallel i} \nabla_{\parallel} v_{\parallel i} - \frac{2}{3n} \nabla_{\parallel} G_i - \frac{1}{n} \nabla_{\parallel} (p_e + \tau p_i) + \mathcal{D}_{v_{\parallel i}}(v_{\parallel i}), \quad (\text{A.68})$$

$$\begin{aligned} \frac{\partial T_e}{\partial t} = & -\frac{R_0}{B} \{\phi, T_e\} - v_{\parallel e} \nabla_{\parallel} T_e + \frac{4T_e}{3B} \left[ \frac{C(p_e)}{n} + \frac{5}{2} C(T_e) - C(\phi) \right] \\ & + \frac{2T_e}{3} \left[ 0.71 \frac{\nabla \cdot (j_{\parallel} \mathbf{b}_0)}{n} - \nabla \cdot (v_{\parallel e} \mathbf{b}_0) \right] + \nabla_{\parallel} (\chi_{\parallel e} \nabla_{\parallel} T_e) + \mathcal{D}_{T_e}(T_e) + S_{T_e}, \end{aligned} \quad (\text{A.69})$$

$$\begin{aligned} \frac{\partial T_i}{\partial t} = & -\frac{R_0}{B} \{\phi, T_i\} - v_{\parallel i} \nabla_{\parallel} T_i + \frac{4T_i}{3B} \left[ \frac{C(p_e)}{n} - \frac{5\tau}{2} C(T_i) - C(\phi) \right] \\ & + \frac{2T_i}{3} \left[ \frac{\nabla \cdot (j_{\parallel} \mathbf{b}_0)}{n} - \nabla \cdot (v_{\parallel i} \mathbf{b}_0) \right] + \nabla_{\parallel} (\chi_{\parallel i} \nabla_{\parallel} T_i) + \mathcal{D}_{T_i}(T_i) + S_{T_i}. \end{aligned} \quad (\text{A.70})$$

Here  $\Omega = \nabla \cdot \omega = \nabla \cdot (n\nabla_{\perp} \phi + \tau\nabla_{\perp} p_i)$  is the scalar vorticity,  $j_{\parallel} = n(v_{\parallel i} - v_{\parallel e})$  the parallel current,  $U_{\parallel e} = v_{\parallel e} + \beta_{e0} m_i \psi / (2m_e)$  the sum of electron inertial and electromagnetic flutter contributions,  $\nu$  is the normalized plasma resistivity, and  $\nabla_{\parallel} A = \mathbf{b}_0 \cdot \nabla + R_0 \beta_{e0} \{\psi, A\} / (2B)$  the parallel derivative of  $A$ , with  $A$  a scalar field. The system is closed by the Poisson and Ampère's equations, which are respectively written as

$$\nabla \cdot (n\nabla_{\perp} \phi) = \Omega - \tau \nabla_{\perp}^2 p_i, \quad (\text{A.71})$$

$$\left( \nabla_{\perp}^2 - \frac{\beta_{e0} m_i}{2 m_e} n \right) v_{\parallel e} = \nabla_{\perp}^2 U_{\parallel e} - \frac{\beta_{e0} m_i}{2 m_e} n v_{\parallel i}, \quad (\text{A.72})$$

with the magnetic flux deduced from  $\psi = \beta_{e0} m_i (U_{\parallel e} - v_{\parallel e}) / (2m_e)$  and  $\nabla_{\perp}^2 = -\nabla \cdot [\mathbf{b}_0 \times (\mathbf{b}_0 \times \nabla)]$  the perpendicular Laplacian operator.

The source terms  $S_n$ ,  $S_{T_e} = 2S_{Q,e} / (3n)$ , and  $S_{T_i} = 2S_{Q,i} / (3n)$  are used to mimic the outflow of hot plasma from the closed flux surface into the SOL region. These sources are assumed poloidally and toroidally constant, with a Gaussian shape in the radial



direction. Small perpendicular diffusion terms of the form  $\mathcal{D}_A(A) = D_A \nabla_{\perp}^2 A$ , with  $D_A$  a constant coefficient, are added for the numerical solution of the equations. The quantities appearing in the present section are normalized according to (tilde denotes a physical quantity in SI units):  $t = \tilde{t} / (\tilde{R} / \tilde{c}_{s0})$ ,  $n = \tilde{n} / \tilde{n}_0$ ,  $T_e = \tilde{T}_e / \tilde{T}_{e0}$ ,  $T_i = \tilde{T}_i / \tilde{T}_{i0}$ ,  $\phi = e\tilde{\phi} / \tilde{T}_{e0}$ ,  $v_{\parallel e} = \tilde{v}_{\parallel e} / \tilde{c}_{s0}$ ,  $U_{\parallel e} = (\tilde{v}_{\parallel e} + e\tilde{\psi} / m_e) / \tilde{c}_{s0}$ ,  $v_{\parallel i} = \tilde{v}_{\parallel i} / \tilde{c}_{s0}$ ,  $B = \tilde{B} / \tilde{B}_0$ ,  $\psi = 2e\tilde{\psi} / (\tilde{\beta}_{e0} m_i \tilde{c}_{s0})$ ,  $R_0 = \tilde{R} / \tilde{\rho}_{s0}$ ,  $\nu = (e^2 \tilde{n}_0 \tilde{R}) / (m_i \tilde{\sigma}_{\parallel} \tilde{c}_{s0})$ , where  $\tilde{\sigma}_{\parallel}$  is the parallel conductivity,  $\tilde{n}_0$ ,  $\tilde{T}_{e0}$ ,  $\tilde{T}_{i0}$ , and  $\tilde{B}_0$  are reference density, electron temperature, ion temperature, and magnetic field,  $\tilde{R}$  is the tokamak major radius, and  $\tilde{c}_{s0}$  and  $\tilde{\rho}_{s0}$  are given by  $\tilde{c}_{s0} = \sqrt{\tilde{T}_{e0} / m_i}$  and  $\tilde{\rho}_{s0} = \tilde{c}_{s0} m_i / (e\tilde{B}_0)$ . Distances perpendicular to  $\mathbf{B}$  are normalized to  $\tilde{\rho}_{s0}$ , while parallel and toroidal distances are normalized to  $\tilde{R}$ . Moreover, we have  $\tau = \tilde{T}_{i0} / \tilde{T}_{e0}$  and  $\beta_{e0} = 2\mu_0 \tilde{n}_0 \tilde{T}_{e0} / \tilde{B}_0^2$ . Finally, the gyroviscous terms are written

$$G_i = -\eta_{0i} \left[ 2\nabla_{\parallel} v_{\parallel i} + \frac{1}{B} C(\phi) + \frac{\tau}{nB} C(p_i) \right], \quad (\text{A.73})$$

$$G_e = -\eta_{0e} \left[ 2\nabla_{\parallel} v_{\parallel e} + \frac{1}{B} C(\phi) - \frac{1}{nB} C(p_e) \right], \quad (\text{A.74})$$

with  $\eta_{0i} = 0.96 \tilde{T}_{i0} \tilde{\tau}_i / (m_i \tilde{R}_0 \tilde{c}_{s0})$  and  $\eta_{0e} = 0.73 \tilde{T}_{e0} \tilde{\tau}_e / (m_e \tilde{R} \tilde{c}_{s0})$ . We note that, while a discussion of plasma-neutral interactions is outside the scope of the present thesis, a model that includes ionization, charge-exchange, and recombination processes, and couples plasma and neutral physics, is implemented in GBS, as detailed in Ref. [202].

### A.3.3 Boundary conditions

Equations (A.65)-(A.70) are completed by a set of boundary conditions at the magnetic pre-sheath (MP) entrance, where the validity of the drift approximation breaks down. Their derivation is detailed in Refs. [94, 158, 177] and the results are summarized here for completeness.

Since at the MP entrance plasma gradients in the direction normal to the walls dominate with respect to other directions, the plasma dynamics can be approximated as one dimensional. Under this condition, one obtains that at the MP entrance

$$v_{\parallel i} = \pm c_s \sqrt{F_T}, \quad (\text{A.75})$$

$$v_{\parallel e} = \pm c_s \exp\left(\Lambda - \frac{\phi}{T_e}\right), \quad (\text{A.76})$$

$$\partial_s n = \mp \frac{n}{c_s \sqrt{F_T}} \partial_s v_{\parallel i}, \quad (\text{A.77})$$

$$\partial_s \phi = \mp \frac{c_s}{\sqrt{F_T}} \partial_s v_{\parallel i}, \quad (\text{A.78})$$

$$\Omega = \mp c_s n \sqrt{F_T} \cos^2 \chi \partial_s^2 v_{\parallel i}, \quad (\text{A.79})$$

$$\partial_s T_e \approx 0, \quad (\text{A.80})$$

$$\partial_s T_i \approx 0, \quad (\text{A.81})$$

where the upper (lower) sign refers to the case of a magnetic field directed towards (away from) the wall,  $c_s = \sqrt{T_e}$  is the ion sound speed,  $F_T = 1 + \tau T_i / T_e$ ,  $\Lambda = \log \sqrt{m_i / (2\pi m_e)} \simeq 3$  for hydrogen plasmas, and  $s$  is a coordinate normal to the wall, which coincides with the poloidal angle in the infinite aspect ratio limit. Here we neglect the drift corrections from radial plasma gradients. The symbol  $\chi$  represents the angle between the magnetic field lines and the wall, with  $\cos \chi \approx 1$  for a toroidal limiter in the infinite aspect ratio limit. Finally, we note that electromagnetic effects are intrinsically three-dimensional, and therefore costly full kinetic simulations would be required to apply the boundary conditions to  $\psi$ . These simulations have not been performed yet and we impose  $\psi = 0$  at the MP entrance.

### A.3.4 Geometry

The differential operators used to express the GBS model equations are written for arbitrary magnetic geometries. Therefore, in order to implement the drift-reduced Braginskii equations in GBS, one has to express the magnetic field in a chosen coordinate system, and compute the corresponding metric tensor coefficients.

The GBS code makes use of the toric  $(\theta_*, r, \varphi)$  coordinate system, where  $r$  is a flux coordinate,  $\varphi$  is the toroidal angle, and  $\theta_*$  is the straight field line angle defined as

$$\theta_*(r, \theta) = \frac{1}{q(r)} \int_0^\theta \frac{\mathbf{B} \cdot \nabla \varphi}{\mathbf{B} \cdot \nabla \theta'} d\theta', \quad (\text{A.82})$$

with  $\theta$  and  $\theta'$  the poloidal angle and

$$q(r) = \frac{1}{2\pi} \int_0^{2\pi} \frac{\mathbf{B} \cdot \nabla \varphi}{\mathbf{B} \cdot \nabla \theta} d\theta \quad (\text{A.83})$$

the safety factor. This toric coordinate system is rescaled as  $y = a\theta_*$ ,  $x = r$ , and  $z = R_0\varphi$ , where  $a$  is the tokamak minor radius in  $\tilde{\rho}_{s0}$  units.

