Université de Strasbourg

HABILITATION A DIRIGER DES RECHERCHES

The GYSELA project: A semi-Lagrangian code addressing gyrokinetic full-f global simulations of flux driven tokamak plasmas

Virginie GRANDGIRARD

Soutenue le mardi 15 Novembre 2016

Pr. Daniel Bouche	*ENS Cachan, Research director*	Referee
Pr. Bruno Despres	*Pierre et Marie Curie University, Professor*	Referee
Pr. Philippe Helluy	*Strasbourg University, Professor*	
Pr. Eric Serre	*Aix-Marseille University, Research director*	Referee
Pr. Eric Sonnendrücker	*Max-Planck Institut für Plasmaphysik, Head of Numerical Methods in Plasma Physics Division*	

Abstract

Dans les machines à fusion telles que les tokamaks, le gain de puissance augmente non-linéairement avec le temps de confinement de l'énergie. La qualité de confinement énergétique du plasma détermine alors largement la taille et donc le coût d'un réacteur à fusion. Ce temps de confinement est principalement gouverné par la turbulence plasma existant dans les tokamaks –induisant des fluctuations d'une amplitude de quelques pourcents dans le coeur du plasma– mais aussi le transport associé. Comprendre l'origine de cette turbulence et ses propriétés en vue d'un possible contrôle est l'un des problèmes critiques pour la fusion par confinement magnétique. Le modèle le plus approprié pour simuler ces plasmas faiblement collisionnels est un modèle cinétique. Dans les descriptions premiers principes des plasmas, une équation à six dimensions (3 coordonnées de position et 3 de vitesse) –de Vlasov ou Fokker-Planck– est résolue pour la fonction de distribution de chaque espèce et couplée de manière auto-consistante aux équations de Maxwell décrivant les champs électromagnétiques. Par chance, dans les plasmas de fusion, les fluctuations turbulentes se développent à des fréquences typiques beaucoup plus faibles que la fréquence cyclotronique. Le problème 6D peut ainsi être réduit à un problème 5D. Malgré cette réduction d'une dimension, la construction d'outils de simulations basés sur ces principes reste très complexe et donc peu répandue. Moins de 15 codes gyrocinétiques 5D sont développés activement dans le monde, dont 4 sont européens. Cette HDR retrace le développement d'un tel code depuis 2001 à l'IRFM (Institut de Recherche sur la Fusion Magnétique) au CEA de Cadarache. Le code GYSELA (pour GYrokinetic SEmi-LAgrangian) permet de résoudre les équations gyrocinétiques pour plusieurs espèces ioniques dans la limite électrostatique (pas de fluctuations du champ magnétique). Les spécificités du code sont d'être global (simulation de l'ensemble du tore), de ne pas faire d'approximation de séparation d'échelle (code full-f) et de forcer la turbulence via des sources de particules, moment et de chaleur (par opposition aux codes qui fixent les gradients de densité, de vitesse et de température). Le code est de plus basé sur une méthode semi-Lagrangienne (qui correspond à un mixe entre l'approche Particle-In-Cell la plus répandue et l'approche eulérienne). Toutes ces propriétés en font un code unique en son genre. Comme tous les autres codes gyrocinétiques, il utilise de manière intensive les super-calculateurs et nécessite les techniques de pointe en calcul massivement parallèle. Un tel projet n'aurait donc pas été possible sans une collaboration forte entre mathématiciens, physiciens et informaticiens. Cette HDR est donc à l'interface entre trois domaines scientifiques que sont la physique de la turbulence plasma, les méthodes numériques pour les équations cinétiques et le calcul haute performance (HPC).

Contents

Abstract **iii**

Introduction **3**

Publications **11**

1 Gyrokinetic model for plasma turbulence **19**
1.1 Brief introduction to plasma fusion . . . 19
1.2 Theoretical Hierarchy of Plasma Physics . . . 21
1.3 Plasma kinetic theory . . . 23
1.4 Gyrokinetic Vlasov-Poisson system . . . 28
1.5 Main differences in gyrokinetic codes . . . 34

2 GYSELA - A global full-f code **39**
2.1 Toroidal geometry with simplified magnetic configuration . . . 40
2.2 Gyrokinetic Vlasov equation . . . 42
2.3 Self-consistent coupling with the quasi-neutrality equation . . . 43
2.4 Gyro-average operator: Padé or integration on the gyro-circles . . . 44
2.5 Initial and boundary conditions . . . 46
2.6 Simplified collision operator recovering neoclassical effects . . . 47
2.7 Collisional transfers between two species . . . 50
2.8 Flux-driven approach with source terms . . . 51

3 A highly parallel semi-Lagrangian code **57**
3.1 Specificity of the GYSELA code: a semi-Lagrangian scheme . . . 57
3.2 An efficient hybrid OpenMP/MPI parallelization . . . 61

4 GYSELA verification **69**
4.1 Normalization . . . 69
4.2 Invariance test . . . 70
4.3 Usual linear and non-linear gyrokinetic benchmarks . . . 73
4.4 Local conservation laws for gyrokinetics . . . 79
4.5 Local charge density conservation . . . 80

Conclusion **89**

A Relationship between $F_s(\mathbf{x}, \mathbf{v}, t)$ and $\bar{F}_s(\mathbf{x}_G, \mathbf{v}_G, t)$ **93**

B Some useful linear algebra expressions **95**
B.1 Some useful expressions of the main vectorial operators for a general metric tensor . . . 95
B.2 Poisson brackets defined as $[F, G] = \mathbf{b} \cdot (\boldsymbol{\nabla} F \times \boldsymbol{\nabla} G)$ . . . 96

C Quasi-neutrality solver **97**
C.1 How to overcome the difficulty due to $\langle U \rangle_{\mathrm{FS}}$ term ? . . . 97
C.2 Solving: Finite differences and Fourier projections . . . 98

D Padé approximation for gyro-average operator **101**

E LU factorization with a modified Thomas algorithm **103**
E.1 Tridiagonal Matrices . . . 103

F Numerical implementation of the collision operator in Gysela **105**
F.1 Expressions of the mean temperature $T_{s,\mathrm{coll}}$ and mean velocity $V_{\|s,\mathrm{coll}}$ for the collision operator . . . 105
F.2 Crank-Nicolson scheme for collision operator solving . . . 107

G Expressions for simplified collisional transfer between two species **111**
G.1 Conservation properties of collisional energy transfer . . . 111
G.2 Conservation properties of collisional parallel momentum transfer . . . 112

H Source terms **115**
H.1 Hermite and Laguerre polynomials . . . 115
H.2 Corresponding sources for fluid moments . . . 116

I A Crank-Nicolson scheme for diffusion terms **125**

J Interpolation by cubic splines **127**
J.1 Interpolation with classical cubic splines . . . 127
J.2 A new interpolation approach for Semi-Lagrangian scheme . . . 130

K Gysela normalized equations **135**
K.1 Gysela normalization . . . 135
K.2 Proofs for Gysela normalized equations . . . 141

L Invariance **153**

M Detailed computations for local conservation laws **155**
M.1 Useful integrals . . . 155
M.2 Expression of $\sum_s q_s \int \mathrm{d}\tau^* \bar{U} \partial_t \bar{F}_s$ for local energy conservation . . . 156
M.3 Effect of the electric potential on the toroidal canonical momentum . . . 157

N Diagnostics for local conservation laws in GYSELA **159**
N.1 Local charge density conservation . . . 160
N.2 Local energy conservation . . . 160
N.3 Local conservation of the toroidal momentum . . . 162
N.4 Input data files . . . 165

O GYSELA diagnostics **169**
O.1 Radial diagnostics . . . 169
O.2 3D diagnostics . . . 173

Bibliography **174**

List of Figures

1 Relative efficiency of the GYSELA code for weak and strong scaling on two bullx machines JADE (CINES, Montpellier, France) and CURIE (TGCC, Saclay, France) and on two Blue Gene supercomputers TURING (IDRIS, Orsay, France) and JUQUEEN (JSC/IAS, Juelich, Germany). 8
2 CPU time allocations for GYSELA on the different accessible supercomputers since 2012. 8
3 Some numbers for four challenging simulations performed with GYSELA: the first one with a big mesh of 272 billion of points for a ITER-like $\rho_* = 1/512$ (where $\rho_* = \rho_s/a$ with ρ_s the Larmor radius of species s and a the small radius of the tokamak), the second and third one which have run several days on more than 16k cores and the last one which correspond to a long time simulation. 9

1.1 Left: Schematic view of the coil system and magnetic field of a tokamak. Right: Corresponding idealized toroidal magnetic geometry and its adopted notations. 21
1.2 Quality factor Q increases with energy confinement time τ_E. Q is proportional to $\tau_E = \tau_E/(\tau_{\text{Lawson}} - \tau_E)$. 22
1.3 Typical space and time range scales in fusion plasmas and applicability of Vlasov, gyrokinetic and MHD models. Here, ω_{ps} is the plasma oscillation frequency, Ω_s is the cyclotron frequency, ω_s^* is the diamagnetic rotation frequency, v_A is the Alfvèn velocity, ν_{ii} is the ion-ion collision frequency, λ_{Ds} is the Debye length, ρ_s is the Larmor radius, L_n is the characteristic gradient length of the equilibrium density profile, a is the plasma size and s denotes the particle species. (figure from [IWS06]) 28
1.4 The phase-space reduction from 6 to 5 dimensions results from the gyro-centre transform, which aims at eliminating the fast gyro-motion. (figure from [IWS06]) 28

2.1 The tokamak magnetic configuration and the toroidal coordinate system (r, θ, φ). . . . 41
2.2 Poloidal cross-section showing flux surfaces and contours of $\theta =$ constant in the case of circular concentric magnetic configuration. 42
2.3 Schematic view of the coupling between the Poisson and Vlasov solvers in the GYSELA code. 45
2.4 The zero-th order Bessel function $J_0(k_\perp \rho_s)$ compare to its Padé approximation $1/\left[1 + (k_\perp \rho_s)^2/4\right]$. 45
2.5 Exact and approximated gyro-average operators applied on an arbitrary funcion F_k exhibiting a broad spectrum ranging from low to large wavelengths as compared with the Larmor radius ρ_c: (a) Representation in the Fourier space, (b) Representation in the real space (figures from [SGF+05]). 46
2.6 Parallel velocity dependence of the diffusive and convective operators $\mathcal{D}_{\|s}$ and $\mathcal{V}_{\|s}$in equations (2.27) and (2.31). 50
2.7 Schematic view of a heat source and buffer regions in the case of a flux driven simulation. Comparison between initial temperature profile (black line) and final temperature (red line). 52
2.8 Schematic view of the energetic particle source as a function of the parallel velocity . . 53

3.1 MPI_COMM_WORLD communicator decomposition for two species, 8 values of μ, $p_r =$ 4 radial sub-domains and $p_\theta = 2$ sub-domains in the poloidal direction. In this case, the number of MPI processes is equal to 128. 62

3.2 Strong scaling performed on the Curie machine from 2048 to 65536 cores: Execution time (a) and relative efficiency (b) for one GYSELA run of 4 iterations for a mesh $(N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}} \times N_\mu) = (512 \times 512 \times 128 \times 128 \times 32)$ with 8 threads and 32 values in μ direction. 65
3.3 Weak scaling performed on the JUQUEEN machine from $64k$ to $458k$ cores: Execution time (a) and relative efficiency (b) for one GYSELA run of 4 iterations for a mesh fixed in 4D as $(N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}}) = (512 \times 1024 \times 128 \times 128)$ but for 7 different values of $\mu = 2,\ 4,\ 8,\ 16,\ 32,\ 48,\ 56$. The number of parallel domains in r, θ directions and the number of threads are fixed such that $(p_r, p_\theta, N_{\text{thread}}) = (16, 32, 64)$. The number of cores varies as $p_r \times p_\theta \times N_\mu \times N_{\text{thread}}/4$ because 4 threads per core are used for BlueGene optimization. 65
3.4 Memory relative efficiency for a GYSELA simulation at the memory peak of time consumption for the four kind of structures used in the code (1D arrays to 4D arrays). The results are extracted from Table 3.2. The reference point corresponds to 128 MPI processes. 66

4.1 Invariance test between three simulations. Left Figure: Time evolution of Zonal Flows Φ_{00} at a fixed radial position 0.5ρ with $\rho = (r_{\max} - r_{\min})/a$. Φ_{00} for `simu1_A1Z1` with Hydrogen must be compared to $2\Phi_{00}$ of `simu2_A4Z2` for Helium and $50\Phi_{00}$ obtained with `simu3_A150Z50` for Tungsten. Right Figures: Relative errors in function of time: (top) between `simu1_A1Z1` and `simu2_A4Z2`, (bottom) between `simu1_A1Z1` and `simu3_A150Z50`. 73
4.2 Comparison of the residual value of the $(0,0)$ mode $\phi_{00}(r_{\text{peak}}, t)/\phi_{00}(r_{\text{peak}}, t = 0) = A_r$ where $A_r = 1/(1 + 1.6q^2/\sqrt{r/R}) = 0.06825$ is given by Rosenbluth-Hinton theory. This result corresponds to simulation 1 described in Table 4.7. 75
4.3 Damping rate (left) and frequency (right) of the electric field for simulation parameters given in table 4.8. Comparison with explicit analytical values given by Sugama and Watanabe [SW06] and Zonca [ZCS96] where FOW effects are taken into account or not. 76
4.4 Residual values divided by the initial value of the electric potential (green stars) or electric field (magenta triangles) for the 5 different values of constant q profile :1.5, 2., 2.5, 3. and 3.5. Comparison with analytical theory given by Rosenbluth-Hinton [RH98] (red line) and Xiao and Catto [XC06] (blue line). 76
4.5 Radial profile of the $(0,0)$ mode of ϕ at the initial time (blue line) and at the time $t = 30000/\Omega_{C_0}$ (red dotted line). 77
4.6 Benchmark between GYSELA and GENE codes for the Cyclone base Case for 6 different $k_\theta \rho_s$ values: (Left) Linear growth rate (plotted with the error bar defined in Table 4.10), (Right) Linear frequency. 78
4.7 Radial structure of the electrostatic potential for $(m, n) = (7, 5)$ and a bandwidth of $m \pm 3$ modes, at different times: (i) initial time, (ii) beginning of the linear phase $t = 7200\Omega_{C_0}^{-1}$, (iii) $t = 36000\Omega_{C_0}^{-1}$ and (iv) end of the linear phase $t = 50400\Omega_{C_0}^{-1}$. 79
4.8 Left: Numerical test of the radial force balance equation (4.1) at time $t = 20040\Omega_{C_0}^{-1}$, comparing the poloidal velocity v_θ^{GYS} directly computed in the code and v_θ^{FB} the sum of the three contributions $-E_r$, $\boldsymbol{\nabla} p/n_e$ and $v_\phi B_\theta$. Right: Relative error between both. The parameters used for this simulation are summarized in Table 4.11. 81
4.9 Numerical test of the charge density conservation for simulation defined in Table 4.11. All the quantities are flux-surface averaged. Left: Comparison of the two terms $-\partial_\chi J^\chi$ and $\mathrm{d}\rho/\mathrm{d}t$ which must be equivalent according to equation eq.(4.5). Right: Relative error (defined as the difference normalized to the quadratic mean). 82
4.10 Contribution of the neoclassical and turbulent parts in the charge density conservation equation for simulation defined in Table 4.11. All the quantities are flux-surface averaged. 82
4.11 Numerical test for local energy conservation for simulation defined in Table 4.11. Left: Comparison between $-\langle\, \partial_\chi Q \,\rangle_{\text{FS}}$ and $\mathrm{d}\langle\, E \,\rangle_{\text{FS}}/\mathrm{d}t$ with $Q = Q_s + Q_{s,\,\text{pot}} + Q_{s,\,\text{pol}}$ and $E = E_{K_s} + E_{p_s}$ defined by eqs.(4.16)-(4.21). Right: Relative error (defined here as the difference normalized to the quadratic mean). 85

4.12 Detailed contribution of each terms in the local energy conservation for simulation defined in Table 4.11. 85
4.13 Numerical test for local toroidal momentum conservation for simulation detailed in Table 4.11. All the quantities are flux-surface averaged. Left: Comparison between $\partial_t \mathcal{L}_\varphi$ and $J^\chi - \partial_\chi \Pi^\chi_\varphi - T^\chi_{\text{pol}}$ which must be equal according to eq.(4.25). Right: Relative error (defined here as the difference normalized to the quadratic mean). 87
4.14 Detailed contributions of neoclassical and turbulent parts in local toroidal momentum conservation for simulation detailed in Table 4.11. All the quantities are flux-surface averaged. 87

N.1 Input data file used in the code for the first simulation for invariance tests (see section 4.2). 165
N.2 Example of input data file used for Rosenbluth-Hinton test (see section 4.3.1). 166
N.3 Example of input data file used for the Cyclone Base Case benchmark with GENE code (see section 4.3.2). 167

Introduction

In magnetic fusion devices, the power gain increases non linearly with the energy confinement time. The performance in plasma energy confinement is a major parameter in determining the size and therefore the cost of a fusion reactor. This confinement time is mainly governed by plasma turbulence as observed in experiments [SCM+07, PDM+07]. The latter sustains fluctuations with relative magnitude of a few percents in the hot core and increasing towards the edge. Turbulent transport is understood to govern heat transport for both ions and electrons. It also plays a key role in flow generation and particle transport [ITE99]. Despite extensive experimental and theoretical effort, plasma turbulence in fusion experiments remains poorly understood and is a major unknown when scaling present experiments to ITER size devices, and, even more troublesome to predict confinement properties of a device like WEST. Two issues are of particular importance, (i) the level of self-organization, and therefore of self-regulation that determines turbulent transport properties in ITER standard conditions, and (ii) the ability of controlling turbulent transport, either towards increasing heat confinement, or towards increasing impurity out-flux due to turbulent transport. Inhomogeneity in density, temperature, and magnetic field drive the plasma out of thermodynamical equilibrium, and tend to excite several micro-instabilities over a wide spectral range. These plasmas exhibit low collisionality so that conventional fluid models are questionable and kinetic descriptions are more appropriate. A kinetic formalism is also needed to account for wave-particle resonant interaction.

In such first-principle descriptions of plasmas, a six dimensional evolution equation for the distribution function – Vlasov or Fokker-Planck equations – is solved for each species, coupled to self-consistent equations for the electromagnetic fields, namely Maxwell's equations [Ich92]. Since turbulent fluctuations are observed to exhibit much lower typical frequencies than the high frequency cyclotron motion, the 6D problem can be reduced to 5D one by removing, using phase space reduction, the gyromotion and other high-frequency dynamics. The useful part of the distribution function then evolves in a five dimensional phase space generated by four slow variables and an adiabatic invariant. This model is known as the *gyrokinetic* model. For detailed gyrokinetic theory see review papers by Brizard & Hahm (2007) [BH07] and Krommes (2012) [Kro12]. However, despite this dimensional reduction, solving 5D non-linear gyrokinetic equations for several ion species remains quite challenging. Moreover, the complex tokamak geometry and boundary conditions, contribute to making such a simulation tool quite demanding with respect to development and use. Consequently, these are not very widespread and, to date, none of the codes addresses routinely all the physics at hand. Less than fifteen 5D first-principle gyrokinetic codes are actively maintained all over the world, four being European codes. Since 2001, I am in charge of developing one of them at IRFM at CEA Cadarache: the Gysela code (for GYrokinetic SEmi-LAgrangian code). Gysela is a 5D gyrokinetic code dedicated to non-linear simulations of Ion Temperature Gradient (ITG) turbulence. The strength of the code is to be a global full-f flux-driven code. Adding to this the fact that it is based on a semi-Lagrangian numerical scheme makes it the only one of its kind in the gyrokinetic worldwide community. As the other gyrokinetic codes, Gysela benefits from an intensive use of massively parallel supercomputers and require state-of-the-art high performance computing (HPC). Such a development would not have been possible without a strong collaboration between mathematicians, physicists and computer scientists. This manuscript therefore reflects this interface between three scientific domains, namely plasma turbulence physics, numerical methods for kinetic equations and high performance computing.

GYSELA a global full-f gyrokinetic semi-Lagrangian code

My first contact with plasma turbulence was during my post-doctoral position (2000-2001) at Centre de recherche en physique des plasmas (CRPP) at Ecole Polytechnique Fédérale de Lausanne (EPFL) where I had developed a 4D drift-kinetic ITG code in cylindrical geometry [37]. This work has been undertaken with M. Brunetti during her post-doc and a positive conservative flux method has been tested [12, 11]. We had shown that the advantage of prescribing positivity of the distribution function was counterbalanced by the drawback of an over-diffusivity.

The first task upon my arrival at CEA Cadarache was to develop a 5D gyrokinetic code in toroidal geometry. The main initial code requirements were to be: (i) *global*, namely simulating the whole plasma domain to be able to study the impact of large scale flows on turbulence ; (ii) *full-f*, meaning that the whole distribution function –with no separation between equilibrium and fluctuation parts– is evolved to take into account the back reaction of turbulent transport in the time evolution of the equilibrium; and (iii) based on the original *Backward Semi-Lagrangian* (BSL) numerical scheme [SRBG99] –a mix between widespread Particle-In-Cell (PIC) and Eulerian methods– for which we had already shown good properties of energy conservation for a simplified 4D drift-kinetic system [38]. Taking into account all these constraints the Gysela code development started in 2005 [41, 40, 42].

Since the first version, we have continuously improved its numerical scheme and conservation properties [48]. This has been performed in tight collaboration with IRMA[1] (Institut de Recherche Mathématique Avancée) institute from Strasbourg university and more recently with the Numerical Methods in Plasma Physics Division[2] from Max-Planck-Institute für Plasmaphysik at Garching (Germany) from its creation in 2012 onwards. Thanks to this ongoing collaborative effort, an innovative Hermite local spline method [MV07, CLS07, CLS09] has been specifically developed to improve the parallel scalability of the BSL scheme in high-dimensional codes as Gysela . Several variants of the BSL scheme for solving Vlasov-type equations have also been tested. Let us for instance mention the Forward Semi-Lagrangian (FSL) scheme [CRS09] or a Conservative-Semi-Lagrangian (CSL) scheme [CMS10b] based on the Parabolic Spline Method [ZWS02]. Each of these new schemes were tested with simplified Vlasov-Poisson models before being tested in a 4D drift-kinetic model [10, 47].

We have learned from this systematic approach, that efficiency of numerical schemes in solving simplified 2D Vlasov-Poisson systems is a necessary but not sufficient condition for applicability with conserved efficiency in more demanding codes, typically the 4D drift-kinetic. Given this observation, the need for using 4D test cases in developing numerical methods has emerged. A simplified 4D test case has therefore been made available. It is organized with independently tested modules in order to facilitate step by step changes in production codes as Gysela.

The decision to develop a platform dedicated to the solution of 4D Vlasov equations including the 4D drift-kinetic case (see for instance description in [38]) has been an important step in developing Gysela. This work has begun in 2010, in the framework of the 4 years INRIA ADT (Technological Development Actions) SeLaLib[3] (Semi-Lagrangian Library). This software development is now pursued in the context of the INRIA Project Lab FRATRES both with the Tonus INRIA Project Team based at the University of Strasbourg and the Max-Planck-Institut für Plasmaphysik (IPP) in Garching. While originally dedicated to the semi-lagrangian method, the structure of the SeLaLib platform is versatile enough to provide a development and test facility for other types of approaches, such as particle-in-cell. The SeLaLib software library contains now a collection of building blocks for the parallel simulation of the Vlasov equations and the gyrokinetic equation either based on semi-Lagrangian or Lagrangian (PIC) schemes. Besides numerical algorithms the library provides low-level utilities such as input-output modules and parallelization strategies. Moreover, a collection of reference simulations for typical test cases with various discretization schemes supplements the library. A goal for SeLaLib is to provide the appropriate numerical environment such that a building block developed in SeLaLib can be implemented in gyrokinetic codes. This has been achieved with a new gyro-average operator based on Hermite interpolation [CMS10a], [63]. This new operator has been implemented in the Gysela code successfully, namely without performance loss. It replaces the Padé

[1] http://www-irma.u-strasbg.fr/
[2] http://www.ipp.mpg.de/ippcms/eng/for/bereiche/numerik
[3] http://selalib.gforge.inria.fr/

approximation first implemented (see section 2.4 for more details) [53].

Presently, we use the modular SeLaLib library to develop a hybrid method based on a coupling between a semi-Lagrangian scheme and an isogeometric approach. The goal is to find an efficient semi-Lagrangian method to address various magnetic configurations in the GYSELA code, favoring comparison to experiments and providing a tool to investigate alternative geometries. As described in section 2.1.2, in GYSELA a generic circular magnetic configuration is currently considered. Such a circular configuration is relevant for the Tore Supra tokamak, as well as the region in the vicinity of the magnetic axis. It is less the case for WEST and ITER, especially when addressing the outer region of the plasma. Simulations with the ORB5 code, have confirmed the importance of plasma shaping, since elongation is shown to impact turbulence self-organization [4, 5]. Addressing turbulence control and confinement optimization thus requires the capability to change the magnetic geometry. This isogeometric approach had already been tested to solve a Poisson-type equation [CRS12]. An hybrid approach based on analytical mapping has also been proposed during CEMRACS 2010 summer school [3] on simplified 2D problems. Coupling with a more general mapping for the 4D drift-kinetic system is under progress using the SeLaLib framework.

Collisional flux-driven plasma turbulence simulations

Main physical results obtained with GYSELA

The physics of turbulent transport in fusion devices is strongly coupled to experiments. Until the breakthrough of tokamaks in the late 60's, small plasma experiments where quite common in fusion laboratories. Kinetic effects where used in particular to identify the role of turbulence and collisions on phenomena akin to the plasma echo. In the late 70's and early 80's the interest had shifted to understanding transport from a global point of view as measured profiles became available. Transport codes emerged which gradually brought the conviction that heat transport, first for electrons, later for ions, was governed by turbulence. At this stage, understanding turbulent transport was essentially cast in the quasilinear framework, assuming constrained gradients and determining transport properties on the basis of the growth rate and most unstable mode number in the so-called mixing length approximation. During the same years, the discovery of the H-mode in ASDEX triggered a new challenge for the theoretical understanding of turbulent transport. Several experiments and theoretical analysis also revealed the occurrence of fast transport events in the radial direction, unfortunately termed non-local transport. These events clearly indicated a departure of transport from the diffusive process at the crux of the quasilinear predictions. At the same time a growing consensus made its way indicating that large scale flows, self-generated by turbulence played a role on the onset of the H-mode. In the late 90's, our theory team together with a few other teams worldwide challenged turbulence modeling with global codes such that the gradients where the result of transport and of the sources and not prescribed. This flux-driven approach revealed complex turbulence self-organization including the non-linear generation of zonal flows. The decay of the latter, was shown to be closely linked to kinetic effects, which proved to be a powerful drive in investigating turbulent transport in the kinetic framework. This very brief summary sets the scene for the development of GYSELA, with a strong emphasis on heat sources rather than constrained gradients as well as multi-scale physics, leading to self-organization, and the idea that no separation between equilibrium and fluctuations could be introduced a priori. In a first step, GYSELA was developed as a full-f global code as recalled earlier in this introduction.

Regarding the physics, an important step was achieved in 2009, when the forcing governed by thermal baths at the two radial boundaries [40] was replaced by a volumetric heat source [57] (based on the same strategy than the one previously tested in a 3D simplified model developed by Darmet [16]). Actually, the implemented source term in the right hand side of the gyrokinetic equation is versatile enough to allow for separate injection of heat, parallel momentum and vorticity [58]. In such a flux-driven regime, turbulent transport exhibits intermittent large scale events, called avalanches, which ballistic propagation on radial distances much larger than the turbulence correlation length. In the analysis of such transport, it was shown in a specific regime, that the flux could be characterized

by a non-local kernel κ: $Q(r) = -\int \kappa(r,r')\nabla_r T(r')\,\mathrm{d}r'$ [18]. Such dynamics of large scales –already observed in fluid codes [56]– have been intensively studied in gyrokinetic simulations and in particular with the GYSELA code [60, 62, 57, 43, 34]. However, the connection with available confinement scaling laws and predictive capability of this approach has not yet been achieved.

GYSELA also addresses turbulent and neoclassical transports on an equal footing. A linearized Fokker-Planck operator was derived [31] and implemented in GYSELA to account for intra-species collisions. The main results of neoclassical theory are recovered at low (banana) and medium (plateau) collisionalities ν_*, i.e. the values of the heat diffusivity, and the ν_* dependency of the ratio between the poloidal flow and the ion temperature gradient [17]. The theoretical analysis is based on the investigation of the extremum of the entropy production [55, 28, 30] where the Maxwellian distribution function plays a leading role as the distribution belonging to the kernel of both the collision operator, by choice of the collision operator, and of the Vlasov convection operator by considering that the Maxwellian parameters are functions of the Vlasov invariants. Recovering known collisional transport regime in the tokamak geometry is regarded as an important result since this transport could be a major contribution to that of heavy impurities such as tungsten. More recently, a new collision operator has been derived and implemented, which models both intra- *and* inter-species collisions. It is particularly suited for trace impurities having reached thermal equipartition with the main ion species. It has also been benchmarked with asymptotic collisional transport formulas in the tokamak geometry, in the high collisionality regime (Pfirsch-Schlüter) [25]. This development has been required by the recent possibility in GYSELA to evolve a second distribution function. Coupling between the species occurs both via the determination of the electric field and via the quasi neutrality constraint: typically the turbulent branch and the collisions. The two distributions can model two ion species of arbitrary masses and charges (e.g. main ions plus impurities, or deuterium-tritium plasmas). The case of kinetic electrons is currently under development.

The properties of the equilibrium flows have been investigated. The order of magnitude of the flux surface averaged poloidal flow has been found comparable to the one predicted by neoclassical theory. However, its profile exhibits radial corrugations generated by turbulence, such that its gradient noticeably differs from the one based on the sole neoclassical theory [31, 20, 58, 32]. Actually, turbulence is shown to globally organize into a so-called "$E \times B$ staircase", characterized by localized regions of strongly sheared zonal flows associated to permeable transport barriers [18]. Their characteristics weakly depend on key plasma parameters, but the departure from the instability threshold [23]. As far as toroidal angular momentum is concerned, the balance of its constitutive equation is well fulfilled in GYSELA [1], and its transport is found to exhibit avalanche-like events correlated with those observed for the heat flux [44, 2] and [DKG+13]. When a net toroidal spin-up is generated, either via a volume source of momentum or through boundary conditions mimicking the coupling to the scrape-off layer, the energy confinement time is found to degrade with increasing rotation for parallel Mach numbers typically exceeding 20% [54].

Finally, improved transport regimes have been considered [61, 59]. First, a vorticity source allows one to efficiently polarize the system. For sufficiently large shearing rates, turbulent transport is suppressed and a transport barrier builds up. The vorticity source also governs a secondary instability – driven by the temperature anisotropy in this case [65, 66]. Turbulence and its associated zonal flows are generated in the vicinity of the barrier, destroying the latter due to the screening of the polarization source by the zonal flows. These barrier relaxations occur quasi-periodically, and generically result from the decoupling between the dynamics of the barrier generation, triggered by the source driven sheared flow, and that of the crash, triggered by the secondary instability. Second, the heat source can be either thermal or supra-thermal. In the latter case, where fast particles are generated, Energetic Geodesic Acoustic Modes or EGAMs can be efficiently excited [70] and [GZD+14]. Their impact on turbulent transport has been evidenced. In the considered range of parameters, the interaction between EGAMs and turbulence is such that turbulent transport is enhanced in the presence of EGAMs [71, 24]. Third, no transport barrier could be triggered in the vicinity of a minimum of the safety factor q, although the radial extent of the gap without resonant modes was larger than the turbulence correlation length [62].

A strong activity of verification and benchmark

The development of the code has been performed with concern of numerical and physical verification. First of all, one of the difficulty in full-f code is the delicate choice of the initial distribution function. The influence of the initial state on turbulence and transport has been addressed. For two strongly differing initial states, it has been found that the steady turbulent regime exhibits nearly identical statistical properties [19, 22]. Conversely, several simulations with strong flow generation have seemed to lock in states with reduced turbulent transport quite distinct from the expected regimes. The possibility of multiple plasma states for the same values of the control parameters cannot be discarded and long simulations hopefully backed by theoretical predictions will be necessary.

The code has been also benchmarked in the linear and non-linear regimes against other codes as well as against theoretical predictions. At each stage, each time it was possible, comparisons with analytical results have been performed: (i) the linear results of the 4D drift-kinetic version have been validated with the linear dispersion relation results [38]; (ii) neoclassical results have been recovered with our simplified collision operator [21, 17, 31, 29]; (iii) the radial force balance equation has been recovered analytically from the conventional first order gyrokinetic equations [31, 1] and successfully recovered numerically [19, 17, 1]; (iv) local conservation equations for density, energy and toroidal momentum have been derived [1, 2]. Conventional verification tests for gyrokinetic codes have been also successfully reproduced: (i) Rosenbluth-Hinton test [42]; (ii) linear and non-linear benchmarks with the classical Cyclone DIII-D base case [42]. In this verification framework we plan to apply, in the near future, the Projection on Proper elements (PoPe) method [13]. This PoPe method, developed by Th. Cartier-Michaud during his PhD, has already proven its capability to verify kinetic codes of plasma turbulence. It will be of interest to use this method in order to investigate the accuracy of GYSELA, and test alternative numerical schemes and resolution.

All along the improvement of the physics of the code, turbulence analysis have been confronted with other gyrokinetic code results [33, 14, 44, 68, 69]. Benchmarking efforts have been also performed since 2007 within (i) participation to the European turbulence code benchmarks [26]; (ii) non-linear benchmarks with the global PIC code ORB5 [40]; (iii) comparison of flux-driven simulations with ORB5 and XGC1 [58]. This benchmark activity continues in the framework of the european Fusion project led by E. Sonnendrücker (2015-2017) on "Verification and development of new algorithms for gyrokinetic codes" [7]. New non-linear simulations have already been compared with the two European ORB5 and GENE codes [35]. A new collaboration has also recently started with JAEA (Japan Atomic Energy Agency) with the objective to confrontate GYSELA results with the Japanese global full-f flux-driven code GT5D.

Finally, in terms of validation, the code has been confronted to tokamak experiments, qualitatively, but alos quantitatively with a comparison with Tore Supra results [14, 15, 27, 23, 64].

GYSELA a massively parallel code preparing for Exascale

The code is developed in Fortran90 with some I/O routines in C. The parallelization is based on an hybrid MPI/OpenMP paradigm since 2007 [46]. GYSELA was one of the pioneer in such parallelization approach, now commonly used to take advantage of the current supercomputer architectures based on a cluster of SMP (symmetric shared memory multiprocessor) nodes. Since 2009 a large effort has been dedicated to improving the parallelization efficiency. For detailed presentation regarding the different optimization steps, we encourage the reader to refer to INRIA reports [49, 45] and papers [50, 48] by Latu et al. as Rozar's paper [53]. The difficulties in developing a code scalable up to several thousands of cores are numerous and call for trade-offs and a close match between some aspects of the code parallelization strategy and the computer architecture. One issue that is seldom considered is the difficulty to access sufficiently large machines. As shown in Figure 1, we have tried to take each opportunity of "Grand Challenge" campaigns –during each installation of new HPCs– to test and improve the parallelization of the code. GYSELA exhibits now an excellent scalability (91% of relative efficiency at 458752 cores) using oll the available computing power of JUQUEEN Blue Gene supercomputer (JSC/IAS, Juelich, Germany) [9]. With such performance, GYSELA is since 2013 a

member of the Hi Q club [4] (Highest Scaling Codes on JUQUEEN) which has been created to promote the idea of exascale capability computing. Presently 23 codes coming from all scientific domains belong to this club. One objective for the GYSELA team is to have access to a bigger machine than JUQUEEN for scalability tests beyond the present 500k core limit.

	Relative efficiency		Number of cores
	Weak scaling	Strong scaling	
Gd Challenge CINES (march 2010)	92 %	82 %	8192
Gd Challenge CURIE (march 2012)	91 %	61 %	65 536
Porting on Blue Gene Architecture => Communication schemes rewritten			
Gd Challenge TURING (january 2013)	92 %	61 %	65 536
Access to totality of JUQUEEN (may 2013)	91 %		458 752

x56

Figure 1: Relative efficiency of the GYSELA code for weak and strong scaling on two bullx machines JADE (CINES, Montpellier, France) and CURIE (TGCC, Saclay, France) and on two Blue Gene supercomputers TURING (IDRIS, Orsay, France) and JUQUEEN (JSC/IAS, Juelich, Germany).

The code is already using intensively Petascale capabilities with production run of 8k cores to 16k cores during several days or weeks. As shown on Figure 2, the use of CPU time has increased by a factor 10 in 6 years, thus leading to a CPU time allocation of about 50 millions of CPU hours in 2016. Obtaining this computing time is another challenge for the team. Every year, this requires submitting applications, all different of course, to be granted the needed CPU time by different HPC facilities. This year, the hours are distributed between the Tier1 Occigen machine at CINES[5], Montpellier, France and the two Tier0 machines : Curie at TGCC[6], Bruyères-le-Châtel, France and Helios at CSC[7], Rokkasho, Japan. Of course the target HPC platforms changes from year to year depending on the opportunities and availability. Since the development code is mainly performed on the Tier2

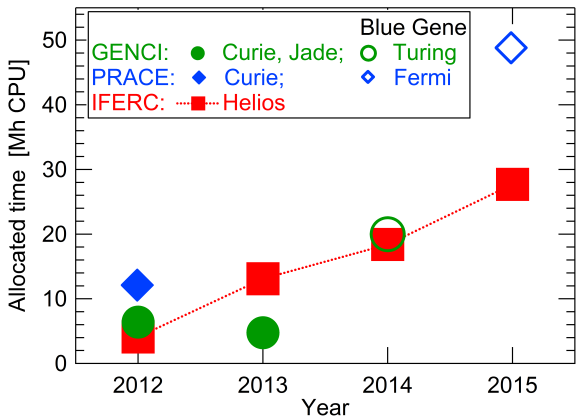


Figure 2: CPU time allocations for GYSELA on the different accessible supercomputers since 2012.

machine Poincaré at IDRIS[8], Orsay, France, one faces the fact that the code is always running on

[4] http://www.fz-juelich.de/ias/jsc/EN/Expertise/High-Q-Club/_node.html
[5] https://www.cines.fr/
[6] http://www-hpc.cea.fr/en/complexe/tgcc.htm
[7] http://www.iferc.org/CSC_Scope.html
[8] http://www.idris.fr/

several HPC platforms at the same time (with little possibility of continuing a run elsewhere than where started). To cope with these constraints, the GYSELA code has been developed with portability in mind. This means that parallelization optimization specific to one machine architecture has been avoided as much as possible. With such strategy GYSELA has already been used on many different HPC architectures: (i) Harpertown / Nehalem INTEL (SGI Altix ICE), (ii) AMD OPTERON (PC cluster + CRAY-XT5), (iii) Intel Itanium2 quadri-proc/dual-core (PC cluster + BULL Novascale), (iv) Power 5 (IBM machine), (v) Nehalem Intel (BULL Novascale) ; the list being far from complete and will most likely grow steadily with years. To facilitate the development and the deployment of the code, Cmake build-system is used for compiling. Code versioning is ensured via the distributed version control system GIT. A complete work-flow relies on quality procedures including human code review and automated testing. The latter has been designed via the tracking system GitLab hosted by Maison de la Simulation[9] and the JENKINS continuous integration tool provided by CI-INRIA[10]. A detailed description of this work-flow is given in Bigot et al. [8].
Despite using petascale computing resources, trade-off between the size of the fusion device and simulation up to energy confinement time must be found. Examples of three large simulations performed since 2010 are presented in Figure 3. For each one, we have tried to stress one of the three main simulation parameters, namely: (i) the mesh size –with a huge mesh of 272 billions of points [2]–; (ii) a long time simulation –for comparison with Tore Supra experiments [23] (46 days on 5520 cores to simulate approximately 1 energy confinement time)– and (iii) a simulation with impurities which has succeed in running 15 days on $16k$ cores followed by 6 days on $32k$ cores after adding a second species, namely tritium. These three simulations –corresponding to 6 million CPU hours for the first and the third one and to 10.6 million CPU hours for the second one– are already extremely challenging. They have all led to advances in the physics of plasma turbulence. A "kitchen sink" extrapolation indicates that a simulation for an ITER grade device and a duration of one energy confinement time, and for two species, would require more than 50 days on $32k$ cores, typically 39 million CPU hours and therefore roughly one year of the present computation budget (combining the allocation time that is effectively spread on different HPCs).

	Number of Points (ρ*=ρ/a)	Time / Ω_c	Number of cores	Number of days of simulation
Gd Challenge CINES 2010	272 billions (ρ*=1/512)	147 840	8192	31
Gd Challenge CURIE 2012	33 billions (ρ*=1/150)	678 510	16 384	15
	=> Adding of tritium		32768	6
Comparison with experiment (in progress)	87 billions (ρ*=1/300)	2 000 000	5520	46

Figure 3: Some numbers for four challenging simulations performed with GYSELA: the first one with a big mesh of 272 billion of points for a ITER-like $\rho_* = 1/512$ (where $\rho_* = \rho_s/a$ with ρ_s the Larmor radius of species s and a the small radius of the tokamak), the second and third one which have run several days on more than 16k cores and the last one which correspond to a long time simulation.

This is to be compared to typical GYSELA simulations at present: these are designed to run with a control parameter $\rho_* = 1/150$ ($\sim$ 33 billions of points for one distribution function), a factor 2 in ρ_* and 8 in number of grid points compared to ITER values, and using 4096 to 8192 cores for a couple of weeks.

[9] https://gitlab.maisondelasimulation.fr/
[10] https://ci.inria.fr/

Stepping kinetic electron is altogether another story. If you consider that the ratio of Larmor radii between ions (deuterium) and electrons is about 60 (scaling like the square root of the mass ratio), one should increase the mesh size by a factor 60^3 and decrease the time step by a factor 60. Therefore, the road is still long for a global flux-driven simulation with both kinetic ions and electrons for ITER parameters. It will not be possible without exascale resources. It will also be very dependent regarding innovative physical frameworks and well adapted as well as more efficient numerical schemes. Regarding this aspect, aligned coordinates –based on a flux-coordinate independent field-aligned approach [Ott11, HO13]– have been recently implemented with the aim of decreasing the number of points required in the toroidal directions. Regarding the physics associated to the electrons, the first step is to investigate the impact of the mass ratio on turbulence. First GYSELA simulations with kinetic electrons are planned with "heavy" electrons ($m_{\text{ion}}/m_{\text{electron}} \sim 400$ instead of ~ 1836). A second approach, where filtering to retain the "slowest" electrons (trapped electrons + part of the passing electrons) as kinetic and the other part being assumed adiabatic, is being developed.

Finally, a computer science effort has been initiated to prepare the code to future exascale machines. The goal is to obtain an efficient parallelization on million of cores. The expertise gained with the GYSELA scalability test up to $500k$ cores on a BlueGene machine has helped identifying several bottlenecks to achieve efficient exascale simulations. Some of them are inherent to GYSELA, such as the requirement for large memory per node, but others are more common to all applications, hence: (i) the probability of crashes, which will inevitably increases with the increase of number of cores; (ii) the choice for exascale architectures, which are still not clear (MIC, GPU or something else?); and (iii) all the questions linked to handling large amounts of data (several hundreds of TBytes for GYSELA), namely data transfer, and data analyses and multidimensional visualization. If options are still open regarding the type of processors that will be used in exascale HPCs, there is a trend towards less memory per node. A strong effort has already been performed during F. Rozar's PhD to improve GYSELA memory scalability [51, 52], but further effort towards increasing the parallel domain decomposition is mandatory to further reduce the memory requirement per node. In the course of preparing GYSELA for future architectures, a simplified prototype (with the main mathematical kernels of GYSELA) has been designed and tested on Xeon-Phi and GPU accelerators [6]. First results show that porting codes on such architectures is not trivial and will require significant rewriting to achieve meaningful performance upgrade. This represents lots of manpower specialized in high performance computing. Regarding fault tolerance, some work has been initiated: (i) with the development of asynchronous writing of the restart files during O. Thomine's post-doc. [67] and (ii) with the implementation of FTI library [BGTK+11] in GYSELA. Both are still under test in the code and efforts will be pursued on this issue in the framework of EoCoE european project[11].

This manuscript has been written keeping in mind that is should prove to be helpful for students and collaborators interested in GYSELA results and running GYSELA simulations. In that spirit, I hope it can be seen as a brief introduction to gyrokinetic plasma turbulence for mathematicians and computer scientists, as well as an introduction to Semi-Lagrangian computing for physicists. In both cases, it would stand as a user's guide of the GYSELA code. In practice, this document is an extension of the reference article for the 5D version of the code [36], completed whenever necessary. It is organized as follows:

- Chapter 1 aims at introducing magnetic confinement in fusion devices. We recall that for strongly magnetized plasmas, the six dimensional evolution equation for the distribution function – Vlasov or Fokker-Planck equations – is solved for each species and coupled to the self-consistent equations for electromagnetic fields, namely Maxwell's equations. We briefly describe the gyrokinetic ordering and the so-called *gyrokinetic* framework *model* following Brizard and Hahm [BH07].
- Chapter 2, is dedicated to the *global full-f flux-driven* code GYSELA including collisions and source terms.

[11] http://www.maisondelasimulation.fr/projects/EoCoE.php

- Chapter 3 is divided into two parts: (i) a first one addresses the numerical description of the code based on the Semi-Lagrangian approach and (ii) the second one is focused on the hybrid MPI/OpenMP parallelization of the code.
- Chapter 4 summarizes the systematic effort for verification and benchmarking performed at each step the code development.

Whenever suitable, technical parts describing precisely what is implemented in the code are presented in Appendices. At the end of this document, the reader can also find a copy of the article published in Société Mathématique de France (SMF) journal "Panoramas et synthèses" [39]. This article stems from the lecture "Gyrokinetic simulations of magnetic fusion plasmas" I gave during CEMRACS 2010 (Centre d'Eté Mathématique de Recherche Avancée en Calcul Scientifique) summer school, dedicated this year to numerical models for fusion. A description of the Particle-In-Cell and Eulerian methods, with a discussion on strengths and weaknesses of the three approaches, is presented there, completing the description of the various numerical schemes used in non-linear gyrokinetic simulations. In comparison, the specific properties of the semi-Lagrangian approach will be found there.

Publications

[1] J. Abiteboul, X. Garbet, V. Grandgirard, S. J. Allfrey, Ph. Ghendrih, G. Latu, Y. Sarazin, and A. Strugarek. Conservation equations and calculation of mean flows in gyrokinetics. *Phys. Plasmas*, 18(8), 2011.

[2] J. Abiteboul, Ph. Ghendrih, V. Grandgirard, T. Cartier-Michaud, G. Dif-Pradalier, X. Garbet, G. Latu, Ch. Passeron, Y. Sarazin, A. Strugarek, O. Thomine, and D. Zarzoso. Turbulent momentum transport in core tokamak plasmas and penetration of scrape-off layer flows. *Plasma Physics and Controlled Fusion*, 55(7):074001, 2013.

[3] J. Abiteboul, G. Latu, V. Grandgirard, A. Ratnani, Sonnendrücker, and A. Strugarek. Solving the Vlasov equation in complex geometries. In *CEMRACS 2010*, volume 32 of *ESAIM: Proc.*, pages 103–117, Luminy, France, 2011.

[4] P. Angelino, X. Garbet, L. Villard, A. Bottino, S. Jolliet, Ph. Ghendrih, V. Grandgirard, B. F. McMillan, Y. Sarazin, G. Dif-Pradalier, and T. M. Tran. The role of plasma elongation on the linear damping of zonal flows. *Phys. Plasmas*, 15(6), 2008.

[5] P. Angelino, X. Garbet, L. Villard, A. Bottino, S. Jolliet, Ph. Ghendrih, V. Grandgirard, B. F. McMillan, Y. Sarazin, G. Dif-Pradalier, and T. M. Tran. Role of Plasma Elongation on Turbulent Transport in Magnetically Confined Plasmas. *Phys. Rev. Lett.*, 102:195002, May 2009.

[6] Y. Asahi, G. Latu, T. Ina, Y. Idomura, V. Grandgirard, and X. Garbet. Optimization of fusion kernels on accelerators with indirect or stride memory access patterns. *submitted to IEEE Trans*, 2016.

[7] A. Biancalani, G. Merlo, and V. Grandgirard. Gyrokinetic investigation of the linear gam dynamics. *Phys. Plasmas (in preparation)*, 2016.

[8] J. Bigot, T Cartier-Michaud, V. Grandgirard, G. Latu, Ch. Passeron, and F. Rozar. An approach to increase reliability of HPC simulation, application to the GYSELA5D code. In *CEMRACS 2014*, volume submitted of *ESAIM: Proc.*, Luminy, France, 2015.

[9] J. Bigot, V. Grandgirard, G. Latu, Ch. Passeron, F. Rozar, and O. Thomine. Scaling gysela code beyond 32K-cores on bluegene/Q. In *CEMRACS 2012*, volume 43 of *ESAIM: Proc.*, pages 117–135, Luminy, France, 2013.

[10] J-Ph. Braeunig, N. Crouseilles, V. Grandgirard, G. Latu, M. Mehrenberger, and E. Sonnendrücker. Some numerical aspects of the conservative PSM scheme in a 4D drift-kinetic code. Inria research report, Dec 2011.

[11] M. Brunetti, V. Grandgirard, P. Bertrand, O. Sauter, J. Vaclavik, and L. Villard. Fine scale structures and negative density regions: Comparison of numerical methods for solving the advection equation. *Transport Theory and Statistical Physics*, 34(3-5):261–274, 2005.

[12] M. Brunetti, V. Grandgirard, O. Sauter, J. Vaclavik, and L. Villard. A semi-lagrangian code for nonlinear global simulations of electrostatic drift-kinetic ITG modes. *Computer Physics Communications*, 163(1):1 – 21, 2004.

[13] T. Cartier-Michaud, P. Ghendrih, Y. Sarazin, J. Abiteboul, H. Bufferand, G. Dif-Pradalier, X. Garbet, V. Grandgirard, G. Latu, C. Norscini, C. Passeron, and P. Tamain. Projection on proper elements for code control: Verification, numerical convergence, and reduced models. application to plasma turbulence simulations. *Physics of Plasmas*, 23(2), 2016.

[14] A. Casati, C. Bourdelle, X. Garbet, F. Imbeaux, J. Candy, F. Clairet, G. Dif-Pradalier, G. Falchetto, T. Gerbaud, V. Grandgirard, Ö.D. Gürcan, P. Hennequin, J. Kinsey, M. Ottaviani, R. Sabot, Y. Sarazin, L. Vermare, and R.E. Waltz. Validating a quasi-linear transport model versus nonlinear simulations. *Nuclear Fusion*, 49(8):085012, 2009.

[15] A. Casati, T. Gerbaud, P. Hennequin, C. Bourdelle, J. Candy, F. Clairet, X. Garbet, V. Grandgirard, Ö. D. Gürcan, S. Heuraux, G. T. Hoang, C. Honoré, F. Imbeaux, R. Sabot, Y. Sarazin, L. Vermare, and R. E. Waltz. Turbulence in the TORE SUPRA tokamak: Measurements and validation of nonlinear simulations. *Phys. Rev. Lett.*, 102:165005, Apr 2009.

[16] G. Darmet, Ph. Ghendrih, Y. Sarazin, X. Garbet, and V. Grandgirard. Intermittency in flux driven kinetic simulations of trapped ion turbulence. *Communications in Nonlinear Science and Numerical Simulation*, 13(1):53 – 58, 2008. Vlasovia 2006: The Second International Workshop on the Theory and Applications of the Vlasov Equation.

[17] G. Dif-Pradalier, P. H. Diamond, V. Grandgirard, Y. Sarazin, J. Abiteboul, X. Garbet, Ph. Ghendrih, G. Latu, A. Strugarek, S. Ku, and C. S. Chang. Neoclassical physics in full distribution function gyrokinetics. *Phys. Plasmas*, 18(6), 2011.

[18] G. Dif-Pradalier, P. H. Diamond, V. Grandgirard, Y. Sarazin, J. Abiteboul, X. Garbet, Ph. Ghendrih, A. Strugarek, S. Ku, and C. S. Chang. On the validity of the local diffusive paradigm in turbulent plasma transport. *Phys. Rev. E*, 82:025401, Aug 2010.

[19] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin, X. Garbet, and Ph. Ghendrih. Defining an equilibrium state in global full-f gyrokinetic models. *Communications in Nonlinear Science and Numerical Simulation*, 13(1):65 – 71, 2008. Vlasovia 2006: The Second International Workshop on the Theory and Applications of the Vlasov Equation.

[20] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin, X. Garbet, and Ph. Ghendrih. Interplay between gyrokinetic turbulence, flows, and collisions: Perspectives on transport and poloidal rotation. *Phys. Rev. Lett.*, 103:065002, Aug 2009.

[21] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin, X. Garbet, Ph. Ghendrih, and P. Angelino. Gyrokinetic simulations of neoclassical transport using a minimal collision operator. *AIP Conference Proceedings*, 1069(1):53–63, 2008.

[22] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin, X. Garbet, Ph. Ghendrih, and P. Angelino. On the influence of initial state on gyrokinetic simulations. *Phys. Plasmas*, 15(4), 2008.

[23] G. Dif-Pradalier, G. Hornung, Ph. Ghendrih, Y. Sarazin, F. Clairet, L. Vermare, P. H. Diamond, J. Abiteboul, T. Cartier-Michaud, C. Ehrlacher, D. Esteve, X. Garbet, V. Grandgirard, Ö.D. Gürcan, P. Hennecquin, Y. Kosuga, G. Latu, P. Maget, P. Morel, C. Norscini, R. Sabot, and A. Storelli. Finding the elusive ExB staircase in magnetised plasmas. *Phys. Rev. Lett.*, 114:085004, 2015.

[24] R. J. Dumont, D. Zarzoso, Y. Sarazin, X. Garbet, A. Strugarek, J. Abiteboul, T. Cartier-Michaud, G. Dif-Pradalier, Ph. Ghendrih, J-B. Girardo, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Interplay between fast ions and turbulence in magnetic fusion plasmas. *Plasma Physics and Controlled Fusion*, 55(12):124012, 2013.

[25] D. Estève, X. Garbet, Y. Sarazin, V. Grandgirard, T. Cartier-Michaud, G. Dif-Pradalier, P. Ghendrih, G.Latu, and C. Norscini. A multi-species collisional operator for full-f gyrokinetics. *Phys. Plasmas*, (22):122506, 2015.

[26] G. L. Falchetto, B. D. Scott, P. Angelino, A. Bottino, T. Dannert, V. Grandgirard, S. Janhunen, F. Jenko, S. Jolliet, A. Kendl, B. F. McMillan, V. Naulin, A. H. Nielsen, M. Ottaviani, A. G. Peeters, M. J. Pueschel, D. Reiser, T. T. Ribeiro, and M. Romanelli. The European turbulence code benchmarking effort: turbulence driven by thermal gradients in magnetically confined plasmas. *Plasma Physics and Controlled Fusion*, 50(12):124015, 2008.

[27] X. Garbet, J. Abiteboul, S. Benkadda, C. Bourdelle, A. Casati, F. Clairet, N. Dubuit, G. Falchetto, C. Fenzi, S. Futatani, V. Grandgirard, R. Guirlet, Ö. D. Gürcan, P. Hennequin, S. Heuraux, G. T. Hoang, C. Honoré, R. Sabot, Y. Sarazin, J. L. Segui, A. Smolyakov, E. Trier, L. Vermare, and D. Villegas. Measurements and modeling of turbulent transport in the Tore Supra tokamak. *AIP Conference Proceedings*, 1308(1):75–84, 2010.

[28] X. Garbet, J. Abiteboul, Y. Sarazin, A. Smolyakov, S. Allfrey, V. Grandgirard, Ph. Ghendrih, G. Latu, and A. Strugarek. Entropy production rate in tokamak plasmas with helical magnetic perturbations. *Journal of Physics: Conference Series*, 260(1):012010, 2010.

[29] X. Garbet, J. Abiteboul, A. Strugarek, Y. Sarazin, G. Dif-Pradalier, Ph. Ghendrih, V. Grandgirard, C. Bourdelle, G. Latu, and A. Smolyakov. Thermodynamics of neoclassical and turbulent transport. *Plasma Physics and Controlled Fusion*, 54(5):055007, 2012.

[30] X. Garbet, J. Abiteboul, E. Trier, Ö. Gürcan, Y. Sarazin, A. Smolyakov, S. Allfrey, C. Bourdelle, C. Fenzi, V. Grandgirard, P. Ghendrih, and P. Hennequin. Entropy production rate in tokamaks with nonaxisymmetric magnetic fields. *Phys. Plasmas*, 17(7), 2010.

[31] X. Garbet, G. Dif-Pradalier, C. Nguyen, Y. Sarazin, V. Grandgirard, and Ph. Ghendrih. Neoclassical equilibrium in gyrokinetic simulations. *Phys. Plasmas*, 16(6), 2009.

[32] X. Garbet, D. Esteve, Y. Sarazin, J. Abiteboul, C. Bourdelle, G. Dif-Pradalier, P. Ghendrih, V. Grandgirard, G. Latu, and A. Smolyakov. Turbulent acceleration and heating in toroidal magnetized plasmas. *Phys. Plasmas*, 20(7), 2013.

[33] X. Garbet, Y. Sarazin, V. Grandgirard, G. Dif-Pradalier, G. Darmet, Ph. Ghendrih, P. Angelino, P. Bertrand, N. Besse, E. Gravier, P. Morel, E. Sonnendrücker, N. Crouseilles, J.-M. Dischler, G. Latu, E. Violard, M. Brunetti, S. Brunner, X. Lapillonne, T.-M. Tran, L. Villard, and M. Boulet. Beyond scale separation in gyrokinetic turbulence. *Nuclear Fusion*, 47(9):1206, 2007.

[34] Ph. Ghendrih, C. Norscini, Th. Cartier-Michaud, G. Dif-Pradalier, J. Abiteboul, Y. Dong, X. Garbet, O. Gürcan, P. Hennequin, V. Grandgirard, G. Latu, P. Morel, Y. Sarazin, A. Storelli, and L. Vermare. Phase space structures in gyrokinetic simulations of fusion plasma turbulence. *The European Physical Journal D*, 68(10):303, 2014.

[35] T. Görler, N. Tronko, W.A Hornsby, R. Kleiber, C. Norscini, V. Grandgirard, F. Jenko, and E. Sonnendrücker. Intercode comparison of gyrokinetic global electromagnetic modes. *submitted*, 2016.

[36] V. Grandgirard, J. Abiteboul, J. Bigot, T. Cartier-Michaud, N. Crouseilles, G. Dif-Pradalier, Ch. Ehrlacher, D. Esteve, X. Garbet, Ph. Ghendrih, G. Latu, M. Mehrenberger, C. Norscini, Ch. Passeron, F. Rozar, Y. Sarazin, E. Sonnendrücker, A. Strugarek, and D. Zarzoso. A 5D gyrokinetic full-f global semi-lagrangian code for flux-driven ion turbulence simulations. *Comp. Phys. Comm*, 207:35–68, 2016.

[37] V. Grandgirard, M. Brunetti, S. Allfrey, A. Bottina, P. Bertrand, X. Garbet, Ph. Ghendrih, A. Ghizzo, G. Manfredi, M. Ottaviani, Y. Sarazin, O. Sauter, J. Vaclavik, and L. Villard. Semi-lagrangian drift-kinetic code for slab-itg turbulence. *29th EPS Conference on Plasma Phys. and Contr. Fusion Montreux*, 26B:4.095, 2002.

[38] V. Grandgirard, M. Brunetti, P. Bertrand, N. Besse, X. Garbet, P. Ghendrih, G. Manfredi, Y. Sarazin, O. Sauter, E. Sonnendrücker, J. Vaclavik, and L. Villard. A drift-kinetic semi-lagrangian 4D code for ion turbulence simulation. *Journal of Computational Physics*, 217(2):395 – 423, 2006.

[39] V. Grandgirard and Y. Sarazin. Gyrokinetic simulations of magnetic fusion plasmas. *Panoramas et synthèses*, (39-40):91–176, 2013.

[40] V. Grandgirard, Y. Sarazin, P. Angelino, A. Bottino, N. Crouseilles, G. Darmet, G. Dif-Pradalier, X. Garbet, Ph. Ghendrih, S. Jolliet, G. Latu, E. Sonnendrücker, and L. Villard. Global full-f gyrokinetic simulations of plasma turbulence. *Plasma Physics and Controlled Fusion*, 49(12B):B173, 2007.

[41] V. Grandgirard, Y. Sarazin, X. Garbet, G. Dif-Pradalier, Ph. Ghendrih, N. Crouseilles, G. Latu, E. Sonnendrücker, N. Besse, and P. Bertrand. GYSELA, a full-f global gyrokinetic semi-lagrangian code for ITG turbulence simulations. *AIP Conference Proceedings*, 871(1):100–111, 2006.

[42] V. Grandgirard, Y. Sarazin, X. Garbet, G. Dif-Pradalier, Ph. Ghendrih, N. Crouseilles, G. Latu, E. Sonnendrücker, N. Besse, and P. Bertrand. Computing ITG turbulence with a full-f semi-lagrangian code. *Communications in Nonlinear Science and Numerical Simulation*, 13(1):81 – 87, 2008. Vlasovia 2006: The Second International Workshop on the Theory and Applications of the Vlasov Equation.

[43] Ö.D. Gürcan, L. Vermare, P. Hennequin, V. Berionni, P.H. Diamond, G. Dif-Pradalier, X. Garbet, P. Ghendrih, V. Grandgirard, C.J. McDevitt, P. Morel, Y. Sarazin, A. Storelli, C. Bourdelle, and the Tore Supra Team. Structure of nonlocality of plasma turbulence. *Nuclear Fusion*, 53(7):073029, 2013.

[44] S. Ku, J. Abiteboul, P.H. Diamond, G. Dif-Pradalier, J.M. Kwon, Y. Sarazin, T.S. Hahm, X. Garbet, C.S. Chang, G. Latu, E.S. Yoon, Ph. Ghendrih, S. Yi, A. Strugarek, W. Solomon, and V. Grandgirard. Physics of intrinsic rotation in flux-driven ITG turbulence. *Nuclear Fusion*, 52(6):063013, 2012.

[45] G. Latu, N. Crouseilles, and V. Grandgirard. Parallel bottleneck in the Quasineutrality solver embedded in GYSELA. INRIA Research Report RR-7595, April 2011.

[46] G. Latu, N. Crouseilles, V. Grandgirard, and E. Sonnendrücker. Gyrokinetic semi-lagrangian parallel simulation using a hybrid OpenMP/MPI programming. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, volume 4757 of *Lecture Notes in Computer Science*, pages 356–364, Paris, France, 2007. Springer.

[47] G. Latu, V. Grandgirard, J. Abiteboul, M. Bergot, N. Crouseilles, X. Garbet, Ph. Ghendrih, M. Mehrenberger, Y. Sarazin, H. Sellama, E. Sonnendrücker, and D. Zarzoso. Accuracy of unperturbed motion of particles in a gyrokinetic semi-lagrangian code. INRIA Research Report RR-8054, Sep 2012.

[48] G. Latu, V. Grandgirard, J. Abiteboul, N. Crouseilles, G. Dif-Pradalier, X. Garbet, P. Ghendrih, M. Mehrenberger, E. Sonnendrücker, and Y. Sarazin. Improving conservation properties of a 5D gyrokinetic semi-lagrangian code. *The European Physical Journal D*, 68(11), 2014.

[49] G. Latu, V. Grandgirard, N. Crouseilles, R. Belaouar, and E. Sonnendrücker. Some parallel algorithms for the Quasineutrality solver of GYSELA. INRIA Research Report RR-7591, April 2011.

[50] G. Latu, V. Grandgirard, N. Crouseilles, and G. Dif-Pradalier. Scalable Quasineutral solver for gyrokinetic simulation. Research Report RR-7611, INRIA, May 2012.

[51] F. Rozar, G. Latu, and J Roman. Achieving memory scalability in the GYSELA code to fit exascale constraints. *Concurrency and computation: Practice and Experience*, 2014.

[52] F. Rozar, G. Latu, J. Roman, and V. Grandgirard. Toward memory scalability of GYSELA code for extreme scale computers. *Concurrency and computation: Practice and Experience*, 27(4):994–1009, 2015.

[53] F. Rozar, G. Steiner, Ch. Latu, M. Mehrenberger, V. Grandgirard, J. Bigot, T. Cartier-Michaud, and J. Roman. Optimization of the gyroaverage operator based on Hermite interpolation. In *CEMRACS 2014*, volume submitted of *ESAIM: Proc.*, Luminy, France, 2015.

[54] Y. Sarazin, J. Abiteboul, G. Dif-Pradalier, B. McMillan, C. Bourdelle, T. Cartier-Michaud, P. Cottier, D. Esteve, X. Garbet, J.-B. Girardo, V. Grandgirard, P. Ghendrih, F. Hariri, G. Latu, D. Newman, C. Norscini, C. Passeron, J. Reynolds-Barredo, R. Sanchez, F. Spineanu, M. Vlad, and L. Villard. Understanding momentum transport in tokamak plasmas. *IAEA, Fusion Energy Conference, Saint Petersburg*, (TH/P4-4), 2014.

[55] Y. Sarazin, G. Dif-Pradalier, D. Zarzoso, X. Garbet, Ph. Ghendrih, and V. Grandgirard. Entropy production and collisionless fluid closure. *Plasma Physics and Controlled Fusion*, 51(11):115003, 2009.

[56] Y. Sarazin, Ph. Ghendrih, S. Benkadda, P. Beyer, G. Falchetto, C. Figarella, X. Garbet, and V. Grandgirard. Impact of large scale flows on turbulent transport. *AIP Conference Proceedings*, 669(1):696–699, 2003.

[57] Y. Sarazin, V. Grandgirard, J. Abiteboul, S. Allfrey, X. Garbet, Ph. Ghendrih, G. Latu, A. Strugarek, and G. Dif-Pradalier. Large scale dynamics in flux driven gyrokinetic turbulence. *Nuclear Fusion*, 50(5):054004, 2010.

[58] Y. Sarazin, V. Grandgirard, J. Abiteboul, S. Allfrey, X. Garbet, Ph. Ghendrih, G. Latu, A. Strugarek, G. Dif-Pradalier, P.H. Diamond, S. Ku, C.S. Chang, B.F. McMillan, T.M. Tran, L. Villard, S. Jolliet, A. Bottino, and P. Angelino. Predictions on heat transport and plasma rotation from global gyrokinetic simulations. *Nuclear Fusion*, 51(10):103023, 2011.

[59] Y. Sarazin, V. Grandgirard, P. Angelino, A. Casati, G. Dif-Pradalier, X. Garbet, Ph. Ghendrih, O. Gürcan, P. Hennequin, and R. Sabot. Turbulence spectra and transport barriers in gyrokinetic simulations. *AIP Conference Proceedings*, 1069(1):325–330, 2008.

[60] Y Sarazin, V Grandgirard, G Dif-Pradalier, E Fleurence, X Garbet, Ph Ghendrih, P Bertrand, N Besse, N Crouseilles, E Sonnendrücker, G Latu, and E Violard. Impact of large scale flows on turbulent transport. *Plasma Physics and Controlled Fusion*, 48(12B):B179, 2006.

[61] Y. Sarazin, V. Grandgirard, G. Dif-Pradalier, X. Garbet, and Ph. Ghendrih. Interplay between transport barriers and density gradient. *Phys. Plasmas*, 13(9), 2006.

[62] Y. Sarazin, A. Strugarek, G. Dif-Pradalier, J. Abiteboul, S. Allfrey, X. Garbet, Ph. Ghendrih, V. Grandgirard, and G. Latu. Flux-driven gyrokinetic simulations of ion turbulent transport at low magnetic shear. *Journal of Physics: Conference Series*, 260(1):012017, 2010.

[63] Ch. Steiner, M. Mehrenberger, N. Crouseilles, V. Grandgirard, G. Latu, and F Rozar. Gyroaverage operator for a polar mesh. *The European Physical Journal D*, 69(1):18, 2015.

[64] A. Storelli, L. Vermare, P. Hennequin, Ö. Gürcan, G. Dif-Pradalier, Y. Sarazin, X. Garbet, T. Görler, R. Singh, P. Morel, V. Grandgirard, Ph. Ghendrih, and the Tore Supra team. Comprehensive comparisons of gam characteristics and dynamics between tore supra experiments and gyrokinetic simulations. *Phys. Plasmas*, 22(6), 2015.

[65] A. Strugarek, Y. Sarazin, D. Zarzoso, J. Abiteboul, A. S. Brun, T. Cartier-Michaud, G. Dif-Pradalier, X. Garbet, Ph. Ghendrih, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Ion transport barriers triggered by plasma polarization in gyrokinetic simulations. *Plasma Physics and Controlled Fusion*, 55(7):074013, 2013.

[66] A. Strugarek, Y. Sarazin, D. Zarzoso, J. Abiteboul, A. S. Brun, T. Cartier-Michaud, G. Dif-Pradalier, X. Garbet, Ph. Ghendrih, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Unraveling quasiperiodic relaxations of transport barriers with gyrokinetic simulations of tokamak plasmas. *Phys. Rev. Lett.*, 111:145001, Oct 2013.

[67] O. Thomine, J. Bigot, V. Grandgirard, G. Latu, Ch. Passeron, and F. Rozar. An asynchronous writing method for restart files in the GYSELA code in prevision of exascale systems. In *CEMRACS 2012*, volume 43 of *ESAIM: Proc.*, pages 108–116, Luminy, France, 2013.

[68] L. Villard, S.J. Allfrey, A. Bottino, M. Brunetti, G.L. Falchetto, V. Grandgirard, R. Hatzky, J. Nührenberg, A.G. Peeters, O. Sauter, S. Sorge, and J. Vaclavik. Full radius linear and nonlinear gyrokinetic simulations for tokamaks and stellarators: zonal flows, applied ExB flows, trapped electrons and finite beta. *Nuclear Fusion*, 44(1):172, 2004.

[69] L. Villard, A. Bottino, S. Brunner, A. Casati, J. Chowdhury, T. Dannert, R. Ganesh, X. Garbet, T. Görler, V. Grandgirard, R. Hatzky, Y. Idomura, F. Jenko, S. Jolliet, S. Khosh Aghdam, X. Lapillonne, G. Latu, B. F. McMillan, F. Merz, Y. Sarazin, T. M. Tran, and T. Vernay. Gyrokinetic simulations of turbulent transport: size scaling and chaotic behaviour. *Plasma Physics and Controlled Fusion*, 52(12):124038, 2010.

[70] D. Zarzoso, X. Garbet, Y. Sarazin, R. Dumont, and V. Grandgirard. Fully kinetic description of the linear excitation and nonlinear saturation of fast-ion-driven geodesic acoustic mode instability. *Phys. Plasmas*, 19(2), 2012.

[71] D. Zarzoso, Y. Sarazin, X. Garbet, R. Dumont, A. Strugarek, J. Abiteboul, T. Cartier-Michaud, G. Dif-Pradalier, Ph. Ghendrih, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Impact of Energetic-Particle-Driven Geodesic Acoustic Modes on turbulence. *Phys. Rev. Lett.*, 110:125002, Mar 2013.

Chapter 1

Gyrokinetic model for plasma turbulence

Contents

1.1 Brief introduction to plasma fusion **19**
1.2 Theoretical Hierarchy of Plasma Physics **21**
1.3 Plasma kinetic theory **23**
1.3.1 Liouville theorem 23
1.3.2 Vlasov-Maxwell system 24
1.3.3 Advective and conservative forms of Vlasov equation 25
1.3.4 Conservation properties of the Vlasov-Maxwell system 26
1.4 Gyrokinetic Vlasov-Poisson system **28**
1.4.1 Gyrokinetic ordering 28
1.4.2 Gyrokinetic Vlasov equation 29
1.4.3 Gyro-average operator 32
1.4.4 Poisson equation with adiabatic electrons 32
1.5 Main differences in gyrokinetic codes **34**
1.5.1 Three numerical approaches : PIC, Eulerian or Semi-Lagrangian 35
1.5.2 δf codes / full-f codes 36
1.5.3 Local geometry / global geometry 36
1.5.4 Fixed gradient / flux driven systems 37

1.1 Brief introduction to plasma fusion

The increase in energy needs and the fact that fossil fuels are running out make indispensable the development of new sources of energies. To produce energy out of matter, it is necessary to carry out a transformation in which, between the initial and final state, a small proportion of the body mass involved disappears. This mass defect may then be found in the form of energy through the well-known formula $E = mc^2$, where E is the produced energy, m the mass that has disappeared and c the speed of light. Chemical reactions involve atoms or molecules and their electrons. As such, the corresponding energies lie in the tens of electron-Volt range (the ionisation energy of hydrogen is 13.6eV). As far as nuclear reactions are concerned, there exist two main types, both in the range of Mega electron-Volt[1]. The fission reaction consists in splitting the nucleus of a sufficiently heavy atom

[1] The ratio between the energy retrieved from chemical and nuclear reactions directly relates to the characteristic interaction lengths of the underlying forces: the Coulomb interaction links electrons to the nucleus on distances $r^{Coulomb}$ of the order of a few tens of Angström ($1\mathring{A} = 10^{-10}m$), while the nuclear force (also called residual strong force) binds neutrons and protons in nuclei at distances $r^{nuclear}$ of the order of one Fermi ($1Fermi = 10^{-15}m$). Since the potential energy associated to each of these central forces (there also exists a weak noncentral component of the nuclear force) decays like $1/r$, with r the distance between the two interacting particles, it comes: $E_{pot}^{Coulomb}/E_{pot}^{nuclear} \sim r^{nuclear}/r^{Coulomb} \sim 10^{-6}$. The precise ratio involves the interaction constants for each of these forces.

(such as the uranium or plutonium atoms) to make lighter atoms. Conversely, very light atomic nuclei are joined together in the fusion reaction to build heavier atoms. Fission is at work in present nuclear reactors while fusion is still at the stage of research.
These researches on controlled fusion are conducted on the most accessible fusion reaction which is the Deuterium-Tritium reaction. In this reaction two nuclei of Deuterium and Tritium (the heavy isotopes of hydrogen, respectively with 1 and 2 neutrons) combine into an alpha particle (Helium nucleus) and a neutron.

$$D + T \rightarrow \ ^4He\ (3.52MeV) + n\ (14.06MeV) \tag{1.1}$$

The total kinetic energy retrieved from the reaction is about $17.6MeV$, one fifth carried out by the Helium nucleus, the remainder by the neutron (this ratio comes from the conservation of momentum during the reaction: $m_{He}v_{He} = m_n v_n$, which implies that $E_{He} = \frac{1}{2}m_{He}v_{He}^2 = \frac{m_{He}}{m_n}\frac{v_{He}^2}{v_n^2}E_n = \frac{1}{4}E_n$). Deuterium is widely available from water, since it constitutes 0.015% of all hydrogen atoms. Tritium is radioactive with a relatively short half-life of 12.3 years, so that its natural occurrence is negligible. It has to be produced. Fortunately, the neutron which is created by the reaction (1.1) can be used to breed new tritium out of lithium directly in the wall of the reaction vessel. In order to overcome the electrostatic repulsion between them, the nuclei must have a temperature greater than a hundred million degrees[2] or about 10keV ($1eV \approx 11\,600\ ^oC$). At such temperatures, electrons are completely detached from the nuclei, such that the hot "gas" is no longer composed of neutral atoms, but of positively (ions) and negatively (electrons) charged particles. This fourth state of matter is called plasma. Due to the presence of these charge carriers, the plasma is electrically conductive so that it strongly responds to electromagnetic fields. Magnetic confinement fusion attempts to create the conditions needed for fusion energy production by confining the plasma with strong magnetic fields. Any static and homogeneous magnetic field restricts the perpendicular (to the magnetic field lines) motion of charged particles to gyro-orbits. Parallel to the field lines, particles move freely (up to magnetic, and possibly electric, mirror effects). In order to keep the volume of the container finite, the field lines are usually bent to a torus. It turns out that charged particles immersed in a curved magnetic field are subject to drifts. Should the magnetic field be purely toroidal (i.e. with circular field lines), these drifts would be strong enough to prevent any confinement on long time durations in a volume of acceptable size [Wes97]. This problem is solved by twisting the magnetic field lines, thanks to the creation of an additional poloidal component of the magnetic field. The average pitch of the field line, defined by the ratio of toroidal revolutions per poloidal revolution of a field line, corresponds to the so-called safety factor q. If q is not a rational number, the field line covers a so-called flux surface. The field lines at different radial positions inside the toroidal plasma vessel define nested flux surfaces. The two most important concepts for magnetic confinement fusion essentially differ in the way the twisting of the field lines is achieved. In stellarators, as Wendelstein 7-X in construction at Greifswald in Germany, the twisted magnetic field that is needed for confinement is entirely generated by the external field coils. In tokamaks (Fig. 1.1), the set of external field coils produces a purely toroidal magnetic field. The poloidal component of the magnetic field is created by the strong electric current induced in the plasma along the toroidal direction. Most of fusion experiments in the world, including the International Thermonuclear Experimental Reactor (ITER), now under construction at Cadarache, France, follow this concept.

In view of producing energy, the ratio between the power from fusion reactions P_{fusion} and the external additional power P_{add} supplied to the plasma by the heating systems must be greater than 1. This ratio is called the quality factor $Q = P_{fusion}/P_{add}$. The present record of fusion power in D-T plasmas has been achieved in the European tokamak JET, with 16 Mega-Watt produced corresponding to $Q = 0.64$. In practice, reaching ignition (condition under which the plasma can be maintained by fusion reactions without external energy input) is not mandatory for viable power plant. Finite values close to $Q = 20$ are often thought to be enough for the economic viability of a reactor. In ITER,

[2] As a matter of fact, the minimal distance between two colliding thermal particles at such a temperature is about $10^{-13}m$, still much too large for the nuclear force to overcome the repulsive Coulomb force. Fusion reactions take place thanks to the tunnel effect of quantum mechanics.

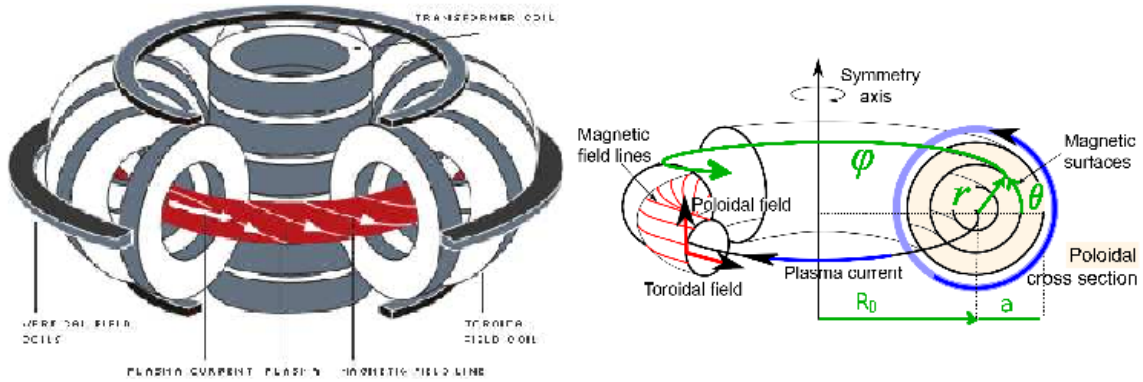


Figure 1.1: Left: Schematic view of the coil system and magnetic field of a tokamak. Right: Corresponding idealized toroidal magnetic geometry and its adopted notations.

whose objective is to demonstrate the practical feasibility of a fusion-based production of energy on an industrial level, the target is $Q = 5$ in steady conditions, and $Q = 10$ during transients. In this latter case, a significant fraction of the heating power of the plasma, namely 2/3, comes from the plasma itself via fusion reactions.
The Lawson criterion is the constraint for reaching self-plasma heating (the *ignition* condition), assuming that the power of Helium ashes is entirely deposited on fuel ions, namely Deuterium and Tritium ions. It is reached when this fraction P_α of the fusion power counterbalances all power losses W/τ_E, with W the plasma internal energy and τ_E the energy confinement time. In this case, the energy confinement time is equal to the Lawson time $\tau_{Lawson} = W/P_\alpha$. Lawson criterion states that the triple product $nT\tau_E$ (n is the density, T the temperature) must be larger than a critical value of the order of $3\,10^{21} m^{-3} keV\ s^{-1}$. In other words, to be able to produce energy from fusion reactions, a sufficiently hot (T) and dense (n) plasma must be confined efficiently (τ_E). The difficulty resides in obtaining the three parameters simultaneously. Indeed, for example, when density is increased by means of gas injection, or when temperature is increased additional heating, the energy confinement time degrades in tokamak plasmas. In magnetic fusion devices, the maximal achievable plasma density n (number of particles per volume unit) is theoretically upper bounded by the strongest achievable magnetic field. Indeed, confinement is effective provided the thermal energy density nT remains lower than the magnetic energy density $B^2/2\mu_0$, with μ_0 the permeability of free space. The ratio of the former on the latter defines the dimensionless beta parameter β. As a matter of fact, tokamak plasmas are subject to large scale instabilities well below the critical value $\beta = 1$ (β is typically of a few percents in tokamaks), which appear redhibitory for the confinement. In such devices, densities are of the order of a few $10^{20} m^{-3}$. Such densities are much lower, typically by 5 orders of magnitudes, than the density of air at normal pressure and temperature. Given the very large temperatures of such plasmas, the pressure is however of a few atmospheres. For these reasons, much effort is devoted to scenarios aiming at improving the confinement time τ_E. As seen on figure 1.2 a small increase of τ_E can have a big impact on the fusion performances. This confinement time, which is basically a thermal relaxation time, is mainly governed by conductive losses. It turns out that these losses are essentially of turbulent nature. *Understanding turbulent transport in magnetized plasma is one of the key open issues in magnetic confinement fusion research.*

1.2 Theoretical Hierarchy of Plasma Physics

The self consistent treatment of plasma evolution requires to solving both the dynamics of the electromagnetic fields, governed by Maxwell's equations, and the plasma response to these fields. Plasma response can be described by a hierarchy of models. The basic and most precise model of this hierarchy is the microscopic description. At this level, the plasma is an ensemble of several species of charged particles (for neutral plasmas, these are electrons with charge $q_e = -e$ and mass m_e and at

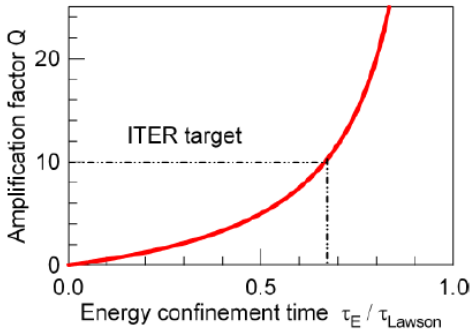


Figure 1.2: Quality factor Q increases with energy confinement time τ_E. Q is proportional to $\tau_E = \tau_E/(\tau_{\text{Lawson}} - \tau_E)$.

least one species of ions with charge $q_i = Z_i e$ and mass m_i) characterized by their positions $\mathbf{x}$ and velocities $\mathbf{v}$. Their individual motion is governed by Newton's equation, the only relevant force being the Lorentz one (Coulomb plus Laplace). For very fast particles, the relativistic formulation should be retained. As shown *e.g.* by Poincaré, the minimal phase space where all the possible trajectories of any dynamical system are represented is six-dimensional: 3D in configuration space (referred as *real space*) and 3D in *velocity space* (or momentum space in the general case). In the following, the vector $\mathbf{x} = (x_1, x_2, x_3)$ refers to position in real space and $\mathbf{v} = (v_1, v_2, v_3)$ in velocity space. The subscript s refers to the various species, with charge q_s and mass m_s. SI units are adopted. Newton's law then reads as follows:

$$m_s \frac{\mathrm{d}\mathbf{v}_s^i}{\mathrm{d}t} = q_s(\mathbf{E} + \mathbf{v}_s^i \times \mathbf{B}) \tag{1.2}$$

with i $(1 \leq i \leq N_s)$ any particle of species s. Here, $\mathbf{E}$ is the electric field and $\mathbf{B}$ the magnetic field. We introduce the scalar ϕ and vector $\mathbf{A}$ potential such that $\mathbf{E} = -\boldsymbol{\nabla}\phi - \partial_t \mathbf{A}$. The dynamics of these fields obey Maxwell's equations:

$$\boldsymbol{\nabla} \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \qquad \text{Gauss} \tag{1.3}$$

$$-\frac{1}{c^2}\frac{\partial \mathbf{E}}{\partial t} + \boldsymbol{\nabla} \times \mathbf{B} = \mu_0 \mathbf{j} \qquad \text{Ampère} \tag{1.4}$$

$$\boldsymbol{\nabla} \cdot \mathbf{B} = 0 \qquad \text{flux conservation} \tag{1.5}$$

$$\frac{\partial \mathbf{B}}{\partial t} + \boldsymbol{\nabla} \times \mathbf{E} = 0 \qquad \text{Faraday} \tag{1.6}$$

The fields depend on the charge density $\rho(\mathbf{x}, t)$ and current density $\mathbf{j}(\mathbf{x}, t)$ of the plasma. Let the plasma consists of N_s particles of s species at positions $(\mathbf{x}_s^i, \mathbf{v}_s^i)$, with $i = 1, \cdots, N_s$. Considering point-like particles, the number density of species s is

$$n_s(\mathbf{x}, t) = \sum_{i=1}^{N_s} \delta(\mathbf{x} - \mathbf{x}_s^i(t))$$

The total charge density is then

$$\rho(\mathbf{x}, t) = \sum_s q_s n_s(\mathbf{x}, t) \tag{1.7}$$

The current density $\mathbf{j}$ is obtained from the mean velocities

$$n_s \mathbf{v}_s(\mathbf{x}, t) = \sum_{i=1}^{N_s} \mathbf{v}_s^i(t)\, \delta(\mathbf{x} - \mathbf{x}_s^i(t))$$

as

$$\mathbf{j}(\mathbf{x}, t) = \sum_s q_s \mathbf{v}_s(\mathbf{x}, t)\, n_s(\mathbf{x}, t) \tag{1.8}$$

The complete description of plasma dynamics is given by Newton's law (Eq. (1.2)) and Maxwell's equations (Eqs. (1.3)-(1.6)). Such a microscopic approach requires to solve N coupled equations in 6D phase space for the $N = \sum_s N_s$ particles of the system. Considering fusion plasmas of about $10^{20} m^{-3}$ ions and electrons, this *many-body model* still remains out of reach for nowadays supercomputers.

Next, the *kinetic models* proceed from the statistical description of the plasma: the particle distribution function $f_s(\mathbf{x}, \mathbf{v}, t)$ is introduced, counting the number of particles of species s in the infinitesimal volume $\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{v}$ of the 6D phase space centered on the position $(\mathbf{x}, \mathbf{v})$. Although the precise locations of individual particles are lost, the detailed knowledge of particle motion is required to evaluate the dynamics of f_s. In this sense kinetic theory is still microscopic, even though statistical averages are employed.

The last step of the hierarchy consists in further reducing the degrees of freedom of the kinetic theory by integrating over the 3D velocity space. Then, a hierarchy of so-called fluid moments can be constructed by weighting the distribution function f_s by tensors of the velocity of rank k before integration. The first moments are the density, the flow velocity (vector), and the pressure (tensor of rank 2). These are macroscopic quantities depending on space and time only. In such *macroscopic or fluid models*, the knowledge of the individual particle motions is no longer required. The major difficulty of the fluid approach resides in the fact that each moment of order k depends on the moment $(k+1)$, such that an additional hypothesis is required to close the system of otherwise infinite set of fluid equations. For weakly collisional media such as fusion plasmas, no satisfactory *closure* exists, conversely to neutral fluids for which the fluid description remains fully appropriate. Still, solving 3D fluid equations is certainly the most convenient and fastest way to solve the problem given the set of well established numerical techniques and the wealth of results obtained in the field of fluid turbulence. However, it is known that the stability threshold given by fluid equations is lower than the actual (kinetic) value [DBB+00]. It is also well established that fluid description usually overestimates turbulent fluxes [DBB+00]. This discrepancy comes partly from the resonant interactions between waves and particles (Landau resonances), which cannot be properly accounted for by fluid models. Also, large scale axi-symmetric flows of fusion plasmas, known as zonal flows, which play an important role in regulating turbulence, tend to be over-damped in fluid models.