For a toroidal limiter at the high field side, with circular magnetic flux surfaces in the infinite aspect ratio limit, the differential operators used in GBS in normalized units are written as

$$\{\phi, A\} = \partial_y \phi \partial_x A - \partial_x \phi \partial_y A, \quad (\text{A.84})$$

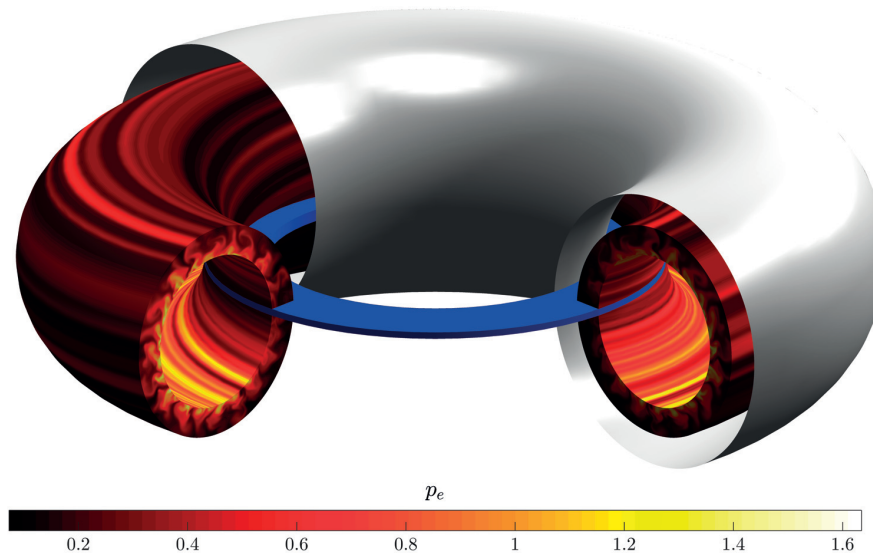
$$C(A) = \sin \theta_* \partial_x A + \cos \theta_* \partial_y A, \quad (\text{A.85})$$

$$\nabla_\perp^2 A = \partial_x^2 A + \partial_y^2 A, \quad (\text{A.86})$$

$$\nabla_\parallel A = \partial_z A + \frac{a}{q} \partial_y A + \beta_{e0} \{\psi, A\} / (2B), \quad (\text{A.87})$$

where  $\theta_* = y/a$  and  $\theta_* = -\pi$  and  $\theta_* = \pi$  at the inner midplane. Here the magnetic geometry is treated with the approximation of being radially local, i.e. geometric coefficients are assumed not to depend on  $x$  in the simulated domain. This approximation corresponds to assume a constant safety factor  $q$  in the radial direction and that  $L_p \ll a$ ,

with  $L_p$  the SOL width. We note that magnetic shear effects are included in GBS by performing the change of coordinate  $\hat{\theta}_* = \theta_* + \hat{s}\theta_*(r - a)/a$ , with  $\hat{s} = (r/q)dq/dr$  the magnetic shear, and assuming  $\partial_{\hat{\theta}_*} \simeq \partial_{\theta_*}$ , as detailed in Ref. [179]. We also note that GBS is able to handle circular magnetic geometries with finite aspect ratios, as detailed in Ref. [179], and that we illustrate the work done to generalize the GBS magnetic geometry to non-circular plasma shapes in Chapter 7.



**Figure A.1** – Snapshot of a section of the three-dimensional  $p_e$  profile obtained from a nonlinear GBS simulation. The domain evolved by GBS is shown, together with the toroidal limiter on the high-field side (in blue).

### A.3.5 Numerics

Herein we summarize the main aspects of the numerical implementation of the GBS model within the GBS code, whereas for a more detailed description we refer to Ref. [177]. The GBS model equations are integrated in time with a fourth order Runge-Kutta algorithm. The toric coordinates are discretized on a Cartesian grid, with spatial derivatives on the right-hand side of Eqs. (A.65)-(A.70) evaluated using second order finite difference schemes, except for the Poisson's brackets, which are computed using the Arakawa scheme [132]. In order to take advantage of the strong anisotropy of SOL turbulence, the grid is aligned to the magnetic field and the parallel gradient is computed along the field line using a second order finite difference approach. The Poisson and Ampère's equations can be solved using second order sparse matrix methods, or using a second order stencil-based multigrid solver, as detailed in Ref. [177] (the multigrid is necessary to avoid the use of the Boussinesq approximation and to perform electromagnetic simulations). The numerical grid where  $(U_{\parallel e}, v_{\parallel e}, v_{\parallel i}, \psi)$  are evaluated is displaced by half a grid point in the toroidal direction with respect to the  $(n, \Omega, \phi, T_e, T_i)$  grid for numerical

## Appendix A. The SOL model and the GBS code

---

stability. The code is parallelized using a domain decomposition technique based both on Message Passing Interface (MPI) and OpenMP approaches, allowing to run a simulation on thousands of CPUs.

An example of the results obtained from a GBS simulation is shown in Fig. A.1. The domain evolved by GBS, together with the toroidal limiter (in blue) are shown. The hot plasma outflowing from the core into the SOL is transported radially outwards because of turbulence, while flowing along the magnetic field and being ultimately lost at the limiter plates. Since a set of boundary conditions describing the plasma interaction with the outer wall and the plasma dynamics at the core does not exist yet, ad hoc boundary conditions are applied at the inner and outer radial boundaries of the simulated domain.

# APPENDIX B

## Estimate of the equilibrium pressure gradient length

The time-averaged plasma pressure gradient scale length  $L_p = -p_e/\nabla p_e$  in the tokamak SOL originates from a balance between the turbulent perpendicular transport of particles and heat, resulting from the nonlinear development of the unstable modes, and the parallel losses at the end of the magnetic field lines. In the limit of a negligible  $\mathbf{E} \times \mathbf{B}$  shear flow and for typical SOL parameters, it is justified to assume that the gradient removal turbulence saturation mechanism, i.e. the local nonlinear flattening of the plasma pressure profile and the resulting removal of the instability drive, is the mechanism that regulates the amplitude of SOL turbulence [100].

The main features of the theory are summarized here. Note that in the following of the present appendix all the equations are expressed in normalized units. The fundamental hypothesis is that the saturation of the growth rate of the linear modes occurs when these are able to remove their own drive, namely, the amplitude of the gradient associated with the fluctuation,  $k_X \delta p_e$ , is comparable to the gradient of the background pressure,  $p_{e0}/L_p$  ( $k_X \sim \sqrt{k_Y/L_p}$  is the radial extension of BMs or DWs obtained from a non-local linear theory [160, 161]). This provides an estimate for the amplitude of the fluctuations  $\delta p_e$ , given by

$$\delta p_e = \frac{p_{e0}}{\sqrt{k_Y L_p}}. \quad (\text{B.1})$$

Then, the leading terms in the linearised pressure continuity equation,

$$\gamma \delta p_e + R_0 k_Y \delta \phi \frac{p_{e0}}{L_p} \simeq 0, \quad (\text{B.2})$$

## Appendix B. Estimate of the equilibrium pressure gradient length

---

provide an estimate of the electromagnetic potential fluctuations

$$\delta\phi \sim \frac{\gamma\delta p_e L_p}{R_0 p_{e0} k_Y} \quad (\text{B.3})$$

and therefore of the turbulent  $\mathbf{E} \times \mathbf{B}$  flux,

$$\Gamma = k_Y \delta\phi \delta p_e \sim \frac{\gamma p_{e0}}{R_0 k_Y}. \quad (\text{B.4})$$

Finally, the balance between the perpendicular turbulent transport,

$$R_0 \partial_X \Gamma \sim R_0 \frac{\Gamma}{L_p} \sim \frac{\gamma p_{e0}}{k_Y L_p}, \quad (\text{B.5})$$

and the parallel losses at the end of magnetic field lines,

$$\nabla_{\parallel} (p_e v_{\parallel e}) \sim \frac{p_{e0} c_s}{q}, \quad (\text{B.6})$$

gives

$$L_p = \frac{q}{c_s} \left( \frac{\gamma}{k_Y} \right)_{\max}, \quad (\text{B.7})$$

where  $\gamma/k_Y$  should be maximized over the unstable modes present in the system. As detailed in Ref. [150], Eq. (B.7) was validated against a large number of inner-wall limited tokamak discharges, showing good agreement between theoretical predictions and experimental measurements. We finally remark that  $\gamma$  and  $k_Y$  depend on  $L_p$  and, therefore, Eq. (B.7) is an implicit equation in  $L_p$  that, in most cases, has to be solved numerically.

# APPENDIX C

## Differential operators for an axisymmetric magnetic field

To obtain the coefficients of the differential operators presented in Section 7.1.2, we proceed as follows. First, we define the covariant basis  $\nabla\xi^i$ , being  $\{\xi^i\} = \{\xi^1, \xi^2, \xi^3\}$  an arbitrary set of coordinates, and we write  $\mathbf{B} = B_i \nabla\xi^i$ , where  $B_i$  are the covariant components of  $\mathbf{B}$  and the Einstein's summation convention is employed. Note that the contravariant components of the vector  $\mathbf{B}$  are evaluated as  $B^i = \mathbf{B} \cdot \nabla\xi^i$ , the covariant components as  $B_i = g_{ij}B^j$ , the contravariant components of the metric tensor associated with the covariant basis  $\nabla\xi^i$  as  $g^{ij} = \nabla\xi^i \cdot \nabla\xi^j$ , and the covariant metric tensor is defined as the inverse of the contravariant metric tensor. The Jacobian associated with the coordinate transformation  $\{\xi^i\}$  is  $\mathcal{J}_{\xi^1\xi^2\xi^3} = 1/\sqrt{\det(g^{ij})}$ , where  $\det(g^{ij})$  is the determinant of the contravariant metric tensor  $g^{ij}$ .