Several attempts have been made to propose optimized closures, in view of accounting as much as possible for some kinetic effects, such as Landau resonances (see e.g. [HP90]-[WSD+97]-[SDPZ+09]). Comparing fluid and kinetic simulations provides a stringent test of these closure schemes. This exercise has revealed much more difficult than expected, pointing out irreconcilable discrepancies [DBB+00]-[SGDP+06a]. Fortunately, the dramatic increase of numerical resources and performances now enables the direct treatment of kinetic models, or at least of the gyro-kinetic ones (see section 1.4).

1.3 Plasma kinetic theory

1.3.1 Liouville theorem

Let us consider a canonical Hamiltonian system: $\mathbf{q} = \{\mathbf{q}_i\}_{1 \le i \le N}$ denote the generalized coordinates, $\mathbf{p} = \{\mathbf{p}_i\}_{1 \le i \le N}$ their conjugate momenta and $H(\{\mathbf{q}_i, \mathbf{p}_i\})$ the Hamiltonian. Let also the phase-space distribution $\mathcal{D}_N(\mathbf{q}, \mathbf{p})$ determine the probability $\mathcal{D}_N(\mathbf{q}, \mathbf{p})\, \mathrm{d}^N q\, \mathrm{d}^N p$ that the system is in the infinitesimal volume of phase-space $\mathrm{d}^N q\, \mathrm{d}^N p$ around the position $(\mathbf{q}, \mathbf{p})$. The equilibrium statistical mechanics of such a canonical Hamiltonian system is governed by Liouville's theorem, which states that the *phase-space distribution function is constant along the trajectories of the system* – that is, the density $\mathcal{D}_N$ of N system points in the vicinity of a given system point travelling through phase-space is constant in time. The proof directly follows from the continuity equation fulfilled by $\mathcal{D}_N$, namely:

$$\frac{\partial \mathcal{D}_N}{\partial t} + \sum_{i=1}^{N} \left[\frac{\partial}{\partial \mathbf{q}_i} \left(\mathcal{D}_N \frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} \right) + \frac{\partial}{\partial \mathbf{p}_i} \left(\mathcal{D}_N \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) \right] = 0 \tag{1.9}$$

and the fact that the generalized velocity field $(\dot{\mathbf{p}}, \dot{\mathbf{q}})$ in phase space is divergenceless, as a direct consequence of the Hamilton equations of motion

$$\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} = \frac{\partial H}{\partial \mathbf{p}_i}$$
$$\frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -\frac{\partial H}{\partial \mathbf{q}_i}$$

Developing eq. (1.9) leads to

$$\frac{\partial \mathcal{D}_N}{\partial t} + \sum_{i=1}^{N} \left(\frac{\partial \mathcal{D}_N}{\partial \mathbf{q}_i} \frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} + \frac{\partial \mathcal{D}_N}{\partial \mathbf{p}_i} \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) + \mathcal{D}_N \sum_{i=1}^{N} \left[\frac{\partial}{\partial \mathbf{q}_i} \left(\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} \right) + \frac{\partial}{\partial \mathbf{p}_i} \left(\frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) \right] = 0$$

The last sum vanishes in virtue of Hamilton's equations. The resulting equation states that the convective derivative of the density $\mathrm{d}\mathcal{D}_N / \mathrm{d}t$ is equal to 0:

$$\frac{\mathrm{d}\mathcal{D}_N}{\mathrm{d}t} = \frac{\partial \mathcal{D}_N}{\partial t} + \sum_{i=1}^{N} \left(\frac{\partial \mathcal{D}_N}{\partial \mathbf{q}_i} \frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} + \frac{\partial \mathcal{D}_N}{\partial \mathbf{p}_i} \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) = 0 \tag{1.10}$$

Integrating by parts Liouville's equation (1.10) over the variables leads to a chain of $(N-1)$ equations where the j-th equation connects the j-th and $(j+1)$-th particle density probability functions, with $\mathcal{D}_j = \mathcal{D}_j(\mathbf{q}_1 \cdots \mathbf{q}_j, \mathbf{p}_1 \cdots \mathbf{q}_j)$. The truncation of this BBGKY hierarchy of equations is a common starting point for many applications of kinetic theory. In particular, truncation at the first equation or the first two equations can be used to derive classical Boltzmann equations and their first order corrections. This derivation is out of the scope of this paper, and can be found in *e.g.* [PGM02]. In the following, we focus on the kinetic description of the plasma turbulence and more precisely on the numerical solving of *Boltzmann equation* and of its collisionless form, namely *Vlasov equation*.

1.3.2 Vlasov-Maxwell system

In a high temperature fusion plasmas, the kinetic energy is much larger than the average potential (Coulombian) energy between particles, such that particles are weakly coupled. In weakly coupled plasmas, multiple particle correlations involving more than two particles are neglected. For each plasma species let us consider $F_s \equiv F_s(t, \mathbf{x}, \mathbf{v})$ the 6D distribution function, which provides the probability to find a particle at position $\mathbf{x}$ with velocity $\mathbf{v}$ at time t. Then the evolution of the density of particles species s in the phase space $(\mathbf{x}, \mathbf{v})$ is governed by Boltzmann equation

$$\frac{\mathrm{d}F_s}{\mathrm{d}t} = \frac{\partial F_s}{\partial t} + \mathbf{v} \cdot \frac{\partial F_s}{\partial \mathbf{x}} + q_s \left(\mathbf{E} + \mathbf{v} \times \mathbf{B} \right) \frac{\partial F_s}{\partial \mathbf{v}} = \sum_{s'} \mathcal{C}_{ss'}[F_s, F_{s'}] + S(F_s) \tag{1.11}$$

where $\mathcal{C}_{ss'}[F_s, F_{s'}]$ is the collision operator between the species s and s' and S is a source operator. Without source term and in the collision-less limit (i.e $\mathcal{C}_{ss'}[F_s, F_{s'}]$ is neglected) equation (1.11) is reduced to the Vlasov equation (or collision-less Boltzmann equation):

$$\frac{\partial F_s}{\partial t} + \mathbf{v} \cdot \frac{\partial F_s}{\partial \mathbf{x}} + q_s \left(\mathbf{E} + \mathbf{v} \times \mathbf{B} \right) \frac{\partial F_s}{\partial \mathbf{v}} = 0 \tag{1.12}$$

Here, $\mathbf{E}$ is the electric field and $\mathbf{B}$ the magnetic field. We introduce the scalar U and vector potential $\mathbf{A}$ such that $\mathbf{E} = -\boldsymbol{\nabla} U - \partial_t \mathbf{A}$. Electromagnetic fields are external but are also generated by the particles. They must be computed self-consistently by the Maxwell's equations,

$$\boldsymbol{\nabla} \cdot \mathbf{E} = \frac{1}{\epsilon_0} \sum_s q_s n_s \qquad \text{Gauss} \tag{1.13}$$

$$-\frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t} + \boldsymbol{\nabla} \times \mathbf{B} = \mu_0 \sum_s \mathbf{j}_s \qquad \text{Ampère} \tag{1.14}$$

$$\boldsymbol{\nabla} \cdot \mathbf{B} = 0 \qquad \text{flux conservation} \tag{1.15}$$

$$\frac{\partial \mathbf{B}}{\partial t} + \boldsymbol{\nabla} \times \mathbf{E} = 0 \qquad \text{Faraday} \tag{1.16}$$

where q_s is the charge of species s (i.e $q_s = Z_s e$). The constants ϵ_0 and μ_0 are respectively the permittivity and permeability of free space. The source terms, the particle density $n_s(\mathbf{x}, t)$ and the current density $\mathbf{j}_s(\mathbf{x}, t)$, correspond to the first two velocity moments of F_s:

$$n_s(\mathbf{x}, t) = \int \mathrm{d}^3\mathbf{v}\, F_s(t, \mathbf{x}, \mathbf{v}) \tag{1.17}$$

$$\mathbf{j}_s(\mathbf{x}, t) = \int \mathrm{d}^3\mathbf{v}\, \mathbf{v} F_s(t, \mathbf{x}, \mathbf{v}) \tag{1.18}$$

The Vlasov-Poisson model is obtained by neglecting the time fluctuations of the magnetic field $\mathbf{B}$. In the electrostatic limit considered in the following the time fluctuations of the vector potential $\mathbf{A}$ are neglected, then $\mathbf{E}$ is directly linked to the fluctuating electrostatic potential U as:

$$\mathbf{E} = -\boldsymbol{\nabla} U \tag{1.19}$$

1.3.3 Advective and conservative forms of Vlasov equation

In this paragraph, let us recall some numerical properties of the Vlasov equation useful for the rest of the manuscript (Proofs can be found for instance in appendix C of [GS13]). First, let us denote $\mathbf{Z}$ the six-dimensional phase space vector $\mathbf{Z} = \{\mathbf{x}, \mathbf{v}\}$ and $\nabla_{(\mathbf{x},\mathbf{v})}$ the six-dimensional phase-space derivative:

$$\nabla_{(\mathbf{x},\mathbf{v})} = \{\nabla_{\mathbf{x}}, \nabla_{\mathbf{v}}\} = \left\{\frac{\partial}{\partial \mathbf{x}}, \frac{\partial}{\partial \mathbf{v}}\right\} = \frac{\partial}{\partial \mathbf{Z}} \tag{1.20}$$

Then the Vlasov equation (1.12) can be transcribed into an *advection equation* in phase-space of the function $F_s : \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}$ (with $d = 6$)

$$\frac{\partial}{\partial t} F_s(\mathbf{Z}, t) + \mathbf{U}(\mathbf{Z}, t) \cdot \nabla_{(\mathbf{x},\mathbf{v})} F_s(\mathbf{Z}, t) = 0 \tag{1.21}$$

Here, we have introduced the six-dimensional phase-space flow $\mathbf{U} : \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}$, defined as the total time derivative of $\mathbf{Z}$:

$$\mathbf{U}(\mathbf{Z}, t) = \{\mathbf{U}_{\mathbf{x}}, \mathbf{U}_{\mathbf{v}}\} = \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} = \left\{\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}, \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\right\} = \{\mathbf{v}, \mathbf{E} + \mathbf{v} \times \mathbf{B}\} \tag{1.22}$$

Let us now consider the differential system

$$\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} = \mathbf{U}(\mathbf{Z}(t), t) \tag{1.23}$$

$$\mathbf{Z}(t_1) = \mathbf{z} \tag{1.24}$$

Definition 1 *The solutions of equation (1.23) are called the characteristics of the advection equation (1.21). Let us denote $\mathbf{Z}(t; \mathbf{z}, t_1)$ the solution of (1.23)-(1.24).*

The existence, uniqueness and regularity of the solutions of the previous differential equations (1.23)-(1.24) derive from the classical theorem of differential equation theory, which reads as follows (proof can be found in *e.g.* [Amm90]):

Theorem 1 *Let us assume $\mathbf{U} \in C^{k-1}(\mathbb{R}^d \times [0, T])$, $\nabla \mathbf{U} \in C^{k-1}(\mathbb{R}^d \times [0, T])$ with $k \in \mathbb{N}, k > 1$ and that, for $\kappa \geq 1$:*

$$|\mathbf{U}(\mathbf{z}, t)| \leq \kappa(1 + |\mathbf{z}|) \qquad \forall t \in [0, T] \quad \forall \mathbf{z} \in \mathbb{R}^d$$

then $\forall t_1 \in [0, T]$ and $\mathbf{z} \in \mathbb{R}^d$, there exists a unique solution $\mathbf{Z} \in C^k([0, T] \times \mathbb{R}^d \times [0, T])$ of equations (1.23)-(1.24).

Proposition 1 *Under the same assumptions as for theorem 1, the following properties hold:*

1. *$\forall t_1, t_2, t_3 \in [0, T]$ and $\mathbf{z} \in \mathbb{R}^d$*

$$\mathbf{Z}(t_3; \mathbf{Z}(t_2; \mathbf{z}, t_1), t_2) = \mathbf{Z}(t_3; \mathbf{z}, t_1)$$

2. *$\forall (t, t_1) \in [0, T]^2$, the application $\mathbf{z} \mapsto \mathbf{Z}(t; \mathbf{z}, t_1)$ is a C^1-diffeomorphism of $\mathbb{R}^d$ with inverse $\mathbf{y} \mapsto \mathbf{Z}(t_1; \mathbf{y}, t)$.*

3. *The jacobian $J(t; 1, t_1) = \det(\nabla_{\mathbf{z}} \mathbf{Z}(t; \mathbf{z}, t_1))$ satisfies $J > 0$ and*

$$\frac{\partial J}{\partial t} = (\nabla \cdot \mathbf{U})(\mathbf{Z}(t; \mathbf{z}, t_1))\, J$$

In particular, if $\nabla \cdot \mathbf{U} = 0$, then $J(t; 1, t_1) = J(t_1; 1, t_1) = \det \mathbb{I}_d = 1$ where $\mathbb{I}_d$ is the identity matrix of order d.

Let us also notice that equation (1.21) can be formulated as:

$$\frac{\mathrm{d}F_s}{\mathrm{d}t} = \frac{\partial F_s}{\partial t} + \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} \cdot \nabla_{\mathbf{z}} F_s = 0 \tag{1.25}$$

The total time derivative of F_s is equal to 0. Said differently, the fundamental property of the Vlasov equation is that *the distribution function F_s is constant along its characteristics.* This property is one of the foundations of many numerical schemes, including both Lagrangian and semi-Lagrangian numerical approaches (see [GS13] for more details).

Since the phase space element is incompressible in the Vlasov equation $\nabla_{(x,v)} \mathbf{U} = 0$ – Liouville's theorem applies – then the advective form of the Vlasov equation (1.21) can be rewritten as

$$\frac{\partial}{\partial t} F_s(\mathbf{Z}, t) + \nabla_{(x,v)} \cdot (\mathbf{U}(\mathbf{Z}, t)\, F_s(\mathbf{Z}, t)) = 0 \tag{1.26}$$

Equation (1.26) corresponds to the *conservative form* of the Vlasov equation. Liouville's theorem guaranties that the advective and the conservative forms of the Vlasov equation are equivalent. Both formulations are used in the numerical treatments, depending on the adopted numerical scheme.

1.3.4 Conservation properties of the Vlasov-Maxwell system

Let us consider the Vlasov-Maxwell's system (1.12)-(1.16). As discussed in this paragraph, it satisfies a certain number of conservation properties. Such exact properties reveal particularly powerful when developing numerical schemes: inherently conserving numerical methods can be envisaged, or, alternatively, they become stringent verification tests for numerical schemes and simulation results.

Proposition 2 *The Vlasov equation insures the conservation of the number N of particles in a closed phase space volume V, apart from exchanges with the exterior. In this case, in agreement with Liouville's theorem:*

$$N = \int_V F_s(\mathbf{x}, \mathbf{v}, t)\, \mathrm{d}\mathbf{x}\, \mathrm{d}\mathbf{v} = \int_V F_s(\mathbf{Z}, t)\, \mathrm{d}V = \text{const} \tag{1.27}$$

As shown below, this simply results from the fact that the total derivative of F_s is equal to 0:

$$\frac{\mathrm{d}N}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t} \int_V F_s(\mathbf{Z}, t)\, \mathrm{d}V = \int_V \frac{\partial F_s}{\partial t}\, \mathrm{d}V = -\int_V \mathbf{U} \cdot \nabla_{(x,v)} F_s\, \mathrm{d}V$$

Using Gauss's theorem, it can be recast as follows

$$\int_V \mathbf{U} \cdot \nabla_{(x,v)} F_s\, \mathrm{d}V = \oint_{S(V)} F_s\, \mathbf{n} \cdot \mathbf{U}\, \mathrm{d}S \tag{1.28}$$

where $S(V)$ stands for the surface of volume V and $\mathbf{n}$ is its normal unit vector. Then

$$\frac{\partial N}{\partial t} = -\oint_{S(V)} F_s(\mathbf{n}\cdot\mathbf{U})\ \mathrm{d}S \tag{1.29}$$

This relation states that the rate of change of the number of particle inside the volume V is equal to the integral of the flux of F_s across the surface $S(V)$ of this volume. Hence, N remains constant for vanishing fluxes at the boundaries.

Proposition 3 *Let the distribution function $F_s(\mathbf{x},\mathbf{v},t)$ in the phase space $(\mathbf{x},\mathbf{v})\in\mathbb{R}^d\times\mathbb{R}^d$, $d = 1,\cdots 3$ be governed by the Vlasov-Maxwell equations. Then, denoting $F_s^0(\mathbf{x},\mathbf{v})$ the distribution function at initial time $t=0$, F_s satisfies the following properties:*

- *Maximum principle*
$$0 \le F_s(\mathbf{x},\mathbf{v},t) \le max_{(x,v)}(F_s^0(\mathbf{x},\mathbf{v})) \quad \forall t\in\mathbb{R}^+ \tag{1.30}$$
- *Conservation of the volume. For all volume V of phase-space*
$$\int_V F_s(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = \int_{\mathbf{U}^{-1}(V)} F_s^0(\mathbf{y},\mathbf{u})\,\mathrm{d}\mathbf{y}\,\mathrm{d}\mathbf{u} \quad \forall t\in\mathbb{R}^+ \tag{1.31}$$
- *Conservation of the L^p norms, $p\in\mathbb{N}$, $1\le p\le\infty$*
$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d\times\mathbb{R}^d} (F_s(\mathbf{x},\mathbf{v},t))^p\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = 0 \quad \forall t\in\mathbb{R}^+ \tag{1.32}$$
- *Conservation of the kinetic entropy*
$$\frac{\mathrm{d}}{\mathrm{d}t}\left(-\int_{\mathbb{R}^d\times\mathbb{R}^d} F_s(\mathbf{x},\mathbf{v},t)\ln F_s(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}\right) = 0 \quad \forall t\in\mathbb{R}^+ \tag{1.33}$$
- *Conservation of the energy,*
$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d\times\mathbb{R}^d} m|\mathbf{v}|^2\,F_s(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} \\ + \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d}\left(\varepsilon_0|E(\mathbf{x},t)|^2 + \frac{1}{\mu_0}|B(\mathbf{x},t)|^2\right)\,\mathrm{d}\mathbf{x} = 0 \quad \forall t\in\mathbb{R}^+\end{aligned} \tag{1.34}$$

The maximum principle especially insures that $F_s(\mathbf{x},\mathbf{v},t)$ remains positive for all time t if the initial data $F_s^0(\mathbf{x},\mathbf{v})$ is positive everywhere. For proofs see for instance appendix D in [GS13].

The entropy conservation is the consequence of the fact that the Vlasov-Maxwell system is *time reversible*. Indeed, let us consider the following time reversal transformation: $t\to -t$; $\mathbf{x}\to\mathbf{x}$; $\mathbf{v}\to-\mathbf{v}$; $\mathbf{E}\to\mathbf{E}$ and $\mathbf{B}\to-\mathbf{B}$. This transformation leaves Vlasov equation invariant. This can be understood as follows: starting from some initial state and evolving F_s in time according to the Vlasov-Maxwell system, and deciding at a certain time to reverse the velocity of all particles (leading to the reversal of the magnetic field $\mathbf{B}$), then the system would trace back to its initial state (see *e.g.* Villani's lecture notes [Viled])

Finally, it is worth noticing that Maxwell's equations imply that charge density ρ and current density $\mathbf{j}$ satisfy the continuity equation

$$\frac{\partial\rho}{\partial t} + \boldsymbol{\nabla}\cdot\mathbf{j} = 0 \tag{1.35}$$

In other words, should this relation not hold, one can easily verify that there is no solution to Maxwell's equations. When the sources of Maxwell's equations are computed numerically by solving the Vlasov equation, they do not necessarily verify a discrete equivalent of the continuity equation, which would be compatible with the discrete form of Maxwell's equations. To address this problem, numerical methods for deriving sources that satisfy the discrete continuity equation have been developed (cf *e.g.* [VB92]). A rigorous mathematical theory for these generalized formulations of Maxwell's equations is derived in [BCS07].

1.4 Gyrokinetic Vlasov-Poisson system

1.4.1 Gyrokinetic ordering

The computational effort to numerically solve the 6-dimensional Vlasov-Maxwell system still remains out of reach for present day supercomputers. As shown in figure 1.3, it involves a wide range of spatio-temporal scales: the typical size of micro-instabilities is of the order of the ion Larmor radius $\rho_{Li} \sim 10^{-3}$ m, but the size of a tokamak is of the order of few meters. The strong magnetic fields

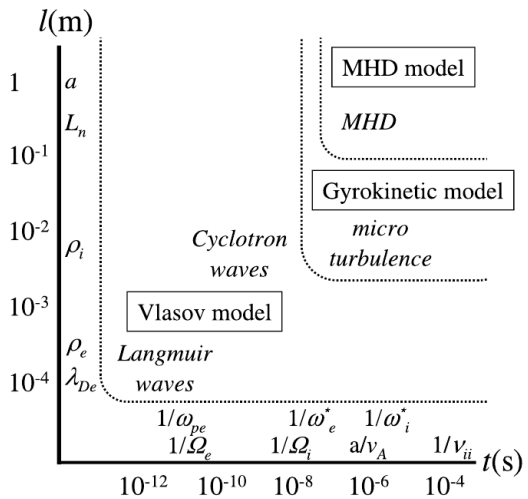


Figure 1.3: Typical space and time range scales in fusion plasmas and applicability of Vlasov, gyrokinetic and MHD models. Here, ω_{ps} is the plasma oscillation frequency, Ω_s is the cyclotron frequency, ω_s^* is the diamagnetic rotation frequency, v_A is the Alfvèn velocity, ν_{ii} is the ion-ion collision frequency, λ_{Ds} is the Debye length, ρ_s is the Larmor radius, L_n is the characteristic gradient length of the equilibrium density profile, a is the plasma size and s denotes the particle species. (figure from [IWS06])

– a few Tesla – used in magnetic confinement fusion devices lead to the fast gyration of charged particles around magnetic field lines. Especially, the cyclotron frequency $\Omega_s = (q_s B_0)/m_s$ is large compared to the characteristic frequency of micro-turbulence, which is of the order of the diamagnetic frequency $\omega \sim \omega_s^* \sim (k_\theta \rho_s) v_{th,s}/L_p$ (with k_θ the poloidal wave vector, $\rho_s = v_\perp/\Omega_s$ the Larmor radius, $v_{th,s} = (T_s/m_s)^{1/2}$ the thermal speed and $L_p = |\nabla \ln p_0|^{-1}$ the characteristic gradient length of the mean pressure profile. In this framework, the single particle motion consists of the superposition of the fast periodic gyro-motion and the slow drift of the guiding-centre, as shown in Fig. 1.4. Besides, experimental observations in core plasmas of magnetic confinement fusion devices suggest

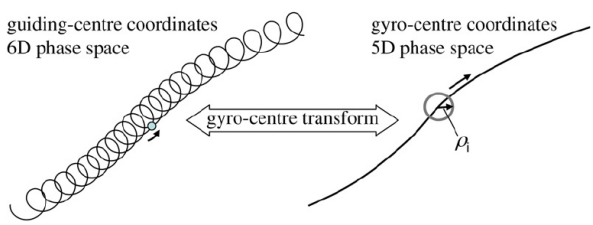


Figure 1.4: The phase-space reduction from 6 to 5 dimensions results from the gyro-centre transform, which aims at eliminating the fast gyro-motion. (figure from [IWS06])

that small scale turbulence, responsible for anomalous transport, obeys the following ordering in a small parameter $\epsilon_g \sim \rho_*$ with $\rho_* = \rho_s/a \ll 1$ (a being the small radius of the tokamak):

$$\frac{\omega}{\Omega_s} \sim \frac{q_s U}{T_s} \sim \frac{\delta n_s}{n_0} \sim \frac{B_1}{B_0} \sim \frac{k_\parallel}{k_\perp} \sim \frac{\rho_s}{L_n} \sim \mathcal{O}(\epsilon_g) \tag{1.36}$$

where the subscript s refers to the particle species. T_s is the temperature, δn_s the perturbed density, n_0 the equilibrium density, $\mathbf{B}_0$ and $\mathbf{B}_1$ are respectively the equilibrium and the perturbed magnetic field, $k_\parallel = \mathbf{k}\cdot\mathbf{b}_0$ and $k_\perp = |\mathbf{k}\times\mathbf{b}_0|$ are parallel and perpendicular components of the wave vector $\mathbf{k}$ with $\mathbf{b}_0 = \mathbf{B}_0/B_0$, $\rho_s = v_\perp/\Omega_s$ is the Larmor radius, $L_n = |\nabla \ln n_0|^{-1}$ is the characteristic length of n_0. Within this *gyro-ordering*, the so-called *gyrokinetic model* can be derived (see *e.g.* [Lee83]) by eliminating high-frequency processes characterized by $\omega > \Omega_s$. The phase space is reduced from 6 to 5 dimensions, while retaining crucial kinetic effects such as finite Larmor radius effects. For detailed gyrokinetic theory see review papers by Brizard & Hahm (2007) [BH07] and Krommes (2012) [Kro12]. Numerically speaking, the computational cost is dramatically reduced because the limitations on the time step and the grid discretization are relaxed from $\omega_{p_s}\Delta t < 1$ and $\Delta x < \lambda_{D_s}$ to $\omega_s^*\Delta t < 1$ and $\Delta x < \rho_s$ (with ω_{p_s} the plasma oscillation frequency and λ_{D_s} the Debye length). It is also important to note that the magnetic moment, $\mu_s = m_s v_\perp^2/(2B)$ becomes an adiabatic invariant. In terms of simulation cost, this last point is convenient because μ_s plays the role of a parameter. This means that the problem to treat is not a true 5D problem but rather a 4D problem parametrized by μ_s. Note that μ_s looses its invariance property in the presence of collisions. Such a numerical drawback can be overcome by considering reduced collisions operators acting in the parallel velocity space only, while still recovering the results of the neoclassical theory [GDPN+09].

1.4.2 Gyrokinetic Vlasov equation

The resulting *gyrokinetic Vlasov equation* is today the most advanced framework to describe plasma micro-turbulence. Let F_s be the particle distribution function of species s and $\bar{F}_s$ the one associated to the guiding-centers. The non-linear time evolution of $\bar{F}_s$ is governed by the gyrokinetic equation described by Brizard and Hahm [BH07]

$$\partial_t \bar{F}_s - \left[H, \bar{F}_s\right]_{\mathrm{GC}} = \mathcal{R}hs(\bar{F}_s) \tag{1.37}$$

where H the Hamiltonian of the system is defined as $H = \frac{1}{2}m_s v_{G\parallel}^2 + \mu_s B + q_s\bar{U}$ and where $[.,.]_{\mathrm{GC}}$ corresponds to generalized expression of the Poisson brackets in the gyro-center coordinates defined as (see equation (150) in [BH07])

$$[X,Y]_{\mathrm{GC}} = \frac{\mathbf{B}_s^*}{m_s B_{\parallel s}^*}\cdot\left(\boldsymbol{\nabla}X\partial_{v_{G\parallel}}Y - \partial_{v_{G\parallel}}X\boldsymbol{\nabla}Y\right) - \frac{\mathbf{b}}{q_s B_{\parallel s}^*}\cdot(\boldsymbol{\nabla}X\times\boldsymbol{\nabla}Y) \tag{1.38}$$

with $\mathbf{b} = \mathbf{B}/\|B\|$ the unit vector along the magnetic field line at the guiding-center position. Here, $\bar{U} = \mathcal{J}_\mu.U$ is the gyro-average of the fluctuating electrostatic potential U. It corresponds to an average over a cyclotron motion: $\mathcal{J}_\mu.U = \oint_0^{2\pi} U\frac{\mathrm{d}\varphi_c}{2\pi}$, where φ_c stands for the cyclotron phase. This gyro-average operator $\mathcal{J}_\mu$ is discussed in more detail in next section 1.4.3. The scalar $B_{\parallel s}^*$ corresponds to the volume element in guiding-center velocity space. It is simply defined as

$$B_{\parallel s}^* = \mathbf{B}_s^*\cdot\mathbf{b} \quad \text{with} \quad \mathbf{B}_s^* \equiv \mathbf{B} + \frac{m_s}{q_s}v_{G\parallel}\boldsymbol{\nabla}\times\mathbf{b} \tag{1.39}$$

Then, using the relation

$$\boldsymbol{\nabla}\times\mathbf{b} = \boldsymbol{\nabla}\left(\frac{1}{B}\right)\times\mathbf{B} + \frac{1}{B}\boldsymbol{\nabla}\times\mathbf{B} = \mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B} + \frac{1}{B}\boldsymbol{\nabla}\times\mathbf{B} \tag{1.40}$$

$B_{\parallel s}^*$ can be expressed as [3]

$$B_{\parallel s}^* \equiv B + \frac{m_s}{q_s}v_{G\parallel}\mathbf{b}\cdot(\boldsymbol{\nabla}\times\mathbf{b}) \tag{1.41}$$

[3] According to (1.39) and (1.40) then

$$B_{\parallel s}^* = \mathbf{B}\cdot\mathbf{b} + \frac{m_s}{q_s}v_{G\parallel}(\boldsymbol{\nabla}\times\mathbf{b})\cdot\mathbf{b} = B + \frac{m_s}{q_s}v_{G\parallel}\left[\mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B} + \frac{1}{B}\boldsymbol{\nabla}\times\mathbf{B}\right]\cdot\mathbf{b} = B + \frac{m_s v_{G\parallel}}{q_s B}\mathbf{b}\cdot(\boldsymbol{\nabla}\times\mathbf{B})$$

For the following, let us notice that (1.39) and (1.40) gives

$$\frac{\mathbf{B}_s^*}{B_{\|s}^*} = \mathbf{b}_s^* + \frac{m_s v_{G\|}}{q_s B_{\|s}^*}\left(\mathbf{b}\times\frac{\boldsymbol{\nabla} B}{B}\right) \tag{1.42}$$

with

$$\mathbf{b}_s^* = \frac{\mathbf{B}}{B_{\|s}^*} + \frac{m_s v_{G\|}}{q_s B_{\|s}^* B}\boldsymbol{\nabla}\times\mathbf{B} \tag{1.43}$$

This Hamiltonian formalism is equivalent to consider that the time evolution of the guiding-center distribution $\bar{F}_s$ is governed by the gyrokinetic conservative equation (see also Eqs (17)-(20) in [GIVW10]):

$$B_{\|s}^*\frac{\partial \bar{F}_s}{\partial t} + \boldsymbol{\nabla}\cdot\left(B_{\|s}^*\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t}\bar{F}_s\right) + \frac{\partial}{\partial v_{G\|}}\left(B_{\|s}^*\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t}\bar{F}_s\right) = 0 \tag{1.44}$$

where $\mathbf{x}_G$ and $v_{G\|}$ are respectively the space coordinates and the parallel velocity of the guiding centers and where, in the electrostatic limit, the motion equations of the guiding centers are given by:

$$B_{\|s}^*\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t} = v_{G\|}\mathbf{B}_s^* + \frac{1}{q_s}\,\mathbf{b}\times\boldsymbol{\nabla}\Xi \tag{1.45}$$

$$B_{\|s}^* m_s\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t} = -\mathbf{B}_s^*\cdot\boldsymbol{\nabla}\Xi \tag{1.46}$$

with

$$\Xi \equiv q_s\bar{U} + \mu_s B \tag{1.47}$$

and $\mathbf{B}_s^*$ and $B_{\|s}^*$ respectively defined by equations (1.39) and (1.41). Then, using relation (1.40), equations (1.45) and (1.46) can be rewritten as [4]

$$\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t} = v_{G\|}\mathbf{b}_s^* + \mathbf{v}_{E\times B_s} + \mathbf{v}_{D_s} \tag{1.48}$$

$$m_s\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t} = -\mu_s\boldsymbol{\nabla}_\|^* B - q_s\boldsymbol{\nabla}_\|^*\bar{U} + m_s v_{G\|}\mathbf{v}_{E\times Bs}\cdot\frac{\boldsymbol{\nabla} B}{B} \tag{1.49}$$

where $\boldsymbol{\nabla}_\|^* \equiv \mathbf{b}_s^*\cdot\boldsymbol{\nabla}$ and where the drift velocities evaluated at the gyro-centre are respectively:

- the '$\mathbf{E}\times\mathbf{B}$' drift $\sim\mathcal{O}(\epsilon_g)$:

$$\mathbf{v}_{E\times B_s} = \frac{1}{B_{\|s}^*}\mathbf{b}\times\boldsymbol{\nabla}\bar{U} \tag{1.50}$$

- the 'grad–B' and 'curvature' drifts $\sim\mathcal{O}(\epsilon_a\epsilon_g)$ with $\epsilon_a = a/R_0$ the inverse aspect ratio:

$$\mathbf{v}_{D_s} = \left(\frac{m_s v_{G\|}^2 + \mu_s B}{q_s B_{\|s}^*}\right)\mathbf{b}\times\frac{\boldsymbol{\nabla} B}{B} \tag{1.51}$$

Proof 1 *(Link between gyrokinetic Poisson brackets and motion equations)*
Using expression (1.38) and the expression of the Hamiltonian, then the different parts of the Poisson

[4] Replacing (1.42) and (1.43) in equation (1.45) leads to equation (1.48). Using again previous expression of $\mathbf{B}_s^*/B_{\|s}^*$ and the fact that $(\mathbf{b}\times\boldsymbol{\nabla} B)\cdot\boldsymbol{\nabla} B$ is null leads to equation (1.49).

brackets $\left[H, \bar{F}_s\right]$ *can be written separately as follows*

$$\left[m_s v_{G\|}^2/2, \bar{F}_s\right]_{\mathrm{GC}} = -v_{G\|}\frac{\mathbf{B}_s^*}{B_{\|s}^*}\cdot\boldsymbol{\nabla}\bar{F}_s = -v_{G\|}\boldsymbol{\nabla}_\|^*\bar{F}_s - \frac{m_s v_{G\|}^2}{q_s B_{\|s}^*}\left(\mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B}\right)\cdot\boldsymbol{\nabla}\bar{F}_s \tag{1.52}$$

$$\begin{aligned}\left[\mu_s B, \bar{F}_s\right]_{\mathrm{GC}} &= \frac{\mu_s\mathbf{B}_s^*}{m_s B_{\|s}^*}\cdot\boldsymbol{\nabla}B\partial_{v_{G\|}}\bar{F}_s - \frac{\mu_s\,\mathbf{b}}{q_s B_{\|s}^*}\cdot\left(\boldsymbol{\nabla}B\times\boldsymbol{\nabla}\bar{F}_s\right)\\ &= \frac{\mu_s}{m_s}\boldsymbol{\nabla}_\|^*B\partial_{v_{G\|}}\bar{F}_s - \frac{\mu_s}{q_s B_{\|s}^*}\left(\mathbf{b}\times\boldsymbol{\nabla}\mathbf{B}\right)\cdot\boldsymbol{\nabla}\bar{F}_s\end{aligned} \tag{1.53}$$

$$\begin{aligned}\left[q_s\bar{U}, \bar{F}_s\right]_{\mathrm{GC}} &= \frac{q_s}{m_s}\frac{\mathbf{B}_s^*}{B_{\|s}^*}\cdot\boldsymbol{\nabla}\bar{U}\partial_{v_{G\|}}\bar{F}_s - \frac{\mathbf{b}}{B_{\|s}^*}\cdot\left(\boldsymbol{\nabla}\bar{U}\times\boldsymbol{\nabla}\bar{F}_s\right)\\ &= \frac{q_s}{m_s}\boldsymbol{\nabla}_\|^*\bar{U}\partial_{v_{G\|}}\bar{F}_s\\ &\quad+\frac{v_{G\|}}{B_{\|s}^*}\left(\mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B}\right)\cdot\boldsymbol{\nabla}\bar{U}\partial_{v_{G\|}}\bar{F}_s - \frac{\left(\mathbf{b}\times\boldsymbol{\nabla}\bar{U}\right)}{B_{\|s}^*}\cdot\boldsymbol{\nabla}\bar{F}_s\end{aligned} \tag{1.54}$$

Combining all the terms (1.52)-(1.54) *we find the gyrokinetic Vlasov equation*

$$\begin{aligned}\partial_t\bar{F}_s &+ \left[v_{G\|}\mathbf{b}_s^* + \left(\frac{m_s v_{G\|}^2+\mu_s B}{q_s B_{\|s}^*}\right)\left(\mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B}\right) + \frac{1}{B_{\|s}^*}\mathbf{b}\times\boldsymbol{\nabla}\bar{U}\right]\cdot\boldsymbol{\nabla}\bar{F}_s\\ &+ \left[-\frac{\mu_s}{m_s}\boldsymbol{\nabla}_\|^*B - \frac{q_s}{m_s}\boldsymbol{\nabla}_\|^*\bar{U} - \frac{v_{G\|}}{B_{\|s}^*}\left(\mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B}\right)\cdot\boldsymbol{\nabla}\bar{U}\right]\partial_{v_{G\|}}\bar{F}_s = 0\end{aligned}$$

which is equivalent to equations (1.44)-(1.46) *due to* $(\mathbf{b}\times\boldsymbol{\nabla}B)\cdot\boldsymbol{\nabla}\bar{U} = -(\mathbf{b}\times\boldsymbol{\nabla}\bar{U})\cdot\boldsymbol{\nabla}B$.

Finally, using the fact that the axi-symmetric equilibrium is determined by three of the ideal MagnetoHydroDynamic (MHD) equations:

$$\boldsymbol{\nabla}p = \mathbf{J}\times\mathbf{B} \quad ; \quad \boldsymbol{\nabla}\times\mathbf{B} = \mu_0\mathbf{J} \quad ; \quad \boldsymbol{\nabla}\cdot\mathbf{B} = 0 \tag{1.55}$$

the vector $\mathbf{b}_s^*$ and the scalar $B_{\|s}^*$ can be expressed as:

$$\mathbf{b}_s^* \equiv \frac{\mathbf{B}}{B_{\|s}^*} + \frac{m_s v_{G\|}}{q_s B_{\|s}^* B}\mu_0\mathbf{J} \tag{1.56}$$

$$B_{\|s}^* \equiv B + \frac{m_s v_{G\|}}{q_s B}\mu_0\mathbf{b}\cdot\mathbf{J} \tag{1.57}$$

Besides, using the relation [5]

$$\mathbf{b}\times\left[\mathbf{b}\times\left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)\right] = \left[(\boldsymbol{\nabla}\times\mathbf{b})\cdot\mathbf{b}\right]\mathbf{b} - \left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right) \tag{1.58}$$

[5] Let $\mathbf{U}$,$\mathbf{V}$ and $\mathbf{W}$ being three vectors, $\mathbf{U}\times(\mathbf{V}\times\mathbf{W}) = (\mathbf{U}\cdot\mathbf{W})\,\mathbf{V} - (\mathbf{U}\cdot\mathbf{V})\,\mathbf{W}$ then

$$\mathbf{b}\times\left[\mathbf{b}\times\left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)\right] = \left[\mathbf{b}\cdot\left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)\right]\mathbf{b} - (\mathbf{b}\cdot\mathbf{b})\left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right) = \left[\left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)\cdot\mathbf{b}\right]\mathbf{b} - \left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)$$

and due to:

$$\boldsymbol{\nabla}\times\mathbf{b} = \boldsymbol{\nabla}\times\left(\frac{\mathbf{B}}{B}\right) = \frac{1}{B}\left(\boldsymbol{\nabla}\times\mathbf{B}\right) + \mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B}$$

then

$$\mathbf{b}\times\left[\mathbf{b}\times\left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)\right] = \left[\left(\boldsymbol{\nabla}\times\mathbf{b} - \mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B}\right)\cdot\mathbf{b}\right]\mathbf{b} - \left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)$$

it can be easily proven that [6]

$$\mathbf{b}_s^* = \mathbf{b} + \frac{m_s v_{G\|}}{q_s B_{\|s}^*}\mathbf{B} \times \left(\frac{\mu_0}{B^2}\boldsymbol{\nabla} p\right) \tag{1.59}$$

Therefore, the velocity $v_{G\|}\mathbf{b}_s^*$ is equal to the parallel motion $v_{G\|}\mathbf{b} \sim \mathcal{O}(1)$ plus a diamagnetic velocity $v_{\mathrm{dia}_s} = (m_s v_{G\|}^2)/(q_s B_{\|s}^*)\mathbf{B} \times \left(\mu_0 \boldsymbol{\nabla} p/B^2\right) \sim \mathcal{O}(\epsilon_a \epsilon_g)$. Let us introduce the dimensionless number $\beta \equiv p/(B^2/2\mu_0)$, the ratio of the plasma kinetic pressure over the magnetic pressure. In tokamak, β is of the order of few percents. Then in the low β limit, the diamagnetic part of $\mathbf{b}_s^*$ can be neglected and $\mathbf{b}_s^* \approx \mathbf{b}$.

1.4.3 Gyro-average operator

The gyro-radius $\boldsymbol{\rho}_s$ is transverse to $\mathbf{b} = \mathbf{B}/B$ and depends on the gyrophase angle φ_c:

$$\boldsymbol{\rho}_s = \frac{\mathbf{b} \times \mathbf{v}}{\Omega_s} = \rho_s \left(\cos\varphi_c\, \mathbf{e}_{\perp 1} + \sin\varphi_c\, \mathbf{e}_{\perp 2}\right) \tag{1.60}$$

Here $\mathbf{e}_{\perp 1}$ and $\mathbf{e}_{\perp 2}$ are the unit vectors of a cartesian basis in the plane perpendicular to the magnetic field direction $\mathbf{b}$. Let $\mathbf{x}_G$ be the guiding-center radial coordinate and $\mathbf{x}$ the position of the particle in the real space. These two quantities differ by a Larmor radius $\boldsymbol{\rho}_s$ as $\mathbf{x} = \mathbf{x}_G + \boldsymbol{\rho}_s$. The gyro-average $\bar{g}$ of any function g depending on the spatial coordinates corresponds to the following operation:

$$\bar{g}(\mathbf{x}_G, v_\perp) = \oint_0^{2\pi} \frac{\mathrm{d}\varphi_c}{2\pi} g(\mathbf{x}) = \left\{\oint_0^{2\pi} \frac{\mathrm{d}\varphi_c}{2\pi} \exp(\boldsymbol{\rho}_s \cdot \boldsymbol{\nabla})\right\} g(\mathbf{x}_G) \tag{1.61}$$

The operator $\mathrm{e}^{\boldsymbol{\rho}_s \cdot \boldsymbol{\nabla}}$ corresponds to the change of coordinates $(\mathbf{x}, \mathbf{p}) \to (\mathbf{x}_G, \mathbf{p}_G)$. This gyro-average process consists in computing an average on the Larmor circle. It weakens fluctuations that develop at sub-Larmor scales. Introducing $\hat{g}(\mathbf{k})$ the Fourier transform of g, with $\mathbf{k}$ the wave vector, it is possible to prove that the gyro-average operation reads [7]:

$$\bar{g}(\mathbf{x}_G, v_\perp) = \int_{-\infty}^{+\infty} \frac{\mathrm{d}^3\mathbf{k}}{(2\pi)^3}\, J_0(k_\perp \rho_s) \hat{g}(\mathbf{k}) \mathrm{e}^{i\mathbf{k}\cdot\mathbf{x}_G} \tag{1.62}$$

where $k_\perp$ is the norm of the transverse component of the wave vector $\mathbf{k}_\perp = \mathbf{k} - (\mathbf{b}\cdot\mathbf{k})\mathbf{b}$, and J_0 is the Bessel function [8] of first order. In the following, we shall denote indistinctly $\mathcal{J}_\mu \phi$ or $\bar{\phi}$ the gyro-average of any scalar field ϕ.

1.4.4 Poisson equation with adiabatic electrons

Now that the gyroaverage operator and the gyrokinetic Vlasov equation have been presented, let us introduce two simplifications of Maxwell's equations which will be used in the following, namely (i) electromagnetic effects are ignored and (ii) electrons are considered adiabatic. For this, let us differentiate the electron species labeled e, who have a charge $q_e = -e$ and a mass m_e, from the ion species – each s species of ion being of charge $q_s = Z_s e$ and of mass m_s– then the Maxwell-Gauss equation (1.13) reads:

$$-\nabla^2 U = \frac{1}{\epsilon_0}\Big(\sum_{\substack{s \\ \text{(except} \\ \text{electrons)}}} q_s n_s - e n_e\Big) \tag{1.63}$$

[6] According to (1.41), B can be expressed as $B = B_{\|s}^* - \frac{m_s}{q_s} v_{G\|}\mathbf{b}\cdot(\boldsymbol{\nabla}\times\mathbf{b})$. Let us then replace this expression in (1.43) and use (1.58), then $\mathbf{b}_s^* = \mathbf{b} + \frac{m_s v_{G\|}}{q_s B_{\|s}^*}\left(\mathbf{b} \times \left[\mathbf{b} \times \left(\frac{\boldsymbol{\nabla}\times\mathbf{B}}{B}\right)\right]\right)$. Finally, using ideal MHD equations (1.55) gives equation (1.59).