Then, to compute the contravariant metric tensor  $g^{ij}$  of the toroic and flux tube coordinate systems, we introduce the standard cylindrical coordinate system  $(R_c, \varphi, Z_c)$ , with the  $Z_c$  axis coinciding with the tokamak symmetry axis and  $R_c$  the distance from this axis. Assuming axisymmetry, that is  $R_c = R_c(r, \theta)$  and  $Z_c = Z_c(r, \theta)$ , the components of the covariant metric tensor associated with the coordinate system  $(\theta, r, \varphi)$  are defined as

$$\begin{aligned}
 g_{\theta\theta} &= \left(\frac{\partial R_c}{\partial\theta}\right)^2 + \left(\frac{\partial Z_c}{\partial\theta}\right)^2, \\
 g_{\theta r} &= \frac{\partial R_c}{\partial r} \frac{\partial R_c}{\partial\theta} + \frac{\partial Z_c}{\partial r} \frac{\partial Z_c}{\partial\theta}, \\
 g_{rr} &= \left(\frac{\partial R_c}{\partial r}\right)^2 + \left(\frac{\partial Z_c}{\partial r}\right)^2, \\
 g_{\varphi\varphi} &= R_c^2, \\
 g_{r\varphi} &= g_{\theta\varphi} = 0.
 \end{aligned} \tag{C.1}$$

## Appendix C. Differential operators for an axisymmetric magnetic field

---

These expressions allow us to express the contravariant metric tensor components associated with the toric and flux-tube coordinate systems, which are

$$\begin{aligned}
 g^{\theta_*\theta_*} &= \left(\frac{\partial\theta_*}{\partial\theta}\right) g^{\theta\theta} + 2\frac{\partial\theta_*}{\partial\theta}\frac{\partial\theta_*}{\partial\theta} g^{\theta r} + \left(\frac{\partial\theta_*}{\partial r}\right)^2 g^{rr}, \\
 g^{\theta_*r} &= \frac{\partial\theta_*}{\partial r} g^{rr} + \frac{\partial\theta_*}{\partial\theta} g^{\theta r}, \\
 g^{\theta_*r} &= 0 \\
 g^{\theta_*\alpha} &= -\hat{s}(r)\theta_*\frac{q(r)}{r}g^{\theta_*r} - q(r)g^{\theta_*\theta_*}, \\
 g^{r\alpha} &= -\hat{s}(r)\theta_*\frac{q(r)}{r}g^{rr} - q(r)g^{\theta_*r}, \\
 g^{\alpha\alpha} &= g^{\varphi\varphi} + q(r)^2g^{\theta_*\theta_*} + 2\frac{q(r)^2\hat{s}(r)\theta_*}{r}g^{\theta_*r} + [\hat{s}(r)\theta_*]^2\frac{q(r)^2}{r^2}g^{rr},
 \end{aligned} \tag{C.2}$$

where  $\hat{s}(r) = r\partial_r q(r)/q(r)$  is the magnetic shear.

Writing the axisymmetric magnetic field as  $\mathbf{B} = F(\psi)\nabla\varphi + \nabla\psi \times \nabla\varphi$ , where  $\psi$  is the poloidal flux and  $F(\psi)$  is the current function, in the toric coordinate system one can write  $\mathbf{B} = B_\varphi\nabla\varphi + \mathcal{J}B^{\theta_*}\nabla r \times \nabla\varphi$ , while in flux-tube coordinates  $\mathbf{B} = \mathcal{J}B^{\theta_*}\nabla r \times \nabla\alpha$ , with  $\mathcal{J} = \mathcal{J}_{\theta_*r\varphi} = \mathcal{J}_{r\alpha\theta_*}$ . The contravariant components of the magnetic field are then given by

$$B^{\theta_*} = \frac{\psi'(r)}{\mathcal{J}}, \quad B^r = 0, \quad B^\varphi = \frac{F(\psi)}{R_c(r, \theta_*)^2} \tag{C.3}$$

in toric coordinates, while they write as

$$B^r = B^\alpha = 0, \quad B^{\theta_*} = \frac{\psi'(r)}{\mathcal{J}} \tag{C.4}$$

in flux-tube coordinates. Moreover, the norm of the magnetic field is computed as

$$B(r, \theta_*) = \frac{F(\psi)}{R_c(r, \theta_*)} \sqrt{1 + \frac{\psi'(r)^2}{F(\psi)^2} g^{rr}}. \tag{C.5}$$



The coefficients in Eqs. (7.9)-(7.12) are then given by

$$\begin{aligned}
\mathcal{P}^{yx} &= \frac{b_\varphi a}{\mathcal{J}}, & \mathcal{P}^{xz} &= \frac{b_{\theta_*}}{\mathcal{J}}, & P^{zy} &= \frac{b_r a}{\mathcal{J}}, \\
\mathcal{D}^y &= aR_0 b^{\theta_*}, & \mathcal{D}^x &= 0, & \mathcal{D}^z &= R_0 b^\varphi, \\
\mathcal{C}^x &= -\frac{R_0 B}{2\mathcal{J}} \frac{\partial c_\varphi}{\partial \theta_*}, & \mathcal{C}^y &= \frac{aR_0 B}{2\mathcal{J}} \frac{\partial c_\varphi}{\partial r}, & \mathcal{C}^z &= \frac{R_0 B}{2\mathcal{J}} \left( \frac{\partial c_r}{\partial \theta_*} - \frac{\partial c_{\theta_*}}{\partial r} \right), \\
\mathcal{N}^{xx} &= g^{rr}, & \mathcal{N}^{xy} &= 2ag^{r\theta_*}, & \mathcal{N}^{yy} &= a^2 \left[ g^{\theta_*\theta_*} - (b^{\theta_*})^2 \right], \\
\mathcal{N}^x &= \nabla^2 r, & \mathcal{N}^z &= -\frac{1}{\mathcal{J}} \frac{\partial}{\partial \theta_*} (\mathcal{J} b^{\theta_*} b^\varphi), & \mathcal{N}^{zz} &= g^{\varphi\varphi} - (b^\varphi)^2, \\
\mathcal{N}^{yz} &= -2ab^{\theta_*} b^\varphi, & \mathcal{N}^y &= a \left\{ \nabla^2 \theta_* - \frac{1}{\mathcal{J}} \frac{\partial}{\partial \theta_*} [\mathcal{J} (b^{\theta_*})^2] \right\},
\end{aligned} \tag{C.6}$$

where  $c_i = b_i/B$ ,  $b_i = B_i/B$ , and  $b^i = B^i/B$ . Since turbulence is characterized by  $\nabla_\perp A / \nabla_\parallel A \gg 1$ , it is possible to make the following approximations when evaluating the  $z$  derivatives

$$\begin{aligned}
\frac{\partial}{\partial z} &= \frac{1}{R_0 b^\varphi} \nabla_\parallel - \frac{a}{q} \frac{\partial}{\partial y} \simeq -\frac{a}{q} \frac{\partial}{\partial y}, \\
\frac{\partial^2}{\partial y \partial z} &= \frac{1}{R_0 b^\varphi} \frac{\partial}{\partial y} \nabla_\parallel - \frac{1}{aR_0 (b^\varphi)^2} \frac{\partial b^\varphi}{\partial \theta_*} \nabla_\parallel - \frac{a}{q} \frac{\partial^2}{\partial y^2} \simeq -\frac{a}{q} \frac{\partial^2}{\partial y^2}, \\
\frac{\partial^2}{\partial z^2} &= \frac{1}{(R_0 b^\varphi)^2} \nabla_\parallel^2 + \frac{a^2}{q^2} \frac{\partial^2}{\partial y^2} + \frac{b^{\theta_*}}{R_0 (b^\varphi)^3} \frac{\partial b^\varphi}{\partial \theta_*} \nabla_\parallel - 2 \frac{ab^{\theta_*}}{R_0 (b^\varphi)^2} \frac{\partial}{\partial y} \nabla_\parallel \simeq \frac{a^2}{q^2} \frac{\partial^2}{\partial y^2},
\end{aligned} \tag{C.7}$$

where we neglect the parallel derivatives with respect to the  $y$  derivatives. Consequently, the coefficients of Eq. (C.6) can be simplified into

$$\begin{aligned}
\mathcal{P}^{yx} &= \frac{a}{\mathcal{J} b^\varphi}, & \mathcal{P}^{xz} &= P^{zy} = 0, \\
\mathcal{C}^x &= -\frac{R_0 B}{2\mathcal{J}} \frac{\partial c_\varphi}{\partial \theta_*}, & \mathcal{C}^y &= \frac{aR_0 B}{2\mathcal{J}} \left[ \frac{\partial c_\varphi}{\partial r} + \frac{1}{q} \left( \frac{\partial c_{\theta_*}}{\partial r} - \frac{\partial c_r}{\partial \theta_*} \right) \right], & \mathcal{C}^z &= 0, \\
\mathcal{N}^{xx} &= g^{rr}, & \mathcal{N}^{xy} &= 2ag^{r\theta_*}, & \mathcal{N}^{yy} &= a^2 \left( g^{\theta_*\theta_*} + \frac{g^{\varphi\varphi}}{q^2} \right), \\
\mathcal{N}^x &= \nabla^2 r, & \mathcal{N}^y &= a \nabla^2 \theta_*, & \mathcal{N}^z &= 0, \\
\mathcal{N}^{yz} &= \mathcal{N}^{zz} = 0.
\end{aligned} \tag{C.8}$$

## Appendix C. Differential operators for an axisymmetric magnetic field

---

For the  $(X, Y, Z)$  coordinate system, the geometric coefficients are

$$\begin{aligned}
 \mathcal{P}^{XY} &= -\frac{b_{\theta_*} a}{\mathcal{J}q}, & \mathcal{P}^{YZ} &= -\frac{ab_r}{\mathcal{J}}, & \mathcal{P}^{ZX} &= -\frac{qb_\alpha}{\mathcal{J}}, \\
 \mathcal{D}^X &= \mathcal{D}^Y = 0, & \mathcal{D}^Z &= qR_0 b^{\theta_*}, \\
 \mathcal{C}^X &= -\frac{R_0 B}{2\mathcal{J}} \frac{\partial c_\alpha}{\partial \theta_*}, & \mathcal{C}^Y &= \frac{aR_0 B}{2\mathcal{J}q} \left( \frac{\partial c_r}{\partial \theta_*} - \frac{\partial c_{\theta_*}}{\partial r} \right), & \mathcal{C}^Z &= \frac{qR_0 B}{2\mathcal{J}} \frac{\partial c_\alpha}{\partial r}, \\
 \mathcal{N}^{XX} &= g^{rr}, & \mathcal{N}^{XY} &= \frac{2g^{\alpha r} a}{q}, & \mathcal{N}^{YY} &= \frac{a^2 g^{\alpha\alpha}}{q^2}, \\
 \mathcal{N}^X &= \nabla^2 r, & \mathcal{N}^Y &= \frac{a}{q} \nabla^2 \alpha, & \mathcal{N}^Z &= qR_0 \left\{ \nabla^2 \theta_* - \frac{1}{\mathcal{J}} \frac{\partial}{\partial \theta_*} \left[ \mathcal{J} (b^{\theta_*})^2 \right] \right\}, \\
 \mathcal{N}^{YZ} &= 2ag^{\theta_*\alpha}, & \mathcal{N}^{XZ} &= 2qg^{r\theta_*}, & \mathcal{N}^{ZZ} &= q^2 \left[ g^{\theta_*\theta_*} - (b^{\theta_*})^2 \right].
 \end{aligned} \tag{C.9}$$

Assuming  $\nabla_\perp A / \nabla_\parallel A \gg 1$ , it is possible to simplify the coefficients listed in Eq. (C.9) and neglect the  $Z$  derivatives with respect to the  $X$  and  $Y$  derivatives.