[7] Due to expression (1.61),

$$\begin{aligned}\bar{g}(\mathbf{x}_G, v_\perp) &= \int_0^{2\pi} \frac{\mathrm{d}\varphi_c}{2\pi} \int_{-\infty}^{+\infty} \frac{\mathrm{d}^3\mathbf{k}}{(2\pi)^3}\, \hat{g}(\mathbf{k}) \exp\{i\mathbf{k}\cdot(\mathbf{x}_G + \boldsymbol{\rho}_s)\} \\ &= \int_{-\infty}^{+\infty} \frac{\mathrm{d}^3\mathbf{k}}{(2\pi)^3} \left[\int_0^{2\pi} \frac{\mathrm{d}\varphi_c}{2\pi} \exp(ik_\perp \rho_s \cos\varphi_c)\right] \hat{g}(\mathbf{k}) \exp(i\mathbf{k}\cdot\mathbf{x}_G)\end{aligned}$$

[8] The Bessel function of the first kind $J_\pm(z)$ are defined as $J_n(z) = \frac{i^{-n}}{\pi}\int_0^\pi \exp(iz\cos\theta)\cos(n\theta)\,\mathrm{d}\theta$.

Ion density

In the gyrokinetic framework, the additional complexity comes from the fact that each particle density $n_s = \int \mathrm{d}^3\mathbf{v} F_s$ needs being related to the distribution function of the gyro-centers $\bar{F}_s$. Both distribution functions relate to each other in the following way:

$$F_s(\mathbf{x},\mathbf{v},t) = \bar{F}_s(\mathbf{x}_G,\mathbf{v}_G,t) + \frac{q_s}{B}\Big\{U(\mathbf{x},t) - \bar{U}(\mathbf{x}_G,\mathbf{v}_G,t)\Big\}\,\partial_\mu \bar{F}_{s,\mathrm{eq}}(\mathbf{x}_G,\mathbf{v}_G) \tag{1.64}$$

with $\bar{F}_{s,\mathrm{eq}}$ the equilibrium distribution function. Such a relationship results from the infinitesimal contact (or canonical) transformation that relates the canonical variables $(\mathbf{x},\mathbf{v})$ and $(\mathbf{x}_G,\mathbf{v}_G)$, as detailed in appendix A. Therefore, after an integration by parts, the particle density n_s reads:

$$n_s(\mathbf{x},t) = \int \mathrm{d}^3\mathbf{v}\bar{F}_s(\mathbf{x}_G,\mathbf{v}_G,t) + \int \mathrm{d}^3\mathbf{v}\frac{q_s}{B}\bar{F}_{s,\mathrm{eq}}(\mathbf{x}_G,\mathbf{v}_G)\,\partial_\mu \bar{U}(\mathbf{x}_G,\mathbf{v}_G,t) \tag{1.65}$$

The first integral on the right hand side corresponds to the gyro-center density $n_{G_s}(\mathbf{x},t)$. The last integral on the right hand side is the polarization density $n_{s,\mathrm{pol}}$. Further recalling that $\mathbf{x}_G = \mathbf{x} - \boldsymbol{\rho}_s$, it follows that

$$n_s(\mathbf{x},t) = n_{G_s}(\mathbf{x},t) + n_{s,\mathrm{pol}}(\mathbf{x},t) \tag{1.66}$$

with:

$$n_{G_s}(\mathbf{x},t) = \int \mathcal{J}_\mathrm{v}\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\;J.\bar{F}_s(\mathbf{x},\mathbf{v},t) \tag{1.67}$$

$$\begin{aligned} n_{s,\mathrm{pol}}(\mathbf{x},t) = & \int \mathcal{J}_\mathrm{v}\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\frac{q_s}{B}\int_0^{2\pi}\frac{\mathrm{d}\varphi_c}{2\pi}\;\Big\{\mathrm{e}^{-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}F_{s,\mathrm{eq}}(\mathbf{x},\mathbf{v}) \\ & \partial_\mu\left[\mathrm{e}^{-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}\left\langle \mathrm{e}^{\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}\right\rangle U(\mathbf{x},\mathbf{v},t)\right]\Big\} \end{aligned} \tag{1.68}$$

where $\mathcal{J}_\mathrm{v} = (2\pi B^*_{\|s}/m_s)$ stands for the Jacobian in the velocity space.

It is interesting to notice that, in the limit of $k_\perp\rho_s \ll 1$, one can restrict the Taylor expansion of the operator to the leading terms of the development only. Further noticing that $\boldsymbol{\rho}_s$ is calculated at $\mathbf{x}_G$, such that $\boldsymbol{\rho}_s(\mathbf{x}_G) \simeq \boldsymbol{\rho}_s(\mathbf{x}) - (\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s$, then (with the convention that $\boldsymbol{\rho}_s$ stands for $\boldsymbol{\rho}_s(\mathbf{x})$ hereafter):

$$\begin{aligned} n_{s,\mathrm{pol}}(\mathbf{x},t) \simeq & \int \mathcal{J}_\mathrm{v}\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\frac{q_s}{B}\int_0^{2\pi}\frac{\mathrm{d}\varphi_c}{2\pi}\;(1-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}) \\ & F_{s,\mathrm{eq}}(\mathbf{x},\mathbf{v})\partial_\mu\left[\left(1-((\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s)\cdot\boldsymbol{\nabla}+\frac{1}{2}\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\right)\right. \\ & \left.\left(1+\frac{1}{4}\rho_s\boldsymbol{\nabla}_\perp.(\rho_s\boldsymbol{\nabla}_\perp)\right)\right]U(\mathbf{x},t) \\ = & \int \mathcal{J}_\mathrm{v}\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\frac{q_s}{B}\Big\{F_{s,\mathrm{eq}}\partial_\mu\left[\frac{1}{4}\left(\boldsymbol{\nabla}_\perp\rho_s^2\right)\cdot\boldsymbol{\nabla}_\perp\right. \\ & \left.+\frac{1}{2}\rho_s\boldsymbol{\nabla}_\perp\cdot(\rho_s\boldsymbol{\nabla}_\perp)\right]U+\frac{1}{2}\rho_s\boldsymbol{\nabla}_\perp F_{s,\mathrm{eq}}\cdot\partial_\mu(\rho_s\boldsymbol{\nabla}_\perp)U\Big\} \\ = & \boldsymbol{\nabla}_\perp\cdot\left(\frac{m_s n_{s,\mathrm{eq}}}{q_s B^2}\boldsymbol{\nabla}_\perp U(\mathbf{x},t)\right) \end{aligned} \tag{1.69}$$

Adiabatic electrons

The polarisation density of electrons can usually be neglected because of their small inertia. Besides, if the electron density is supposed to follow an adiabatic (Boltzmann) response in a flux surface, the density of electrons reads:

$$n_e(\mathbf{x},t) = n_{e_0}(r)\exp\left(e\,\frac{[U(\mathbf{x},t) - \langle\, U\,\rangle_{\mathrm{FS}}(r,t)]}{T_e(r)}\right) \tag{1.70}$$

where the density n_{e_0} is the density defined when $U = \langle U \rangle_{\rm FS}$ with $\langle U \rangle_{\rm FS}$ representing the flux surface average of U defined as:

$$\langle U \rangle_{\rm FS} = \frac{\int U(r,\theta,\varphi)\mathcal{J}_{\rm x}(r,\theta)\ \mathrm{d}\theta\,\mathrm{d}\varphi}{\int \mathcal{J}_{\rm x}(r,\theta)\ \mathrm{d}\theta\,\mathrm{d}\varphi} \tag{1.71}$$

where $\mathcal{J}_{\rm x}(r,\theta)$ stands for the jacobian in space. Due to this hypothesis, the electron density fluctuations vanish for axi-symmetric zonal modes. Provided U is small, the previous electron density equation (1.70) can be linearized, hence:

$$n_e \approx n_{e_0} + e\, n_{e_0} \frac{[U - \langle U \rangle_{\rm FS}]}{T_e} + \mathcal{O}\left[\left(\frac{eU}{T_e}\right)^2\right] \tag{1.72}$$

Using the definition of the ion guiding-center density (1.66), (1.67) and the definition of the electron density (1.72) then the Poisson's equation (1.63) reads in the electron adiabatic limit:

$$\begin{aligned} -\epsilon_0 \nabla^2 U \quad &- \quad \sum_s Z_s e \nabla_\perp \cdot \left(\frac{n_{s,\rm eq}}{B\Omega_s}\nabla_\perp U(\mathbf{x},t)\right) \\ &+ \quad e^2 n_{e_0}\left[\frac{U - \langle U \rangle_{\rm FS}}{T_e}\right] = e\left(\sum_s Z_s n_{G_s} - n_{e_0}\right) \end{aligned} \tag{1.73}$$

where the expression of n_{G_s} is given by (1.67).
Several gyrokinetic codes, among which GYSELA code, are based on this "simplified" Vlasov-Poisson model. But even with such simplifications, solving the gyrokinetic Vlasov equation (1.44) –where the distribution function evolves in a five dimensional phase space– non-linearly coupled to this 3D Poisson equation (1.73) is still extremely difficult. Solving 5D non-linear gyrokinetic equations for several ion species proves so challenging that to date no code is able to treat all the physics involved. Several strategies based on different simplifications –both in terms of physics or numerical approaches– have been developed to decrease these extreme numerical costs. These main differences are summarized in the next section.

1.5 Main differences in gyrokinetic codes

As previously shown, the minimal set of equations to describe the gyrokinetic turbulence in tokamak plasmas can be summarized as: (i) an evolution equation of the distribution function F_s in a 4D phase space parametrized by the adiabatic invariant μ; (ii) a set of four coupled ordinary differential equations (ODE) for the characteristics; (iii) a 3D integro-differential equations for the fields, namely Poisson's equation (or the quasi-neutrality equation) and (iv) the Ampère law (if magnetic perturbations are taken into account). This set of equations is nonlinear, the dominant quadratic nonlinearity being due to the $\mathbf{E} \times \mathbf{B}$ advection terms. The quasi-neutrality equation is generally solved by using Fourier projection in all the periodic directions and finite differences or finite elements in the others. Concerning the global algorithm, the challenge consists in finding numerical schemes which preserve the first principles such as the conservation of the Casimir invariants, the phase space volume and the total energy. Various numerical schemes have been explored until now. They can be classified in Lagrangian, Eulerian and Semi-Lagrangian. In paragraph 1.5.1, we briefly recall the concepts of each numerical approaches as well as their advantages and drawbacks. More details can be found in paper [GS13] published in Panorama & Synthèse review after the tutorial I had given at CEMRACS summer school in 2010 at Marseille in France (attached at the end of this manuscript). The interest of developing concurrently these three approaches is crucial for the gyrokinetic community. Indeed, due to the extremely challenging computing requirements, each gyrokinetic code runs at the limit of its applicability range. Benchmarks between the different approaches are therefore crucial and more and more encouraged (*e.g.* [TGL+08, FSA+08]). They are all the more important that gyrokinetic codes also differ in terms of physics assumptions. As briefly described in paragraphs 1.5.2 to 1.5.4, differences are mainly due to choices in terms of: (i) distribution function representation; (ii) geometry and (iii) boundary conditions. We encourage the reader to refer to the paper by Garbet et al. (2010)

[GIVW10] for an overview of these different strategies for numerical simulations and their comparisons with fusion experiments.

1.5.1 Three numerical approaches : PIC, Eulerian or Semi-Lagrangian

Particle In Cell (PIC) method

The existing gyrokinetic codes differ by their numerical schemes which have evolved all along the last twenty-five years, in direct link with the evolution of HPC resources. Historically, particle in cell methods (PIC) [BL85] –pioneered for gyrokinetics by Lee [Lee83]– have been most popular, and represent widely adopted approaches to numerical simulations of kinetic plasmas. The Lagrangian-PIC approach looks for solutions of the Vlasov equation in terms of the solution of the ODE of motion for macro-particles. Each macro-particle represents a large number of the plasma particles. The particle orbits are the characteristics of the Vlasov equation. In this case, the phase space density along the trajectories is preserved by construction (Lagrange), while charge and current densities need being computed, by extrapolation, on Eulerian (i.e. fixed in space) grid points. PIC codes have the enormous advantages of being simple, robust and easily scalable; which explains that many gyrokinetic codes are PIC codes. Let us mention, Parker's code [PL93], Sydora's code [SDD96], PG3EQ [DWBC96], GTC [LHL+98], GT3D [ITKW01], GTS [WHL+07] and for the 5D gyrokinetic codes: ORB5 [BPH+07, JBA+07], ELMFIRE [HKKS+01] and XGC1 [CKD+09]. However, it is well known that the relative numerical noise inherent to PIC methods constitutes a strong limiting factor to accurately describe the distribution function in phase space on long time scales. The main problem for non-linear gyrokinetic simulations is that the noise level *a priori* accumulates in time [NHD+05]. Even small errors in the evaluation of these moments can cause a systematic corruption of the simulation results on relatively short periods of time. Consequently, the reduction of numerical noise has been right from the start a matter of intense research, and many improvements have taken place during the past ten years. Let us specifically mention the achievements made on the ORB5 gyrokinetic PIC-code summarized in [VBB+10].

Eulerian method

Another approach to avoid the issue of marker sampling noise is the Eulerian approach. It consists in discretizing the phase space on a fixed grid, and in applying finite differences, finite volumes and/or Fourier transforms to model the differential and integral operators (see *e.g.* [Bö7] for a review). Eulerian schemes are not subject to the issue of marker sampling noise which is critical in Lagrangian-PIC methods. Conversely, when explicit time integration is performed, they are limited by the Courant-Friedrichs-Lewy (CFL) stability condition, which constraints the maximum time step as a function of grid space resolution. Comparison between PIC and Eulerian approaches in terms of operation numbers is detailed in *e.g.* [Bö7, FB05]. Several gyrokinetic codes are based on this approach as proved by this non-exhaustive list of Eulerian codes: GS2 [DJKR00], GYRO [CW03b], GENE [JDKR00], GKW [PS04] and GT5D [IIK+08].

Semi-Lagrangian method

The purpose of the semi-Lagrangian method is to take advantage of both the Lagrangian and Eulerian approaches, with an accurate description of the phase space, in particular regions where the density is low, as well as an enhanced numerical stability. This method was primarily developed by Cheng and Knorr [CK76]. It has been cast in more general framework of SL by Sonnendrücker in 1998 [SRBG99] and Nakamura in 1999 [NY99]. In this approach, the phase-space mesh grid is kept fixed in time (Eulerian method) and the Vlasov equation is integrated along the trajectories (Lagrangian method) using the invariance of the distribution function along the trajectories. The GYSELA code is based on this standard Semi-Lagrangian approach [SRBG99]. So a complete description of this numerical scheme will be given in section 3.1.

1.5.2 δf codes / full-f codes

Gyrokinetic models can be split in two distinct families with respect to the adopted representation of the distribution function: either full-f or δf models. In the δf model, only perturbations with respect to some prescribed background equilibrium (usually Maxwellian in velocity) are computed. This method [DK95] has been widely used from the beginning of gyrokinetic code development. Its main advantage is to reduce statistical noise, which can become prohibitive in PIC codes. Technically, the distribution function F_s is decomposed into the time-independent background $F_{s,0}$ and the time-dependent perturbation δF:

$$F_s = F_{s,0} + \delta F$$

The background $F_{s,0}$ is the prescribed distribution function at time $t = t_0$. It is one of the stationary solutions of the gyrokinetic equation at vanishing gyro-averaged potential $J_{0_s} \cdot \phi = 0$. Perturbations δF are then governed by the following equation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\delta F = -\frac{\mathrm{d}}{\mathrm{d}t}F_{s,0} = -\frac{\partial}{\partial t}F_{s,0} - \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}\cdot\frac{\partial}{\partial \mathbf{Z}}F_{s,0} = -\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}\cdot\frac{\partial}{\partial \mathbf{Z}}F_{s,0} \tag{1.74}$$

It is clear that, contrary to the full distribution function F_s, δF is not conserved along the trajectories. Indeed, the right-hand side of eq.(1.74) is not equal to zero. As far as $F_{s,0}$ is concerned, it is usually assumed Maxwellian $F_{s,0} = F_{s,\mathrm{eq}}$ ($F_{s,\mathrm{eq}}$ being an equilibrium Maxwellian distribution function). Trajectories are decomposed into equilibrium and perturbed components, i.e.

$$\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} = \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{eq}} + \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{perturb}}$$

The usual form of the $\dot{\delta F}$ equation then reads as follows:

$$\frac{\partial}{\partial t}F_{s,\mathrm{eq}} = \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{eq}}\cdot\frac{\partial}{\partial \mathbf{Z}}F_{s,\mathrm{eq}} = 0 \tag{1.75}$$

$$\text{and} \quad \frac{\mathrm{d}}{\mathrm{d}t}\delta F = -\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{perturb}}\cdot\frac{\partial}{\partial \mathbf{Z}}F_{s,\mathrm{eq}} \tag{1.76}$$

Should all terms be retained in these equations, they would be rigorously equivalent to the originate gyrokinetic equation. However, let us mention a few simplifications likely leading to discrepancies with respect to the exact gyrokinetic equation: (i) $F_{s,0}$ may not be a *true* equilibrium of the system at vanishing electric field, such that $\partial F_{s,0}$ is not exactly zero (eq.(1.75)); (ii) a few non-linear terms in the left-hand-side of eq. (1.76) are sometimes neglected, assuming that they are high order with respect to the right-hand-side terms. In particular, if the flux-surface averaged part of δF is frozen in time, one is left with the so-called gradient-driven regime (see section 1.5.4).

Conversely, in full-f models, the whole distribution function is evolved. Especially, the back reaction of turbulent transport is accounted for in the time evolution of the equilibrium. Full-f models are specifically capable of addressing the relaxation dynamics of equilibrium profiles, either due to fast non-linear transients or governed by the slow drift of the initial profiles towards their relaxed state in the presence of saturated turbulence. Notice that transients are of great importance in confinement devices. The formation of Internal Transport Barrier in the plasma core is one of such examples. In full-f simulations, the turbulent regime is evanescent if no free energy is injected in the system to prevent the inevitable relaxation of equilibrium profiles below the – linear or non-linear – threshold of the underlying instability. A heat source is mandatory in view of exploring the long time – on energy confinement times – behavior of turbulence and transport.

1.5.3 Local geometry / global geometry

Accounting for the toroidal magnetic geometry is not trivial and introduces strong anisotropy in the low frequency perturbations. While $k_\parallel \rho_s$ is ordered as a small quantity in the gyrokinetic ordering (1.36), the perpendicular wavenumber is not assumed small, $k_\perp \rho_s \sim \mathcal{O}(1)$. Fluctuations typically have parallel wavelengths of the order of the system size, whereas perpendicular wavelengths are of the order of a few Larmor radii. Moreover, core plasmas are weakly collisional, characterized by mean

free paths larger than the system size. The geometry of the magnetic configuration strongly affects micro-instabilities and turbulence. This strong anisotropy can become an advantage in numerical simulations using field-aligned coordinates instead of toric ones. These coordinates can lead to an order of magnitude improvement of numerical schemes. A further simplification is made in *flux-tube* codes, in which the considered domain remains in the vicinity of a magnetic field line. Scale separation is assumed, fluctuations being at smaller scale than the equilibrium. In this framework, the equilibrium profiles, and more precisely their gradients, are taken constant in time. In such codes (*e.g.* GS2 [DJKR00], GYRO [CW03b]), periodicity is assumed for the fluctuations along the radial direction. Conversely, *global codes* take the geometry of the whole plasma domain into account. In this case, the radial periodicity assumption is irrelevant. As a result, the delicate problem of radial boundary conditions has to be solved.

The flux-tube approach allows for an efficient reduction of CPU time and memory size consumption. Conversely, modeling the whole plasma domain requires huge meshes. This drawback is counterbalanced by the fact that global codes can describe phenomena such as profile shearing, profile relaxation and large scale transport events such as avalanches (see section 1.5.4), although the latter have recently been also observed in local simulations. The wealth of such physics has encouraged the development of a new generation of global codes since 15 years. Let us mention, for instance, ORB5 [JBA+07], GT5D [IIK+08] or GYSELA [GSA+07] for core plasmas, and XGC1 [CK08, CKD+09] for edge plasmas. Even the flux-tube code GENE [JDKR00] has recently given birth to a global version [LBD+08].

1.5.4 Fixed gradient / flux driven systems

Let us close this section with differences in the treatment of the boundary conditions. As already mentioned, flux-tube codes use periodic boundary conditions in all directions. As far as global codes are concerned, periodicity is lost in the radial direction. Until recently, gyrokinetic simulations set the system out of equilibrium by imposing two thermal baths as radial boundary conditions. As shown in [GSA+07], the relaxation of the mean profiles in the center of the radial domain can lead to strong gradients at the radial edges, unless buffer regions are added. This evolution stops when the core gradients reach the instability threshold. It typically takes place on energy confinement time scales. Limited statistics are available to investigate the physics of scaling laws in this case, unless running very small ρ_* simulations. Alternatively, an ad-hoc heat source can be added to force the gradients out of thermodynamical equilibrium. The impact of the characteristics of the source on the turbulent transport properties is certainly an issue. Several types of sources have been developed. A Krook operator is used in [MJT+08, MJT+09]. In this case, however, the driving flux is not prescribed *a priori*: since the Krook term depends on the actual distribution function, it evolves in time. Conversely, implementing a heat source which is independent of the distribution function provides a way to study forced turbulence at constant-in-time incoming flux. Gradients then self-adjust in response to this flux as a result of turbulent (and possibly collisional) transport. Such a forcing was successfully exploited in fluid simulations of turbulent transport. Let us remark that, in gyrokinetics, the source is *a priori* 5-dimensions. In particular, its expression in the velocity space has to be considered. Its precise choice depends on the number of fluid moments one wishes to excite.

In this chapter the gyrokinetic models commonly used to describe plasma turbulence in tokamaks have been briefly reviewed. As already said, even with several simplifications, developing such gyrokinetic applications is very challenging and not very widespread. for sure, the wide variety of gyrokinetic codes, coming from all these different choices, is a strength for the fusion community. Indeed, due to the extremely challenging computing requirements, each gyrokinetic code runs at the limit of its applicability range. For instance significant advances like taking into account kinetic electrons and electromagnetic effects are extremely time consuming for global codes. This has been firstly achieved in the GYRO code [CW03a], then in the GENE code [GLB+11] and in the ORB5/NEMORB[9] code

[9]NEMORB seems to have been the first electromagnetic branch of ORB5. Apparently, the ORB groups have meanwhile decided to avoid distinction between the different branches. So for the rest of the paper, we will refer to the code

[BSB+10] and more recently in the GKW code [HMB+15]; this list is not exhaustive. Conversely, the full-f and flux-driven regimes are two necessary ingredients to investigate, among others, the mechanism underlying self-consistent transport barrier creation. Benchmarks between the different approaches are therefore primordial (*e.g.* [TGL+08, FSA+08]). Difficulties reside generally in finding common domains of validity (set of parameters, initial and boundary conditions,...).
After these generalities, the rest of the manuscript is more specifically dedicated to one of these application: the electrostatic non-linear 5D gyrokinetic code GYSELA . Within the community of the 5D gyrokinetic codes, GYSELA is close to GT5D code [INJ14] in the sense that they are both global full-f flux-driven codes. They mainly differ by: (i) their numerical schemes – GT5D is based on a fourth-order non-dissipative conservative finite difference scheme [IITV07, IIK+08] which conserves both L^1 and L^2 norms–; (ii) and their flux driven choice where the source term is compensated by a sink term in GT5D [Ido14]. In the gyrokinetic code jargon, GYSELA is a *global full-f flux-driven* based on a *Semi-Lagrangian* approach. The two next chapters are dedicated to a detailed description of the choices performed for the code both in terms of physics (chapter 2) as well as in terms of numerics and parallelization (chapter 3).

with the unique ORB5 name.

Chapter 2

GYSELA - A global full-f code

Contents

2.1 Toroidal geometry with simplified magnetic configuration 40
2.1.1 Toroidal coordinate system 40
2.1.2 Simplified magnetic configuration 40
2.1.3 Current definition 41
2.2 Gyrokinetic Vlasov equation 42
2.3 Self-consistent coupling with the quasi-neutrality equation 43
2.4 Gyro-average operator: Padé or integration on the gyro-circles 44
2.4.1 A Padé approximation for the gyro-average operator 44
2.4.2 Integration on the gyro-circles by using hermite interpolation 45
2.5 Initial and boundary conditions 46
2.6 Simplified collision operator recovering neoclassical effects 47
2.7 Collisional transfers between two species 50
2.7.1 Collisional energy transfer 50
2.7.2 Parallel momentum transfer 51
2.8 Flux-driven approach with source terms 51
2.8.1 Sources of heating, momentum and vorticity 51
2.8.2 Energetic particle source 52
2.8.3 Source of impurities 53
2.8.4 Diffusion terms in buffer regions 54

In this chapter we focus on a complete description of the actual validated version of the GYSELA code in terms of equations. The present version of the code deals with a toroidal geometry with a simplified concentric circular magnetic configuration (section 2.1.2) similar to the Tore Supra equilibria. In the following we consider adiabatic electrons but the possibility to address transport of impurities. A kinetic treatment of the electrons is under development in the code but will not be addressed in this manuscript. Then, the time evolution of the full distribution function of each ion species (major species as e.g Deuterium + one minor impurity) is governed by a 5D non-linear gyrokinetic Vlasov equation (section 2.2) self-consistently coupled to a 3D Poisson equation (section 2.3). The required gyro-average operator (section 2.4) that used to be approximated by a Padé expansion in the past versions of GYSELA can now be computed with a direct average on gyro-circles. The intra-species collisions are taken into account via a simplified linearized collision operator (section 2.6) which recovers neoclassical effects [GDPN+09]. We also describe the first inter-species operator implemented in the code (section 2.7). It corresponds to a highly simplified version which ensures momentum and energy transfers between species. A more complete version, satisfying neoclassical results for impurity transport has been recently developed but is still under test. It will not be presented in this manuscript

but more details can be found in [EGS+15]. Problem of initialization and radial boundary conditions inherent to global full-f codes like GYSELA are addressed in section 2.5. Concerning boundary conditions, three modes are available in the code: (i) the *fixed-gradient* mode where the temperature profile is fixed at both radial boundaries, corresponding to decaying turbulence regimes (relaxation of equilibrium profiles below instability thresholds cannot be avoided); (ii) the *gradient-driven* mode, equivalent to the previous one but where gradient profiles are maintained by an artificial Krook-type operator and finally (iii) the *flux-driven mode* (the most often used) where temperature is still fixed at the outer boundary but can evolve freely at the inner one. In the gradient-driven mode, the strength of the drag force of the Krook operator governs the dynamics of the mean (flux surface averaged) gradient profiles: they remain all the more sticked to their initial value since the Krook coefficient ν_k is large. More precisely, significant departures of the mean profiles w.r.t. their initial value are only possible on short time scales ($t \ll \nu_k^{-1}$), while the long time behaviour ensures that they remain unchanged when time averaged. Concerning the flux-driven mode, the turbulence is forced with a constant-in-time incoming flux generated by a heat source independent of the distribution function (section 2.8) leading to possible long-time simulation. A simulation over several confinement times has been recently performed for comparison to Tore-Supra experiments [DPHG+15].

2.1 Toroidal geometry with simplified magnetic configuration

2.1.1 Toroidal coordinate system

Let us introduce the notations used in the paper. We consider a set of coordinates labelled $\{x^i\}$, the metric tensor $\{g_{ij}\}$ is the product of the transposed Jacobian matrix J^T and the Jacobian matrix J, *i.e* $\{g_{ij}\} = J^T J$. For a set of cartesian coordinates X^i, the elements J_{ij} of the Jacobian matrix are defined as $J_{ij} = \partial_{x^j} X^i$. Let g represents the determinant of the metric tensor (*i.e* $g = \det\{g_{ij}\}$), then the Jacobian in space $\mathcal{J}_{\mathrm{x}}$ is defined as $\mathcal{J}_{\mathrm{x}} = \sqrt{g}$ and is equal to $\mathcal{J}_{\mathrm{x}} = \left[\left(\boldsymbol{\nabla} x^1 \times \boldsymbol{\nabla} x^2\right) \cdot \boldsymbol{\nabla} x^3\right]^{-1}$, *i.e* the volume element is $\mathcal{J}_{\mathrm{x}}\, \mathrm{d}^3\mathbf{x}$. The tensor $\{g^{ij}\}$ is the inverse of the tensor $\{g_{ij}\}$. The element of the contravariant metric tensor verifies the relation $g^{ij} = \boldsymbol{\nabla} x^i \cdot \boldsymbol{\nabla} x^j$. With these notations, each vector $\mathbf{A}$ can be defined in terms of its covariant components A_i as $\mathbf{A} = A_i \boldsymbol{\nabla} x^i$ and the equivalent norm is given by $\|\mathbf{A}\| = \sqrt{(A_1)^2 g^{11} + (A_2)^2 g^{22} + (A_3)^2 g^{33}}$. At present, in the code, the coordinate system used is the toroidal one, *i.e* the set of coordinates (x^1, x^2, x^3) is equal to (r, θ, φ) where r is the radial position, θ is the poloidal angle and φ the toroidal angle. Therefore, $g^{11} = g^{rr} = 1$, $g^{22} = g^{\theta\theta} = 1/r^2$, $g^{33} = g^{\varphi\varphi} = 1/R^2$ and $g^{ij} = 0$ for all $i \neq j$. $R(r, \theta) = R_0 + r\cos\theta$ with R_0 the major radius of the torus at the magnetic axis. The Jacobian $\mathcal{J}_{\mathrm{x}}$ is then equal to $\mathcal{J}_{\mathrm{x}} = rR$.

2.1.2 Simplified magnetic configuration

The general form of the magnetic field in an axisymmetric tokamak is

$$\mathbf{B} = I(\chi)\boldsymbol{\nabla}\varphi + \boldsymbol{\nabla}\varphi \times \boldsymbol{\nabla}\chi \tag{2.1}$$

where χ is the opposite of the poloidal magnetic flux [DHCS91], which is a label of magnetic flux surfaces, φ is the geometric angle in the (axisymmetric) toroidal direction and I is a flux function. We can define a poloidal angle θ such that the safety factor q is only a function of χ,

$$q(\chi) \equiv \frac{\mathbf{B} \cdot \boldsymbol{\nabla}\varphi}{\mathbf{B} \cdot \boldsymbol{\nabla}\theta} \tag{2.2}$$

The safety factor q describes the pitch of the magnetic field lines, and can be understood as the number of toroidal revolutions performed for one poloidal revolution following a magnetic field line. An adequate system of toroidal coordinates in tokamak geometry can then be defined by (χ, θ, φ). Note that the poloidal angle obtained here is not the geometric poloidal angle and depends on the structure of the magnetic field. This corresponds to so-called *flux coordinates*, designed for the magnetic field lines to be straight on a given flux-surface. As manipulating a flux as a variable is not always practical, the coordinate χ can be replaced by a radial coordinate r, also a label of flux surfaces, such that χ is

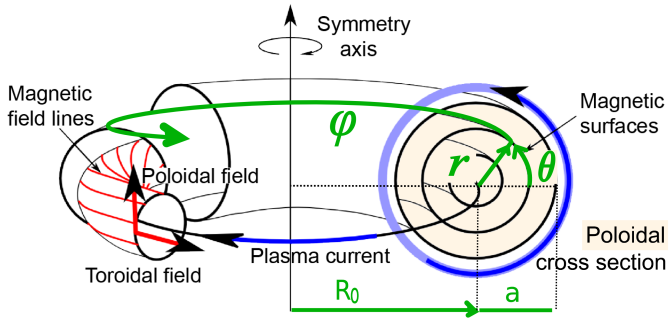


Figure 2.1: The tokamak magnetic configuration and the toroidal coordinate system (r, θ, φ).

a function of r only. This system of toroidal coordinates is represented in Fig. 2.1. The geometry of the torus can be described by its minor radius a and major radius R_0 (at the magnetic axis). For the coordinate system (r, θ, φ), the metric tensor is

$$g^{ij} = \begin{pmatrix} |\nabla r|^2 & \nabla r \cdot \nabla \theta & 0 \\ \nabla r \cdot \nabla \theta & |\nabla \theta|^2 & 0 \\ 0 & 0 & \frac{1}{R^2} \end{pmatrix} \tag{2.3}$$

In Gysela code, a simplified magnetic geometry is adopted, where the poloidal cross-sections of the magnetic surfaces (see Fig. 2.1) are taken as circular and concentric. The magnetic field $\mathbf{B}$ is defined as

$$\mathbf{B} = \frac{B_0 R_0}{R(r,\theta)} \left[\zeta(r)\mathbf{e}_\theta + \mathbf{e}_\varphi\right] \quad \text{with} \quad \zeta(r) = \frac{r}{qR_0} \tag{2.4}$$

where B_0 and R_0 correspond to the magnetic field and the major radius of the torus computed at the magnetic axis while $R(r, \cos\theta) = R_0 + r\cos\theta$. The vectors $\mathbf{e}_\theta = r\nabla\theta$ and $\mathbf{e}_\varphi = R\nabla\varphi$ are the unit vectors in the poloidal and toroidal periodic directions. An example of concentric, circular magnetic flux surfaces is shown in Fig. 2.2 for an aspect ratio $R/a = 3.2$. Let us notice that with this choice of angles the local field line pitch $\mathbf{B} \cdot \nabla\varphi / \mathbf{B} \cdot \nabla\theta$ does not only depend on r but also on θ as

$$\frac{\mathbf{B} \cdot \nabla\varphi}{\mathbf{B} \cdot \nabla\theta} = q(r) \frac{R_0}{R(r,\theta)}$$

In this special case of circular concentric surfaces, $\nabla r \cdot \nabla\theta = 0$ and the metric tensor (2.3) is diagonal and the Jacobian of the metric reads

$$\mathcal{J}_{\mathrm{x}}(r,\theta) = \frac{1}{\mathbf{B} \cdot \nabla\theta} \tag{2.5}$$

In the circular concentric case, the contravariant and covariant components of the magnetic field can be expressed as

$$\begin{array}{lll} B^r = 0, & B^\theta = \mathcal{J}_{\mathrm{x}}(r,\theta)^{-1}, & B^\varphi = I/R^2 \\ B_r = 0, & B_\theta = g^{rr}\mathcal{J}_{\mathrm{x}}(r,\theta)/R^2, & B_\varphi = I \end{array}$$

In the code, the radial safety factor profile $q(r)$ is defined by three parameters q_1, q_2, and q_3 as $q(r) = q_1 + q_2 \exp\left(q_3 \log(r/a)\right)$.

2.1.3 Current definition

As first approximation, the current is decoupled from the field and the magnetic field is assumed to satisfy the Ampère equation, but not the force balance equation. Then the Ampère equation leads to a current of the form:

$$\mu_0 \mathbf{J} = \mu_0 J_T R \nabla\varphi \quad \text{with} \quad \mu_0 J_T = \frac{B_0 R_0}{R} \frac{\zeta}{r} \left(1 + \frac{r}{\zeta}\frac{\mathrm{d}\zeta}{\mathrm{d}r} - \frac{r}{R}\cos\theta\right) \tag{2.6}$$

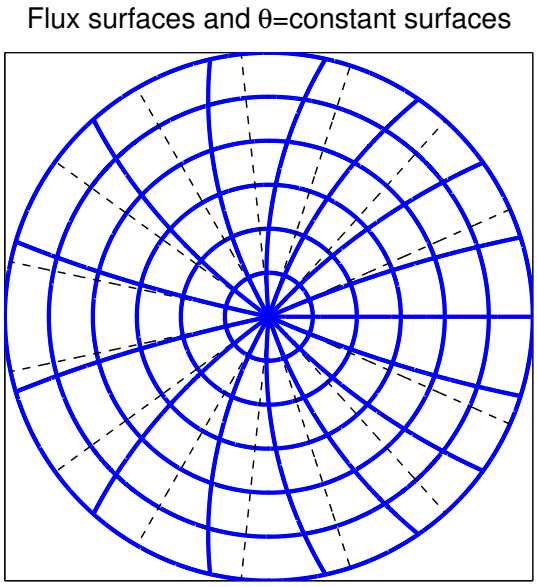


Figure 2.2: Poloidal cross-section showing flux surfaces and contours of θ = constant in the case of circular concentric magnetic configuration.

2.2 Gyrokinetic Vlasov equation

At the opposite of δf codes, GYSELA is a full-f code, namely the full guiding-center distribution function $\bar{F}_s$ is evolved for each species s with no separation between equilibrium and perturbation. Let us consider the gyro-center coordinate system $(\mathbf{x}_G, v_{G\|}, \mu)$ where $\mathbf{x}_G$ corresponds to 3D space coordinates, $v_{G\|}$ is the velocity parallel to the magnetic field line and $\mu_s = m_s v_{G\perp}^2/(2B)$ the magnetic moment. Conventional covariant and contravariant notations are adopted in the following. For reminders on Einstein notations see appendix B.1. In GYSELA code, the non-linear time evolution of $\bar{F}_s$ is governed by the 5D collisional gyrokinetic equation (described by Brizard and Hahm [BH07])

$$B^*_{\|s}\frac{\partial \bar{F}_s}{\partial t} + \boldsymbol{\nabla}\cdot\left(B^*_{\|s}\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t}\bar{F}_s\right) + \frac{\partial}{\partial v_{G\|}}\left(B^*_{\|s}\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t}\bar{F}_s\right) = \mathcal{R}hs(\bar{F}_s) \tag{2.7}$$

with the right hand side $\mathcal{R}hs$ defined as

$$\mathcal{R}hs(\bar{F}_s) = B^*_{\|s}\left(\mathcal{D}_r(\bar{F}_s) + \mathcal{K}(\bar{F}_s) + \mathcal{C}(\bar{F}_s) + \mathcal{S}\right)$$

where $\mathcal{D}_r$ and $\mathcal{K}$ are respectively a diffusion term and a Krook operator applied on a radial buffer region (see section 2.8.4), $\mathcal{C}$ corresponds to a collision operator (see section 2.6) and $\mathcal{S}$ refers to source terms which are detailed in section 2.8.
Let us define the Poisson brackets as $[F, G] = \mathbf{b}\cdot(\boldsymbol{\nabla}F \times \boldsymbol{\nabla}G)$ (see appendix B.2 for detailed expression), and the parallel projection as $\boldsymbol{\nabla}^*_{\|}F = \mathbf{b}^*_s\cdot\boldsymbol{\nabla}F$. Let b_k represents the covariant components of the unitary magnetic field vector $\mathbf{b} = \mathbf{B}/\|B\|$ and $\mathcal{J}_{\mathrm{x}}$ the jacobian in space of the system. Then it can be easily checked that $[F, G] = \mathcal{J}_{\mathrm{x}}^{-1}\epsilon^{ijk}\partial_i F\partial_j G b_k$ where ϵ^{ijk} is the Levi-Civita symbol and $\mathbf{b}^*_s\cdot\boldsymbol{\nabla}F = \mathbf{b}^{*i}_s\partial_i F$. By using this formalism, the evolution of the gyro-center coordinates of species s, involved in the Boltzmann equation (2.7) above, are given (within the electrostatic limit) by:

$$\frac{\mathrm{d}x^i_G}{\mathrm{d}t} = v_{G\|}\mathbf{b}^*_s\cdot\boldsymbol{\nabla}x^i_G + \mathbf{v}_{E\times B_s}\cdot\boldsymbol{\nabla}x^i_G + \mathbf{v}_{D_s}\cdot\boldsymbol{\nabla}x^i_G \tag{2.8}$$

$$m_s\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t} = -\mu_s\mathbf{b}^*_s\cdot\boldsymbol{\nabla}B - q_s\mathbf{b}^*_s\cdot\boldsymbol{\nabla}\bar{U} + K_{\nabla B}\frac{m_s v_{G\|}}{B}\mathbf{v}_{E\times Bs}\cdot\boldsymbol{\nabla}B \tag{2.9}$$

where $\mathbf{b}^*_s$ is defined by equations (1.43) and (1.41) as

$$\mathbf{b}^*_s \equiv \frac{\mathbf{B}}{B^*_{\|s}} + \frac{m_s v_{G\|}}{q_s B^*_{\|s} B}\boldsymbol{\nabla}\times\mathbf{B} \qquad \text{with} \qquad B^*_{\|s} \equiv B + \frac{m_s}{q_s}v_{G\|}\mathbf{b}\cdot(\boldsymbol{\nabla}\times\mathbf{b})$$

The i-th contravariant components of the '$\mathbf{E}\times\mathbf{B}$' drift are given by

$$\mathbf{v}_{E\times B_s}\cdot\boldsymbol{\nabla}x_G^i = \mathbf{v}^i_{E\times B_s} = \frac{1}{B^*_{\|s}}\left[\bar{U}, x_G^i\right] \tag{2.10}$$

and the i-th contravariant components of the 'grad–B' and 'curvature' drifts read (at low $\beta = nT/(B^2/2\mu_0)$ limit)

$$\mathbf{v}_{D_s}\cdot\boldsymbol{\nabla}x_G^i = \mathbf{v}^i_{D_s} = K_{\nabla B}\left(\frac{m_s v_{G\|}^2 + \mu_s B}{q_s B^*_{\|s} B}\right)\left[B, x_G^i\right] \tag{2.11}$$

The constant $K_{\nabla B}$ has been added for tests in the code. $K_{\nabla B}$ is equal to 1 if the curvature of the magnetic field is taken into account and 0 otherwise. Besides, using the fact that the axi-symmetric equilibrium is determined by three of the ideal MagnetoHydroDynamic (MHD) equations $\boldsymbol{\nabla}p = \mathbf{J}\times\mathbf{B}$, $\boldsymbol{\nabla}\times\mathbf{B} = \mu_0\mathbf{J}$ and $\boldsymbol{\nabla}\cdot\mathbf{B} = 0$, the i-th contravariant components of $\mathbf{b}^*_s$ appearing in equation (2.8) read

$$\mathbf{b}^*_s\cdot\boldsymbol{\nabla}x_G^i = \mathbf{b}^{*i}_s = \frac{B\cdot\boldsymbol{\nabla}x^i}{B^*_{\|s}} + \frac{m_s v_{G\|}}{q_s B^*_{\|s}}\frac{\mu_0\mathbf{J}\cdot\boldsymbol{\nabla}x_G^i}{B} \tag{2.12}$$

2.3 Self-consistent coupling with the quasi-neutrality equation

Fusion plasma turbulence typically develops at Larmor scales, which are larger than the Debye length λ_D (i.e $\lambda_{D_s} \ll \rho_{L_s}$). The Larmor radius of thermal Deuterium nuclei is $\rho_D \approx 4.6\,10^{-3}\,T^{1/2}_{[keV]}/B_{[T]}\,m \approx 4.1\,10^{-3}\,m$, for $T = 20keV$ and $B = 5T$, while

$$\lambda_D \approx 2.35\,10^{-5}(T_{[keV]}/n_{10^{20}m^{-3}})^{1/2}m \approx 10^{-4}m$$

for $n = 10^{20}m^{-3}$. The electron Larmor radius ρ_e may eventually compete with λ_D, but both experimental observations and numerical simulations suggest that electron turbulence saturates at scales larger than ρ_e. In such a case, the term $\epsilon_0\nabla^2 U$ in the Poisson's equation (1.73) can be dropped out[1]. This approximation is called the quasi-neutrality constraint and leads to the simple equality $n_e = \sum_s Z_s n_s$. Therefore, assuming that $n_{e_0} = \sum_s Z_s n_{s,\mathrm{eq}}$ and that $n_{s,\mathrm{eq}} \approx n_{G_s,\mathrm{eq}}$ (where $n_{G_s,\mathrm{eq}}$ is the guiding-center equilibrium density of species s), equation (1.73) becomes:

$$-\frac{1}{n_{e_0}}\sum_s Z_s\nabla_\perp\cdot\left(\frac{n_{s,\mathrm{eq}}}{B\Omega_s}\nabla_\perp U(\mathbf{x},t)\right) + e\left(\frac{U - \langle U\rangle_{\mathrm{FS}}}{T_e}\right) = \frac{1}{n_{e_0}}\sum_s Z_s\left(n_{G_s} - n_{G_s,\mathrm{eq}}\right) \tag{2.13}$$

with $\Omega_s = q_s B_0/m_s$ and $\boldsymbol{\nabla}_\perp = \left(\partial_r, \frac{1}{r}\partial_\theta\right)$. Here, the polarization density (first term of eq. (2.13)) is approximated by its expression in the limit of large wavelengths with respect to the Larmor radius (limit $k_\perp\rho_s \ll 1$). The gyro-center density n_{G_s} of species s is defined by equation

$$n_{G_s}(\mathbf{x},t) = \int \mathcal{J}_{\mathrm{v}}\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\;J.\bar{F}_s(\mathbf{x},\mathbf{v},t) \tag{2.14}$$

while the equilibrium part $n_{G_s,\mathrm{eq}}$ corresponds to the same equation where $\bar{F}_s$ is replaced by $\bar{F}_{s,\mathrm{eq}}$, i.e

$$n_{G_s,\mathrm{eq}}(\mathbf{x},t) = \int \mathcal{J}_{\mathrm{v}}\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\;J.\bar{F}_{s,\mathrm{eq}}(\mathbf{x},\mathbf{v},t) \tag{2.15}$$

The concentration $c_s \equiv n_s/n_{e_0}$ is defined such that $\sum_s c_{s_0}Z_s = 1$. The flux surface average $\langle U\rangle_{\mathrm{FS}}$ is given by relation (1.71). In practice, the right hand side ρ of (2.13) is computed as

$$\rho(\mathbf{x},t) = \frac{1}{n_{e_0}}\sum_s Z_s\int \mathrm{d}\mu\mathcal{J}_\mu.\left[\int \mathcal{J}_{\mathrm{v}}\,\mathrm{d}v_{G\|}\,\left(\bar{F}_s(\mathbf{x},\mathbf{v},t) - \bar{F}_{s,\mathrm{eq}}(r,\theta,v_{G\|})\right)\right] \tag{2.16}$$

[1]Due to $\lambda_{D_{is}} = \sqrt{\epsilon_0 k_B T_e/(q_s^2 n_{i_0})}$ then $\epsilon_0 = \frac{m_s}{B_0^2}\frac{T_i}{T_e}n_{i_0}\left(\frac{\lambda_{D_{is}}}{\rho_{L_s}}\right)^2$ because $\rho_{L_s} = \sqrt{T_s m_s}/(q_s B_0)$

To avoid the problem of the singularity in $1/r$, the problem is solved within a ring $r_{\min} \leq r \leq r_{\max}$ where $r_{\min} \geq 1e^{-5}$. The boundary conditions are Dirichlet boundary conditions for U and $\langle U \rangle_{\mathrm{FS}}$ at $r_{\min}$ and $r_{\max}$. One difficulty with equation (2.13) is to deal with the flux surface average term $\langle U \rangle_{\mathrm{FS}}$. This term is non-linear in θ, because the space Jacobian depends on θ. Therefore, it does not allow one to project simply in 2D Fourier space. To overcome this problem the solving of (2.13) is performed in two steps as proposed in [CRS12]. Let us define the differential operator $\mathcal{L} = -\frac{1}{n_{e_0}} \sum_s Z_s \nabla_\perp \cdot \left(\frac{n_{s_0}}{B_0 \Omega_s} \nabla_\perp \cdot \right)$ and $\langle \cdot \rangle_{\theta,\varphi}$ the average on θ and φ directions. Then, the first step consists in finding $\mathcal{U}$ solution of the following differential system,

$$\left(\mathcal{L} + \frac{e}{T_e} \right) \mathcal{U} = \rho - \langle \rho \rangle_{\theta,\varphi} \quad \text{with} \quad \mathcal{U}(r,\theta,\varphi) = U(r,\theta,\varphi) - \langle U \rangle_{\theta,\varphi}(r) \tag{2.17}$$

The second step consists in solving the following 1D radial differential equation

$$\mathcal{L} \langle U \rangle_{\mathrm{FS}} = \langle \rho \rangle_{\theta,\varphi} + \left(\mathcal{L} + \frac{e}{T_e} \right) \langle \mathcal{U} \rangle_{\mathrm{FS}} \tag{2.18}$$

In this equation, φ plays the role of a parameter. Fourier projection is performed in θ direction. In radial direction, for both differential systems (2.17) and (2.18), finite differences are used. Finally, the electrostatic potential is reconstructed with the formula $U = \mathcal{U} - \langle U \rangle_{\mathrm{FS}} + \langle \mathcal{U} \rangle_{\mathrm{FS}}$. For both equations (2.17) and (2.18), Dirichlet boundary conditions are applied at the outer boundary at r_{max}, while Dirichlet or Neumann can be chosen at the inner boundary at r_{min}. Notice that these boundary conditions then apply to $\mathcal{U}$, not to the actual electric potential U. Given the relationship between U and $\mathcal{U}$, imposing the Dirichlet condition $\mathcal{U}(r_{BC}) = 0$, with $r_{BC} = \{r_{min}, r_{max}\}$, is then equivalent to the following conditions on U: $U(r_{BC}) = \langle U \rangle_{\theta,\varphi}(r_{BC})$ and $\langle U \rangle_{FS}(r_{BC}) = 0$. The Neumann condition proceeds in a similar way. In GYSELA, it can be only applied to the inner boundary provided $r_{min} \leq 10^{-2}$. In this case, $J_x(r_{min}, \theta)$ is fairly independent of θ, so that the flux surface average is almost equal to the average over both angles: $\langle \cdot \rangle_{FS} \approx \langle \cdot \rangle_{\theta,\varphi}$. In the framework of this approximation, imposing the Neumann condition $\partial_r \mathcal{U}(r_{min}) = 0$ is equivalent to $\partial_r U(r_{min}) = \partial_r \langle U \rangle_{\theta,\varphi}(r_{min}) \approx \partial_r \langle U \rangle_{FS}(r_{min}) = 0$. The fact that this solution is equivalent to solve directly equation (2.13) and the detailed numerical scheme associated are explained in Appendix C.