# APPENDIX D

## Derivation of a shaped magnetic equilibrium

The magnetic equilibrium presented in Section 7.1.3 is obtained by solving the Grad-Shafranov equation in the  $\bar{\epsilon} \rightarrow 0$  limit, being  $\bar{\epsilon} = r/R_0$  the local inverse aspect ratio. The main steps of the derivation are summarized here, while for a detailed discussion we refer to Ref. [176]. Note that in the remainder of this appendix we work in SI units. The magnetic surfaces  $(R_c, Z_c)$  are modelled by

$$R_c(r, \theta) = R_0 \left\{ 1 + \bar{\epsilon} \cos \theta + \frac{\Delta(r)}{R_0} + \sum_{m=2}^3 \frac{S_m(r)}{R_0} \cos[(m-1)\theta] - \frac{P(r)}{R_0} \cos \theta \right\}, \quad (\text{D.1})$$

$$Z_c(r, \theta) = R_0 \left\{ \bar{\epsilon} \sin \theta - \sum_{m=2}^3 \frac{S_m(r)}{R_0} \sin[(m-1)\theta] - \frac{P(r)}{R_0} \sin \theta \right\}, \quad (\text{D.2})$$

where  $\Delta(r)$  is the Shafranov's shift, being  $\Delta(a) = 0$ ,  $S_2(r)$  and  $S_3(r)$  shaping coefficients related to  $\kappa$  and  $\delta$  according to  $\kappa = [a - S_2(a)]/[a + S_2(a)]$  and  $\delta = 4S_3(a)/a$ , and  $P(r)$  is a correction factor used to ensure  $\psi'(r)/F(\psi) = \bar{\epsilon}/q(r) + \mathcal{O}(\bar{\epsilon}^4)$ . Assuming  $\Delta(r)/R_0 \sim S_m(r)/R_0 \sim P(r)/r \sim \mathcal{O}(\bar{\epsilon}^2)$ , with  $m = \{2, 3\}$ , we expand  $\mathcal{J}_{\theta r \varphi}/R_c^2 = (\partial_r R_c \partial_\theta Z_c - \partial_\theta R_c \partial_r Z_c)/R_c$  to third order in  $\bar{\epsilon}$ , and, imposing  $\psi'(r) = \bar{\epsilon} F(\psi)/q(r) + \mathcal{O}(\bar{\epsilon}^4)$ , we obtain

$$\frac{P(r)}{R_0} = \bar{\epsilon} \left( \frac{\bar{\epsilon}^2}{8} - \frac{\Delta(r)}{2R_0} \right) + \sum_{m=2}^3 \frac{1-m}{2\bar{\epsilon}} \left[ \frac{S_m(r)}{R_0} \right]^2 + \mathcal{O}(\bar{\epsilon}^4). \quad (\text{D.3})$$

In cylindrical coordinates  $(R_c, \varphi, Z_c)$ , the Grad-Shafranov operator  $\Delta^*$  is given by

$$\Delta^* \psi = R_c^2 \nabla \cdot \left( \frac{\nabla \psi}{R_c^2} \right) \quad (\text{D.4})$$

## Appendix D. Derivation of a shaped magnetic equilibrium

---

and, using the  $(\theta, r, \varphi)$  coordinate system, it can be written as

$$\Delta^* \psi = \frac{R_c(r, \theta)^2}{\mathcal{J}_{\theta r \varphi}} \left[ \frac{\partial}{\partial r} \left( \frac{\psi'(r) g_{\theta\theta}}{\mathcal{J}_{\theta r \varphi}} \right) - \frac{\partial}{\partial \theta} \left( \frac{\psi'(r) g_{r\theta}}{\mathcal{J}_{\theta r \varphi}} \right) \right]. \quad (\text{D.5})$$

Combining Eqs. (D.1)-(D.2) with Eq. (D.5), expanding the result to first order in  $\bar{\epsilon}$ , and writing separately each one of the  $m$  Fourier components, we obtain

$$\begin{aligned} \frac{1}{2r^2} \left[ r^2 \psi'(r)^2 \right]' + \mu_0 R_0^2 p'(\psi) + F(\psi) F'(\psi) &= 0, \\ \Delta''(r) + \Delta'(r) \left[ 2 \frac{\psi''(r)}{\psi'(r)} + \frac{1}{r} \right] + \frac{1}{R_0} - \frac{2R_0 r p'(\psi)}{\psi'(r)^2} &= 0, \\ S_m''(r) + \left[ 2 \frac{\psi''(r)}{\psi'(r)} + \frac{1}{r} \right] S_m'(r) + \frac{1-m^2}{r} S_m &= 0, \end{aligned} \quad (\text{D.6})$$

where  $p(\psi)$  is the plasma pressure at the flux surface  $\psi$ . Assuming  $F(r) = B_0 R_0 [1 + F_2(r)]$ , where  $F_2(r) \sim \bar{\epsilon}^2$ , and expanding Eq. (D.6) to first order in  $\bar{\epsilon}$ , we obtain

$$\begin{aligned} F_2'(r) &= \mu_0 \frac{p'(r)}{B_0^2} - \frac{r}{R_0^2} \frac{2 - \hat{s}(r)}{q(r)^2}, \\ \Delta'(r) &= -\bar{\epsilon} \left( \beta_p(r) + \frac{l_i(r)}{2} \right), \\ r^2 S_m''(r) + r[3 - 2\hat{s}(r)] S_m'(r) + (1 - m^2) S_m &= 0, \end{aligned} \quad (\text{D.7})$$

with

$$l_i(r) = \frac{2q(r)^2}{r^4} \int_0^r \frac{\bar{r}^3}{q(\bar{r})^2} d\bar{r}, \quad \beta_p(r) = -\frac{2\mu_0 R_0^2 q(r)^2}{r^4 B_0^2} \int_0^r p'(\bar{r}) \bar{r}^2 d\bar{r}. \quad (\text{D.8})$$

Equation (D.7) is solved assuming a safety factor with parabolic profile and neglecting the plasma pressure contribution, i.e assuming  $q(r) = q_0 + (q - q_0)(r/a)^2$  and  $p'(r) = 0$ . This gives

$$\begin{aligned} F_2(r) &= -\frac{r^2}{q(r)^2 R_0^2} \left[ 2 + \frac{q - q_0}{q_0} \left( \frac{r}{a} \right)^2 \right], \\ l_i(r) &= \frac{a^2 q(r)^2}{r^2 (q - q_0)} \left[ \frac{a^2}{r^2 (q - q_0)} \log \left( \frac{q(r)}{q_0} \right) - \frac{1}{q(r)} \right], \\ S_m(r) &= S_m(a) \left( \frac{r}{a} \right)^{m-1} \frac{q(r) \hat{s}(r) + 2q_0 \frac{m+1}{m-1}}{q \hat{s} + 2q_0 \frac{m+1}{m-1}}. \end{aligned} \quad (\text{D.9})$$

The Shafranov's shift is therefore

$$\Delta(r) = -\frac{1}{2R_0} \int_a^r \bar{r} l_i(\bar{r}) d\bar{r}. \quad (\text{D.10})$$

Equations (D.1)-(D.3), (D.9)-(D.10), and (7.23)-(7.24) define the magnetic equilibrium used in Chapter 7.

# APPENDIX E

## Curvature operator in the $\epsilon = 0$ limit

To obtain an analytical expression for the curvature operator  $C(A) = B[\nabla \times (\mathbf{b}_0/B)] \cdot \nabla A/2$  that takes into account shaping effects in the  $\epsilon = 0$  limit, we simplify the  $\mathcal{C}^Y$  coefficient in Eq. (7.17) as follows. First, we assume  $\mathcal{C}^Y \simeq -\partial_r R_c(r, \theta)|_{\theta_*}$ , according to a numerical investigation of the leading order term in the curvature operator. Second, having observed that BMs are mainly destabilized at  $\theta_* = 0$  (as discussed in Section 7.2), we assume  $\mathcal{C}^Y \simeq -\partial_r R_c(r, \theta)|_{\theta_*=0}$ . Third, combining Eq. (D.1) with Eq. (D.3), expanding the result in  $\epsilon$ , and retaining  $\epsilon$  zeroth and first order terms only, we obtain

$$R_c(r, \theta) = R_0 \left\{ 1 + \frac{r}{R_0} \cos \theta + \sum_{m=2}^3 \frac{S_m(r)}{R_0} \cos[(m-1)\theta] - \sum_{m=2}^3 \frac{1-m}{2rR_0} [S_m(r)]^2 \cos \theta \right\}. \quad (\text{E.1})$$

Fourth, expressing  $\partial_r R_c(r, \theta)|_{\theta_*=0} = \partial_r R_c(r, 0) + \partial_\theta R_c(r, 0) \partial_r \theta|_{\theta_*=0}$  and noting that  $\partial_r \theta|_{\theta_*=0} = 0$  for  $\epsilon = 0$ , we have

$$\partial_r R_c(r, \theta)|_{\theta_*=0} = 1 + \sum_{m=2}^3 S'_m(a) - \sum_{m=2}^3 \frac{1-m}{a} \left[ S_m(a) S'_m(a) - \frac{S_m(a)^2}{2a} \right]. \quad (\text{E.2})$$

Finally, substituting the  $S_m(r)$  expression given in Eq. (D.9) into Eq. (E.2) and using the definition of  $\kappa$  and  $\delta$ , we deduce

$$\partial_r R_c(r, \theta)|_{\theta_*=0} = 1 + \frac{\delta q}{1+q} + \frac{\delta^2(7q-1)}{16(1+q)} - \frac{\kappa-1}{2(\kappa+1)} - \frac{(\kappa-1)(5q-2)}{(\kappa+1)^2(2+q)}. \quad (\text{E.3})$$

This is the result presented in Eq. (7.43).



# Bibliography

- [1] V. Smil. *Energy Transitions: History, Requirements, Prospects*. Praeger (2010).
- [2] D. Ahuja and M. Tatsutani. Sustainable energy for developing countries. *Surveys and Perspectives Integrating Environment and Society* **2**, 1–16 (2009).
- [3] J. D. Lawson. Some Criteria for a Power Producing Thermonuclear Reactor. *Proceedings of the Physical Society. Section B* **70**, 6–10 (1957).
- [4] G. H. Miller, E. I. Moses, and C. R. Wuest. The National Ignition Facility: enabling fusion ignition for the 21st century. *Nuclear Fusion* **44**, S228–S238 (2004).
- [5] T. Klinger *et al.* Performance and properties of the first plasmas of Wendelstein 7-X. *Plasma Physics and Controlled Fusion* **59**, 014018 (2017).
- [6] J. Jacquinot. Fifty years in fusion and the way forward. *Nuclear Fusion* **50**, 014001 (2010).
- [7] M. Shimada *et al.* Chapter 1: Overview and summary. *Nuclear Fusion* **47**, S1–S17 (2007).
- [8] J. Wesson and D. Campbell. *Tokamaks*. International Series of Monographs on Physics. OUP Oxford (2011).
- [9] F. Wagner and U. Stroth. Transport in toroidal devices-experimentalist’s view. *Plasma Physics and Controlled Fusion* **34**, 1803–1803 (1992).
- [10] J. Ongena, R. Koch, R. Wolf, and H. Zohm. Magnetic-confinement fusion. *Nature Physics* **12**, 398–410 (2016).
- [11] A. Loarte *et al.* Chapter 4: Power and particle control. *Nuclear Fusion* **47**, S203–S263 (2007).
- [12] A. Fasoli *et al.* Computational challenges in magnetic-confinement fusion physics. *Nature Physics* **12**, 411–423 (2016).
- [13] S. I. Braginskii. Transport Processes in a Plasma. *Reviews of Plasma Physics* **1**, 205 (1965).

## Bibliography

---

- [14] A. Zeiler, J. F. Drake, and B. Rogers. Nonlinear reduced Braginskii equations with ion thermal dynamics in toroidal plasma. *Physics of Plasmas* **4**, 2134 (1997).
- [15] O. E. Garcia, V. Naulin, A. H. Nielsen, and J. J. Rasmussen. Computations of Intermittent Transport in Scrape-Off Layer Plasmas. *Physical Review Letters* **92**, 165003 (2004).
- [16] V. Naulin, T. Windisch, and O. Grulke. Three-dimensional global fluid simulations of cylindrical magnetized plasmas. *Physics of Plasmas* **15**, 012307 (2008).
- [17] T. T. Ribeiro and B. Scott. Gyrofluid turbulence studies of the effect of the poloidal position of an axisymmetric Debye sheath. *Plasma Physics and Controlled Fusion* **50**, 055007 (2008).
- [18] B. D. Dudson *et al.* BOUT++: A framework for parallel plasma fluid simulations. *Computer Physics Communications* **180**, 1467–1480 (2009).
- [19] P. Tamain *et al.* TOKAM-3D: A 3D fluid code for transport and turbulence in the edge plasma of Tokamaks. *Journal of Computational Physics* **229**, 361–378 (2010).
- [20] P. Ricci *et al.* Simulation of plasma turbulence in scrape-off layer conditions: the GBS code, simulation results and code validation. *Plasma Physics and Controlled Fusion* **54**, 124047 (2012).
- [21] F. D. Halpern *et al.* Three-dimensional simulations of blob dynamics in a simple magnetized torus. *Physics of Plasmas* **21**, 13–16 (2014).
- [22] J. Loizu *et al.* Effect of the limiter position on the scrape-off layer width, radial electric field and intrinsic flows. *Nuclear Fusion* **54**, 083033 (2014).
- [23] J. Loizu *et al.* Intrinsic toroidal rotation in the scrape-off layer of tokamaks. *Physics of Plasmas* **21**, 062309 (2014).
- [24] F. D. Halpern *et al.* Theory of the scrape-off layer width in inner-wall limited tokamak plasmas. *Nuclear Fusion* **54**, 043003 (2014).
- [25] W. L. Oberkampf and T. G. Trucano. Verification and validation in computational fluid dynamics. *Progress in Aerospace Sciences* **38**, 209–272 (2002).
- [26] W. L. Oberkampf and C. J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, New York, NY, USA (2010).
- [27] P. J. Roache. *Verification and Validation in Computational Science and Engineering*. Hermosa Publishers, Albuquerque, NM, USA (1998).
- [28] A. M. Dimits *et al.* Comparisons and physics basis of tokamak transport models and turbulence simulations. *Physics of Plasmas* **7**, 969 (2000).



- [29] J. Birn *et al.* Geospace Environmental Modeling (GEM) Magnetic Reconnection Challenge. *Journal of Geophysical Research* **106**, 3715–3719 (2001).
- [30] P. Ricci, J. U. Brackbill, W. Daughton, and G. Lapenta. Collisionless magnetic reconnection in the presence of a guide field. *Physics of Plasmas* **11**, 4102 (2004).
- [31] G. L. Falchetto *et al.* The European turbulence code benchmarking effort: turbulence driven by thermal gradients in magnetically confined plasmas. *Plasma Physics and Controlled Fusion* **50**, 124015 (2008).
- [32] R. V. Bravenec *et al.* A verification of the gyrokinetic microstability codes GEM, GYRO, and GS2. *Physics of Plasmas* **20**, 104506 (2013).
- [33] C. S. Chang *et al.* Whole-volume integrated gyrokinetic simulation of plasma turbulence in realistic diverted-tokamak geometry. *Journal of Physics: Conference Series* **180**, 012057 (2009).
- [34] D. Twarog, R. Stankiewicz, and K. Drozdowicz. Test of the European Transport Solver in the frame of Integrated Tokamak Modelling. Technical Report 2051/AP, Institute of nuclear physics, Polish Academy of Sciences (2011).
- [35] C. Holland *et al.* Advances in validating gyrokinetic turbulence models against L- and H-mode plasmas. *Physics of Plasmas* **18**, 056113 (2011).
- [36] N. T. Howard *et al.* Measurement of plasma current dependent changes in impurity transport and comparison with nonlinear gyrokinetic simulation. *Physics of Plasmas* **19**, 056110 (2012).
- [37] A. E. White *et al.* Multi-channel transport experiments at Alcator C-Mod and comparison with gyrokinetic simulations. *Physics of Plasmas* **20**, 056106 (2013).
- [38] N. Howard *et al.* Validation of the gyrokinetic model in ITG and TEM dominated L-mode plasmas. *Nuclear Fusion* **53**, 123011 (2013).
- [39] P. W. Terry *et al.* Validation in fusion research: Towards guidelines and best practices. *Physics of Plasmas* **15**, 062503 (2008).
- [40] M. Greenwald. Verification and validation for magnetic fusion. *Physics of Plasmas* **17**, 058101 (2010).
- [41] C. Holland. Validation metrics for turbulent plasma transport. *Physics of Plasmas* **23**, 060901 (2016).
- [42] S. Steinberg and P. J. Roache. Symbolic manipulation and computational fluid dynamics. *Journal of Computational Physics* **57**, 251–284 (1985).
- [43] P. J. Roache. Code Verification by the Method of Manufactured Solutions. *Journal of Fluids Engineering* **124**, 4 (2002).

## Bibliography

---

- [44] C. J. Roy. Review of code and solution verification procedures for computational simulation. *Journal of Computational Physics* **205**, 131–156 (2005).
- [45] L. F. Richardson. The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **210**, 307–357 (1911).
- [46] L. F. Richardson and J. A. Gaunt. The Deferred Approach to the Limit. Part I. Single Lattice. Part II. Interpenetrating Lattices. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **226**, 299–361 (1927).
- [47] P. J. Roache. Perspective: A Method for Uniform Reporting of Grid Refinement Studies. *Journal of Fluids Engineering* **116**, 405 (1994).
- [48] A. Fasoli *et al.* Electrostatic turbulence and transport in a simple magnetized plasma. *Physics of Plasmas* **13**, 055902 (2006).
- [49] I. Furno *et al.* Plasma turbulence, suprathermal ion dynamics and code validation on the basic plasma physics device TORPEX. *Journal of Plasma Physics* **81**, 345810301 (2015).
- [50] P. Sonato, R. Piovan, and A. Luchetta. Control of non-axisymmetric magnetic fields for plasma enhanced performances: The RFX contribution. *Fusion Engineering and Design* **74**, 97–107 (2005).
- [51] M. M. Turner *et al.* Simulation benchmarks for low-pressure plasmas: Capacitive discharges. *Physics of Plasmas* **20**, 013507 (2013).
- [52] J. Brackbill. On energy and momentum conservation in particle-in-cell plasma simulation. *Journal of Computational Physics* **317**, 405–427 (2016).
- [53] W. L. Oberkampf and T. G. Trucano. Verification and validation benchmarks. *Nuclear Engineering and Design* **238**, 716–743 (2008).
- [54] F. Riva *et al.* Verification methodology for plasma simulations and application to a scrape-off layer turbulence code. *Physics of Plasmas* **21**, 062301 (2014).
- [55] F. Riva, C. F. Beadle, and P. Ricci. A methodology for the rigorous verification of Particle-in-Cell simulations. *Physics of Plasmas* **24**, 055703 (2017).
- [56] B. D. Dudson *et al.* Verification of BOUT++ by the method of manufactured solutions. *Physics of Plasmas* **23**, 062303 (2016).
- [57] P. Tamain *et al.* The TOKAM3X code for edge turbulence fluid simulations of tokamak plasmas in versatile magnetic geometries. *Journal of Computational Physics* **321**, 606–623 (2016).