2.4 Gyro-average operator: Padé or integration on the gyro-circles

The gyro-average operation is essential for gyrokinetic simulations. This operator ensures the link between the Poisson equation which is solved with the charge density of particles and the Vlasov equation which describe the guiding-center evolution. As schematically described in Figure 2.3 the gyro-average operator is not only applied to the 3D electrostatic potential U (used for advection equations), but also to the 5D guiding-center distribution function $\bar{F}_s$ (for right hand side of Poisson equation). Here is used the fact that $\int J_{0_s} \cdot \bar{F}_s \, \mathrm{d}^3\mathbf{v}$ corresponds to the particle density. Historically, this operator was approximated by a Padé expansion (section 2.4.1). A computation based on direct integration on the gyro-circles has been more recently implemented in the code (section 2.4.2).

2.4.1 A Padé approximation for the gyro-average operator

Considering the expression (1.62), in Fourier space the gyro-average reduces to the multiplication by the Bessel function of argument $k_\perp \rho_s$. This operation is straightforward in simple geometry with periodic boundary conditions, such as in local codes. Conversely, in the case of global codes, the use of Fourier transform is not applicable for two main reasons: (i) radial boundary conditions are non periodic, and (ii) the radial dependence of the Larmor radius has to be accounted for. Several approaches have been developed to overcome this difficulty in the gyrokinetic codes. The first one, currently used in the code, consists in simplifying the treatment of the gyro-average operator by approximating the Bessel function with a Padé expansion $J_{\text{Padé}}(k_\perp \rho_s) = 1/\left[1 + (k_\perp \rho_s)^2/4\right]$ (*e.g.* see [SGF+05]). The advantage of this Padé representation is that it does no longer requires to use the Fourier space as required by the Bessel function. Indeed, since it involves $k_\perp^2$, it can easily be treated

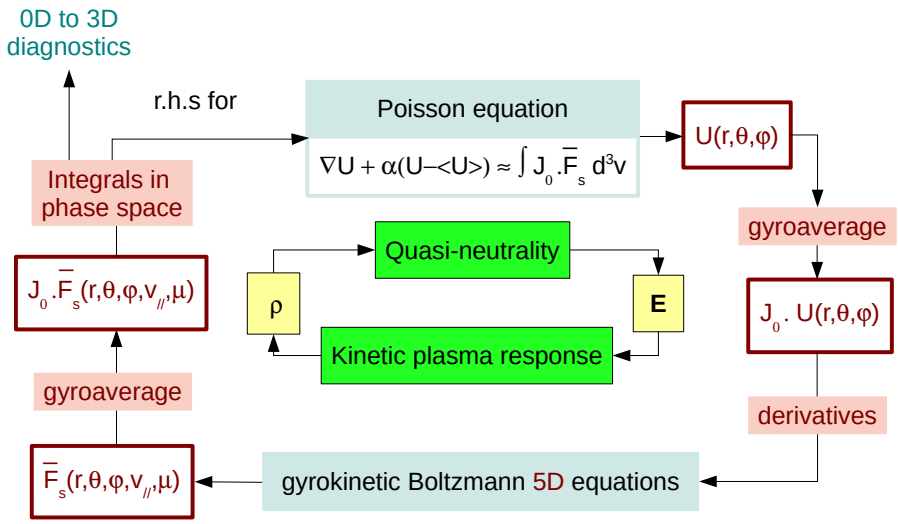


Figure 2.3: Schematic view of the coupling between the Poisson and Vlasov solvers in the GYSELA code.

in the configuration space by using the relation $\nabla_\perp^2 \leftrightarrow -k_\perp^2$. Using this Laplacian equivalence, the gyro-average operation of any g function is defined such that each m Fourier mode of $\bar{g}$ is solution of the equation

$$\left[1 - \frac{1}{2\Omega_s^2}\frac{B_0}{m_s}\mu_s\left(\frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} - \frac{m^2}{r^2}\right)\right]\bar{g}^m(r,\varphi) = g^m(r,\varphi) \tag{2.19}$$

where at first approximation $B(r,\theta)$ has been replaced by B_0 to be consistent with the quasi-neutrality equation. In this differential equation, first and second derivatives are computed using a Taylor expansion of second order leading to a tridiagonal matrix system (see Appendix D for details of implementation in the code). This Padé approximation is asymptotically correct in the large wavelength limit $k_\perp\rho_s \ll 1$ (indeed: $J_0(k_\perp\rho_s) \approx 1 - k_\perp^2\rho_s^2/4$ for $k_\perp\rho_s \ll 1$), while keeping $J_{\text{Padé}}$ finite in the opposite limit $k_\perp\rho_s \to \infty$. The drawback is a filtering of small scales: in the limit of large arguments $x \to \infty$, $J_{\text{Padé}}(x) \approx 4/x^2$, whereas $J_0(x) \approx (2/\pi x)^{1/2}\cos(x - \pi/4)$ (see Figure 2.5).

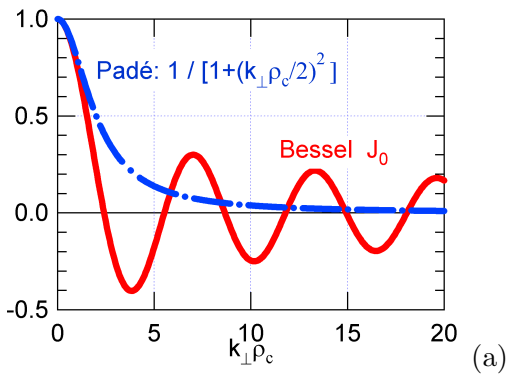


Figure 2.4: The zero-th order Bessel function $J_0(k_\perp\rho_s)$ compare to its Padé approximation $1/\left[1 + (k_\perp\rho_s)^2/4\right]$.

2.4.2 Integration on the gyro-circles by using hermite interpolation

A second widespread method for this gyro-averaging process is to use a quadrature formula. In this context, the integral over the gyro-ring is usually approximated by a sum over four points or more on the gyro-ring [Lee83]. This is rigorously equivalent to considering the Taylor expansion of the Bessel function at order two in the small argument limit, namely $J_0(k_\perp\rho_s) \simeq 1 - (k_\perp\rho_s)^2/4$, and equivalent

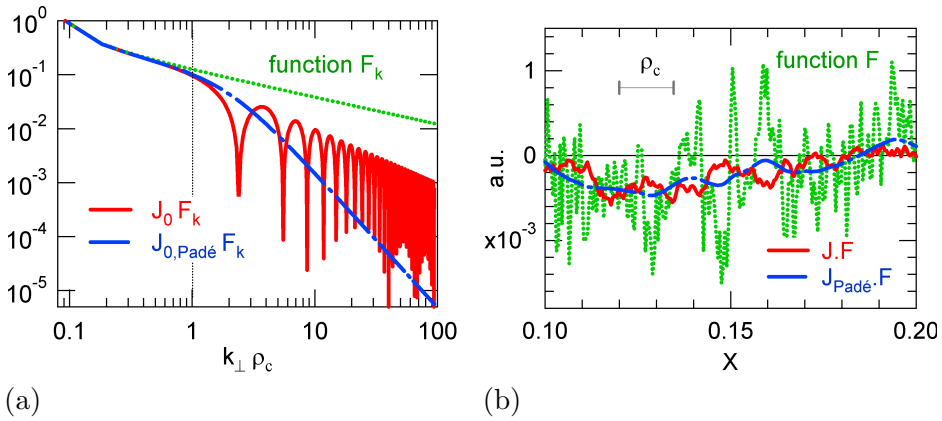


Figure 2.5: Exact and approximated gyro-average operators applied on an arbitrary funcion F_k exhibiting a broad spectrum ranging from low to large wavelengths as compared with the Larmor radius ρ_c: (a) Representation in the Fourier space, (b) Representation in the real space (figures from [SGF+05]).

to computing the transverse Laplacian at second order using finite differences. This method has been extended to achieve accuracy for large Larmor radius [HTK+02], *i.e* the number of points (starting with four) is linearly increased with the gyro-radius to guarantee the same number of points per arclength on the gyro-ring. In this approach –used *e.g.* in [ITKW01] and [JBA+07]– the points that are equidistantly distributed over the ring are rotated for each particle (or marker) by a random angle calculated every time step. This is performed on a finite element formalism and enables therefore high order accuracy by keeping the matricial formulation. In [CMS10a] the influence of the interpolation operator (which is of great importance when the quadrature points do not coincide with the grid points) has been studied and it is shown that the cubic splines are appropriate. The direct integration on gyro-circles proposed in [CMS10a] has been recently generalized to arbitrary coordinates [SMC+15] and implemented in the code. For the distribution function the gyro-average operator is applied on $\bar{F}_s - \bar{F}_{s,\text{eq}}$ to deal with values close to 0 at the domain boundaries. Two interpolations –cubic splines and Hermite polynomial– have been tested on analytical cases and basic gyrokinetic simulations with a 4D drift-kinetic model, one Larmor radius and the standard linear Cyclone benchmark case (see [SMC+15] for more details and comparison with Padé approximation). Both appear to give the same results. However, the Hermite interpolation is slightly faster and its local character is more favourable for parallelization. The number of points per circle is an input parameter comprised between 4 and 16. There is no adaptive number of points depending on the Larmor radius value because the CPU time is in fact determined by the maximum number. As shown in [SMC+15] the method converges with the number of points, so there is no interest in decreasing the number of points at small radius. It is shown in [RSM+15] that 16 points is a good compromise between accuracy and CPU time consumption (2 times slower than the previous Padé approximation due to its higher algorithmic complexity). All numerical results presented in the following are performed with the Padé approximation. The reason for not choosing the Hermite interpolation is just chronological: this latter method has been developed only recently. First analyses of new simulations with gyro-average operators based on Hermite show similar results in terms of conservation laws. The impact on non-linear cases will be addressed in a future paper.

2.5 Initial and boundary conditions

Boundary conditions are periodic in θ and φ directions. Concerning the radial direction, GYSELA is a *global* code, *i.e* it considers a large fraction of the plasma radius. This is in contrast to flux-tube codes which focus on a small volume around magnetic field lines by proceeding from a scale separation assumption, the fluctuation scale length being smaller than that of the equilibrium. In such codes, periodicity is almost always assumed along the radial direction. Conversely, global codes as GYSELA face the delicate problem of radial boundary conditions. Non-axisymmetric fluctuations of the electric potential and of the distribution function –*i.e* $(m, n) \neq (0, 0)$ modes, with m and n the poloidal and

toroidal wave numbers– are forced to zero at both radial boundaries of the simulated domain. As far as the axisymmetric component is concerned, the value of the potential is prescribed at the outer boundary, while the radial electric field is set to zero at the inner boundary. In addition, so as to avoid possible numerical instabilities, which might occur in the case where turbulent fluctuations reach the frontiers of the simulation domain, buffer regions have been added at both radial boundaries. They are characterized by non-vanishing ad-hoc dissipative coefficients, which aim at keeping all gradients finite and damping out all fluctuations (see section 2.8.4). Initial conditions consist of an equilibrium distribution function $\bar{F}_{s,\mathrm{eq}}$ perturbed by a sum of accessible (m, n) Fourier modes (m and n being the poloidal and toroidal wave numbers, respectively). That means, $\bar{F}_s = \bar{F}_{s,\mathrm{eq}} + \delta\bar{F}_s$ where the perturbation part $\delta\bar{F}_s$ reads $\delta\bar{F}_s = \bar{F}_{s,\mathrm{eq}}\, g(r)\, h(v_{G\|})\delta p(\theta, \varphi)$ with $\delta p(\theta, \varphi) = \sum_{m,n} \epsilon \cos(m\theta + n\varphi + \delta_{mn})$ where the amplitude ϵ is fixed and the phases δ_{mn} have arbitrary values. The radial function $g(r)$ (resp. $h(v_{G\|})$) has a polynomial dependence and vanishes at both radial (resp. $v_{G\|}$) boundaries. Concerning the initialization of the equilibrium distribution function two choices are possible: (i) the first one is a local conventional Maxwellian, (ii) and the second one is a *canonical* Maxwellian, *i.e* depends on the motion invariants. The Maxwellian distribution function is defined as

$$\bar{F}_{s,\mathrm{eq}}(r, E) = n_{s_0}(r) \times [2\pi T_s(r)/m_s]^{-\frac{3}{2}} \exp\left(-\frac{E}{T_s(r)}\right) \tag{2.20}$$

where E stands for the kinetic energy $m_s v_{G\|}^2/2 + \mu B(r, \theta)$ which is the second invariant of the system (at vanishing electric potential). The initial radial profiles of the ion temperature and density (respectively $T_s(r)$ and $n_{s_0}(r)$) are deduced by numerical integration of their gradient profiles given by the two parameters κ and Δr: $\mathrm{d}\log \mathrm{T_s(r)}/\mathrm{d}r = -\kappa_{T_s} \cosh^{-2}((r - r_p)/\Delta r_{T_s})$ with r_p corresponding to the middle of the radial box. $\bar{F}_{s,\mathrm{eq}}$ is constant on a magnetic surface labelled by the radial coordinate r. As shown in [DPGS+08a], such initial state does not constitute an equilibrium of the system solved by GYSELA at vanishing electric field. A stationary equilibrium of the collisionless equations of the code must depend on the three motion invariants, namely the adiabatic invariant μ, the total energy E and the toroidal kinetic momentum $P_\varphi = q_s\psi + m_s R v_\varphi$ with ψ the poloidal flux and v_φ the toroidal fluid velocity. In GYSELA, a convenient choice for this equilibrium is provided by the canonical Maxwellian (2.20) in which the radial coordinate r is replaced –as proposed in [ABe06]– by an effective radial coordinate $\bar{r}$, with the dimension of a length, derived from P_φ

$$\bar{r} = r_p - \frac{q_p}{r_p}\Big[\psi(r) - \psi(r_p)\Big] - \frac{m_s q_p}{e B_0 r_p}\Big[R v_{G\|} - R_0 \bar{v_{G\|}}\Big] \tag{2.21}$$

where $\psi(r) = -B_0 \int_0^r r'/q \, \mathrm{d}r'$. The last term $\bar{v_{G\|}}$ in (2.21) is defined as,

$$\bar{v_{G\|}} = \mathrm{sign}(v_{G\|})\sqrt{2/m_s}\sqrt{E - \mu B_{\max}}\,\mathcal{H}(E - \mu B_{\max})$$

with $\mathcal{H}$ the Heaviside function and $B_{\max}$ the maximum of the magnetic field on the whole simulation box. It has been chosen to minimize parallel flows. With this expression the difference between $\bar{r}$ and r is of order ρ_*, the ratio of ρ_s the Larmor radius of species s and the minor radius a. In the case of a decaying turbulence it is important to choose $\bar{F}_{s,\mathrm{eq}}$ as a function of the motion invariants, especially for studying zonal flows. It had been observed in [ITKW01, ABe06] that breaking this rule leads to the development of large scale steady flows, which can prevent the onset of turbulence. This phenomena has also been observed in GYSELA simulations where a study of the difference between both equilibrium initialization has been performed in detail [DPGS+08a, DPGS+08b]. In [DPGS+08b] it has also been shown that when the system is driven by an external source of free energy, the choice of a canonical equilibrium is less crucial. The turbulence onset is only delayed and its ultimate nature is unaltered; namely characterized by the same level of parallel and axisymmetric flows, the same level of transport and the same correlation times and lengths.

2.6 Simplified collision operator recovering neoclassical effects

Although fusion plasmas are weakly collisional, so that a kinetic approach is mandatory, the effect of collisions cannot be fully neglected. Collisional transport plays an important role in regions where

turbulent transport is low, such as transport barriers. Even more importantly, collisional friction damps low frequency flows, and hence controls the equilibrium radial electric field. Collisions also regularize fine structures in velocity space. Finally, collisions have also an impact on microinstabilities, e.g via particle detrapping that is partially stabilizing Trapped Electron Modes (TEMs). Let F_s be the particle distribution function of species s and $F_{s'}$ be the particle distribution function of species s'. The Fokker-Planck equation, which corresponds to the collisional version of the Vlasov equation, can be written as:

$$\mathcal{C}_s(F_s) = \sum_{s'} \mathcal{C}_{ss'}(F_s, F_{s'}) \tag{2.22}$$

to take into account the collisions inter and intra-species. In tokamak configuration, these collisions depend on the Coulomb binary interactions between particles (see Landau [Lan36] for calculations). Then it is shown in [HM03] that the collision operator applied to the distribution functions can take the classical Fokker-Planck operator form:

$$\mathcal{C}_{ss'}(F_s, F_{s'}) = \frac{\partial}{\partial \mathbf{v}} \cdot \left\{ \left(\mathcal{D}(F_{s'}) \cdot \frac{\partial}{\partial \mathbf{v}} + \mathcal{V}(F_{s'}) \right) F_s \right\} \tag{2.23}$$

where $\mathcal{V}$ is a dynamical friction term and $\mathcal{D}$ is a diffusion term of the order of $\mathcal{D} \sim v_{T_s}^2 \nu_{ss'}$ with the collision frequency $\nu_{ss'}$ being defined as

$$\nu_{ss'} \sim \frac{\mathcal{D}}{v_{T_s}^2} \sim \frac{n_{s'}}{m_s} \left(\frac{1}{m_s} + \frac{1}{m_{s'}} \right) \frac{q_s^2 q_{s'}^2 \log \Lambda}{(1+\epsilon_0)^2} \left(v_{T_s}^2 + v_{T_s'}^2 \right)^{-3/2} \tag{2.24}$$

where $n_{s'}$ is the density of species s', q_s (resp. $q_{s'}$) is the particle charge of species s (resp. s'), m_s is the particle mass of species s, ϵ_0 the permittivity of free space, $\log \Lambda \approx 17$ the Coulomb logarithm and $v_{T_s} = \sqrt{T_s/m_s}$ is the thermal velocity of species s. Let us first underline that the frequency $\nu_{ss'}$ is different from $\nu_{s's}$. Besides, let us compare the collision frequencies for the different following particle populations: (i) ion-ion collisions: $\nu_{ii} \propto Z^4 n_i/\sqrt{m_i} T_i^{3/2}$, (ii) electron-electron collisions: $\nu_{ee} \propto n_e/\sqrt{m_e} T_e^{3/2}$, (iii) electron-ion collisions: $\nu_{ei} \approx Z\nu_{ee}$ and (iv) ion-electron collisions: $\nu_{ie} \approx \frac{m_e}{m_i} Z^2 \nu_{ee} \ll \nu_{ii} \ll \nu_{ee}$. Therefore, the ion-electron collisions can be neglected. At the moment in the code, the electrons are considered adiabatic, therefore only the ion-ion collisions and impurity-ion collisions (see section 2.7.1) are taken into account. The full gyro-averaged and linearized Landau operator has been derived in [XR91] but such a full Coulomb collisional operator is difficult to implement in GYSELA without severe loss of parallelisation efficiency. As described in section 3.2, the GYSELA parallelization takes advantage of the fact that the magnetic moment μ is an adiabatic invariant, that plays the role of a parameter in Boltzmann equation. A unique value of μ is assigned to each processor. It was shown that the predictions of the neoclassical theory at low collisionality could be entirely recovered with a reduced collision operator acting in the $v_\parallel$ direction only [DPDG+11]. In short, this results from the fact that the main ions of tokamak plasmas are weakly collisional. In this so-called banana regime, collisions essentially perturb the banana orbits at their turning points, where the parallel velocity of trapped particles vanishes. This corresponds to the trapped-passing boundary in the $(v_\parallel, v_\perp)$ plane. Accounting for diffusion in $v_\parallel$ then reveals sufficient to model such a transport, governed by the broadening of the trapped-passing boundary due to collisions. It also has the advantage of keeping μ invariant, hence not degrading the efficiency of the code parallelization. The operator implemented in the code, is a simplified version of the Lenard-Bernstein operator [LB58]. This simplified version has been derived in [GDPN+09] where it is especially shown that it recovers the exact neoclassical transport in the banana and plateau regimes[2] (see Helander's book [HS05] p.149 for complete neoclassical transport regime description). This generic energy and momentum-conserving collision operator has been implemented and successfully tested in the code (see [AGG+11],

[2]The physics of neoclassical transport depends on the collisionality ν_*. If the collisionality is low, $\nu_* < 1$, the particle orbits are completed by a typical thermal particle. In this so-called *banana* regime, trapped particles almost fully determine the transport coefficients. In the opposite limit, $\nu_* > \epsilon^{-3/2}$, the particle orbit is not fully completed because its motion is disturbed by collisions before. This high collisionality regime is called the *Pfirsch-Schülter* regime or fluid regime. In-between, the *plateau* regime is characterized by a weak dependance of the transport on the collisionality of the plasma.

[DPDG+11]). A new version also valid for the Pfirsch-Schülter regime has been recently derived [EGS+15] and is still under tests in the code. The current collision operator is expressed as a simplified Lorentz-type operator where only the $v_{G\|}$ contribution is taken into account:

$$\mathcal{C}_s(\bar{F}_s) = \frac{1}{B^*_{\|s}} \frac{\partial}{\partial v_{G\|}} \left\{ B^*_{\|s} \mathcal{D}_{\|s} \, \bar{F}_{M_s} \frac{\partial}{\partial v_{G\|}} \left(\frac{\bar{F}_s}{\bar{F}_{M_s}} \right) \right\} \tag{2.25}$$

where $\bar{F}_{M_s}$ is a shifted Maxwellian distribution –chosen such that $\mathcal{C}_s(\bar{F}_{M_s}) = 0$– defined as:

$$\bar{F}_{M_s} = \frac{n_{s_0}}{(2\pi T_{s,\text{coll}}/m_s)^{3/2}} \exp\left(-m_s \frac{(v_{G\|} - V_{\|s,\text{coll}})^2}{2T_{s,\text{coll}}} - \frac{\mu_s B}{T_{s,\text{coll}}} \right) \tag{2.26}$$

The collision term $\mathcal{D}_{\|s}$ reads for each μ_s:

$$\mathcal{D}_{\|s}(r,v) = \mathcal{A}_s(r) \left(\frac{\Phi(v) - G(v)}{2v} \right) \qquad \text{with} \qquad \mathcal{A}_s(r) = 3\frac{\sqrt{\pi}}{2} \frac{v^3_{T_{s,\text{coll}}} \epsilon^{3/2}}{qR_0} \nu_{*s} \tag{2.27}$$

where $v(r, v_{G\|}) = \sqrt{E/T_{s,\text{coll}}}$. The scalar ν_{*s}, associated to the main s species, is a dimensionless ion-ion collisionality parameter depending on the ion-ion collision frequency ν_{ss} such that:

$$\nu_{*s} = \frac{qR_0}{v_{T_{s,\text{coll}}} \epsilon^{3/2}} \nu_{ss} \quad \text{with} \quad \nu_{ss} = \frac{4\sqrt{\pi}}{3} \frac{n_s \, e^4 \log \Lambda}{(4\pi\epsilon_0)^2 m_s^2 v^3_{T_{s,\text{coll}}}} \tag{2.28}$$

where $v_{T_{s,\text{coll}}}$ denotes the initial thermal velocity $v_{T_{s,\text{coll}}} = (T_{s,\text{coll}}/m_s)^{1/2}$, q the safety factor, $R = R_0 + r\cos\theta$ the major radius, $\epsilon = r/R_0$ the inverse aspect ratio and n_s the density of ion species s. The explicit expression (2.27) also involves the error function Φ and the Chandrasekhar function G defined as

$$G(v) = \frac{\Phi(v) - v\Phi'(v)}{2v^2} \quad \text{with} \quad \Phi(v) = \frac{2}{\sqrt{\pi}} \int_0^v e^{-x^2} \mathrm{d}x \quad \text{and} \quad \Phi'(v) = \frac{2}{\sqrt{\pi}} e^{-v^2} \tag{2.29}$$

Considering that the ν_{*s} scalar which is given (as input data) in the code corresponds to the main species, the collision frequencies $\nu_{s's'}$ for minority species s' are deduced from this value as

$$\nu_{s's'} = \frac{v_{T_{s',\text{coll}}} \epsilon^{3/2}}{qR_0} \nu_{*s'} \quad \text{with} \quad \nu_{*s'} = \left(\frac{n_{s'}}{n_s} \right) \left(\frac{Z_{s'}}{Z_s} \right)^4 \left(\frac{T_{s,\text{coll}}}{T_{s',\text{coll}}} \right)^2 \nu_{*s} \tag{2.30}$$

Let us express for each μ_s, the operator $\mathcal{V}_{\|s}$ as

$$\mathcal{V}_{\|s}(r,v) = -\frac{(v_{G\|} - V_{\|s,\text{coll}})}{v^2_{T_{s,\text{coll}}}} \mathcal{D}_{\|s}(r,v) \qquad \text{with} \qquad v_{T_{s,\text{coll}}} = \sqrt{T_{s,\text{coll}}/m_s} \tag{2.31}$$

then the collision operator (2.25) can be expressed with a more classical Fokker-Planck structure as:

$$\mathcal{C}_s(\bar{F}_s) = \frac{1}{B^*_{\|s}} \frac{\partial}{\partial v_{G\|}} \left[B^*_{\|s} \left(\mathcal{D}_{\|s} \frac{\partial \bar{F}_s}{\partial v_{G\|}} - \mathcal{V}_{\|s} \, \bar{F}_s \right) \right] \tag{2.32}$$

where the operators $\mathcal{D}_{\|s}$ and $\mathcal{V}_{\|s}$ respectively model a diffusion and a drag in the parallel velocity direction (see figure 2.6). The conservation properties of parallel momentum and energy are ensured by constraining $\mathcal{D}_{\|s}$ to depend on μ_s only and defining the local fluid velocity $V_{\|s,\text{coll}}$ and ion temperature $T_{s,\text{coll}}$ as follows (see Appendix F for more details)

$$\begin{aligned} V_{\|s,\text{coll}} &= P^{-1} \left[\left\langle \frac{m_s}{B^*_{\|s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s} v_{G\|}) \right\rangle \left\langle m_s \mathcal{D}_{\|s} v_{G\|} \right\rangle \right. \\ &\quad \left. - \left\langle \frac{1}{B^*_{\|s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s}) \right\rangle \left\langle m_s^2 \mathcal{D}_{\|s} v^2_{G\|} \right\rangle \right] \end{aligned} \tag{2.33}$$

$$T_{s,\text{coll}} = P^{-1} \left[\left\langle m_s \mathcal{D}_{\|s} \right\rangle \left\langle m_s^2 \mathcal{D}_{\|s} v^2_{G\|} \right\rangle - \left\langle m_s \mathcal{D}_{\|s} v_{G\|} \right\rangle \left\langle m_s^2 \mathcal{D}_{\|s} v_{G\|} \right\rangle \right] \tag{2.34}$$

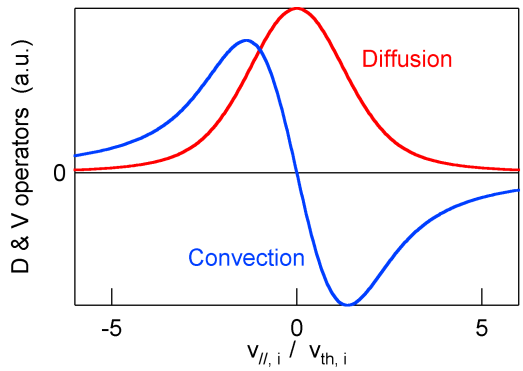


Figure 2.6: Parallel velocity dependence of the diffusive and convective operators $\mathcal{D}_{\|s}$ and $\mathcal{V}_{\|s}$in equations (2.27) and (2.31).

where

$$P = \langle m_s \mathcal{D}_{\|s} \rangle \left\langle \frac{m_s}{B^*_{\|s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s} v_{G\|}) \right\rangle - \langle m_s^2 \mathcal{D}_{\|s} v_{G\|} \rangle \left\langle \frac{1}{B^*_{\|s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s}) \right\rangle \tag{2.35}$$

with the brackets $\langle \cdot \rangle$ corresponding to the velocity space integral $\langle \cdot \rangle = \int \cdot \, \mathcal{J}_{\mathrm{v}} \, \mathrm{d}\mu_s \, \mathrm{d}v_{G\|} F_s$. The impact of the collisions on the evolution of the distribution function $\bar{F}_s$ is taken into account by stepping the evolution of the distribution function $\partial_t \bar{F}_s = \mathcal{C}_s(\bar{F}_s)$ with a Crank-Nicolson scheme. This collision operator forces the system to relax towards the Maxwellian distribution function, calculated from the instantaneous and local parallel flow $V_{\|s,\mathrm{coll}}$ and the isotropic temperature $T_{s,\mathrm{coll}} \sim T_{s,\mathrm{coll}\,\|}$.

2.7 Collisional transfers between two species

The inter-species operator currently implemented in the code is highly simplified. It only ensures the moment transfers and energy transfers between species. A more complete version, satisfying neoclassical results for impurity transport, is under development.

2.7.1 Collisional energy transfer

The energy exchange between two species is approached by the following reduced collision operator

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = -\frac{\eta^{\mathcal{E}}_{ss'}}{(2\pi T_{\mathrm{mean}}/m_s)^{3/2}} \frac{\Delta T_{ss'}}{T_{\mathrm{mean}}} \left(\frac{\mathcal{E}_{ss'}}{T_{\mathrm{mean}}} - \frac{3}{2} \right) \exp\left(-\frac{\mathcal{E}_{ss'}}{T_{\mathrm{mean}}} \right) \equiv C^{\mathcal{E}}_{ss'}(\mathcal{E}_{ss'}) \tag{2.36}$$

$$\frac{\mathrm{d}\bar{F}_{s'}}{\mathrm{d}t} = -\frac{\eta^{\mathcal{E}}_{ss'}}{(2\pi T_{\mathrm{mean}}/m_{s'})^{3/2}} \frac{\Delta T_{s's}}{T_{\mathrm{mean}}} \left(\frac{\mathcal{E}_{s's}}{T_{\mathrm{mean}}} - \frac{3}{2} \right) \exp\left(-\frac{\mathcal{E}_{s's}}{T_{\mathrm{mean}}} \right) \equiv C^{\mathcal{E}}_{s's}(\mathcal{E}_{s's}) \tag{2.37}$$

where $\mathrm{d}/\mathrm{d}t$ stands for the phase space Lagrangian derivative and the following definitions have been adopted

$$T_{\mathrm{mean}} = \frac{T_s + T_{s'}}{2} \quad ; \quad \Delta T_{ss'} = T_s - T_{s'} = -\Delta T_{s's} \tag{2.38}$$

$$V_{\mathrm{mean}} = \frac{V_{\|s} + V_{\|s'}}{2} \quad ; \quad \Delta V_{ss'} = V_{\|s} - V_{\|s'} \tag{2.39}$$

$$\mathcal{E}_{ss'} = \frac{m_s \left(v_{G\|} - V_{\mathrm{mean}} \right)^2}{2} + \mu_s B \quad ; \quad \mathcal{E}_{s's} = \frac{m_{s'} \left(v_{G\|} - V_{\mathrm{mean}} \right)^2}{2} + \mu_{s'} B \tag{2.40}$$

The temperatures and fluid velocities which enter these definitions are flux surface averaged, *i.e.* by considering $\langle \cdot \rangle_{\mathrm{FS}} = \int \cdot \mathcal{J}_{\mathrm{x}} \mathrm{d}\theta \mathrm{d}\varphi / \int \mathcal{J}_{\mathrm{x}} \mathrm{d}\theta \mathrm{d}\varphi$. Then velocities correspond to $V_{\|s}(r) = \langle \int v_{G\|} \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}} / N_s(r)$ with density $N_s(r) = \langle \int \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$, the volume element being defined as $\mathrm{d}^3\mathbf{v} = \mathcal{J}_{\mathrm{v}} \, \mathrm{d}v_{G\|} \, \mathrm{d}\mu_s$. Temperatures are defined as $T_s(r) = 2/3 \langle \int \mathcal{E}_s \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}} / N_s(r)$ where $\mathcal{E}_s = (m_s/2) \left(v_{G\|} - V_{\|s} \right)^2 + \mu_s B$.

The parameter $\eta^{\mathcal{E}}_{ss'}$ has been designed so that particles and parallel momentum are conserved

$$\eta^{\mathcal{E}}_{ss'} = \frac{8\,\varepsilon^{3/2}}{3\sqrt{\pi}}\,\frac{m_s}{m_{s'}}\left(\frac{Z_{s'}}{Z_s}\right)^2 \frac{n_{s'}\,\sqrt{T_{s,\mathrm{coll}}/m_s}}{qR_0}\left(1+\frac{v^2_{T'_s}}{v^2_{T_s}}\right)^{-3/2}\nu_{*s} \tag{2.41}$$

where the normalized ion-ion collision frequency ν_{*s} is given by eq.(2.28) (for detailed calculation see Appendix G.1). Numerically, equation (2.36), is solved as $\bar{F}_s(t+\Delta t) = \bar{F}_s(t) + \Delta t\, C^{\mathcal{E}}_{ss'}(\mathcal{E}_{ss'}, t)$.

2.7.2 Parallel momentum transfer

Parallel momentum exchange between two species can be modelled by the following approximate collision operator:

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = -\eta^{v_{G\|}}_{ss'}\,\Delta V_{ss'} v_{G\|}\exp\left(-\frac{E_s}{T_{\mathrm{mean}}}\right) \equiv C^{v_{G\|}}_{ss'}(E_s) \quad \text{with } E_s = \frac{1}{2}m_s v^2_{G\|} + \mu_s B \tag{2.42}$$

$$\frac{\mathrm{d}\bar{F}_{s'}}{\mathrm{d}t} = -\eta^{v_{G\|}}_{s's}\,\Delta V_{s's} v_{G\|}\exp\left(-\frac{E_{s'}}{T_{\mathrm{mean}}}\right) \equiv C^{v_{G\|}}_{s's}(E_{s'}) \tag{2.43}$$

where T_{mean} is defined by equation (2.38). $\Delta V_{ss'}$ is equivalent to eq.(2.39). The parameters $\eta^{v_{G\|}}_{s's}$ and $\eta^{v_{G\|}}_{ss'}$ are designed such that only parallel momentum exchanges are induced by this operator, thus leading to

$$\eta^{v_{G\|}}_{ss'} = n_{s'} m_s^2 \left(\frac{\varepsilon}{2\pi}\right)^{3/2}\frac{1}{qR_0}\frac{\sqrt{T_{s,\mathrm{coll}}}}{T^{5/2}_{\mathrm{mean}}}\left(\frac{Z_{s'}}{Z_s}\right)^2\frac{v_{T_s}}{v_{T_>}}\nu_{*s} \quad \text{and} \quad \eta^{v_{G\|}}_{s's} = \left(\frac{m_{s'}}{m_s}\right)^{3/2}\eta^{v_{G\|}}_{ss'} \tag{2.44}$$

where the velocity $v_{T_>}$ corresponds to the maximum value between v_{T_s} and $v_{T'_s}$ (for more details see Appendix G.2).

2.8 Flux-driven approach with source terms

2.8.1 Sources of heating, momentum and vorticity

As introduced previously GYSELA is a *full-f* code, namely the back reaction of turbulent transport is accounted for in the time evolution of the equilibrium. In such a framework, the turbulence regime is evanescent if no free energy is injected in the system. Turbulent transport results in the flattening of the temperature profile, which would ultimately reach marginal stability in the absence of any forcing. A heat source is mandatory in view of exploring the long time –on energy confinement times– behaviour of turbulence and transport. In GYSELA, the possibility to drive the system by a prescribed source was added in 2009 [SGA+10]. This source consists of the sum of the product of Hermite and Laguerre polynomials in $v_{G\|}$ and μ_s, respectively, in the spirit of the pioneering work by Darmet *et al* [DGS+08]. It is versatile enough to allow for separate injection of heat, parallel momentum and vorticity. Such a versatility imposes serious constraints on the expression of the source in phase space. The separation between these three kinds of sources is achieved using projections onto the bases of orthogonal Hermite polynomials in $v_{G\|}$ and orthogonal Laguerre polynomials in μ_s. The retained expression for the source is the following (see Appendix H for detailed description)

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = S_{\mathrm{heat}}(r,\theta,v_{G\|},\mu_s) + S_{\mathrm{moment}}(r,\theta,v_{G\|},\mu_s) + S_{\mathrm{vorticity}}(r,\theta,v_{G\|},\mu_s) \tag{2.45}$$

where the pure heating source is defined as:

$$S_{\mathrm{heat}} = \left[\bar{v}^2_{G_\| s} - \frac{1}{2} - \frac{J_{\|B}}{2-J^2_{\|B}}\,(2-\bar{\mu}_s)\left(2\bar{v}_{G_\| s} - J_{\|B}\right)\right]\frac{2S^{\mathcal{E}}_0}{(2\pi T_{s,\mathrm{srce}}/m_s)^{3/2}\,T_{s,\mathrm{srce}}}\,S^{\mathcal{E}}_r\,\mathrm{e}^{-\bar{v}^2_{G_\| s}-\bar{\mu}_s} \tag{2.46}$$

with $\bar{\mu}_s = \mu_s B/T_{s,\mathrm{srce}}$, $\bar{v}_{G_\parallel s} = v_{G\parallel}/\sqrt{2T_{s,\mathrm{srce}}/m_s}$, $J_{\parallel B} \equiv \sqrt{2m_s T_{s,\mathrm{srce}}}/(q_s B^2)\,\mu_0 J_\parallel$ and $\alpha = (m_s/q_s^2)\times T_{s,\mathrm{srce}}/(2B^2)$. S_{moment} is a pure momentum source expressed as

$$S_{\mathrm{moment}} = \left[2\bar{v}_{G_\parallel s}(2-\bar{\mu}_s) - J_{\parallel B}\left(1+2\bar{v}_{G_\parallel s}-\bar{\mu}_s\right)\right] \frac{S_0^{v_{G\parallel}}}{4\pi^{3/2}\left(T_{s,\mathrm{srce}}/m_s\right)^2}\, S_r^{v_{G\parallel}}\, \mathrm{e}^{-\bar{v}^2_{G_\parallel s}-\bar{\mu}_s} \tag{2.47}$$

while $S_{\mathrm{vorticity}}$ is a pure source of vorticity

$$S_{\mathrm{vorticity}} = -\left[2\bar{v}^2_{G_\parallel s} - \bar{\mu}_s\right] \frac{S_0^{\Omega}}{\alpha\left(2\pi T_{s,\mathrm{srce}}/m_s\right)^{3/2}}\, S_r^{\Omega}\, \mathrm{e}^{-\bar{v}^2_{G_\parallel s}-\bar{\mu}_s} \tag{2.48}$$

The prescribed radial envelopes $S_r^{\mathcal{E}}$, $S_r^{v_{G\parallel}}$ and S_r^{Ω} are chosen as the sum of two hyperbolic tangents but could be any radial function under the constraint that the integral over the minor radius is normalized to 1 while $S_0^{\mathcal{E}}$, $S_0^{v_{G\parallel}}$ and S_0^{Ω} correspond to the source amplitudes and $T_{s,\mathrm{srce}}$ to the temperature of the source. They are usually localized close to the inner boundary of the simulation domain [SGA+10]. The heat source is an isotropic source that takes particles at a vanishing velocity and accelerates them up to $v \approx 1.5 v_{T_h}$. See schematic view with figure 2.8. As a remark, there is another choice possible for the heat source S_{heat} in the code, which corresponds to

$$S_{\mathrm{heat}} = \frac{S_0^{\mathcal{E}}}{3\sqrt{2}\left(\frac{\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}}} \left(\frac{E}{T_{s,\mathrm{srce}}} - \frac{3}{2}\right) \exp\left(-\frac{E}{T_{s,\mathrm{srce}}}\right) S_r^{\mathcal{E}} \tag{2.49}$$

There are two main advantages in dealing with a prescribed heat source: (i) the forcing of turbulence can mimic that in experiments, in contrast to simulations where the mean gradient is prescribed and (ii) the sum of the spatially and time (on the τ_E time scale) averaged turbulent and neoclassical heat fluxes must balance the prescribed driving flux. In this case, the response is the temperature gradient, which ultimately governs the internal energy and therefore the performance of the discharge. An example of initial and final temperature profiles is shown in figure 2.7. Flux driven simulations then allow investigating the impact of heating power on the energy confinement time [SGA+11]. The source of vorticity described before has been efficiently used to polarize the plasma [SSZ+13a] inducing the development of sheared electric fields in the turbulent core. The creation of ion transport barriers by these externally induced sheared $E \times B$ flows has been studied in details in [SSZ+13b, SSZ+13a].

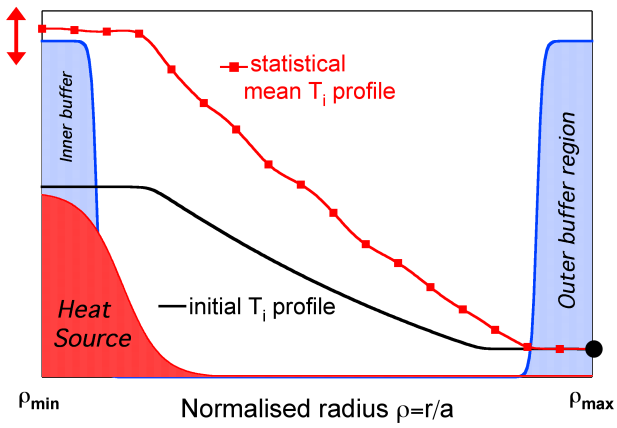


Figure 2.7: Schematic view of a heat source and buffer regions in the case of a flux driven simulation. Comparison between initial temperature profile (black line) and final temperature (red line).

2.8.2 Energetic particle source

A source of energetic particles (EPs) has been also implemented in the code to study the interaction between EPs and turbulence. EPs are characterized by energies larger than the thermal energy. The excitation by EPs of the geodesic acoustic modes (GAMs) –corresponding to the oscillatory

component of large scale $E \times B$ zonal flows– creates modes which are called energetic geodesic acoustic modes (EGAMs) [Fu08, NFA+08]. For more details on the impact of EGAMs on turbulence see [ZGS+12, ZSG+13, DZS+13]. In practice this source is coupled to the heating source, such that $\mathrm{d}_t \bar{F}_s = S_{\mathrm{heat}} + S_{\mathrm{EP}}$ where S_{EP} is also built, as for the previous sources, by using projection onto the Laguerre and Hermite polynomial bases, with the constraint to inject only parallel energy. For symmetry reason, the energetic particle source, is built as

$$S_{\mathrm{EP}}(r, \theta, v_{G\|}, \mu_s, t) = S_0^{\mathrm{EP}}(t) S_r^{\mathrm{EP}}(r)\,(S_+ + S_-) \quad \text{with} \quad S_-(\theta, v_{G\|}, \mu_s) = S_+(\theta, -v_{G\|}, \mu_s)$$

where S_0^{EP} is the source amplitude and S_r^{EP} the radial profile is normalized such that $\int r\,\mathrm{d}r S_r^{\mathrm{EP}} = 1$. After the same kind of calculation as in the previous sources, the final expression reads

$$S_\pm = \left[(\bar{v}_{G_\| s} \pm \bar{v}_0)^2 - \frac{1}{2} - \mathcal{Q}_{\mathrm{EP}}(2 - \bar{\mu}_s)(2(\bar{v}_{G_\| s} \pm \bar{v}_0) - J_{\| B})\right] \mathrm{e}^{-(\bar{v}_{G_\| s} \pm \bar{v}_0)^2} \mathrm{e}^{-\bar{\mu}_s} \tag{2.50}$$

where $\bar{\mu}_s = \mu_s B(r,\theta)/T_{S\perp}$ and $\mathcal{Q}_{\mathrm{EP}} = J_{\| B}/\left(2 - J_{\| B}^2(1 + 2\bar{v}_0^2)\right)$ with $\bar{v}_0 = v_0/\sqrt{2T_{S\|}}$ an arbitrary normalized velocity. The expressions of the parallel current $J_{\| B}$ and the velocity $\bar{v}_{G_\| s}$ are the same as in equation (2.46) where $T_{s,\mathrm{srce}}$ is replaced by $T_{S\|}$. $T_{S\|}$ and $T_{S\perp}$ correspond to the normalized parallel and perpendicular temperatures of the energetic particle source. Each of the terms S_+ and S_- does not inject neither mass –which is essential because electrons are considered adiabatic in the code– nor vorticity. See figure 2.8 for a representative view of the source in parallel velocity direction. The

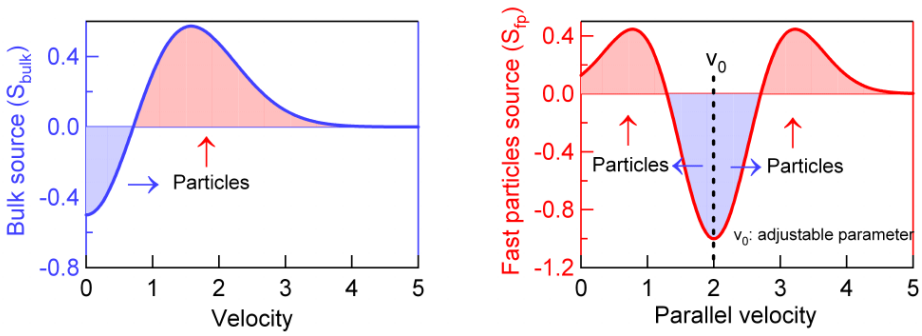


Figure 2.8: Schematic view of the energetic particle source as a function of the parallel velocity

source mimics the effects of two tangential neutral beam injectors, oriented in the co-and counter-current directions. It is localized around the mid position $r = 0.5\,(r_{\min} + r_{\max})$ and brings the distribution function out of the equilibrium by creating a positive slope in energy. As explained in [ZSG+13] v_0 and $T_{S\|}$ are both critical parameters in view of exciting EGAMs. GYSELA results for EGAMs excitation have been successfully compared to analytical theory [ZSG+13] and benchmarked more recently with ORB5 code.

2.8.3 Source of impurities

Finally, it is also possible to add a source of impurities s' of the form

$$S_{n_{s'}} = \frac{S_0^{n_{s'}} S_r}{\left(\frac{2\pi T_{s',\mathrm{srce}}}{m_{s'}}\right)^{3/2}} \left(\frac{5}{2} - \frac{\mu_{s'} B}{T_{s',\mathrm{srce}}} - \frac{m_{s'} v_{G\|}^2}{2T_{s',\mathrm{srce}}}\right) \exp\left(-\frac{m_{s'} v_{G\|}^2}{2T_{s',\mathrm{srce}}} - \frac{\mu_{s'} B}{T_{s',\mathrm{srce}}}\right) \tag{2.51}$$

Let us notice that this source of matter is not a pure source, due to the fact that it also injects some amount of vorticity. This injection is balanced by a sink for the major species s, such that

$$Z_s \int S_{n_s} \mathcal{J}_{\mathrm{v}_s}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_s + Z_{s'} \int \mathcal{J}_{\mathrm{v}_{s'}} S_{n_{s'}}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_{s'} = 0$$

2.8.4 Diffusion terms in buffer regions

Finally, to avoid strong gradients at the boundaries, radial diffusion and artificial damping can be added in buffer regions. These buffer regions are defined at each side of the radial domain $r \in [r_{\min}, r_{\max}]$ as a sum of hyperbolic tangents:

$$H_{\text{buff}}(r) = 1 + \frac{1}{2}\left[\tanh\left(\frac{r - r_{\max} + B_L\, L_r}{B_S\, L_r}\right) - \tanh\left(\frac{r - r_{\min} - B_L\, L_r}{B_S\, L_r}\right)\right] \tag{2.52}$$

where L_r is the length of the radial domain. B_L and B_S are respectively the location and the stiffness of the buffer regions. The function H_{buff} plays the role of a mask which is equal to 1 in the buffer regions and 0 elsewhere. The diffusion term which is applied in the buffer regions is of the form

$$\mathcal{D}_r(\bar{F}_s) = \frac{1}{B^*_{\|s}}\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\chi(r)B^*_{\|s}\frac{\partial}{\partial r}\bar{F}_s\right)\right] \tag{2.53}$$

with $\chi(r) = \chi_0 H_{\text{buff}}(r)$ (χ_0 being the diffusion coefficient). The equation $\partial_t \bar{F}_s = \mathcal{D}_r(\bar{F}_s)$ is solved by using a Crank-Nicolson scheme (see appendix I). An artificial damping term ν_0 is introduced in the buffer regions by defining a Krook operator

$$\mathcal{K}_r(\bar{F}_s) = -\nu(r)(\bar{F}_s - \bar{F}_{s,\text{eq}}) \quad \text{with } \nu(r) = \nu_0 H_{\text{buff}}(r) \tag{2.54}$$

and solving $\partial_t \bar{F}_s = \mathcal{K}_r(\bar{F}_s)$. Let Δt be the time step, then an analytic solution of the previous equation is given by [3]

$$\bar{F}_s(t + \Delta t) = \bar{F}_{s,\text{eq}} + (\bar{F}_s(t) - \bar{F}_{s,\text{eq}}) \exp\left(-\nu(r)\Delta t\right) \tag{2.55}$$

This mechanism restores the distribution function to its initial equilibrium state $\bar{F}_{s,\text{eq}}$, in the buffer regions, by slowly damping all the turbulent modes of the system. It also plays the role of a heat sink by effectively coupling the plasma with the outer thermal baths.

Let us summarize the main characteristic of the GYSELA code with its strengths and weaknesses. First of all, it is *global*. It simulates the whole plasma in toroidal geometry (r, θ, φ) but for a simplified circular magnetic configuration. The 2D velocity space is $(v_{G\|}, \mu)$ with μ the magnetic momentum. Periodic boundary conditions are imposed both in poloidal (θ) and toroidal (φ) directions. Homogeneous Dirichlet boundary conditions are considered in parallel velocity ($v_{G\|}$) direction. In radial direction (r), boundary conditions are non-homogeneous Dirichlet in $r = r_{\max}$ while they vary according to the forcing choice in $r = r_{\min}$ (non-homogeneous Dirichlet if thermal baths and Neumann conditions for gradient-driven or flux-driven options).
Secondly, GYSELA is a *full-f* code. It considers for each ion species the time evolution of the full 5D distribution function $F_s(r, \theta, \varphi, v_{G\|}, \mu)$ (where μ as adiabatic invariant plays the role of a parameter) but electrons are assumed adiabatic. The minimal set of GYSELA equations is then

- a 4D+1D gyrokinetic Vlasov equations for each ion species (2.7)
- with four advection equations (2.8)-(2.9)
- self-consistently coupled to a 3D quasi-neutrality equation (2.13).

Thirdly, GYSELA can *take into account neoclassical transport effects* with:

- a linearized intra-species collision operator (2.32) and
- a simplified inter-species collision operator including energy transfers (2.36)-(2.37) and parallel momentum transfers (2.42)-(2.43).

Fourthly, additional source terms are available for *flux-driven* simulations:

[3] Let G be a function defined as $G(t) = \bar{F}_s(t) - \bar{F}_{s,\text{eq}}$. As $\bar{F}_{s,\text{eq}}$ is constant in time, $\partial_t \bar{F}_s = \mathcal{K}_r(\bar{F}_s)$ is equivalent to $\partial_t G(t) = -\nu(r)G(t)$ which has for solution $G(t) = \exp(-\nu(r)t)$. Therefore, $G(t + \Delta t) = \bar{F}_s(t + \Delta t) - \bar{F}_{s,\text{eq}} = \exp(-\nu(r) \times (t + \Delta t)) = G(t)\exp(-\nu(r)\Delta t)$.

- a pure source of heating (2.46),
- a pure source of momentum (2.47),
- a pure source of vorticity (2.48) and
- a source of energetic particles (2.50).

Finally, GYSELA is the unique 5D global full-f flux-driven code in the world based on a *Semi-Lagrangian* numerical scheme. This originality is detailed in the next chapter.

Chapter 3

A highly parallel semi-Lagrangian code

Contents

3.1 Specificity of the GYSELA code: a semi-Lagrangian scheme 57
3.1.1 Backward Semi-Lagrangian (BSL) concept 58
3.1.2 Time-splitting 59
3.1.3 GYSELA global algorithm 60
3.2 An efficient hybrid OpenMP/MPI parallelization 61
3.2.1 Parallel Vlasov solver: How to treat non-local interpolation aspect ? 62
3.2.2 A scalable quasi-neutral solver 63
3.2.3 Performing weak and strong scaling 63
3.2.4 Memory scalability 64

As said before, one originality of the GYSELA code is to be based on a Semi-Lagrangian scheme which is a mix between PIC and Eulerian approaches. Each of these three numerical methods has its advantages and its drawbacks. This objective of this chapter is not at all to prove that the Semi-Lagrangian approach is better or not than the two others. We are convinced that having the three type of gyrokinetic codes (PIC, Eulerian and Semi-Lagrangian) is a strength for Fusion community. We first detailed here how the Backward Semi-Lagrangian (BSL) method is applied to the 5D gyrokinetic Boltzmann equation. For this, we start by reminding the BSL concept. Then, we describe the global numerical scheme, implemented in the code, which is based on a time-splitting of Strang [Str68]. All the numerical methods associated to each operators and each equations are completely described in appendices to act as user's guide of the code. As already mentioned, GYSELA requires state-of-the-art in HPC to deal with non-linear simulations on huge 5D meshes. The second part of Chapter 3 is therefore dedicated to these parallel computing aspects. Its eulerian character is an advantage because there is no problem of load-balancing. However, a difficulty when parallelizing the semi-Lagrangian Vlasov solver is due to the cubic spline interpolation. Indeed, cubic splines are a good compromise between simplicity and accuracy but are non-local. We detail how this difficulty has been overcome and what has been done to improve the scalability of the Poisson solver. The associated hybrid OpenMP/MPI parallelization results in a remarkable scalability on several hundreds of thousands of cores.

3.1 Specificity of the GYSELA code: a semi-Lagrangian scheme

Semi-Lagrangian (SL) schemes have been first used for the advection of vorticity in simplified models of large scale flows. It has gained maturity when the discretization approach was introduced in the relevant context of atmospheric flows. A comprehensive review of semi-Lagrangian methods in this meteorological context until 1990 is due to Staniforth [SC91]. It is also applied to geophysical fluid dynamics (cf. [Dur98]). In magnetized plasma turbulence area, the SL method has been first

applied to calculate a turbulence driven by passing ions in 2D (1D in space, 1D in velocity) [SRBG99] and trapped ions in 3D (2D in space, 1D in velocity) [DGBG00, SGF+05]. This method was then extended to the 4D model (3D in space and $v_\parallel$ (with $\mu = 0$)) of Ion Temperature Gradient (ITG) driven turbulence in cylindrical geometry with the development of the GYSELA code (for GYrokinetic SEmi-LAgrangian code) [GBB+06] and the CYGNE code [BGS+04]. The 4D drift-kinetic slab-ITG version of the GYSELA code has shown good properties of energy conservation in non-linear regime [GBB+06] as well as accurate description of fine spatial scales [SGDP+06b]. In the CYGNE code the standard Taylor expansion is replaced by a Bürlisch-Stoer scheme (for the 2D advection) to increase the spatial accuracy and the logarithmic interpolation technique is used to ensure the positivity of the distribution function. Brunetti et al. [BGS+04] have shown that (i) the positivity can be preserved but at the cost of larger diffusion and (ii) that non-equidistant meshes in radial and parallel velocity directions are a key tool for obtaining accurate results. Due to the good conservation property obtained in 4D, the gyrokinetic 5D version of the GYSELA code [GSA+07] has been based on the same numerical scheme. The global algorithm for the new 5D gyrokinetic multi-ion species version of the code will be described in the following both in terms of numerics (section 3.1.3) and parallelisation (section 3.2). Advantages and drawbacks of such a 5D semi-Lagrangian code will be discussed. The purpose of the semi-Lagrangian method is to take advantage of both the Lagrangian and Eulerian approaches, with an accurate description of the phase space, in particular regions where the density is low, as well as an enhanced numerical stability. It is based on the fact that the most precise way to solve convection (or advection) hyperbolic PDE is to use their characteristics along which the distribution function remains constant. This method was primarily developed by Cheng and Knorr [CK76]. It has been cast in more general framework of SL by Sonnendrücker in 1998 [SRBG99] and Nakamura in 1999 [NY99]. In this approach, the phase-space mesh grid is kept fixed in time (Eulerian method) and the Vlasov equation is integrated along the trajectories (Lagrangian method) using the invariance of the distribution function along the trajectories. The GYSELA code is based on this standard semi-Lagrangian approach [SRBG99]. This approach has been recently renamed *Backward semi-Lagrangian approach (BSL)* by its author to make the distinction with the emergence of new schemes: (i) *Forward semi-Lagrangian approach (FSL)* firstly proposed in 2009 [CRS09] and (ii) *Conservative semi-Lagrangian approach (CSL)* proposed in [CMS10b, BCMS12] through Parabolic Spline Method [ZWS07]. The main difference between BSL and FSL approaches is that the advection equations are solved backward in time in the first case and forward in the second one. They are both based on solving the advective form of Vlasov equation while the CSL methods deal with the conservative form of the Vlasov equation. FSL and CSL schemes have both been tested in GYSELA [LGA+12, BCG+11] but the actual version of the code is the standard BSL approach as described in the following.

3.1.1 Backward Semi-Lagrangian (BSL) concept

Let us consider the 5D gyrokinetic Vlasov equation

$$B^*_{\parallel s}\frac{\partial \bar{F}_s}{\partial t} + \boldsymbol{\nabla}\cdot\left(B^*_{\parallel s}\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t}\bar{F}_s\right) + \frac{\partial}{\partial v_{G\parallel}}\left(B^*_{\parallel s}\frac{\mathrm{d}v_{G\parallel}}{\mathrm{d}t}\bar{F}_s\right) = 0 \tag{3.1}$$

which corresponds to equation (2.7) without collisions and source terms. By using the incompressibility property of the gyrocenter flow [1] in the 4D phase space, *i.e* $\boldsymbol{\nabla}\left(B^*_{\parallel s}\mathrm{d}_t\mathbf{x}_G\right) + \partial_{v_{G\parallel}}\left(B^*_{\parallel s}\mathrm{d}_t v_{G\parallel}\right) = 0$, the previous conservative Vlasov equation (3.1) is equivalent to its advective form

$$\frac{\partial \bar{F}_s}{\partial t} + \frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t}\boldsymbol{\nabla}\cdot\bar{F}_s + \frac{\mathrm{d}v_{G\parallel}}{\mathrm{d}t}\frac{\partial \bar{F}_s}{\partial v_{G\parallel}} = 0 \tag{3.2}$$

Let $\boldsymbol{\Gamma} = (\mathbf{x}_G, v_{G\parallel})$ be a position vector in the phase space solution of the characteristic equations (2.8)-(2.9). Then equation (3.2) leads to $\mathrm{d}_t\bar{F}_s(\boldsymbol{\Gamma}(t), t) = 0$. The semi-Lagrangian method uses this invariance

[1] According to equations (1.45)-(1.41), $B^*_{\parallel s}\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t} = v_{G\parallel}\mathbf{B} + \frac{1}{q_s}\left(m_s v_{G\parallel}^2\nabla\times\mathbf{b} + \mu_s\mathbf{b}\times\nabla B\right) + \mathbf{b}\times\nabla\bar{U}$, then using the fact that $\nabla B = 0$ and $\nabla\cdot(\nabla\times\mathbf{b}) = 0$, $\nabla\cdot\left(B^*_{\parallel s}\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t}\right) = \nabla\cdot(\mathbf{b}\times\nabla\bar{U}) + \frac{1}{q_s}\nabla\cdot(\mathbf{b}\times\mu_s\nabla B)$. Finally due to $\nabla\times\nabla\bar{U} = \nabla\times\nabla B = 0$, $\nabla\cdot\left(B^*_{\parallel s}\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t}\right) = \nabla\bar{U}\cdot\nabla\times\mathbf{b} + \frac{\mu_s}{q_s}\nabla B\cdot\nabla\times\mathbf{b}$. Besides, $\frac{\partial}{\partial v_{G\parallel}}\left(B^*_{\parallel s}\frac{\mathrm{d}v_{G\parallel}}{\mathrm{d}t}\right) = -\frac{1}{q_s}\nabla\times\mathbf{b}\cdot\nabla\left(q_s\bar{U} + \mu_s B\right) = \nabla\times\mathbf{b}\cdot\left(\frac{\mu_s}{q_s}\nabla B + \nabla\bar{U}\right)$ which proves the free divergence property.

of the distribution function $\bar{F}_s$ along its characteristics. Let us consider the computational 5D domain $[x^1_{G\,\min}, x^1_{G\,\max}] \times [x^2_{G\,\min}, x^2_{G\,\max}] \times [x^3_{G\,\min}, x^3_{G\,\max}] \times [v_{G\|\,\min}, v_{G\|\,\max}] \times [\mu_{s\,\min}, \mu_{s\,\max}]$ and the associated grid (fixed in time) defined by the finite set of mesh points $\mathbf{\Gamma}_{ijklm} = \left(x^1_{G_i}, x^2_{G_j}, x^3_{G_k}, v_{G\|_l}, \mu_{s_m}\right)$ with $x^p_{G_q} = q\Delta x^p_G$ for all $q = 0, \cdots, N_p$ with $p = 1, 2, 3$; $v_{G\|_l} = l\Delta v_{G\|}$ for all $l = 0, N_{v_\|}$ and μ_{s_m} for all $m = 0, N_\mu$. N_1, N_2, N_3 are the number of cells in each spatial directions and $N_{v_\|}$ the number of cells in $v_{G\|}$ direction. The $(N_\mu + 1)$ values for μ_s are not necessary equidistributed. Actually, the standard choice in GYSELA corresponds to an equidistributed grid in $\sqrt{\mu}$. This choice leads to a better accuracy when computing integrals in μ. Let us also assume that $\bar{F}_s$ is known at each point $\mathbf{\Gamma}_{ijklm}$ of this grid at time t_n. Therefore the distribution function can be computed at the next time $t_n + \Delta t$, with Δt the time step, on the same grid by using the invariance property $\bar{F}_s\left(\mathbf{\Gamma}_{ikjlm}(t_n + \Delta t), t_n + \Delta t\right) = \bar{F}_s\left(\mathbf{\Gamma}(t_n, \mathbf{\Gamma}_{ijklm}, t_n + \Delta t), t_n\right)$ where $\mathbf{\Gamma}(t_n, \mathbf{\Gamma}_{ijklm}, t_n + \Delta t)$ corresponds to the solution of the characteristic at time step t_n which is equal to $\mathbf{\Gamma}_{ijklm}$ at time $t_n + \Delta t$. The method consists in (i) finding the foot of the characteristic at the time t_n: $\mathbf{\Gamma}(t_n, \mathbf{\Gamma}_{ijklm}, t_n + \Delta t)$ by solving backward in time the advection equations (2.8)-(2.9) and (ii) computing $\bar{F}_s\left(\mathbf{\Gamma}(t_n, \mathbf{\Gamma}_{ijklm}, t_n + \Delta t), t_n\right)$ by interpolation, using the fact that at this time t_n the distribution function is known over the whole fixed grid. Cubic spline interpolations are used in the code, because it offers a good compromise between accuracy (small diffusivity) and simplicity (numerical cost) [FSB01, BM08] (see appendix J for more details).

3.1.2 Time-splitting

In low-dimensional systems, the semi-Lagrangian method is very efficient. When applied to higher dimensional problems, one faces the problem of multidimensional interpolation, which is extremely expensive for high dimensional problems. However, this problem has been partially cured by using the time-splitting idea of Cheng and Knorr [CK76]. Using the incompressibility property, Strang's operator decomposition into space and velocity can be applied, replacing equation (2.7) by a set of two conservative equations. Besides, to avoid dealing with a 3D space operator, the latter is also divided into two parts. Let us denote $\mathcal{X}_G = (x^1_G, x^2_G) = (r, \theta)$ and remind that in our case $x^3_G = \varphi$ then the Boltzmann equation (2.7) is solved by applying a splitting of Strang [Str68] as

$$B^*_{\|s}\frac{\partial \bar{F}_s}{\partial t} + \boldsymbol{\nabla}_\perp \cdot \left(B^*_{\|s}\frac{\mathrm{d}\mathcal{X}_G}{\mathrm{d}t}\bar{F}_s\right) = 0 \qquad \text{at } (\varphi, v_{G\|}) \text{ fixed} \tag{3.3}$$

$$B^*_{\|s}\frac{\partial \bar{F}_s}{\partial t} + \frac{\partial}{\partial \varphi}\left(B^*_{\|s}\frac{\mathrm{d}\varphi}{\mathrm{d}t}\bar{F}_s\right) = 0 \qquad \text{at } (\mathcal{X}_G, v_{G\|}) \text{ fixed} \tag{3.4}$$

$$B^*_{\|s}\frac{\partial \bar{F}_s}{\partial t} + \frac{\partial}{\partial v_{G\|}}\left(B^*_{\|s}\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t}\bar{F}_s\right) = 0 \qquad \text{at } \mathbf{x}_G \text{ fixed} \tag{3.5}$$

This splitting into three equations was introduced in the 4D version of the code. As explained in [GBB+06], in the 4D drift-kinetic slab case the conservative and advective forms of the equations are equivalent in the (r, θ) direction and z direction separately due to the independent vanishing divergence property, see equations (9)-(10) in [GBB+06] due to $\boldsymbol{\nabla} \cdot v_{GC} = 0$ and $\partial_z v_\| = 0$. In the 5D gyrokinetic case $\boldsymbol{\nabla}_\perp \cdot (B^*_{\|s}\mathrm{d}_t\mathcal{X}_G) \neq 0$ and $\partial_\varphi(B^*_{\|s}\mathrm{d}_t\varphi) \neq 0$ so that these terms should be taken into account as source terms of the advective form of equations (3.3) and (3.4). However, they are presently set to zero. This simplification may alter the accuracy of the conservation properties of the code (see section 4.4). Also, it likely has an impact on the maximal value acceptable for the discretization time step. A solution to overcome this problem could be to use a conservative scheme instead of the BSL one but the first tests we have performed [BCG+11] are not conclusive. This constraint on the numerical value of Δt is acceptable for ion turbulence simulations but could become problematic when addressing kinetic electrons. The development of more efficient semi-Lagrangian schemes is still an active axis of research. An idea currently under evaluation is to separate and to treat differently the linear and non-linear parts. Encouraging results have been presented in [LGA+14]. The splitting operation stays a drawback of the semi-Lagrangian method. An alternative method without splitting (based on a 4D advection and 4D cubic spline interpolation) is currently developed. The first drawback is an increase of the numerical diffusion due to the 4D interpolation which will require to be quantified.

In the current version, the advections in φ and $v_{G\|}$ directions are straightforward, but that in the $\mathcal{X}_\mathcal{G}$ direction requires more attention. If we consider the 2D advection in (r,θ) direction between times t and $t+\Delta t$, the value of the electric field $\mathbf{E}$ at time $t+\Delta t/2$ is required in second order time scheme. This value is calculated by using a predictor-corrector method. Besides, computing these 2D trajectories is equivalent to solving $\mathrm{d}\mathcal{X}_G/\mathrm{d}t = \mathbf{V}(\mathcal{X}_G,\varphi,t)$, $\mathbf{V}$ being the advection field. This system is solved by using the parabolic assumption developed in [SRBG99]. Let $\mathcal{X}_{G_{ij}}$ be the position of $\mathcal{X}_G(t_n+\Delta t)$ at time $t_n+\Delta t$, then there exists a displacement $\mathbf{d}_{ij}=(\alpha_{ij},\beta_{ij})$ tangent to the parabola such that $\mathcal{X}_G(t_n)=\mathcal{X}_{G_{ij}}-\mathbf{d}_{ij}$ and $\mathcal{X}_G(t_n-\Delta t)=\mathcal{X}_{G_{ij}}-2\mathbf{d}_{ij}$. The displacement $\mathbf{d}_{ij}$ can be calculated by solving the implicit equation $\mathbf{d}_{ij}=\Delta t\,\mathbf{V}(\mathcal{X}_{G_{ij}}-\mathbf{d}_{ij},t_n)$ (see p. 129 in [GS13]). This is done with a Taylor expansion which is equivalent at second order to a Newton algorithm.

3.1.3 GYSELA global algorithm

Concerning now the complete Boltzmann equation (2.7) the right hand side is also split to treat separately the collision operator, the Krook operator, the diffusion and source terms. Let $\tilde{\mathcal{X}}_\mathcal{G}$ denote the shift operator in the poloidal cross-section (r,θ) associated to equation (3.3) over a time step Δt. Similarly, $\tilde{\varphi}$ and $\tilde{v}_{G\|}$ denote the shift operators respectively in the φ (equation (3.4)) and $v_{G\|}$ directions (equation (3.5)). As described in the previous paragraph, each of these three shift operators are based on a backward semi-Lagrangian scheme which means two steps for each mesh point: (i) first the computation of the characteristic feet and (ii) second an interpolation by cubic splines. Let us denote $\tilde{\mathcal{C}}$ the collision operator corresponding to solving $\partial_t\bar{F}_s=\mathcal{C}(\bar{F}_s)$ and $\tilde{\mathcal{D}}$ the operator associated to the radial diffusion $\partial_t\bar{F}_s=\mathcal{D}_r(\bar{F}_s)$. They are both solved by applying a Crank-Nicolson scheme (see Appendices F.2 and I). The Krook operator $\tilde{\mathcal{K}}$ corresponding to the Krook diffusion equation $\partial_t\bar{F}_s=\mathcal{K}(\bar{F}_s)$ is trivial while solving $\partial_t\bar{F}_s=\mathcal{S}$ associated to the source operator $\tilde{\mathcal{S}}$ is described in Appendix H.2.5. Then, using these notations the following sequence is used to solve the 5D Boltzmann equation (2.7)

$$\widetilde{\mathrm{Boltz}} \equiv \left[\left(\frac{\tilde{\mathcal{K}}_r}{2},\frac{\tilde{\mathcal{D}}_r}{2},\frac{\tilde{\mathcal{C}}}{2},\frac{\tilde{\mathcal{S}}}{2}\right),\left(\frac{\tilde{v}_{G\|}}{2},\frac{\tilde{\varphi}}{2},\tilde{\mathcal{X}}_\mathcal{G},\frac{\tilde{\varphi}}{2},\frac{\tilde{v}_{G\|}}{2}\right),\left(\frac{\tilde{\mathcal{S}}}{2},\frac{\tilde{\mathcal{C}}}{2},\frac{\tilde{\mathcal{D}}_r}{2},\frac{\tilde{\mathcal{K}}_r}{2}\right)\right] \tag{3.6}$$

where the factor 1/2 means that the operator is applied on half a time step. The choice of the sequence (3.6) is not unique but some constraints are imposed in the code: (i) the first one is to impose a symmetry to keep second order accuracy in the splitting (Strang splitting [Str68]), (ii) the second one is to fix the 2D operator $\tilde{\mathcal{X}}_\mathcal{G}$ which is the most costly at the middle of the algorithm; (iii) finally the operators coupled to the right hand side, respectively to the Vlasov equation, are contiguous. Let us also define the operator $\tilde{Q}$ (corresponding to the Poisson solving) which denotes symbolically the four steps: (i) computation of right hand side of the quasi-neutrality equation (2.13) using expression (2.16), (ii) solving the QN equation (2.13) to deduce the electrostatic potential U, (iii) computation of the gyro-averaged electric potential $\bar{U}=\mathcal{J}_\mu.U$ with a Padé approximation (2.19) or with an integration on the gyro-circles as described in section 2.4 and (iv) computation of the electric field as $\mathbf{E}=-\boldsymbol{\nabla}\bar{U}$.
Finally, the global numerical algorithm of the GYSELA code can be summarized (see schematic view in Figure 2.3) as follows,

1. **Initialization**
   Considering a prescribed magnetic field $B(r,\theta)$ (see section 2.1.2) and equilibrium profiles of density $n_0(r)$, ion temperatures $T_s(r)$ and safety factor $q(r)$ (see section 2.5), then
   (a) Computation of the equilibrium distribution function $\bar{F}_{s,\mathrm{eq}}$ as a local or canonical Maxwellian by using (2.20) and (2.21).
   (b) Initialization of $\bar{F}_s(t=0)$ as $\bar{F}_s(t=0)=\bar{F}_{s,\mathrm{eq}}(1+\mathrm{perturbation})$ as described in section 2.5.
2. **For each time iteration,**
   Considering the distribution function $\bar{F}_s^n=\bar{F}_s(t=t_n)$ at time t_n known on the 5D mesh grid, then the distribution function $\bar{F}_s^{n+1}$ at the next time $t_{n+1}=t_n+\Delta t$ on the same mesh grid is computed by using a predictor-algorithm as

(a) Computation of the electric field $\mathbf{E}(t_n)$ by using the $\tilde{Q}$ sequence.

(b) **Prediction on Δt/2**:

- Computation of $\bar{F}_s(t = t_n + \Delta t/2)$ by solving $\widetilde{\text{Boltz}}/2$ sequence with $\mathbf{E}(t_n)$.
- Computation of $\mathbf{E}(t_n + \Delta t/2)$ by solving $\tilde{Q}$.

(c) **Correction on Δt**:

- Starting from $\bar{F}_s(t = t_n)$ –given that this 5D distribution function has been stored before prediction–, computation of $\bar{F}_s(t = t_n + \Delta t)$ by solving $\widetilde{\text{Boltz}}$ on a time step Δt with the electric field $\mathbf{E}(t_n + \Delta t/2)$ at time $t_{n+1/2}$.

3.2 An efficient hybrid OpenMP/MPI parallelization

The code is developed in Fortran 90 with some I/O routines in C (47k lines of Fortran 90 and 2.3k lines of C code) using GIT as version control system. Doxygen is used to generate documentation from the source code. The only external library dependence is the HDF5 library. HDF5 is the chosen format for all output saving, both 0D to 3D diagnostics and 5D restart files. Diagnostic analyses are performed with Python. The parallelization is based on a hybrid MPI/OpenMP paradigm. This hybrid approach is suitable for cluster of SMP (symmetric shared memory multiprocessor) nodes where MPI provides communication capability across nodes and OpenMP exploits loop level parallelism within a node. Let us denote by N_{species} the number of ion species and by N_μ the number of μ values. As the magnetic momentum μ is an adiabatic invariant it plays the role of a parameter. So for each species we have N_μ independent Boltzmann equations (2.7) to solve. Let N_r, N_θ, N_φ and $N_{v_{G\|}}$ be the number of points in each directions r, θ, φ and $v_{G\|}$. Large data structures are used in GYSELA: 5D data of size $N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}} \times N_\mu$ for distribution functions and 3D data of size $N_r \times N_\theta \times N_\varphi$ for the electrostatic potential and its derivatives as for first moments of the distribution function (used for diagnostics). An MPI parallelization is mandatory to treat such large amount of data. Let us take the example of a typical 5D mesh used for our simulations [2], namely $(N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}} \times N_\mu) = (256 \times 128 \times 128 \times 128 \times 16)$. It corresponds to a mesh of almost 8.6 billion points. The size of one 5D array for the distribution function is of the order of 68 Gbytes, which is not tractable on a single node. For information, the biggest simulation run so far with GYSELA was an ITER simulation [AGG+13] with 272 billion points. Taking into account the fact that two distribution functions are necessary for the numerical integration over time due to predictor-corrector scheme, more than 1 Tbytes of data (just for 5D arrays) were manipulated. So, as described in the following, we use a domain decomposition so that a MPI process never contains the complete 5D distribution function.

Concerning the MPI parallelization, an MPI communicator is defined per species. Inside each one of the MPI_SPECIES communicators, an MPI communicator is defined for each value of the magnetic moment μ. Within each MPI_MU communicator a 2D domain decomposition allows us to assign to each MPI process a sub-domain in (r, θ) dimensions. Let us consider p_r (resp. p_θ) the number of sub-domains in r (resp. θ) direction. The number of MPI processes used during one run is equal to $N_{MPI} = p_r \times p_\theta \times N_\mu \times N_{\text{species}}$. This MPI decomposition of the default MPI_COMM_WORLD communicator is summarized in Figure 3.1. Thus, each MPI process is then responsible for the sub-domain of the distribution function $\bar{F}_s(r = [i_{\text{start}}, i_{\text{end}}], \theta = [j_{\text{start}}, j_{\text{end}}], \varphi = *, v_{G\|} = *, \mu = \mu_{\text{id}})$ with the integer $\mu_{\text{id}} \in [0, N_\mu - 1]$. The local values i_{start}, i_{end}, j_{start} and j_{end} associated to the parallel decomposition are initially set by using a classical domain decomposition in $(p_r \times p_\theta)$ blocks. The OpenMP paradigm is used in addition to MPI. Let us denote by N_{thread} the number of threads in each MPI process, then the number of cores for a simulation corresponds to $N_{\text{thread}} \times N_{MPI}$.

[2] For current GYSELA simulations, N_μ is chosen equal to 16 or 32 while the choice of $N_{v_{G\|}}$ is much larger (typically, $N_{v_{G\|}} \sim 4N_\mu$). Notice however that N_μ should be compared to $N_{v_{G\|}}/2$, since the grid in μ goes from 0 to μ_{max}, while that in $v_{G\|}$ covers the range $-v_{G\|,max}$ to $+v_{G\|,max}$. The choice of $N_{v_{G\|}} > 128$ is necessary to take correctly into account the trapping and de-trapping of particles and also to solve accurately the collision operator (at this time, only in parallel velocity). The same number of points for μ direction will be required when collision operator effects will be added in perpendicular direction (mandatory step for kinetic electrons).

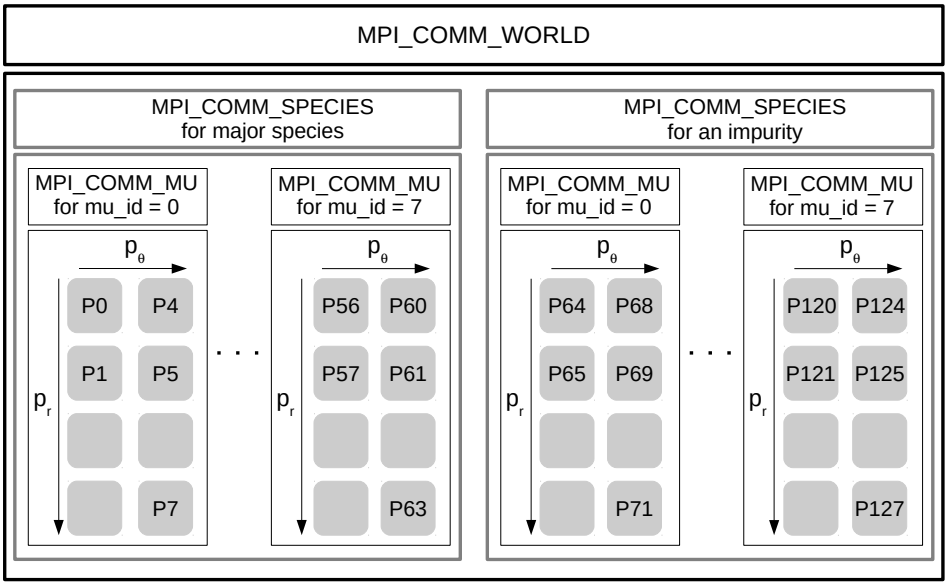


Figure 3.1: MPI_COMM_WORLD communicator decomposition for two species, 8 values of μ, $p_r = 4$ radial sub-domains and $p_\theta = 2$ sub-domains in the poloidal direction. In this case, the number of MPI processes is equal to 128.

3.2.1 Parallel Vlasov solver: How to treat non-local interpolation aspect ?

A difficulty when parallelizing the semi-Lagrangian Vlasov solver is due to the cubic spline interpolation. Cubic splines are a good compromise between simplicity and accuracy but a drawback is that they are non-local. Indeed, a lot of the values of the distribution function $\bar{F}_s$ are required to reconstruct the interpolated value of the function at any position in the domain. Two strategies are available in the code to overcome the problem: (i) local cubic spline interpolation or (ii) transposition. The Hermite spline interpolation on patches [CLS07, CLS09, LCGS07] has been specifically developed for dealing with 2D domains distributed on several MPI processes. The idea is to compute local cubic spline coefficients on each 2D (r, θ) sub-domains by solving reduced linear systems. Then one ensures a $\mathcal{C}^1$ global interpolator similar to the sequential one by imposing Hermite boundary conditions at the interface of each patch [CLS07]. The first limitation of this technique is that a minimum of 32 points per directions is needed per MPI process [LCGS07] (*i.e* $N_r/p_r \geq 32$ and $N_\theta/p_\theta \geq 32$) to provide good numerical stability and small communication overhead (in 2D, each processor has to exchange derivatives with its 8 neighboring processors). The second constraint is that the shift at one point on the border of a sub-domain, which results from the motion along the trajectories in the 4D phase space, must not exceed the elementary cell width. This constraint is linked to the choice made to limit the size of the interface transferred between processors. This CFL condition can be extremely restrictive specially in the θ direction where large shifts can occur but also in the radial direction when a source is imposed in the case of flux-driven simulations. For these reasons the choice of a 4D data transposition is now often preferred. This transposition consists in modifying in each MPI_COMM_MU communicator the parallel decomposition of $\bar{F}_s$ such that each processor then contains only part of the data in φ and $v_{G\|}$ direction but all the information in the poloidal cross-section. Standard cubic spline interpolation in (r, θ) plane are then possible. Let us define the transposition operation T_F and its inverse T_F^{-1} as

$$\bar{F}_s(r_{\text{block}}, \theta_{\text{block}}, \varphi = *, v_{G\|} = *, \mu = \mu_{\text{id}}) \underset{T_F^{-1}}{\overset{T_F}{\rightleftarrows}} \bar{F}_s(r = *, \theta = *, \varphi_{\text{block}}, v_{G_\|\text{block}}, \mu = \mu_{\text{id}})$$

then the sequence (3.6) described above for solving the Boltzmann equation is replaced by the following one

$$\widetilde{\text{Boltz}}_{\text{transp}} \equiv \left[\left(\frac{\tilde{\mathcal{K}}_r}{2}, \frac{\tilde{\mathcal{D}}_r}{2}, \frac{\tilde{\mathcal{C}}}{2}, \frac{\tilde{\mathcal{S}}}{2} \right), \left(\frac{\tilde{v}_{G\|}}{2}, \frac{\tilde{\varphi}}{2} \right), T_F \left(\tilde{\mathcal{X}}_{\mathcal{G}} \right) T_F^{-1}, \left(\frac{\tilde{\varphi}}{2}, \frac{\tilde{v}_{G\|}}{2} \right), \left(\frac{\tilde{\mathcal{S}}}{2}, \frac{\tilde{\mathcal{C}}}{2}, \frac{\tilde{\mathcal{D}}_r}{2}, \frac{\tilde{\mathcal{K}}_r}{2} \right) \right]$$

3.2.2 A scalable quasi-neutral solver

The parallel quasi-neutrality algorithm presently used in the code is summarized in algorithm 1. For more details on the different improvements which have been performed to obtain this parallel solver see [LGC+11, LCG11]. The presence of the non-local term $\langle U \rangle_{\text{FS}}(r)$ couples the θ and φ directions and penalizes the parallelization but the most important cost is the communication induced by the computation of the r.h.s, namely task 2 in algorithm 1. Indeed this calculation requires a collective communication step that involves all MPI processes. Instead of broadcasting $U(r = *, \theta = *, \varphi = *)$ to all MPI processes, a refined strategy has been setup to reduce the large communication cost. It consists in sending to each process only a sub-domain in φ direction of U. With this decomposition the gyro-average computation of U as the partial derivatives in r and θ directions are straightforward. Afterwards, a transposition is performed to calculate $\partial_\varphi U$. For a complete performance analysis of the different steps of the algorithm see [LGCDP12].