- 
- [58] F. Harlow. The particle-in-cell computing method for fluid dynamics. *Methods in Computational Physics* **3**, 319–343 (1964).
- [59] J. Dawson. One-Dimensional Plasma Model. *Physics of Fluids* **5**, 445 (1962).
- [60] C. K. Birdsall and A. B. Langdon. *Plasma physics via computer simulation*. Series in plasma physics. Taylor & Francis, New York (2005).
- [61] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. Taylor & Francis, Inc., Bristol, PA, USA (1988).
- [62] Y. Grigoryev, V. Vshivkov, and M. Fedoruk. *Numerical "Particle-in-Cell" Methods: Theory and Applications*. De Gruyter (2002).
- [63] J. M. Dawson. Computer modeling of plasma: Past, present, and future. *Physics of Plasmas* **2**, 2189 (1995).
- [64] J. Villasenor and O. Buneman. Rigorous charge conservation for local electromagnetic field solvers. *Computer Physics Communications* **69**, 306–316 (1992).
- [65] G. Chen, L. Chacón, and D. Barnes. An energy- and charge-conserving, implicit, electrostatic particle-in-cell algorithm. *Journal of Computational Physics* **230**, 7018–7036 (2011).
- [66] G. Lapenta and S. Markidis. Particle acceleration and energy conservation in particle in cell simulations. *Physics of Plasmas* **18**, 072101 (2011).
- [67] H. Denavit. Time-Filtering Particle Simulations. *Jouranl of Computational Physics* **366**, 337–366 (1981).
- [68] R. J. Mason. Implicit moment particle simulation of plasmas. *Journal of Computational Physics* **41**, 233–244 (1981).
- [69] J. Brackbill and D. Forslund. An implicit method for electromagnetic plasma simulation in two dimensions. *Journal of Computational Physics* **46**, 271–308 (1982).
- [70] A. Langdon, B. I. Cohen, and A. Friedman. Direct implicit large time-step particle simulation of plasmas. *Journal of Computational Physics* **51**, 107–138 (1983).
- [71] G. Lapenta, J. U. Brackbill, and P. Ricci. Kinetic approach to microscopic-macroscopic coupling in space and laboratory plasmas. *Physics of Plasmas* **13**, 055904 (2006).
- [72] D. W. Forslund and C. R. Shonk. Formation and Structure of Electrostatic Collisionless Shocks. *Physical Review Letters* **25**, 1699–1702 (1970).
- [73] D. W. Forslund, K. B. Quest, J. U. Brackbill, and K. Lee. Collisionless dissipation in quasi-perpendicular shocks. *Journal of Geophysical Research* **89**, 2142 (1984).

## Bibliography

---

- [74] B. Lembege and J. M. Dawson. Formation of double layers within an oblique collisionless shock. *Physical Review Letters* **62**, 2683–2686 (1989).
- [75] P. L. Pritchett. Geospace Environment Modeling magnetic reconnection challenge: Simulations with a full particle electromagnetic code. *Journal of Geophysical Research: Space Physics* **106**, 3783–3798 (2001).
- [76] J. F. Drake *et al.* Formation of electron holes and particle energization during magnetic reconnection. *Science* **299**, 873–877 (2003).
- [77] C. Joshi *et al.* Ultrahigh gradient particle acceleration by intense laser-driven plasma density waves. *Nature* **311**, 525–529 (1984).
- [78] S. P. D. Mangles *et al.* Monoenergetic beams of relativistic electrons from intense laser–plasma interactions. *Nature* **431**, 535–538 (2004).
- [79] S. C. Wilks, W. L. Kruer, M. Tabak, and A. B. Langdon. Absorption of ultra-intense laser pulses. *Physical Review Letters* **69**, 1383–1386 (1992).
- [80] D. Tskhakaya and S. Kuhn. Particle-in-cell simulations of the plasma-wall transition with a magnetic field almost parallel to the wall. *Journal of Nuclear Materials* **313-316**, 1119–1122 (2003).
- [81] J. Loizu, P. Ricci, and C. Theiler. Existence of subsonic plasma sheaths. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* **83**, 016406 (2011).
- [82] A. Y. Aydemir. A unified Monte Carlo interpretation of particle simulations and applications to non-neutral plasmas. *Physics of Plasmas* **1**, 822 (1994).
- [83] S. Allfrey and R. Hatzky. A revised  $\delta f$  algorithm for nonlinear PIC simulation. *Computer Physics Communications* **154**, 98–104 (2003).
- [84] G. Hu and J. A. Krommes. Generalized weighting scheme for  $\delta f$  particle-simulation method. *Physics of Plasmas* **1**, 863 (1994).
- [85] F. James. *Statistical Methods in Experimental Physics*. World Scientific Publishing Co Inc (2006).
- [86] L. Lista. *Statistical Methods for Data Analysis in Particle Physics*. Lecture Notes in Physics. Springer International Publishing (2015).
- [87] A. N. Kolmogorov. Sulla Determinazione Empirica di una Legge di Distribuzione. *Giornale dell’Istituto Italiano degli Attuari* **4**, 83–91 (1933).
- [88] J. A. Peacock. Two-dimensional goodness-of-fit testing in astronomy. *Monthly Notices of the Royal Astronomical Society* **202**, 615–627 (1983).
- [89] R. H. C. Lopes, P. R. Hobson, and I. D. Reid. Computationally efficient algorithms for the two-dimensional Kolmogorov–Smirnov test. *Journal of Physics: Conference Series* **119**, 042019 (2008).

- 
- [90] G. Fasano and A. Franceschini. A multidimensional version of the Kolmogorov-Smirnov test. *Monthly Notices of the Royal Astronomical Society* **225**, 155–170 (1987).
- [91] D. A. Russell, D. A. D’Ippolito, and J. R. Myra. On relaxing the Boussinesq approximation in scrape-off layer turbulence (SOLT) model simulations. *Bulletin of the American Physical Society, 54th Annual Meeting of the APS Division of Plasma Physics*, vol 57, BP8.159. Providence, Rhode Island, USA (2012).
- [92] G. Q. Yu, S. I. Krasheninnikov, and P. N. Guzdar. Two-dimensional modelling of blob dynamics in tokamak edge plasmas. *Physics of Plasmas* **13**, 042508 (2006).
- [93] K. Bodi *et al.* Impact of the Boussinesq approximation in tokamak scrape-off layer turbulence. *38th EPS Conference on Plasma Physics*, P1.121. Strasbourg, France (2011).
- [94] J. Loizu, P. Ricci, F. D. Halpern, and S. Jolliet. Boundary conditions for plasma fluid models at the magnetic presheath entrance. *Physics of Plasmas* **19**, 122307 (2012).
- [95] W. Research, Inc. *Mathematica*. Champaign, Illinois (2010).
- [96] P. J. Roache. Quantification of Uncertainty in Computational Fluid Dynamics. *Annual Review of Fluid Mechanics* **29**, 123–160 (1997).
- [97] F. Stern, R. V. Wilson, H. W. Coleman, and E. G. Paterson. Comprehensive Approach to Verification and Validation of CFD Simulations—Part 1: Methodology and Procedures. *Journal of Fluids Engineering* **123**, 793 (2001).
- [98] P. J. Roache and P. M. Knupp. Completed Richardson extrapolation. *Communications in Numerical Methods in Engineering* **9**, 365–374 (1993).
- [99] S. A. Richards. Completed Richardson extrapolation in space and time. *Communications in Numerical Methods in Engineering* **13**, 573–582 (1997).
- [100] P. Ricci and B. N. Rogers. Plasma turbulence in the scrape-off layer of tokamak devices. *Physics of Plasmas* **20**, 010702 (2013).
- [101] W. L. Oberkampf, T. G. Trucano, and C. Hirsch. Verification, validation, and predictive capability in computational engineering and physics. *Applied Mechanics Reviews* **57**, 345 (2004).
- [102] B. Peherstorfer, K. Willcox, and M. Gunzburger. Optimal Model Management for Multifidelity Monte Carlo Estimation. *SIAM Journal on Scientific Computing* **38**, A3163–A3194 (2016).
- [103] R. Myers, D. Montgomery, and C. Anderson-Cook. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. Wiley Series in Probability and Statistics. Wiley (2016).
-

## Bibliography

---

- [104] P. Ricci *et al.* Methodology for turbulence code validation: Quantification of simulation-experiment agreement and application to the TORPEX experiment. *Physics of Plasmas* **18**, 032109 (2011).
- [105] J. Mason and D. Handscomb. *Chebyshev Polynomials*. CRC Press (2002).
- [106] V. Naulin, O. Garcia, A. Nielsen, and J. Rasmussen. Statistical properties of transport in plasma turbulence. *Physics Letters A* **321**, 355–365 (2004).
- [107] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A Zang. *Spectral Methods in Fluid Dynamics*. Scientific Computation. Springer Berlin Heidelberg (2012).
- [108] V. Naulin and A. H. Nielsen. Accuracy of Spectral and Finite Difference Schemes in 2D Advection Problems. *SIAM Journal on Scientific Computing* **25**, 104–126 (2003).
- [109] J. Scheffel. A Spectral Method in Time for Initial-Value Problems. *American Journal of Computational Mathematics* **02**, 173–193 (2012).
- [110] B. Finlayson. *The Method of Weighted Residuals and Variational Principles*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics (2013).
- [111] K. W. Gentle and H. He. Texas Helimak. *Plasma Science and Technology* **10**, 284–289 (2008).
- [112] MATLAB. *version 9.1 (R2016b)*. The MathWorks Inc., Natick, Massachusetts (2016).
- [113] F. D. Halpern *et al.* Theory-based scaling of the SOL width in circular limited tokamak plasmas. *Nuclear Fusion* **53**, 122001 (2013).
- [114] D. A. D’Ippolito, J. R. Myra, and S. J. Zweben. Convective transport by intermittent blob-filaments: Comparison of theory and experiment. *Physics of Plasmas* **18**, 060501 (2011).
- [115] S. I. Krasheninnikov, D. A. D’Ippolito, and J. R. Myra. Recent theoretical progress in understanding coherent structures in edge and SOL turbulence. *Journal of Plasma Physics* **74**, 679–717 (2008).
- [116] O. E. Garcia *et al.* Mechanism and scaling for convection of isolated structures in nonuniformly magnetized plasmas. *Physics of Plasmas* **12**, 090701 (2005).
- [117] A. Y. Aydemir. Convective transport in the scrape-off layer of tokamaks. *Physics of Plasmas* **12**, 062503 (2005).
- [118] N. Bian, S. Benkadda, J.-V. Paulsen, and O. E. Garcia. Blobs and front propagation in the scrape-off layer of magnetic confinement devices. *Physics of Plasmas* **10**, 671 (2003).



- 
- [119] D. A. D'Ippolito and J. R. Myra. Blob stability and transport in the scrape-off-layer. *Physics of Plasmas* **10**, 4029 (2003).
- [120] G. Q. Yu and S. I. Krasheninnikov. Dynamics of blobs in scrape-off-layer/shadow regions of tokamaks and linear devices. *Physics of Plasmas* **10**, 4413 (2003).
- [121] S. Sugita, M. Yagi, S. I. Itoh, and K. Itoh. Propagation Velocity Analysis of a Single Blob in the SOL. *Plasma and Fusion Research* **3**, 040–040 (2008).
- [122] I. Furno *et al.* Blob current structures in TORPEX plasmas: experimental measurements and numerical simulations. *Plasma Physics and Controlled Fusion* **53**, 124016 (2011).
- [123] L. Easy *et al.* Three dimensional simulations of plasma filaments in the scrape off layer: A comparison with models of reduced dimensionality. *Physics of Plasmas* **21**, 122515 (2014).
- [124] A. H. Nielsen *et al.* Simulation of transition dynamics to high confinement in fusion plasmas. *Physics Letters A* **379**, 3097–3101 (2015).
- [125] J. J. Rasmussen *et al.* Numerical modeling of the transition from low to high confinement in magnetically confined plasma. *Plasma Physics and Controlled Fusion* **58**, 014031 (2016).
- [126] P. Tamain *et al.* 3D Properties of Edge Turbulent Transport in Full-Torus Simulations and their Impact on Poloidal Asymmetries. *Contributions to Plasma Physics* **54**, 555–559 (2014).
- [127] F. Riva *et al.* Blob dynamics in the TORPEX experiment: a multi-code validation. *Plasma Physics and Controlled Fusion* **58**, 044005 (2016).
- [128] C. Theiler *et al.* Cross-Field Motion of Plasma Blobs in an Open Magnetic Field Line Configuration. *Physical Review Letters* **103**, 065001 (2009).
- [129] I. Furno *et al.* Mechanism for blob generation in the TORPEX toroidal plasma. *Physics of Plasmas* **15**, 055903 (2008).
- [130] I. Furno *et al.* Experimental Observation of the Blob-Generation Mechanism from Interchange Waves in a Plasma. *Physical Review Letters* **100**, 055004 (2008).
- [131] P. Stangeby. *The Plasma Boundary of Magnetic Fusion Devices*. Series in Plasma Physics and Fluid Dynamics. Taylor & Francis (2000).
- [132] A. Arakawa. Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. Part I. *Journal of Computational Physics* **1**, 119–143 (1966).
- [133] G. D. Byrne and A. C. Hindmarsh. PVODE, an ODE Solver for Parallel Computers. *Int. J. High Perform. Comput. Appl.* **13**, 354–365 (1999).
-

## Bibliography

---

- [134] C. G. Theiler. *Basic Investigation of Turbulent Structures and Blobs of Relevance for Magnetic Fusion Plasmas*. Ph.D. thesis, SB, Lausanne (2011).
- [135] G. E. Karniadakis, M. Israeli, and S. A. Orszag. High-order splitting methods for the incompressible Navier-Stokes equations. *Journal of Computational Physics* **97**, 414–443 (1991).
- [136] R. Donat and A. Marquina. Capturing Shock Reflections: An Improved Flux Formula. *Journal of Computational Physics* **125**, 42–58 (1996).
- [137] W. Fundamenski *et al.* Dissipative processes in interchange driven scrape-off layer turbulence. *Nuclear Fusion* **47**, 417–433 (2007).
- [138] I. Furno *et al.* Pre-sheath density drop induced by ion-neutral friction along plasma blobs and implications for blob velocities. *Physics of Plasmas* **21**, 012305 (2014).
- [139] J. R. Myra, D. A. D'Ippolito, S. I. Krasheninnikov, and G. Q. Yu. Convective transport in the scrape-off-layer by nonthermalized spinning blobs. *Physics of Plasmas* **11**, 4267 (2004).
- [140] C. Theiler *et al.* Blob motion and control in simple magnetized plasmas. *Physics of Plasmas* **18**, 055901 (2011).
- [141] I. Furno *et al.* Direct Two-Dimensional Measurements of the Field-Aligned Current Associated with Plasma Blobs. *Physical Review Letters* **106**, 245001 (2011).
- [142] J. T. Omotani, F. Militello, L. Easy, and N. R. Walkden. The effects of shape and amplitude on the velocity of scrape-off layer filaments. *Plasma Physics and Controlled Fusion* **58**, 014030 (2016).
- [143] R. Pitts *et al.* Physics basis and design of the ITER plasma-facing components. *Journal of Nuclear Materials* **415**, S957–S964 (2011).
- [144] G. Arnoux *et al.* Scrape-off layer properties of ITER-like limiter start-up plasmas in JET. *Nuclear Fusion* **53**, 073016 (2013).
- [145] A. Masetto *et al.* Turbulent regimes in the tokamak scrape-off layer. *Physics of Plasmas* **20**, 092308 (2013).
- [146] A. Masetto, F. D. Halpern, S. Jolliet, and P. Ricci. Low-frequency linear-mode regimes in the tokamak scrape-off layer. *Physics of Plasmas* **19**, 112103 (2012).
- [147] F. Nespoli *et al.* Non-linear simulations of the TCV Scrape-Off Layer. *Nuclear Materials and Energy* **0**, 1–4 (2016).
- [148] F. D. Halpern *et al.* Comparison of 3D flux-driven scrape-off layer turbulence simulations with gas-puff imaging of Alcator C-Mod inner-wall limited discharges. *Plasma Physics and Controlled Fusion* **57**, 054005 (2015).



- 
- [149] R. Jorge *et al.* Plasma turbulence in the scrape-off layer of the ISTTOK tokamak. *Physics of Plasmas* **23**, 102511 (2016).
- [150] F. D. Halpern, J. Horacek, R. A. Pitts, and P. Ricci. A theoretical interpretation of the main scrape-off layer heat-flux width scaling for tokamak inner-wall limited plasmas. *Plasma Physics and Controlled Fusion* **58**, 084003 (2016).
- [151] P. Martin *et al.* Overview of the RFX fusion science program. *Nuclear Fusion* **51**, 094023 (2011).
- [152] P. Piovesan *et al.* RFX-mod: A multi-configuration fusion facility for three-dimensional physics studies. *Physics of Plasmas* **20**, 056112 (2013).
- [153] M. Spolaore *et al.* H-mode Achievement and Edge Features in RFX-mod Tokamak Operation. *26th IAEA Fusion Energy Conference*, EX/P5–24. Kyoto, Japan (2016).
- [154] M. Spolaore *et al.* Magnetic and electrostatic structures measured in the edge region of the RFX-mod experiment. *Journal of Nuclear Materials* **390-391**, 448–451 (2009).
- [155] N. Vianello *et al.* Drift-Alfvén vortex structures in the edge region of a fusion relevant plasma. *Nuclear Fusion* **50**, 042002 (2010).
- [156] H. Y. W. Tsui *et al.* A new scheme for Langmuir probe measurement of transport and electron temperature fluctuations. *Review of Scientific Instruments* **63**, 4608–4610 (1992).
- [157] F. D. Halpern *et al.* Ideal ballooning modes in the tokamak scrape-off layer. *Physics of Plasmas* **20**, 052306 (2013).
- [158] A. Masetto *et al.* Finite ion temperature effects on scrape-off layer turbulence. *Physics of Plasmas* **22**, 012308 (2015).
- [159] B. D. Scott. Free-energy conservation in local gyrofluid models. *Physics of Plasmas* **12**, 102307 (2005).
- [160] B. N. Rogers and W. Dorland. Noncurvature-driven modes in a transport barrier. *Physics of Plasmas* **12**, 062511 (2005).
- [161] P. Ricci, B. N. Rogers, and S. Brunner. High- and Low-Confinement Modes in Simple Magnetized Toroidal Plasmas. *Physical Review Letters* **100**, 225002 (2008).
- [162] F. Halpern and P. Ricci. Velocity shear, turbulent saturation, and steep plasma gradients in the scrape-off layer of inner-wall limited tokamaks. *Nuclear Fusion* **57**, 034001 (2017).
- [163] E. Sánchez *et al.* Statistical characterization of fluctuation wave forms in the boundary region of fusion and nonfusion plasmas. *Physics of Plasmas* **7**, 1408–1416 (2000).
-

## Bibliography

---

- [164] Y. H. Xu, S. Jachmich, R. R. Weynants, and the TEXTOR team. On the properties of turbulence intermittency in the boundary of the TEXTOR tokamak. *Plasma Physics and Controlled Fusion* **47**, 1841–1855 (2005).
- [165] I. Nanobashvili, J. Gunn, and P. Devynck. Radial profiles of plasma turbulent fluctuations in the scrape-off layer of the Tore Supra tokamak. *Journal of Nuclear Materials* **363-365**, 622–627 (2007).
- [166] P. Ricci *et al.* Langmuir probe-based observables for plasma-turbulence code validation and application to the TORPEX basic plasma physics experiment. *Physics of Plasmas* **16**, 055703 (2009).
- [167] C. P. Ritz *et al.* Advanced plasma fluctuation analysis techniques and their impact on fusion research (invited). *Review of Scientific Instruments* **59**, 1739–1744 (1988).
- [168] C. A. F. Varandas *et al.* Engineering Aspects of the Tokamak ISTTOK. *Fusion Technology* **29**, 105–115 (1996).
- [169] Y. Camenen *et al.* Impact of plasma triangularity and collisionality on electron heat transport in TCV L-mode plasmas. *Nuclear Fusion* **47**, 510–516 (2007).
- [170] J. E. Kinsey, R. E. Waltz, and J. Candy. The effect of plasma shaping on turbulent transport and ExB shear quenching in nonlinear gyrokinetic simulations. *Physics of Plasmas* **14**, 102306 (2007).
- [171] E. A. Belli, G. W. Hammett, and W. Dorland. Effects of plasma shaping on nonlinear gyrokinetic turbulence. *Physics of Plasmas* **15**, 092303 (2008).
- [172] A. Marinoni *et al.* The effect of plasma triangularity on turbulent transport: modeling TCV experiments by linear and non-linear gyrokinetic simulations. *Plasma Physics and Controlled Fusion* **51**, 055016 (2009).
- [173] W. Wan, Y. Chen, S. E. Parker, and R. J. Groebner. Effects of the magnetic equilibrium on gyrokinetic simulations of tokamak microinstabilities. *Physics of Plasmas* **22**, 062502 (2015).
- [174] J. Horacek *et al.* Multi-machine scaling of the main SOL parallel heat flux width in tokamak limiter plasmas. *Plasma Physics and Controlled Fusion* **58**, 074005 (2016).
- [175] F. Hofmann *et al.* Creation and control of variably shaped plasmas in TCV. *Plasma Physics and Controlled Fusion* **36**, B277–B287 (1994).
- [176] J. P. Graves. Toroidal drift precession and wave–particle interaction in shaped tokamaks with finite beta and neoclassical equilibrium effects. *Plasma Physics and Controlled Fusion* **55**, 074009 (2013).