Algorithm 1: Quasi-neutrality algorithm in the GYSELA code

Input: $\bar{F}_s(r_{\text{block}}, \theta_{\text{block}}, \varphi = *, v_{G\|} = *, \mu = \mu_{\text{id}})$

1. $v_{G\|}$ integration of $\bar{F}_s$ to compute within each MPI_COMM_MU communicator

$$\texttt{intdvpar_Fs}(r_{\text{block}}, \theta_{\text{block}}, \varphi = *, \mu = \mu_{\text{id}}) = \int \mathcal{J}_{\text{v}} \, \mathrm{d}v_{G\|} \left(\bar{F}_s - \bar{F}_{s,\text{eq}} \right)$$

2. Remapping within each MPI_COMM_MU communicator of `intdvpar_Fs` –because the gyro-average operation requires to have all the data for each (r, θ) plane– as

$$\texttt{intdvpar_Fs}(r_{\text{block}}, \theta_{\text{block}}, \varphi = *, \mu = \mu_{\text{id}}) \Rightarrow \texttt{intdvpar_Fs}(r = *, \theta = *, \varphi_{\text{block}}, \mu_{\text{block}}).$$

3. For all φ, computation of the gyro-average of `intdvpar_Fs` and integration over μ to obtain

$$\rho_s(r = *, \theta = *, \varphi_{\text{block}}) = \int \mathrm{d}\mu_s \mathcal{J}_\mu . \cdot (\texttt{intdvpar_Fs}(r = *, \theta = *, \varphi_{\text{block}}))$$

4. MPI reduction towards the N_{species} MPI_COMM_SPECIES communicators to finally compute $\rho = \sum_s Z_s \rho_s$, the right hand side of the quasi-neutrality equation (2.13).

5. Solving (2.17) and (2.18) to deduce $U(r = *, \theta = *, \varphi_{\text{block}})$ and broadcast to the $N_{\text{species}} \times N_\mu$ communicators.

Output: $U(r = *, \theta = *, \varphi_{\text{block}})$ on each MPI process.

3.2.3 Performing weak and strong scaling

The weak and strong scaling, presented in this section, have been performed on two different high performance computers. The strong scaling has been tested on the thin nodes of the Curie machine at TGCC[3], Bruyères-le-Châtel, France which are based on bullx B510 architecture. Each node contains two 8 cores processors (INTEL Xeon E5-2680 Sandy-Bridge) running at 2.7 GHz with 64 GB of 1.6 GHz memory. The weak scaling has been performed on the IBM Blue Gene/Q machine JUQUEEN at

[3] http://www-hpc.cea.fr/en/complexe/tgcc.htm

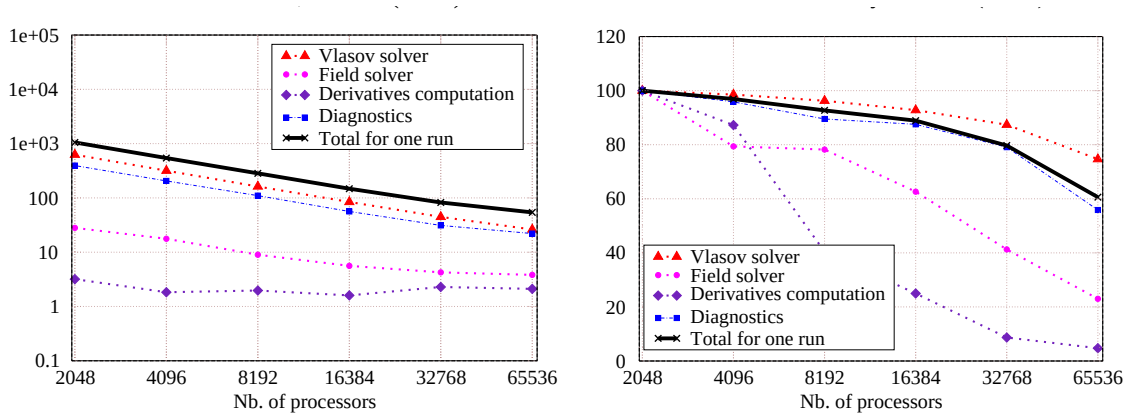
JSC/IAS[4], Juelich, Germany which is composed of 24 racks grouping 1024 nodes. Each node contains a single 17-cores processor running at 1.6 GHz with 16 GB of 1.33 GHz memory. These cores are twice slower compared to Sandy Bridge cores. The improvements of the code to adapt it efficiently to the BlueGene architecture are detailed in [BGL+13]. The scaling results are detailed in Figures 3.2 and 3.3 for the 4 main components of the code, namely (i) *Vlasov solver*, solving the Boltzmann equation as described in section 3.2.1; (ii) *Field solver*, solving the quasi-neutrality equation as summarized in algorithm 1; (iii) *Derivatives computation*, the computation of the first derivatives of the gyro-averaged electrostatic potential $\mathcal{J}_\mu.U$ in the three r, θ and φ directions and (iv) *Diagnostics* corresponding to all physical quantities from 0D to 3D computed and saved in HDF5 format like densities, parallel and perpendicular temperatures, velocities, fluxes, energies et caetera. The strong scaling has been performed with the mesh size parameters $N_r = 512$, $N_\theta = 512$, $N_\varphi = 128$, $N_{v_{G\|}} = 128$ and $N_\mu = 32$. The number of threads was kept constant $N_{\text{thread}} = 8$ to assign two MPI process per node. The couple (p_r, p_θ) of processors in r and θ directions take the following values $(2,4)$, $(8,2)$, $(8,4)$, $(16,4)$, $(16,8)$ and $(16,16)$ so that the scaling spans from 2048 to 65536 cores. For the weak scaling the testbed case was composed from 64k to 458k cores. The considered meshes vary from 17.18 to 481 billion points defined with $N_r = 512$, $N_\theta = 1024$, $N_\varphi = 128$, $N_{v_{G\|}} = 128$ and 7 different values of $N_\mu = 2$, 4, 8, 16, 32, 48 and 56. The triplet $(p_r, p_\theta, N_{\text{thread}})$ is kept constant equal to $(16, 32, 64)$. The number of threads is chosen equal to 64 such that a single MPI process is mapped per node and 4 threads are dedicated per core as determined for BlueGene/Q optimization. The results of both scalings are summarized in Figures 3.2 and 3.3. Let us first remark that the weak scaling test (Fig. 3.3) exhibits an excellent scalability of the code with 90.9% of relative efficiency at 458752 cores –which corresponds to the totality of the JUQUEEN computer– compared to 16k. Concerning the strong scaling GYSELA globally scales with a relative efficiency of of 89% at 16k cores and 60.6% on 65k cores compared to 2048 cores (Fig. 3.2). This is already a very good result for such a semi-Lagrangian code. Looking into more detail, we see that the deterioration from 89% at 16k to 60.6% at 65k is mainly due to the diagnostics and field solver. Indeed, the Vlasov solver which represents 60% of the application at 2k cores and 48% at 65k cores exhibits a good efficiency of 74.6% at 65k processors. Conversely, the diagnostics which correspond to 37.5% of the total time for the reference case, end taking as much time as the Vlasov solver due to the decreased efficiency at 55.8%. A lot of work has already been done to improve the field solver [LCG11, LGC+11, LGCDP12] but this work must continue because an efficiency of 41% impacts the global scalability. Let us finally notice that even if the computation time of the derivatives remains negligible until 16k, a further effort is needed to prepare the code to future exascale machines.

The execution times are not comparable in the two scalings presented here because the considered meshes are not the same. Performance comparisons between BlueGene/Q and Bullx architectures are found in [BGL+13]. Production runs are commonly running on both architectures. In general, depending on simulation parameters, the code is a factor 2.5 to 3 times faster on bullx machines (as Curie or Helios at CSC, Rokkasho, Japan) than on BlueGene machines (as JUQUEEN or Turing at IDRIS, Orsay, France or Fermi at CINECA, Bologna, Italy). This behavior is consistent with that observed with many other codes.

3.2.4 Memory scalability

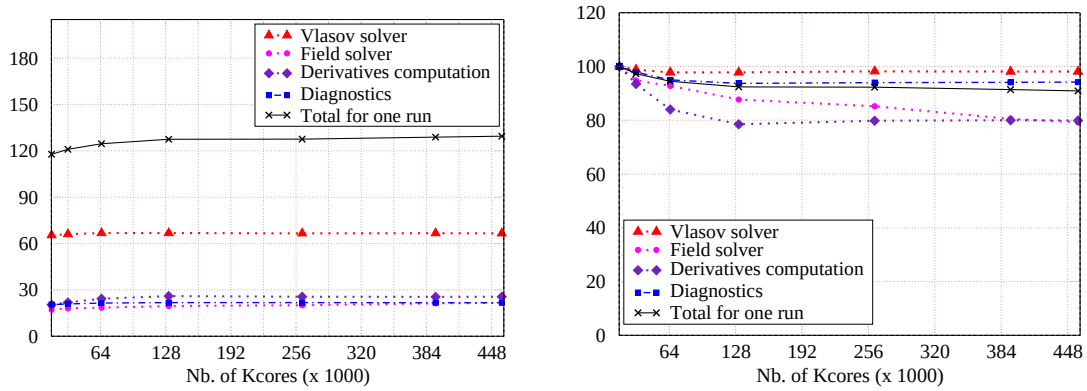
Due to the previous scalability results GYSELA uses frequently from 8k to 32k cores for one ion species with adiabatic electrons and the twice these values when an impurity is taken into account. Besides, a simulation often runs during several weeks. The annual time consumption on HPC machines is currently of 51 millions of core hours. So the code already benefits from petascale computational power of the current high performance computers. We also know that at short term when adding kinetic electrons, a simulation with ITER parameters on several confinement times will require exascale HPC capabilities. Among the Exascale challenges, the reduced memory per core has been identified as one of the most critical. This is particularly true for the GYSELA code due its global character which requires huge 5D meshes. The mesh discretization is already constrained by the memory required per node.

[4] http://www.fz-juelich.de/ias/jsc/EN/Home/home_node.html



(a) Execution time (b) Relative efficiency

Figure 3.2: Strong scaling performed on the Curie machine from 2048 to 65536 cores: Execution time (a) and relative efficiency (b) for one GYSELA run of 4 iterations for a mesh $(N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}} \times N_\mu) = (512 \times 512 \times 128 \times 128 \times 32)$ with 8 threads and 32 values in μ direction.



(a) Execution time (b) Relative efficiency

Figure 3.3: Weak scaling performed on the JUQUEEN machine from $64k$ to $458k$ cores: Execution time (a) and relative efficiency (b) for one GYSELA run of 4 iterations for a mesh fixed in 4D as $(N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}}) = (512 \times 1024 \times 128 \times 128)$ but for 7 different values of $\mu = 2, 4, 8, 16, 32, 48, 56$. The number of parallel domains in r, θ directions and the number of threads are fixed such that $(p_r, p_\theta, N_{\text{thread}}) = (16, 32, 64)$. The number of cores varies as $p_r \times p_\theta \times N_\mu \times N_{\text{thread}}/4$ because 4 threads per core are used for BlueGene optimization.

The C/Fortran MTM (Modelization & Tracing Memory consumption) library has been developed [RLRG15] to investigate in detail the memory consumption of the code. This library provides an Application Programming Interface (API) which replaces the standard calls to allocation/de-allocation routines. This intrusive technique permits to retrieve precisely the peak of memory consumption and all the arrays involved. External Python scripts have been designed to analyze these results and provide memory prediction. Continued efforts are made to reduce the memory footprint of the code and to improve its memory scalability (see [RLRG15] for details) but Table 3.1 shows that lots of work remains to be done to be able to run ITER simulations on machines with only 16GB per node. Indeed, for a minimum ITER mesh, namely $(N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}} \times N_\mu) = (1024 \times 1024 \times 256 \times 128 \times 16)$, Table 3.1 shows that such simulations can run on Curie Machine (64 GB/node) with 8192 cores but would require 524288 cores on the BlueGene current architecture (16 GB/node)- knowing that the optimal number of threads is 64. This number of cores is still out of reach on European high performance computers. Currently, as shown in Table 3.2 and in Figure 3.4, the global memory scalability of the code is of 47.8%. This strong scaling was performed with a constant 5D mesh of $(1024 \times 4096 \times 1024 \times 128 \times 2)$ points using MTM prediction mode and varying the number of MPI processes from 128 to 2048 by increasing the number of points in r and θ directions. The 4D structures are very scalable (relative

MPI procs \ OpenMP threads	16	32	64	128
128	126.1 GB	126.9 GB	128.7 GB	132.2 GB
512	35.2 GB	36.1 GB	37.8 GB	41.3 GB
2048	16.4 GB	16.4 GB	16.5 GB	23.2 GB
8192	12.5 GB	12.5 GB	12.5 GB	19.1 GB

Table 3.1: Memory peak (in GBytes) depending on the number of MPI procs and of OpenMP threads for a 5D mesh $(N_r \times N_\theta \times N_\varphi \times N_{v_{G\|}} \times N_\mu) = (1024 \times 1024 \times 256 \times 128 \times 16)$.

efficiency of 89.9%) compared to the others (see Fig.3.4). Indeed, 3D structures with 23.8% represent 32% of the global cost for 32k cores against 16% for 2048 cores. The 3D structures are then no more negligible. The scalability of the 1D structures is with 19% of the order of the 3D arrays but the cost is still not significant. At the opposite, the incompressibility of 2D structures between 256 and 2k MPI processes is extremely penalizing leading to an increase of the cost percentage by a factor 10. The fact that the memory for 2D structures remains constant (equal to 7.1 GBytes) is probably due to incompressible temporary arrays allocated for OpenMP loops. Improvement of memory footprint of 3D and 2D structures will be pursued in the future.

Number of cores Number of MPI processes		2k 128	4k 256	8k 512	16k 1024	32k 2048
4D structures	Memory size	207.2 GB	104.4 GB	53.7 GB	27.3 GB	14.4 GB
	Memory percentage	79.2%	71.5%	65.6%	52.2%	42.0%
3D structures	Memory size	42.0 GB	31.1 GB	18.6 GB	15.9 GB	11.0 GB
	Memory percentage	16.1%	21.3%	22.7%	30.4%	32.1%
2D structures	Memory size	7.1 GB	7.1 GB	7.1 GB	7.1 GB	7.1 GB
	Memory percentage	2.7%	4.9%	8.7%	13.6%	20.8%
1D structures	Memory size	5.2 GB	3.3 GB	2.4 GB	2.0 GB	1.7 GB
	Memory percentage	2.0%	2.3%	3.0%	3.8%	5.1%
Total per MPI process in GBytes		261.5	145.9	81.8	52.3	34.2

Table 3.2: Strong scaling for each kind of data for a 5D mesh $(1024 \times 4096 \times 1024 \times 128 \times 2)$: (first lines) memory allocation size in GBytes and (second lines) percentage with respect to the total memory at the peak of the memory consumption. (Table from [RLRG15]).

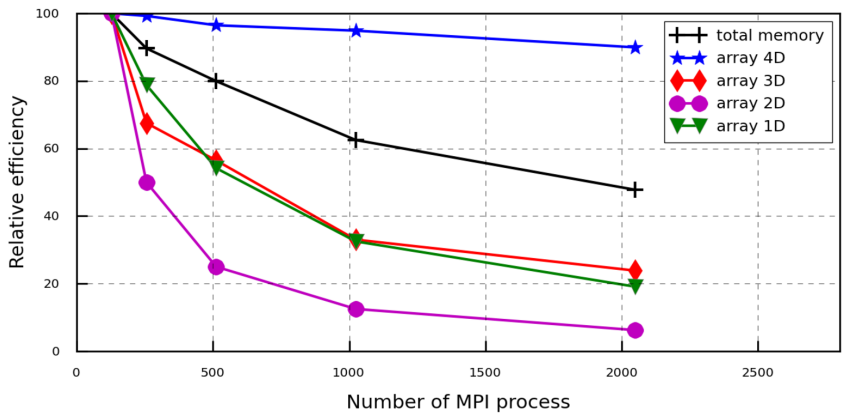


Figure 3.4: Memory relative efficiency for a GYSELA simulation at the memory peak of time consumption for the four kind of structures used in the code (1D arrays to 4D arrays). The results are extracted from Table 3.2. The reference point corresponds to 128 MPI processes.

Another bottleneck for Exascale applications will be the possibly increased crash probability following that of the number of cores of future machines [Cap09]. The Weibull law [SG10] gives an estimate of the time between two crashes which is of the order of the minute for a number on nodes larger than 10^5. Two approaches have been explored until now to try to improve the fault tolerance of the code. The first one consists in employing an asynchronous method to increase the frequency of the restart file writing. A GYSELA simulation runs typically several days or even weeks. So the complete simulation is split into a series of jobs of approximately 10 hours with automatic re-submission. The restart files are not only saved at the end of each job but several times during the job in parallel to calculations (see [TBG+13] for complete description). Another checkpointing using the FTI library [BGTK+11] (high performance Fault Tolerance Interface for hybrid systems) is under investigation. The main idea is to benefit from fast access local SSD disks available on some HPC architectures.

Chapter 4

GYSELA verification

Contents

4.1 Normalization **69**
4.2 Invariance test **70**
4.3 Usual linear and non-linear gyrokinetic benchmarks **73**
4.3.1 Rosenbluth-Hinton test 73
4.3.2 Linear benchmark based on Cyclone DIII-D case for global codes 76
4.4 Local conservation laws for gyrokinetics **79**
4.4.1 Radial force balance equation 80
4.5 Local charge density conservation **80**
4.5.1 Local energy conservation 83
4.5.2 Local toroidal momentum conservation 85

Since the first steps 15 years ago, the code has evolved significantly, including more and more physics with more and more numerical complexity and high level of parallelism. Portability of the code on several HPC architectures has been achieved. In such a complex code it becomes very difficult to track or to propagate changes by being sure to reproduce previous results and simulations. Over the past two years, a large effort has been made to improve the development process with due regard to modularity, reproducibility and efficiency. This approach is based on the joint use of a version control system (GIT[1]) together with that of a continuous integration platform such as JENKINS[2]. At each commit on the GIT reference branch, automatic compiling and executing are submitted. For more details on our strategy the reader can refer to [BCMG+15]. A database of non-regression tests is also under construction with the objective to be run less frequently (every night or week) but to ensure that new changes have no impact on well-established results. All the tests proposed in this section, which were used for GYSELA verification[3], are detailed with the objective to become part of the non-regression database. They all correspond to 5D gyrokinetic simulations for one species. Other verification tests can be founded on simpler 4D models in [GBB+06, LGA+14].

4.1 Normalization

The GYSELA code is written in normalized units based on the following normalization choices. We use SI units and a thermal energy scale in electron volts ($1eV = 1.6022\ 10^{-19}J$). The four fundamental dimensional normalizing quantities are: a reference ionic mass $m_0 = A_0 m_p$ (Kilogram), a reference ionic charge $q_0 = Z_0 e$ (Coulomb), a reference magnetic induction B_0 (Tesla) and a reference thermal energy T_0 (eV). Here, A_0 and Z_0 are the (dimensionless) mass number and charge state of the main

[1] http://git-scm.com/
[2] http://jenkins-ci.org
[3] The numerical results presented in the following are based on the GIT release 17.0 of the code.

ion species and e the modulus of the electron charge. These quantities are used to define the reference ion cyclotron frequency Ω_{c_0}, the reference thermal speed v_{T_0} and the reference Larmor-radius ρ_0 as

$$\Omega_{c_0} = \frac{Z_0\, e\, B_0}{m_0} \quad ; \quad v_{T_0} = \sqrt{\frac{T_0}{m_0}} \quad ; \quad \rho_0 = \frac{v_{T_0}}{\Omega_{c_0}} = \frac{\sqrt{T_0 m_0}}{Z_0\, e\, B_0}$$

Finally, we choose the equilibrium electron density at mid radius n_0 as reference density. Physical quantities (mass, length, time, charge and density) can be recovered from the normalized quantities used in the code (denoted with a hat symbol) by choosing values for $[A_0, Z_0, B_0, T_0, n_0]$ and applying $m_s = A_s m_p = m_0 \hat{A}_s$ with $A_s = A_0 \hat{A}_s$, $q_s = Z_0\, e\, \hat{Z}_s$, $l = \rho_0\, \hat{l}$, $t = \frac{\hat{t}}{\Omega_{c_0}}$ and $n_s = n_0\, \hat{n}_s$. The velocities are normalized to the corresponding thermal velocities $v_{T_{s0}} = \sqrt{T_0/m_s}$, *i.e* $v = v_{T_{s0}} \hat{v}_s = \frac{v_{T_0}}{\sqrt{\hat{A}_s}}\, \hat{v}_s$. The main normalizations are thus $U = \frac{T_0}{Z_0\, e} \hat{\phi}$, $B = B_0 \hat{B}$, $T_s = T_0 \hat{T}_s$ while $\mu_s = \frac{T_0}{B_0} \hat{\mu}_s$ with $\hat{\mu}_s = \frac{\hat{v}_{\perp s}^2}{2\hat{B}}$ and $\mu_0 \mathbf{J} = \frac{B_0}{\rho_0} \mu_0 \hat{\mathbf{J}}$ with $\mu_0 \hat{\mathbf{J}} = \hat{\boldsymbol{\nabla}} \times \hat{\mathbf{B}}$. By deduction[4], $\mathbf{E} = E_0 \hat{\mathbf{E}}$ with $E_0 = v_{T_0} B_0$ and the normalized distribution function $\hat{F}_s$, which evolves in the code, is defined as $\hat{F}_s = F_s v_{T_{s0}}^3 / n_0$. Finally, the energy is normalized to the reference thermal energy T_0. The code normalization is summarized in Table 4.1. The subsequent normalized equations used in the code are presented in Appendix K. In this chapter,

$$\boxed{\begin{array}{rcl} m_s & = & m_0 \hat{A}_s \\ q_s & = & Z_0\, e \hat{Z}_s \\ l & = & \rho_0 \hat{l} \\ t & = & \hat{t}/\Omega_{c_0} \\ n_s & = & n_0\, \hat{n}_s \\ T_s & = & T_0 \hat{T}_s \\ B & = & B_0 \hat{B} \end{array}} \quad \Rightarrow \quad \boxed{\begin{array}{rcl} v_{G\|} & = & v_{T_{s0}} \hat{v}_{G\|s} \\ & = & (v_{T_0}/\sqrt{\hat{A}_s}) \hat{v}_{G\|s} \\ \mu_s & = & (T_0/B_0) \hat{\mu}_s \\ U & = & [T_0/(Z_0\, e)] \hat{\phi} \\ E & = & (v_{T_0} B_0) \hat{E} \\ F_s & = & (n_0/v_{T_{s0}}^3) \hat{F}_s \end{array}}$$

Table 4.1: Links between physical and normalized quantities.

all the quantities considered are normalized coordinates, but hat symbols are omitted for the sake of readability.

4.2 Invariance test

In the present work the source terms are not taken into account. Let us call `simu1` a first fixed gradient simulation (*i.e* no source term) for one unique species of charge Z_1 and mass A_1 and `simu2` a second one for a species of charge $Z_2 = \alpha_Z Z_1$ and mass $A_2 = \alpha_A A_1$. Then it is possible to define the other parameters of the second simulation, only depending on the multiplying factors $\alpha_Z \in \mathbb{N}$ and $\alpha_A \in \mathbb{N}$ such that both simulations `simu1` and `simu2` are identical. The idea is to define an invariance test which permits to verify that charge and mass are correctly taken into account in the code. Notice that such a transformation departs from a simple ρ_* scaling, where only the ρ_* parameter is modified (via e.g. a change of the minor radius at constant aspect ratio) [CWD04, MLB+10]. Such a test can be defined as a non-regression test of any gyrokinetic code but it is not generic. It depends on the equations treated in each code. For any code, the solved equations (gyrokinetic + quasi-neutrality) are invariant under certain groups of transformations. Within these groups, the invariance is not approximate; it is exact provided that the code actually solves the equations it is supposed to. We have derived in Appendix L such a group of transformations which leaves the solution unchanged for the GYSELA code. Seven control parameters are required α_x, α_t, α_{T_e}, α_ϕ, α_{coll}, α_{diff} and α_{Krook} that respectively rescale space, time, electron temperature, electrostatic potential, collision operator, diffusion term and Krook operator. The only issue which cannot be accounted for (i.e. which cannot be

[4] $\mathbf{E} = -\boldsymbol{\nabla} U = -\frac{T_0}{\rho_0\, e\, Z_0} \hat{\boldsymbol{\nabla}} \hat{\phi} = -\frac{1}{\sqrt{T_0 m_0}} \frac{e\, Z_0 B_0 T_0}{e\, Z_0} \hat{\boldsymbol{\nabla}} \hat{\phi} = -v_{T_0} B_0 \hat{\boldsymbol{\nabla}} \hat{\phi}$

rescaled) in this operation is the impact of the boundary conditions. The invariance constraints (*i.e* the relationships between the various scaling factors which leave the Boltzmann equation invariant under these transformations), as summarized in Table 4.2, are $\alpha_x = \sqrt{\alpha_A}/\alpha_Z$, $\alpha_t = \alpha_A/\alpha_Z$, $\alpha_{T_e} = 1/\alpha_Z$, $\alpha_{\text{coll}} = 1$, $\alpha_{\text{diff}} = 1/\alpha_Z$ and $\alpha_{\text{Krook}} = \alpha_Z/\alpha_A$. Then, the relation between the electrostatic potential ϕ_1 solution of the first simulation and ϕ_2 the electrostatic potential of the second one is $\phi_2 = \alpha_\phi \phi_1$ with $\alpha_\phi = 1/\alpha_Z$.

	Simulation 1	Simulation 2
Charge	Z_1	$Z_2 = \alpha_Z Z_1$
Mass	A_1	$A_2 = \alpha_A A_1$
Mesh discretization	Δx_1	$\Delta x_2 = \frac{\sqrt{\alpha_A}}{\alpha_Z}\Delta x_1$
Time step Δt	Δt_1	$\Delta t_2 = \frac{\alpha_A}{\alpha_Z}\Delta t_1$
Temperature T_e	T_{e1}	$T_{e2} = \frac{1}{\alpha_Z}T_{e1}$
Collision coefficient	K_{coll1}	$K_{\text{coll2}} = K_{\text{coll1}}$
Krook coefficient	K_{krook1}	$K_{\text{krook2}} = \frac{\alpha_Z}{\alpha_A}K_{\text{krook1}}$
Diffusion coefficient	K_{diff1}	$K_{\text{diff2}} = \frac{1}{\alpha_Z}K_{\text{diff1}}$

Table 4.2: Parameter dependence between two equivalent simulations. α_Z and α_A are the multiplying factors between both cases respectively for charge and mass.

Three different simulations were run to test the invariance property of the code, considering first a reference plasma with Hydrogen (mass $A_s = 1$ and charge $Z_s = 1$), then Helium ($A_s = 4$ and $Z_s = 2$) and finally Tungsten ($A_s = 150$ and $Z_s = 50$). For the following, let us respectively call `simu1_A1Z1`, `simu2_A4Z2` and `simu3_A150Z50` these three simulations. These cases are not relevant in terms of physics but have been designed for numerical tests. The idea was to define small cases tractable as non-regression tests. So the reference simulation is based on a small plasma ($\rho_* = 1/150$) for a radial domain between 0.15ρ and 0.85ρ (with $\rho = r/a$) and for half a torus. The parallel velocity space is defined as $-7v_{T_{s0}} \leq v_{G\|} \leq 7v_{T_{s0}}$ and the perpendicular direction is represented by 16 values of μ between 0 and 12. The radial profiles of density $n_{s_0}(r)$, temperature $T_{s_0}(r)$ and safety factor $q(r)$ are analytically prescribed as $\mathrm{d}\log n_{s_0}(r)/\mathrm{d}r = -2.2\cosh^{-2}((\rho - 0.5)/0.04)$, $\mathrm{d}\log T_{s_0}(r)/\mathrm{d}r = -8\cosh^{-2}((\rho - 0.5)/0.04)$ with $\rho = r - r_{\min}/(r_{\max} - r_{\min})$ and $q(r) = 1.5 + 1.7\exp(2.8\log(r/a))$. A Krook operator (see section 2.8.4) of amplitude 0.01 and a diffusion (eq.(2.53)) of 0.015 are applied in a buffer region defined by eq.(2.52) with a location $B_L = 0.06$ and a stiffness of 0.017635. The collision operator (eq.(2.32)) is applied every iterations while $T_{s,\text{coll}}$ (eq.(2.34)) and $V_{\|s,\text{coll}}$ (eq.(2.33)) are refreshed every 10 iterations. All the numerical parameters of this `simu1_A1Z1` simulation are summarized in Tables 4.3 and 4.4. For the complete description of the case see the associated input data file of the code (Figure N.1) in Appendix N.4.
So taking into account the equivalences defined in Table 4.2, the second simulation `simu2_A4Z2` is defined with the same parameters as `simu1_A1Z1` except the fact that Helium is considered instead of Hydrogen and (i) $\tau = 2.$, (ii) $\Omega_{c_0}\Delta t = 10.$, (iii) $D_{\text{coeff}} = 0.0075$, (iv) $\text{Krook}_{\text{coeff}} = 0.005$ while (v) the diagnostic time step $\Omega_{c_0}\Delta t_{\text{diag}} = 100$ instead of 50. The species concentration c_s has been divided by 2 to satisfy the constrainst $\sum_s c_s Z_s = 1$. The third simulation `simu3_A150Z50` is performed for a Tungsten species $A_s = 150$ and $Z_s = 50$ and differs from `simu1_A1Z1` by the following parameters: (i) $1/\rho_* = 36.7423$ ($1/\rho_{*\text{simu1}} \times \sqrt{150}/50$), (ii) $c_s = 0.02$, (iii) $\tau = 50$, (iv) $\Omega_{c_0}\Delta t = 15$, (v) $\Omega_{c_0}\Delta t_{\text{diag}} = 150$, (vi) $D_{\text{coeff}} = 0.0003$ and (vii) $\text{Krook}_{\text{coeff}} = 0.01/3$. The differences between the three considered simulations are highlighted in Table 4.5. The time evolution of the zonal flow component ($m = 0$, $n = 0$), where m and n are respectively poloidal and toroidal mode number in the middle of the radial domain is presented in Figure 4.1 for the three simulations. It exhibits a relative error of 10^{-15} when Hydrogen case is compared to Helium case and of 10^{-7} when compared with Tungsten simulation. The loss of accuracy for Tungsten is probably due to the small size of the plasma considered ($a = 36.7423/\rho_0$). In this case boundary conditions can play a more important role. For each test

N_r	N_θ	N_φ	$N_{v_{G\|}}$	N_μ	nb_{vth0}	$\mu_{\max}$	
256	256	128	128	16	7.	12.	
ρ_*	R_0/a	r_{int}/a	r_{ext}/a	Torus	Z_s	A_s	$\Omega_{c_0}\Delta t$
1/150	3.3	0.15	0.75	1/2	1.	1.	5.
q_1	q_2	q_3	κ_{ns_0}	Δ_{ns_0}	κ_{Ts_0}	Δ_{Ts_0}	T_i/T_e
1.5	1.7	2.8	2.2	0.04	8.	0.04	1.

Table 4.3: Main parameters for reference simulation `simu1_A1Z1`. The velocity phase space is defined by $-\mathrm{nb}_{vth0}v_{T_{s0}} \leq v_{G\|} \leq \mathrm{nb}_{vth0}v_{T_{s0}}$ and $0 \leq \mu \leq \mu_{\max}T_0/B_0$. Torus indicates the fraction of the torus simulated. The safety factor radial profile is defined as $q(r) = q_1 + q_2\exp(q_3\log(r/a))$. The radial density profile is defined by its gradient as $\mathrm{d}\log n_{s_0}(r)/\mathrm{d}r = -\kappa_{ns_0}\cosh^{-2}((r-0.5)/\Delta r_{ns_0})$. The same analytical expression is used for the temperature with κ_{Ts_0} and Δr_{Ts_0}.

Buffer region					Collision operator		
B_L	B_S	D_{coeff}	$\mathrm{Krook}_{\mathrm{coeff}}$	$\nu_{*,\mathrm{buff}}$	$\nu_{*,\mathrm{coeff}}$	$\mathrm{nbstep}_{\mathrm{coll}}$	$\mathrm{nbrefresh}_{\mathrm{coll}}$
0.06	0.017635	0.015	0.01	20.	0.28	1	10

Table 4.4: Main parameters for `simu1_A1Z1` simulation concerning buffer region and collisions. B_L and B_S correspond respectively to the location and the stiffness of the buffer regions where both diffusion and Krook operator are applied with respective amplitude D_{coeff} and $\mathrm{Krook}_{\mathrm{coeff}}$. The collision operator is applied every $\mathrm{nbstep}_{\mathrm{coll}}$ iterations but refreshed every $\mathrm{nbrefresh}_{\mathrm{coll}}$ iterations.

the results are sufficiently accurate to consider that the charge and mass factors have been correctly implemented in the code equations. The same order of magnitude is obtained when the distribution function values are compared.

	A_s	Z_s	$1/\rho_*$	τ	$\Omega_{c_0}\Delta t$	$\Omega_{c_0}\Delta t_{\mathrm{diag}}$	D_{coeff}	$\mathrm{Krook}_{\mathrm{coeff}}$
`simu1_A1Z1`	1.	1.	150.	1.	5.	50.	0.015	0.01
`simu2_A4Z2`	4.	2.	150.	2.	10.	100.	0.0075	0.005
`simu3_A150Z50`	150.	50.	36.7423	50.	15.	150.	0.0003	0.00333333

Table 4.5: Modified parameters according to the scaling law for the three simulations used in invariance tests.

The same invariance exercise was performed for a smaller reference case: $\rho_* = 1/75$, $N_r = 128$, $N_\theta = 128$ and a full torus (all the other simulation parameters being identical to the ones described in Tables 4.3 and 4.4). The two others simulations defined for comparison follow the same rules as previously which corresponds to a ρ_* of 1/18.37 for the Tungsten case. The relative error is also of the order of 10^{-15} when comparing Hydrogen and Helium but of 10^{-6} when considering the Tungsten. The two cases $\rho_* = 1/75$ for Hydrogen and Helium will be good candidates as non-regression tests. Regarding the Tungsten simulations more work could be done. The first idea would be to consider a $\rho_* = 1/75$ plasma for Tungsten, but this would imply a reference case with $\rho_* \sim 1/306.205$ for the Hydrogen (*i.e* a mesh of the order of 68 billion of points with $(N_r, N_\theta, N_\varphi, N_{v_{G\|}}, N_\mu) = (512, 512, 128, 128, 16)$). Such a simulation is feasible but extremely time consuming for a verification test (several hours on 8192 cores). It would not be possible on the development cluster Poincare (92 nodes of 8 Sandy Bridge E5-2670 bi-processors (2.60GHz)) at IDRIS french supercomputing center that has been used for all tests presented above (performed on 512 cores).

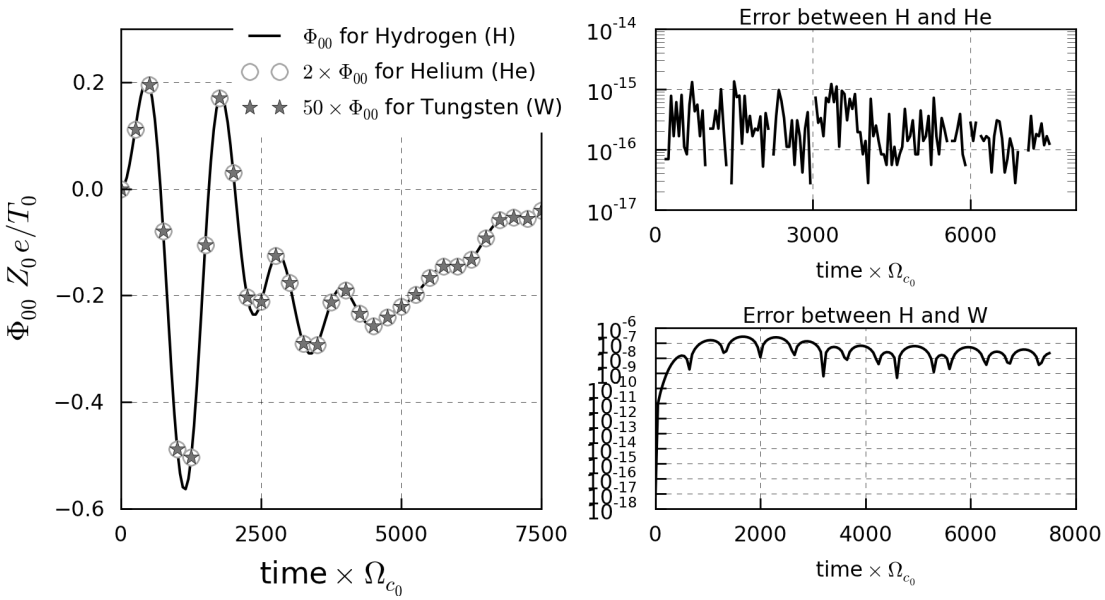


Figure 4.1: Invariance test between three simulations. Left Figure: Time evolution of Zonal Flows Φ_{00} at a fixed radial position 0.5ρ with $\rho = (r_{\max} - r_{\min})/a$. Φ_{00} for `simu1_A1Z1` with Hydrogen must be compared to $2\Phi_{00}$ of `simu2_A4Z2` for Helium and $50\Phi_{00}$ obtained with `simu3_A150Z50` for Tungsten. Right Figures: Relative errors in function of time: (top) between `simu1_A1Z1` and `simu2_A4Z2`, (bottom) between `simu1_A1Z1` and `simu3_A150Z50`.

4.3 Usual linear and non-linear gyrokinetic benchmarks

Together with the verification invariance test described above, several other benchmarks have already been performed in the past to validate the code. The so-called "Rosenbluth-Hinton test" (RH) [RH98] –which became an essential test for gyrokinetic codes to check the validity of zonal flows and Geodesic-Accoustic-Modes (GAMs) treatment– was verified in 2008 [GSG+08]. Linear and non-linear benchmarks were successfully achieved [GSG+08] with the classical Cyclone DIII-D base case (CBC) [DBB+00]. Non-linear benchmarks were also performed [GSA+07] with the global PIC code ORB5 [JBA+07]. We also participated to the European turbulence code benchmarking effort [FSA+08]. Finally, flux-driven simulations have been compared [SGA+11] between GYSELA, ORB5 and XGC1 [CK08, CKD+09]. In the following we present only the tests which have been recently investigated again with the new multi-species version of the code to be added as non-regression tests, namely the RH test and the linear CBC test. These tests have been re-designed with the objective of being sufficiently small to be automatically launched on the continuous integration platform(JENKINS) at a reasonable frequency with the aim of recovering the predicted results. Here are described the results of our investigation. This work is still in progress and we plan to continue decreasing the CPU time consumption of such tests in the future.

4.3.1 Rosenbluth-Hinton test

This test consists in studying the collisionless relaxation of an initial $E \times B$ poloidal flow, including the transient GAM oscillation (Geodesic Acoustic Mode), towards a non vanishing residual value. Such simulations are interesting validation tests for gyrokinetic codes because the damping, the frequency and the residual value of the GAM can be compared to theoretical predictions [SW06, ZCS96]. In practice, a zonal perturbation in ion density is initialized, with a radial profile of the form $\sin(\pi r/a)$. This initial state leads to the development of GAMs which correspond to the $(m, n) = (0, 0)$ mode coupled to sidebands $(m, n) = (\pm 1, 0)$ as a result of toroidal coupling. These GAMS are Landau-damped because of the finite poloidal wavenumber of the sideband while the zonal flows relax towards a residual value which has been analytically predicted in the case of large aspect ratio and small ρ_* [HR99] as $\phi_{00}(t_\infty) = \phi_{00}(t_0) \times A_r$ with $A_r = 1/(1 + 1.6q^2/\sqrt{r/R})$, r being the minor radius of the

considered magnetic surface, R and q corresponding respectively to the major radius and the safety factor on this surface of interest.

Rosenbluth-Hinton test as a non-regression test

Eight simulations (detailed in Table 4.7) were performed varying both time and phase space discretizations. The initial distribution function is defined as $\bar{F}_s = \bar{F}_{s,\mathrm{eq}}(1 + \epsilon \sin(2\pi(r - r_{\min})/2L_r))$ with a perturbation amplitude ϵ equal to 10^{-3}. For all simulations (see Table 4.6 for common parameters), the safety factor profile is constant ($q(r) = 1.9 \forall r \in [r_{\min}, r_{\max}]$). The density and temperature profiles are quasi-constant with $\kappa_{Ts_0} = \kappa_{ns_0} = 10^{-7}$. For a complete description of the case see the associated input data file of the code (Figure N.2) in Appendix N.4. Time evolution of the $(0, 0)$ mode is plotted in Figure 4.2 for the highest discretized simulation (simu 1 in Table 4.7). The theoretically predicted residual value A_r is found to be recovered up to 20% in this case. A much better agreement can be found by decreasing the ρ_* value of the simulation as noticed by Biancalani (see figure 4 in [BBLZ14]). A detailed analysis of both the transient GAM oscillation and damping on the one hand, and of the residual value of zonal flows on the other hand, is presented in next paragraph 4.3.1. The numerical echo appearing at time $t = 15000\Omega_{C_0}^{-1}$ is due to finite discretization of the velocity phase space. The numerical damping rates γ_{GAM} and frequencies ω_{GAM} reported in Table 4.7 are all computed between $t \in [0, 15000\,\Omega_{C_0}^{-1}]$. We observe that all the values are equal to $\gamma_{\mathrm{GAM}} = 2.46459\,10^{-4}\Omega_{C_0}$ and $\omega_{\mathrm{GAM}} = 9.11061\,10^{-3}\Omega_{C_0}$ with an error smaller than 3% except for simulations number 4, 6 and 8 in Table 4.7. The error larger than 7% for both simulations 4 and 6 is due to the fact that the numerical echo appears at time $t = 9000\,\Omega_{C_0}^{-1}$ instead of $t = 15000\Omega_{C_0}^{-1}$, making the numerical residual estimation impossible. Error in simulation 4 is due to the fact that four values of μ are not sufficient for μ integral computation required for the r.h.s of the quasi-neutrality equation. Results for simulation 6 shows that 64 in parallel velocity direction are not sufficient. The CPU times reported as results in Table 4.7 correspond to mono-core hours on the Poincaré machine where all simulations were performed with 4 threads and $(2 \times N_\mu)$ MPI processes for a final time equal to $t = 20000\Omega_{C_0}^{-1}$. Simulation 7 corresponding to 30 minutes on 64 cores is a non-regression test for the code.

ρ_*	A_s	Z_s	R_0/a	r_{int}/a	r_{ext}/a	Torus	nb_{vth0}	$\mu_{\max}$
1/100	1.	1.	2.78	0.2	0.8	1.	7.	12.
q_1	q_2	q_3	r_{peak}/a	κ_{ns_0}	Δ_{ns_0}	κ_{Ts_0}	Δ_{Ts_0}	T_i/T_e
1.9	0.	0.	0.5	$1.e^{-7}$	0.2	$1.e^{-7}$	0.2	1.