- 
- [177] F. D. Halpern *et al.* The GBS code for tokamak scrape-off layer simulations. *Journal of Computational Physics* **315**, 388–408 (2016).
- [178] F. Riva, E. Lanti, S. Jolliet, and P. Ricci. Plasma shaping effects on tokamak scrape-off layer turbulence. *Plasma Physics and Controlled Fusion* **59**, 035001 (2017).
- [179] S. Jolliet *et al.* Aspect ratio effects on limited scrape-off layer plasma turbulence. *Physics of Plasmas* **21**, 022303 (2014).
- [180] B. Scott. Three-dimensional computation of drift Alfvén turbulence. *Plasma Physics and Controlled Fusion* **39**, 1635–1668 (1997).
- [181] J. W. Connor, R. J. Hastie, H. R. Wilson, and R. L. Miller. Magnetohydrodynamic stability of tokamak edge plasmas. *Physics of Plasmas* **5**, 2687 (1998).
- [182] A. Zeiler. Tokamak Edge Turbulence. Technical Report IPP 5/88, Max-Planck-Institut für Plasmaphysik (1999).
- [183] J. R. Myra, D. A. D’Ippolito, X. Q. Xu, and R. H. Cohen. Resistive modes in the edge and scrape-off layer of diverted tokamaks. *Physics of Plasmas* **7**, 4622 (2000).
- [184] D. D. Ryutov and R. H. Cohen. Instability Driven by Sheath Boundary Conditions and Limited to Divertor Legs. *Contributions to Plasma Physics* **44**, 168–175 (2004).
- [185] J. W. Connor, R. J. Hastie, and J. B. Taylor. Shear, Periodicity, and Plasma Ballooning Modes. *Physical Review Letters* **40**, 396–399 (1978).
- [186] G. Bateman and D. B. Nelson. Resistive-Ballooning-Mode Equation. *Physical Review Letters* **41**, 1804–1807 (1978).
- [187] D. R. McCarthy *et al.* Stability of resistive and ideal ballooning modes in the Texas Experimental Tokamak and DIII-D. *Physics of Fluids B: Plasma Physics* **4**, 1846 (1992).
- [188] S. V. Novakovskii *et al.* New unstable branch of drift resistive ballooning modes in tokamaks. *Physics of Plasmas* **2**, 781 (1995).
- [189] T. Rafiq, G. Bateman, A. H. Kritz, and A. Y. Pankin. Development of drift-resistive-inertial ballooning transport model for tokamak edge plasmas. *Physics of Plasmas* **17**, 082511 (2010).
- [190] K. Mima and A. Hasegawa. Nonlinear instability of electromagnetic drift waves. *Physics of Fluids* **21**, 81 (1978).
- [191] M. Wakatani and A. Hasegawa. A collisional drift wave description of plasma edge turbulence. *Physics of Fluids* **27**, 611 (1984).

## Bibliography

---

- [192] H. Sugama, M. Wakatani, and A. Hasegawa. Study of resistive drift and resistive interchange modes in a cylindrical plasma with magnetic shear. *Physics of Fluids* **31**, 1601 (1988).
- [193] W. Horton. Drift waves and transport. *Reviews of Modern Physics* **71**, 735–778 (1999).
- [194] P. H. Diamond, A. Hasegawa, and K. Mima. Vorticity dynamics, drift wave turbulence, and zonal flows: a look back and a look ahead. *Plasma Physics and Controlled Fusion* **53**, 124001 (2011).
- [195] S. Coda for the TCV Team. Overview of recent and current research on the TCV tokamak. *Nuclear Fusion* **53**, 104011 (2013).
- [196] J.-M. Moret *et al.* Tokamak equilibrium reconstruction code LIUQE and its real time implementation. *Fusion Engineering and Design* **91**, 1–15 (2015).
- [197] J. A. Boedo *et al.* Fast scanning probe for the NSTX spherical tokamak. *Review of Scientific Instruments* **80**, 123506 (2009).
- [198] R. Smith. *Uncertainty Quantification: Theory, Implementation, and Applications*. Computational Science and Engineering. SIAM (2013).
- [199] B. N. Rogers and P. Ricci. Low-Frequency Turbulence in a Linear Magnetized Plasma. *Physical Review Letters* **104**, 225002 (2010).
- [200] P. Ricci and B. N. Rogers. Transport scaling in interchange-driven toroidal plasmas. *Physics of Plasmas* **16**, 062303 (2009).
- [201] P. Ricci and B. N. Rogers. Turbulence Phase Space in Simple Magnetized Toroidal Plasmas. *Physical Review Letters* **104**, 145001 (2010).
- [202] C. Wersal and P. Ricci. A first-principles self-consistent model of plasma turbulence and kinetic neutral dynamics in the tokamak scrape-off layer. *Nuclear Fusion* **55**, 123014 (2015).

## Fabio Riva

Birth date: 06 February 1988

Phone: +41(21)6934560

Cell phone: +41(79)6488509

E-mail: [fabio.riva@epfl.ch](mailto:fabio.riva@epfl.ch)

Web: <http://people.epfl.ch/fabio.riva>

Address: EPFL SB SPC, Station 13, CH-1015 Lausanne

## EMPLOYMENT

- **Doctoral Assistant at the Swiss Plasma Center (SPC) of the École Polytechnique Fédérale de Lausanne (EPFL)** 05/2013 – 06/2017

## EDUCATION

- **PhD in plasma physics at the SPC of the EPFL** 05/2013 – 07/2017  
- PhD thesis title: "Verification and validation procedures with applications to plasma-edge turbulence simulations". Advisor: Prof. Paolo Ricci.
- **Master of Science MSc in Physics at the EPFL, accredited with a European Fusion Master's Certificate** 09/2011 – 02/2013  
- Master thesis title: "Experimental investigation of filamentary current structures associated with blobs in TORPEX plasmas". Grade: 6/6. Advisor: Dr. M.E.R. Ivo Furno.
- **Bachelor of Science BSc in Physics at the EPFL** 09/2007 – 07/2011
- **Matura** (Swiss university entrance qualification), **Liceo di Lugano 2** 2007

## TEACHING EXPERIENCE

- Co-supervision of a Master student during the 4 months Research Internship "Rigorous Verification of Particle-in-Cell Codes" 2015
- Co-supervision of a Master student during the 9 months Research Project "Development of a fully-spectral Global Braginskii Solver" 2014 – 2015
- Co-supervision of the Master thesis "Shaping effects on tokamak scrape-off layer turbulence" 2014
- Teaching assistantship for the following Bachelor and Master courses: 2013 – 2016
  - General Physics I, Prof. Ansermet
  - General Physics II, Prof. Ricci
  - Plasma physics II, Prof. Ricci
- Prize for exceptional assistantship work at EPFL 2016

## LANGUAGE KNOWLEDGE

Italian	<i>native speaker</i>
French	<i>fluent</i>
English	<i>fluent</i>
German	<i>basic knowledge</i>

## JOURNAL PUBLICATIONS

### • First author

- F. Riva, C. Beadle and P. Ricci, "A methodology for the rigorous verification of Particle-in-Cell simulations", **Physics of Plasmas** **24**, 055703 (2017);
- F. Riva, E. Lanti, S. Jolliet and P. Ricci, "Plasma shaping effects on tokamak scrape-off layer turbulence", **Plasma Physics and Controlled Fusion** **59**, 035001 (2017);
- F. Riva, C. Colin, J. Denis, L. Easy, I. Furno, J. Madsen, F. Militello, V. Naulin, A.H. Nielsen, J.M.B. Olsen, J.T. Omotani, J.J. Rasmussen, P. Ricci, E. Serre, P. Tamain and C. Theiler, "Blob dynamics in the TORPEX experiment: a multi-code validation", **Plasma Physics and Controlled Fusion** **58**, 044005 (2016);
- F. Riva, P. Ricci, F.D. Halpern, S. Jolliet, J. Loizu and A. Masetto, "Verification methodology for plasma simulations and application to a scrape-off layer turbulence code", **Physics of Plasmas** **21**, 062301 (2014).

### • Co-author

- F. Nespoli, F. Avino, I. Furno, F.D. Halpern, B. Labit, F. Musil, P. Ricci and F. Riva, "Blob properties in nonlinear simulations of the TCV Scrape-Off Layer", **Plasma Physics and Controlled Fusion** **59**, 055009 (2017);
- F. Nespoli, I. Furno, F.D. Halpern, B. Labit, J. Loizu, P. Ricci and F. Riva, "Non-linear simulations of the TCV Scrape- Off Layer", **Nuclear Materials and Energy** **0**, 1 (2016);
- F. Militello, N.R. Walkden, T. Farley, W.A. Gracias, J. Olsen, F. Riva, L. Easy, N. Fedorczak, I. Lupelli, J. Madsen, A.H. Nielsen, P. Ricci, P. Tamain and J. Young, "Multi-code analysis of Scrape-Off Layer filament dynamics in MAST", **Plasma Physics and Controlled Fusion** **58**, 115002 (2016);
- F.D. Halpern, P. Ricci, S. Jolliet, J. Loizu, J. Morales, A. Masetto, F. Musil, F. Riva, T.M. Tran and C. Wersal, "The GBS code for tokamak scrape-off layer simulations", **Journal of Computational Physics** **315**, 388 (2016);
- P. Ricci, F. Riva, C. Theiler, A. Fasoli, I. Furno, F.D. Halpern and J. Loizu, "Approaching the investigation of plasma turbulence through a rigorous verification and validation procedure: A practical example", **Physics of Plasmas** **22**, 055704 (2015);
- S. Jolliet, F.D. Halpern, J. Loizu, A. Masetto, F. Riva and P. Ricci, "Numerical approach to the parallel gradient operator in tokamak scrape-off layer turbulence simulations and application to the GBS code", **Computational Physics Community** **188**, 21 (2015);
- A. Fasoli, F. Avino, A. Bovet, I. Furno, K. Gustafson, S. Jolliet, J. Loizu, D. Malinverni, P. Ricci, F. Riva, C. Theiler, M. Spolaore and N. Vianello, "Basic investigations of electrostatic turbulence and its interaction with plasma and suprathermal ions in a simple magnetized toroidal plasma", **Nuclear Fusion** **53**, 063013 (2013).

## CONFERENCE CONTRIBUTIONS AND INVITED TALKS (1<sup>st</sup> AUTHOR)

- "A methodology for the rigorous verification of plasma simulation codes", invited talk presented at the 58th Annual Meeting of the APS Division of Plasma Physics, 2016, San Jose, California;
- "Progress in simulating SOL plasma turbulence with the GBS code", poster presented at the 21st EU-US Transport Task Force Meeting, 2016, Leysin, Switzerland;
- "Effects of plasma shaping on tokamak scrape-off layer turbulence", poster presented at the 16th European Fusion Theory Conference, 2015, Lisbon, Portugal;
- "Verification methodology for plasma simulations and application to a scrape-off layer turbulence code", poster presented at the 19th Joint EU-US Transport Task Force Meeting, 2014, Culham, UK.

## BOOKS

Co-author of the chapter "Verification and Validation in plasma physics" in "Plasma Modeling: Methods and Application", edited by Gianpiero Colonna and Antonio D'Angola, IOP Publishing Ltd of Temple Circus, Temple Way, Bristol (2016).