Table 4.6: Common parameters for Rosenbluth-Hinton test. The velocity phase space is defined by $-\mathrm{nb}_{vth0}v_{T_{s0}} \leq v_{G\|} \leq \mathrm{nb}_{vth0}v_{T_{s0}}$ and $0 \leq \mu \leq \mu_{\max}T_0/B_0$. Torus indicates the fraction of the torus simulated. The safety factor radial profile is defined as $q(r) = q_1 + q_2 \exp(q_3 \log(r/a))$. The radial density profile is defined by its gradient as $\mathrm{d}\log n_{s_0}(r)/\mathrm{d}r = -\kappa_{ns_0} \cosh^{-2}((r - r_{\mathrm{peak}}/a)/\Delta r_{ns_0})$. The same analytical expression is used for the temperature with κ_{Ts_0} and Δr_{Ts_0}.

Comparison with theoretical values for $\rho_* = 1/160$

The parameters used for the following simulations are the same as those used by Biancalani [BBLZ14], namely: (i) an analytical circular equilibrium with large aspect ratio ($\epsilon = a/R = 0.1$); (ii) flat density and temperature profiles with $\tau = T_i/T_e = 1$ and (iii) flat q-profiles varying from $q = 1.5$ to 3.5. All simulations are performed for a mesh $(N_r, N_\theta, N_\varphi, N_{v_\|}, N_\mu) = (256, 256, 16, 128, 32)$. The results obtained with GYSELA are displayed on figs.4.3,4.4, where the GAM frequency, the damping rate and the residual value are plotted as a function of the safety factor q. They are comparable to those published by the ORB5 team (see figures 2 and 3 in [BBLZ14]). Consistently with the theory, the FOW (Finite Orbit Width) effects are already significant at moderate values of q (typically for $q \leq 2$, cf. fig.4.3a). Regarding the residual values, two ratio are considered: either the ratio of the finite over the initial zonal electric potential, or the ratio of the radial electric fields. Both ratios would be equal if the radial profiles of the electric potential would not evolve in time, as assumed by the theory. As

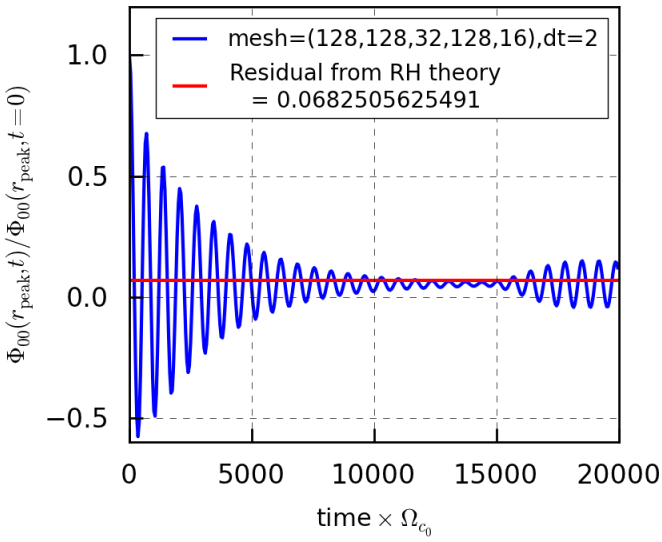


Figure 4.2: Comparison of the residual value of the $(0,0)$ mode $\phi_{00}(r_{\rm peak},t)/\phi_{00}(r_{\rm peak},t=0) = A_r$ where $A_r = 1/(1+1.6q^2/\sqrt{r/R}) = 0.06825$ is given by Rosenbluth-Hinton theory. This result corresponds to simulation 1 described in Table 4.7.

simu.	N_r	N_θ	N_φ	$N_{v_{G\|}}$	N_μ	$\Omega_{C_0}\Delta t$	$\gamma_{\rm GAM}$	$\omega_{\rm GAM}$	CPU time (monoproc)
1	128	128	32	128	16	2.	$2.46459\,10^{-4}$	$9.11061\,10^{-3}$	7219. h.
2	128	128	32	128	16	5.	$2.46459\,10^{-4}$	$9.11061\,10^{-3}$	2921. h.
3	64	64	16	128	8	5.	$2.46459\,10^{-4}$	$9.11061\,10^{-3}$	159. h.
4	64	64	16	128	4	5.	$2.63456\,10^{-4}$	$9.73893\,10^{-3}$	79. h.
5	64	64	16	128	8	10.	$2.46459\,10^{-4}$	$9.11061\,10^{-3}$	41. h.
6	64	64	16	64	8	10.	$2.54957\,10^{-4}$	$9.42477\,10^{-3}$	83. h.
7	64	64	16	128	8	25.	$2.46459\,10^{-4}$	$9.11061\,10^{-3}$	37. h.
8	64	64	16	128	8	50.	$2.54957\,10^{-4}$	$9.42477\,10^{-3}$	21. h.

Table 4.7: Rosenbluth-Hinton test: Values of numerical damping $\gamma_{\rm GAM}$ and frequency $\omega_{\rm GAM}$ for 8 GYSELA simulations varying according to 5D mesh size $(N_r, N_\theta, N_\varphi, N_{v_{G\|}}, N_\mu)$ and time discretization.

evident on fig.4.4, this is actually not the case in these global simulations.

ρ_*	A_s	Z_s	ϵ_0	$r_{\rm int}/a$	$r_{\rm ext}/a$	Torus	nb_{vth0}
1/160	1.	1.	0.1	0.01	1.	1.	7.
$\mu_{\rm max}$	$r_{\rm peak}/a$	κ_{ns_0}	Δ_{ns_0}	κ_{Ts_0}	Δ_{Ts_0}	T_i/T_e	
12.	0.5	$1.e^{-7}$	0.2	$1.e^{-7}$	0.1	1.	

Table 4.8: Common parameters for GAM test. The velocity phase space is defined by $-\mathrm{nb}_{vth0}v_{T_{s0}} \leq v_{G\|} \leq \mathrm{nb}_{vth0}v_{T_{s0}}$ and $0 \leq \mu \leq \mu_{\rm max}T_0/B_0$. Torus indicates the fraction of the torus simulated. The radial density profile is defined by its gradient as $\mathrm{d}\log n_{s_0}(r)/\mathrm{d}r = -\kappa_{ns_0}\cosh^{-2}((r-r_{\rm peak}/a)/\Delta r_{ns_0})$. The same analytical expression is used for the temperature with κ_{Ts_0} and Δr_{Ts_0}.

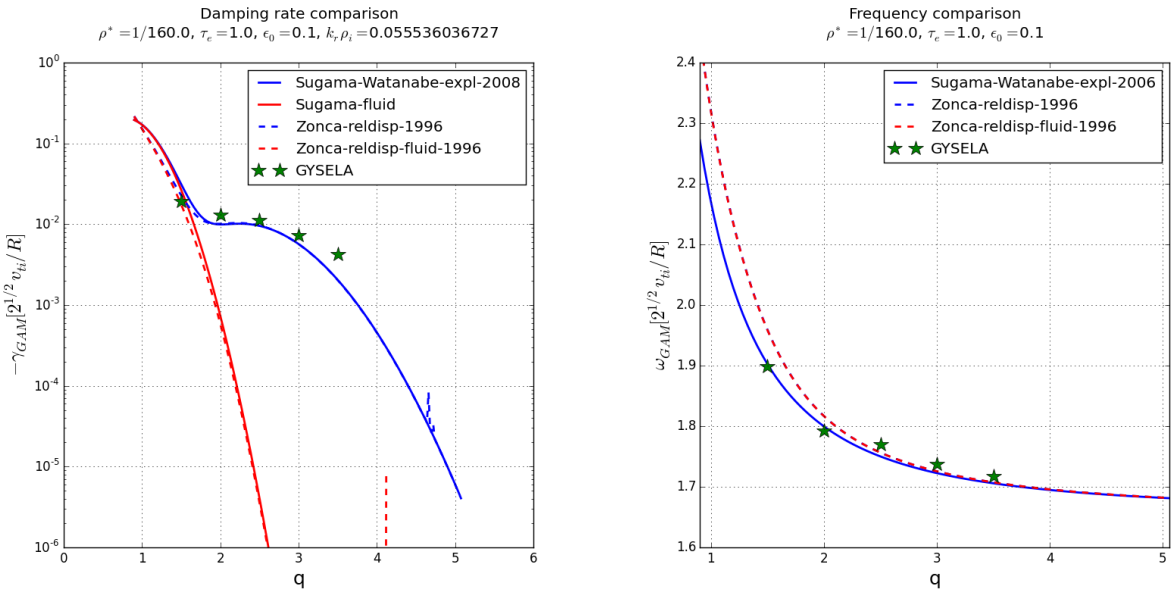


Figure 4.3: Damping rate (left) and frequency (right) of the electric field for simulation parameters given in table 4.8. Comparison with explicit analytical values given by Sugama and Watanabe [SW06] and Zonca [ZCS96] where FOW effects are taken into account or not.

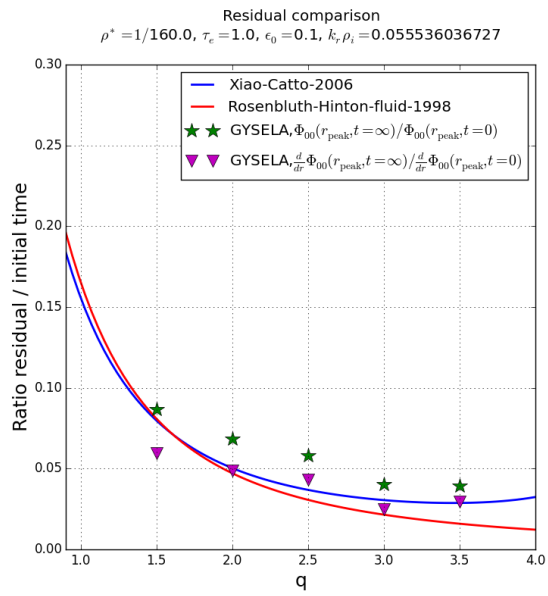


Figure 4.4: Residual values divided by the initial value of the electric potential (green stars) or electric field (magenta triangles) for the 5 different values of constant q profile :1.5, 2., 2.5, 3. and 3.5. Comparison with analytical theory given by Rosenbluth-Hinton [RH98] (red line) and Xiao and Catto [XC06] (blue line).

4.3.2 Linear benchmark based on Cyclone DIII-D case for global codes

For the present benchmark, the considered physical parameters are the same as the ones defined in Lapillone's paper [LMG+10] corresponding to the standard linear Cyclone base case (CBC) [DBB+00]. The circular concentric magnetic equilibrium is defined with an aspect ratio of $R/a = 2.78$ and a safety factor profile $q(r) = 0.86 - 0.16r/a + 2.52(r/a)^2$. This corresponds to a local safety factor $q(r_{\rm peak}) = 1.4$ and a local magnetic shear $s(r_{\rm peak}) = 0.8$ with $r_{\rm peak} = r_{\rm min} + 0.5(r_{\rm max} - r_{\rm min})$ and $s(r) = (r/q)\,{\rm d}q/{\rm d}r$. The initial density and temperature profiles are defined with the radial form

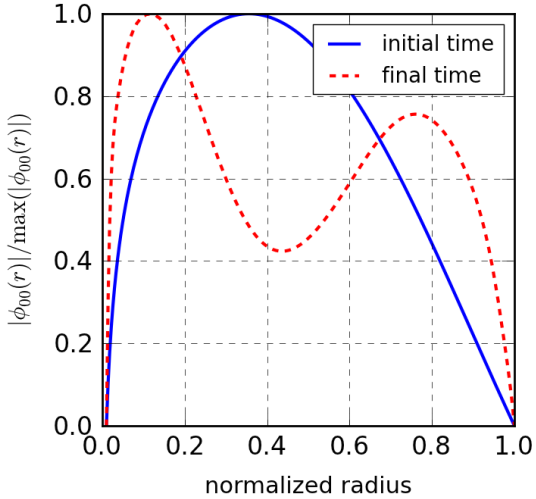


Figure 4.5: Radial profile of the $(0,0)$ mode of ϕ at the initial time (blue line) and at the time $t = 30000/\Omega_{C_0}$ (red dotted line).

function $f(r) = \exp(-\kappa_x \Delta_{r_x} \tanh((r - r_{\text{peak}})/\Delta_{r_x})$. The κ_x and Δ_{r_x} parameters are chosen to obtain peaked profiles at the middle of the radial box r_{peak} with $\Delta_{ns_0} = \Delta_{Ts_0} = 0.3$, $\kappa_{ns_0} = 2.23$ and $\kappa_{Ts_0} = 6.96$ (see Figure 1 in [LMG+10]). To be consistent with the DIII-D shot ρ_* is chosen equal to 1/180.For a complete description of the case see the associated input data file of the code (Figure N.3) in Appendix N.4.

ρ_*	R_0/a	r_{int}/a	r_{ext}/a	nb_{vth0}	$\mu_{\max}$	Z_s	A_s
1/180	2.78	0.1	0.9	6.	7.	1.	1.
q_1	q_2	q_3	κ_{ns_0}	Δ_{ns_0}	κ_{Ts_0}	Δ_{Ts_0}	T_i/T_e
0.86	−0.16	2.52	2.23	0.3	6.96	0.3	1.

Table 4.9: Main parameters of the $\rho_* = 1/180$ linear CBC simulation. The velocity phase space is defined by $-\text{nb}_{vth0} v_{T_{s0}} \leq v_{G\|} \leq \text{nb}_{vth0} v_{T_{s0}}$ and $0 \leq \mu \leq \mu_{\max} T_0/B_0$. The safety factor radial profile is defined as $q(r) = q_1 + q_2(r/a) + q_3(r/a)^2$. The radial density profile is defined by its gradient as $\mathrm{d}\log n_{s_0}(r)/\mathrm{d}r = -\kappa_{ns_0} \cosh^{-2}((r - 0.5)/\Delta r_{ns_0})$. The same analytical expression is used for the temperature with κ_{Ts_0} and Δr_{Ts_0}.

The first difficulty with a full-f code is that the separation between linear and non-linear terms is not possible. So non-linear mode coupling cannot be avoided, which implies that the time window for linear growth rate estimation is limited. To prevent this coupling all the toroidal mode numbers n are filtered except the one n initialized as perturbation in the initial distribution function $\bar{F}_s(t = 0) = \bar{F}_{s,\text{eq}}(1 + \epsilon \sum_{m=1}^{m_{\max}} \cos(m\theta + n\varphi + \delta_{mn}))$ with a perturbation amplitude $\epsilon = 10^{-6}$. This filtering is performed by applying, after the quasi-neutrality equation solving, the condition $\mathcal{F}(\phi)_{mn} = 0$ for all $1 \leq m \leq m_{\max}$ where $\mathcal{F}$ denotes the 2D Fourier transform in (θ, φ). Besides, the global aspect of the code implies that large toroidal mode numbers n are hardly accessible because a large mesh discretization is then necessary. For the 6 cases which were run for $|n|$ ranging from 5 to 30 (see Table 4.10) the discretization of the velocity space was kept constant $(N_{v_{G\|}}, N_\mu) = (64, 16)$ but increased in real space for $|n| > 15$. The number of radial points is kept constant $(N_r = 128)$ but doubled in both poloidal and toroidal directions for $|n| \geq 25$. This corresponds to a mesh of 8 billions of points for the smallest poloidal mode numbers but of 32 billions of points for the largest. The time step Δt is equal to $40./\Omega_{C_0}$ for $|n| \geq 15$ and divided by a factor 4. The linear growth rate is estimated by a linear fit

of the exponential growth of $\int \phi^2 \, \mathrm{d}^3\mathbf{x} \, \mathrm{d}^3\mathbf{v}$ during the linear phase. An example of this exponential growth is plotted at the top of Figure 4.7 for the smallest $k_\theta \rho_s = |n| q(r_{\text{peak}}) \rho_* / r_{\text{peak}} = 0.078$. So the linear fit is performed on a time interval $\Omega_{C_0}[t_{\text{init}}, t_{\text{end}}]$ depending on the duration of this linear phase. This interval is taken equal to $[5000, 50000]$ for $k_\theta \rho_s = 0.078$ and $[4000, 18000]$ for the others. The frequency is estimated on the same time intervals by spectra analysis. Both have been compared with GENE results [LMG+10] and show a very good agreement (see Figure 4.6). This work is in progress in the framework of the Eurofusion project [GTH+16]. The smallest runs were performed on Poincare machine on 512 cores (with $(p_r, p_\theta, p_\mu, N_{\text{thread}}) = (2, 4, 16, 4)$) with a CPU time cost of 9k hours/monoprocessor (for 1300 iterations). The two largest cases ran on Turing (IDRIS-France) BlueGene machine using 32768 processors ($(p_r, p_\theta, p_\mu, N_{\text{thread}}) = (8, 4, 16, 64)$) with an expensive CPU time cost of 1.7 millions of monoprocessor hours for 2000 iterations (equivalent to $1.3\ 10^6$ h. on an INTEL machine as Curie (CCRT-France)). Such simulations are definitively more expensive for a global full-f code than for δf flux-tube codes. This explains why the two last points performed by GENE for $k_\theta \rho_* > 0.5$ (Fig. 4.6) have not been simulated with GYSELA. Nevertheless, the global aspect can give access to useful information like the time evolution of the radial structure of poloidal modes (for toroidal mode $|n|$=5). This radial structure is plotted at 4 different times on Figure 4.7, going from initialization to the end of the linear phase. It shows that the linear phase, *i.e* the exponential growth, starts as soon as the global eigenmode –characterized by a single n and several m mode numbers– acquires its radial structure. The process of mode reorganization and the main physical parameters involved (as q profile, $1/L_T$ shape or ...) would be interesting to investigate in more detail.

n	$m_{\max}$	N_θ	N_φ	$k_\theta \rho_s$	$\Omega_{C_0} \Delta t$	$\Omega_{C_0}[t_{\text{init}}, t_{\text{end}}]$	γ	err_γ	ω
-5	7	256	128	0.078	40.	$[5000, 50000]$	0.045	0.00258	0.106
-10	14	256	128	0.156	40.	$[4000, 18000]$	0.158	0.0069	0.238
-15	21	256	128	0.233	40.	$[4000, 18000]$	0.217	0.00123	0.476
-20	25	256	128	0.311	10.	$[4000, 18000]$	0.257	0.00127	0.636
-25	35	512	256	0.389	10.	$[4000, 18000]$	0.271	0.00468	0.827
-30	42	512	256	0.467	10.	$[4000, 18000]$	0.253	0.002	0.978

Table 4.10: CBC results: Linear growth rate γ and frequencies ω estimated in the time interval $\Omega_{C_0}[t_{\text{init}}, t_{\text{end}}]$ for 6 different toroidal mode numbers n. The poloidal wave number k_θ is computed as $k_\theta \rho_s = |n| q(r_{\text{peak}}) \rho_* / r_{\text{peak}} = |n| \times 1.4/(2 \times 180)$, r_{peak} being the middle of the radial box. These results are compared to GENE results in Figure 4.6

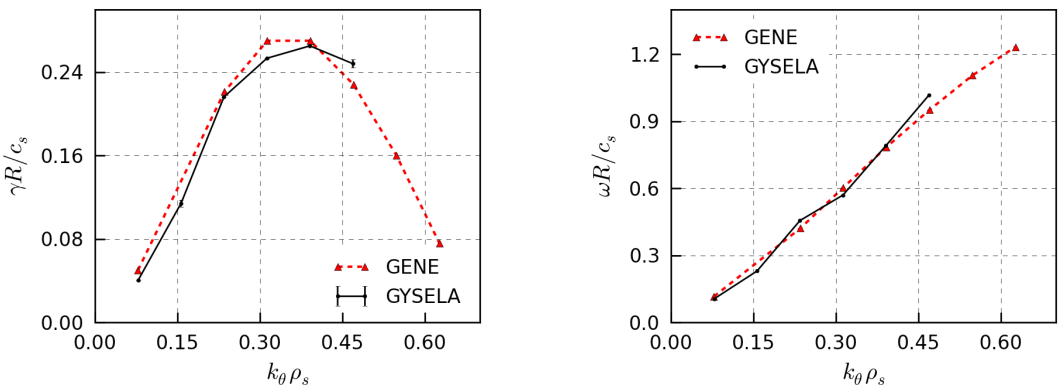


Figure 4.6: Benchmark between GYSELA and GENE codes for the Cyclone base Case for 6 different $k_\theta \rho_s$ values: (Left) Linear growth rate (plotted with the error bar defined in Table 4.10), (Right) Linear frequency.

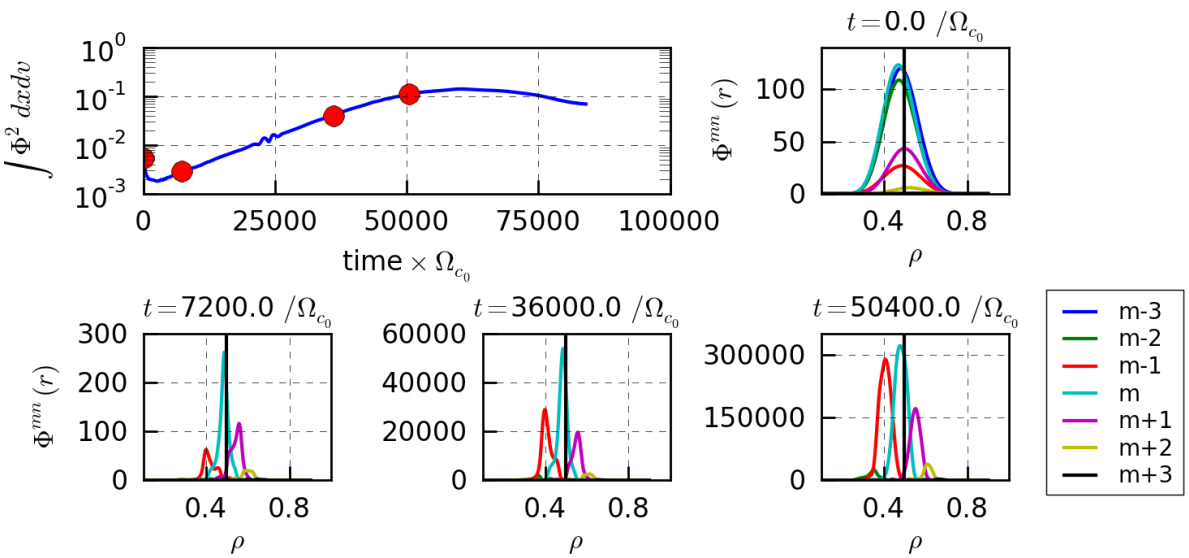


Figure 4.7: Radial structure of the electrostatic potential for $(m, n) = (7, 5)$ and a bandwidth of $m \pm 3$ modes, at different times: (i) initial time, (ii) beginning of the linear phase $t = 7200\Omega_{C_0}^{-1}$, (iii) $t = 36000\Omega_{C_0}^{-1}$ and (iv) end of the linear phase $t = 50400\Omega_{C_0}^{-1}$.

4.4 Local conservation laws for gyrokinetics

Verifying adequate conservation laws is an essential step in providing a correct description of mean flows. Since the controversy on the suitability of gyrokinetic codes for describing toroidal momentum transport [PC08, PC10] a specific effort has been devoted to address conservation equations both in the context of a reduced gyrofluid model [Bri10] or gyrokinetic field theory [SS10, BT11]. As shown in this section, it is also possible [AGG+11] to derive local conservation equations for density, energy and toroidal momentum from the gyrokinetic electrostatic model (2.7)-(2.13) implemented in the GYSELA code (more detailed calculations can be found in Abiteboul's PhD [Abi12]). The radial force balance is presented in section 4.4.1 while the conservation laws are detailed for: (i) charge density in section 4.5, (ii) energy in section 4.5.1 and toroidal momentum in section 4.5.2. As explained in [AGG+11], the force balance equation added to the energy and toroidal momentum conservation equations ensures a self-consistent treatment of the radial electric field and flows. We present for each property an example of numerical results, (see Figures 4.8 to 4.13) for a simulation close to the non-linear CBC simulation proposed in [LMG+10] for $\rho_* = 1/180$. It corresponds to a collisionless simulation, with Dirichlet boundary conditions and without diffusion or Krook operators. Contrary to the linear case, there is no filtering on the toroidal modes. The initial perturbation is defined as $p(r, \theta, v_{G\|}) = \epsilon \sum_{m=1}^{m_{\max}} \sum_{n=1}^{n_{\max}} \cos(m\theta + n\varphi + \delta_{mn}) f_{\text{eq}}(r, \theta, v_{G\|})/(m_{\max} n_{\max})$ for $m_{\max} = 28$ and $n_{\max} = 20$ and random values for δ_{mn} phases. The perturbation amplitude ϵ is equal to 10^{-6}. The equilibrium distribution function f_{eq} is a local Maxwellian function given by eq.(2.20). The other numerical parameters of this simulation are summarized in Table 4.11. The conservation equations are derived at second order in ρ_*. Accurate results (error of 2%) were already shown with the code for force balance and toroidal momentum for a simulation with $\rho_* = 1/512$ (see Figure 2 in [AGG+11]). For a larger $\rho_* = 1/180$ the results are still accurate with a relative error of 5 to 10%. As expected, boundary conditions play a more important role leading to a degradation of the accuracy outside an internal region $0.4 \leq r/a \leq 0.7$. Comparing to relative errors for all conservation properties (Figures 4.8 to 4.13), the local energy conservation (Fig. 4.11) is the least accurate. It was not possible to analyze the energy behaviour for smaller ρ_* (as 1/300 and 1/512) because such simulations are extremely expensive (several millions of mono-processor hours) and the diagnostic was not fully implemented in large simulations until present. This will be investigated in more details on the upcoming big simulations. Complete flux driven simulations with source terms and collisions have also been analyzed (but not presented here). Even for large $\rho_* = 1/150$, the force balance equation, local charge density and toroidal momentum are conserved with less than 10% error even at times

N_r	N_θ	N_φ	$N_{v_{G\parallel}}$	N_μ	nb_{vth0}	$\mu_{\max}$	
256	256	128	64	16	7.	12.	
ρ_*	R_0/a	r_{int}/a	r_{ext}/a	Torus	Z_s	A_s	$\Omega_{c_0}\Delta t$
1/180	2.78	0.1	0.9	1/2	1.	1.	10.
q_1	q_2	q_3	κ_{ns_0}	Δ_{ns_0}	κ_{Ts_0}	Δ_{Ts_0}	T_i/T_e
0.86	−0.16	2.52	2.23	0.3	6.96	0.3	1.

Table 4.11: Main parameters of the $\rho_* = 1/180$ simulation used to check conservation law properties. The velocity phase space is defined by $-\mathrm{nb}_{vth0}v_{T_{s0}} \leq v_{G\parallel} \leq \mathrm{nb}_{vth0}v_{T_{s0}}$ and $0 \leq \mu \leq \mu_{\max}T_0/B_0$. Torus indicates the fraction of the torus simulated. The safety factor radial profile is defined as $q(r) = q_1 + q_2(r/a) + q_3(r/a)^2$. The radial density profile is defined by its gradient as $\mathrm{d}\log \mathrm{n}_{\mathrm{s}_0}(\mathrm{r})/\mathrm{d}r = -\kappa_{ns_0}\cosh^{-2}((r-0.5)/\Delta r_{ns_0})$. The same analytical expression is used for the temperature with κ_{Ts_0} and Δr_{Ts_0}.

when turbulence is fully developed. The fact that this requirement is not met for energy is still under investigation.

4.4.1 Radial force balance equation

In the fluid description, the radial electric field and the flows are related via the force balance equation. It was verified that this relation holds also in gyrokinetics [DPGS+08a, GDPN+09]. Indeed the radial force balance can be recovered analytically from the conventional first order gyrokinetic equations (see Appendix E in [AGG+11]), yielding the standard fluid expression

$$\partial_\chi \phi_{eq} + \frac{\partial_\chi P_{eq}}{n_{eq}e} + \frac{B}{I}V_{\parallel eq} = q\frac{B^2R^2}{I^2}\left(\mathbf{V}\cdot\nabla\theta\right) \tag{4.1}$$

where the equilibrium density n_{eq}, pressure P_{eq}, velocity $V_{\parallel eq}$ and potential ϕ_{eq} are functions of $-P_\varphi/e$ (which is approximately equal to χ at first order in ρ_s). The label of flux surfaces χ is chosen as the opposite of the poloidal flux of the magnetic field, *i.e* $\chi \equiv -\psi_{\mathrm{pol}}$ with $\psi_{\mathrm{pol}} = -(2\pi)^{-1}\int_{S_\theta} dS\,\mathbf{B}\cdot\nabla\theta/|\nabla\theta|$. Considering the form of the magnetic field used presently in the code $\mathbf{B} = (B_0R_0/R(r,\theta))\left[\zeta(r)\mathbf{e}_\theta + \mathbf{e}_\varphi\right]$ with $\zeta(r) = r/(qR_0)$, the label χ is determined by $\chi = B_0\int_0^r r'/q(r')\mathrm{d}r'$ which implies $\mathrm{d}\chi/\mathrm{d}r = B_0r/q$.

In order to check that the force balance equation is verified numerically, we compare v_θ^{GYS} the poloidal velocity directly computed within Gysela –from the distribution function as the sum of the poloidal ExB, curvature, grad-B and magnetization flows– with the expected velocity from the force balance equation, namely v_θ^{FB} corresponding to the left-hand side of eq.(4.1) (see Figure 4.8 (left)). This agreement was robustly verified in Gysela simulations for a wide range of parameters for temperature gradient, collisionality and normalized gyroradius ρ_s [DPGS+09, DPDG+11]. A precision of 2% was obtained for a ITER parameter case with $\rho_* = 1/512$ [AGG+11]. In Figure 4.8 (right), we recover this good agreement for the case $\rho_* = 1/180$ described in Table 4.11. Indeed, even for a larger $\rho_* = 1/180$, the relative error –defined here as the difference normalized to the quadratic mean– is smaller than 0.08 at time $t = 20040\Omega_{C_0}^{-1}$ which corresponds to the beginning of the non-linear phase.

4.5 Local charge density conservation

To obtain local conservation equations, we perform integrations of the conservative form of the gyrokinetic equation (2.7) over the velocity space and over the magnetic flux surfaces, i.e. over $\mathrm{d}\tau^* \equiv \mathcal{J}_{\mathrm{x}}\,\mathrm{d}\theta\,\mathrm{d}\varphi\,\mathrm{d}^3\mathbf{v}$ with $\mathrm{d}^3\mathbf{v} = \mathcal{J}_{\mathrm{v}}\,\mathrm{d}v_{G\parallel}\,\mathrm{d}\mu_s$. We recall that $\mathcal{J}_{\mathrm{x}} = 1/(\mathbf{B}\cdot\nabla\theta)$ is the Jacobian in real space and $\mathcal{J}_{\mathrm{v}} = 2\pi B^*_{\parallel s}/m_s$ is the Jacobian in gyro-center velocity-space.

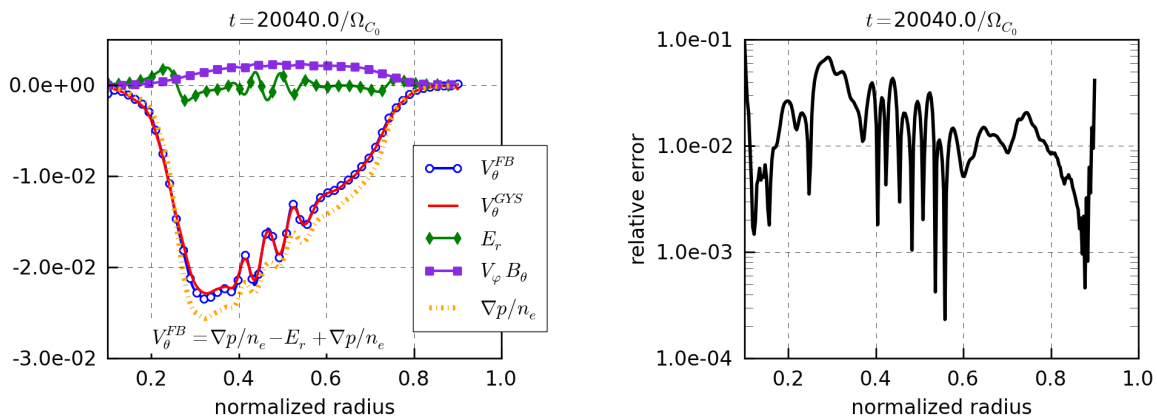


Figure 4.8: Left: Numerical test of the radial force balance equation (4.1) at time $t = 20040\Omega_{C_0}^{-1}$, comparing the poloidal velocity v_θ^{GYS} directly computed in the code and v_θ^{FB} the sum of the three contributions $-E_r$, $\boldsymbol{\nabla} p/n_e$ and $v_\phi B_\theta$. Right: Relative error between both. The parameters used for this simulation are summarized in Table 4.11.

We consider *a radial region outside the buffer region and without source terms*, i.e we consider the following simplified version of the conservative Boltzmann equation (2.7)

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = \frac{\partial \bar{F}_s}{\partial t} + \frac{1}{B_{\|s}^*}\nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B_{\|s}^* \bar{F}_s\right) = \mathcal{C}(\bar{F}_s) \tag{4.2}$$

where $\mathbf{z} = (\chi, \theta, \varphi, v_{G\|}, \mu_s)$ and $\dot{\mathbf{z}} = \frac{\mathrm{d}\mathbf{z}}{\mathrm{d}t}$. We integrate Eq. (4.2) in a small phase-space volume between two surfaces χ and $\chi + \delta\chi$ and apply a divergence theorem[5]. The label of flux surfaces χ is chosen as the opposite of the poloidal flux of the magnetic field, i.e

$$\chi \equiv -\psi_{\mathrm{pol}} \quad \text{with} \quad \psi_{\mathrm{pol}} = -\frac{1}{2\pi}\int_{S_\theta} dS\, \mathbf{B} \cdot \frac{\nabla\theta}{|\nabla\theta|} \tag{4.3}$$

which is equivalent to[6]

$$\chi = B_0 \int_0^r \frac{r'}{q(r')}\,\mathrm{d}r' \quad \text{which implies} \quad \frac{\mathrm{d}\chi}{\mathrm{d}r} = B_0 \frac{r}{q} \tag{4.4}$$

by considering the form of the magnetic field imposed in the code $\mathbf{B} = \frac{B_0 R_0}{R(r,\theta)}\left[\zeta(r)\mathbf{e}_\theta + \mathbf{e}_\varphi\right]$ with $\zeta(r) = \frac{r}{qR_0}$. Using the collision operator conservation property $\int \mathcal{C}(\bar{F}_s)\,\mathrm{d}\tau^* = 0$ and summing over all species, this leads directly to the usual expression for local transport of charge density [7]

$$\partial_t \bar{\rho} + \partial_\chi J^\chi = 0 \tag{4.5}$$

[5] If V is a volume enclosed by a surface S and $\mathrm{d}\mathbf{S} = \mathbf{n}\mathrm{d}S$, where $\mathbf{n}$ is the unit normal outward from V, $\int_V \mathrm{d}V \nabla \cdot \mathbf{A} = \int_S \mathrm{d}\mathbf{S} \cdot \mathbf{A}$ for all vector $\mathbf{A}$.

[6] According to equation (4.3) and the fact that $dS = R\,\mathrm{d}r\,\mathrm{d}\varphi$ and $|\nabla\theta| = 1/r$,

$$\psi = -\int_0^r (\mathbf{B}\cdot\nabla\theta)\, r' R\,\mathrm{d}r' = -\int_0^r (\mathbf{B}\cdot\mathbf{e}_\theta)\, R\,\mathrm{d}r' = -\int_0^r \frac{B_0 R_0}{R}\frac{r'}{q\,R_0} R\,\mathrm{d}r' = -B_0\int_0^r \frac{r'}{q(r')}\,\mathrm{d}r'$$

[7] Integrating (4.2) over $\mathrm{d}\tau^*$, using the property $\int \mathrm{d}\tau^* \mathcal{C}(\bar{F}_s) = 0$ and summing over all species leads to $\sum_s \int \mathrm{d}\tau^* \frac{\partial \bar{F}_s}{\partial t} + \sum_s \int \mathrm{d}\tau^* \frac{1}{B_{\|s}^*}\nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B_{\|s}^* \bar{F}_s\right) = 0$. Integrating in a small phase-space volume between two surfaces χ and $\chi + \delta\chi$ and applying a divergence theorem, the second term of the previous equation reads $\sum_s q_s \int \mathrm{d}\tau^* \frac{1}{B_{\|s}^*}\nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B_{\|s}^* \bar{F}_s\right) = \partial_\chi \int_\chi^{\chi+\delta\chi} \sum_s q_s \int \mathrm{d}\tau^* \frac{1}{B_{\|s}^*}\nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B_{\|s}^* \bar{F}_s\right) = \partial_\chi \sum_s q_s \left[\int \mathrm{d}\tau^* \frac{1}{B_{\|s}^*}\left(\dot{\mathbf{z}} B_{\|s}^* \bar{F}_s\right) \cdot \nabla\chi\right]$.

where $\bar{\rho}$ is the charge density and J^χ is the radial current of gyrocenters:

$$\bar{\rho} = \sum_{species} q_s \int \mathrm{d}\tau^* \bar{F}_s \tag{4.6}$$

$$J^\chi = \sum_{species} q_s \int \mathrm{d}\tau^* (\dot{\mathbf{z}} \cdot \nabla\chi) \bar{F}_s \tag{4.7}$$

In the case of electrostatic simulations with adiabatic electron response, boundary conditions impose a vanishing radial current at the edges of the simulation domain. Therefore the radial current J^χ is expected to be small in such simulations. Figure 4.9 shows that equation (4.5) is numerically satisfied with an error of less than 1%. Part of this error is due to the fact that the time derivative $\partial_t \bar{\rho}$ is computed from post-processed data ($\bar{\rho}$ is not saved at each time step Δ_t but at each diagnostic time step ($\Delta_{t\,\mathrm{diag}} = 12\Delta_t$ for this simulation)). Another error source is the fact that the terms $\nabla_\perp \cdot (B^*_{\|s} \mathrm{d}_t \chi_G)$ and $\partial_\varphi (B^*_{\|s} \mathrm{d}_t \varphi)$ are neglected in the splitting algorithm (see section 3.1). For information the contribution of the two terms (4.6) and (4.7) are plotted in figure 4.10.

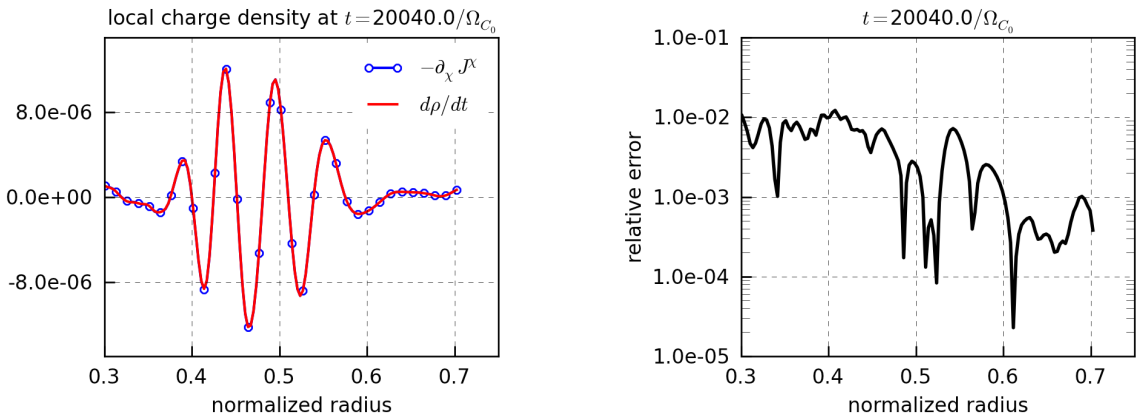


Figure 4.9: Numerical test of the charge density conservation for simulation defined in Table 4.11. All the quantities are flux-surface averaged. Left: Comparison of the two terms $-\partial_\chi J^\chi$ and $\mathrm{d}\rho/\mathrm{d}t$ which must be equivalent according to equation eq.(4.5). Right: Relative error (defined as the difference normalized to the quadratic mean).

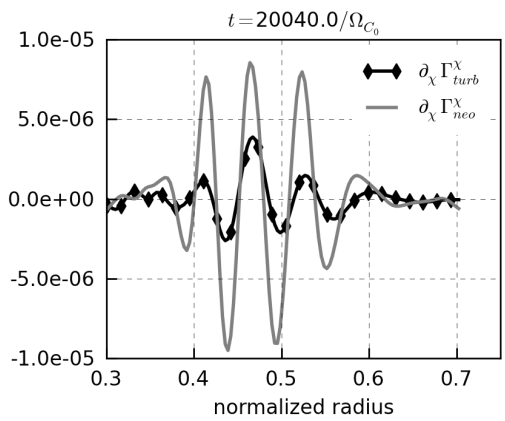


Figure 4.10: Contribution of the neoclassical and turbulent parts in the charge density conservation equation for simulation defined in Table 4.11. All the quantities are flux-surface averaged.

4.5.1 Local energy conservation

Similarly, a conservation equation for the total energy can be derived by multiplying the gyrokinetic equation (4.2) by the gyrocenter Hamiltonian, which reads

$$\bar{H}_s = \frac{1}{2} m_s v_{G\parallel}^2 + \mu_s B + q_s \bar{U} \quad \text{with} \quad \bar{U} = J_0 \cdot U \tag{4.8}$$

Using the fact that our *collision operator* is constructed to *conserve the total Hamiltonian* (see appendix F.1 for more details) and integrating over $\mathrm{d}\tau^*$ yields

$$\partial_t \int \mathrm{d}\tau^* \left(\frac{1}{2} m_s v_{G\parallel}^2 + \mu_s B \right) \bar{F}_s + \int \mathrm{d}\tau^* q_s \bar{U} \partial_t \bar{F}_s + \int \mathrm{d}\tau^* \bar{H}_s \frac{1}{B_{\parallel s}^*} \nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B_{\parallel s}^* \bar{F}_s \right) = 0 \tag{4.9}$$

The first term in Eq. (4.9) is equal to $\partial_t E_{K_s}$ where E_{K_s} corresponds to the kinetic energy of the gyrocenters for the species considered and is defined as

$$E_{K_s} \equiv \int \mathrm{d}\tau^* \left(\frac{1}{2} m_s v_{G\parallel}^2 + \mu_s B \right) \bar{F}_s \tag{4.10}$$

To compute the third term, we integrate again over a phase-space volume between χ and $\chi + \delta\chi$ and we use again the divergence theorem. Because $\dot{\mathbf{z}} \cdot \nabla_{\mathbf{z}} \bar{H}_s = 0$, this leads to [8]

$$\int \mathrm{d}\tau^* \bar{H}_s \frac{1}{B_{\parallel s}^*} \nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B_{\parallel s}^* \bar{F}_s \right) = \partial_\chi \int \mathrm{d}\tau^* \bar{H}_s \left(\dot{\mathbf{z}} \cdot \nabla \chi \right) \bar{F}_s \tag{4.11}$$

This term yields the radial flux of energy Q_s defined as

$$Q_s = \int \mathrm{d}\tau^* \bar{H}_s \left(\dot{\mathbf{z}} \cdot \nabla \chi \right) \bar{F}_s \tag{4.12}$$

Then, for each species s, equation (4.9) can be rewritten as an equation for the radial energy transport as

$$\partial_t E_{K_s} + \partial_\chi Q_s = W_s \tag{4.13}$$

where the right-hand side appears as a kinetic energy source $W_s = -q_s \int \mathrm{d}\tau^* \bar{U} \partial_t \bar{F}_s$. This term corresponds to an exchange of energy between a given species and the turbulence, and is generally referred to as turbulent heating [MOT77, WSD+97, HW06, WS08, GES+13]. It corresponds to a transfer of energy between particles and the electromagnetic field. The numerical computation of this term is expensive because it requires saving the 5D distribution function of each species at two successive time steps which corresponds to a large amount of memory. So to obtain a local conservation equation with no source term, we consider the total energy by summing (4.13) over all species. Then,

$$\partial_t E_K + \partial_\chi Q = W \quad \text{with} \quad E_K = \sum_s E_{K_s} \quad , \quad Q = \sum_s Q_s \quad \text{and} \quad W = \sum_s W_s \tag{4.14}$$

The term W is decomposed into two parts as

$$\begin{aligned} \sum_s q_s \int \mathrm{d}\tau^* \bar{U} \partial_t \bar{F}_s = & \sum_s q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \partial_t \bar{F}_s \right) \\ & + \sum_s q_s \int \mathrm{d}\tau^* \left\{ \left(J_0 \cdot U \right) \partial_t \bar{F}_s - U \left(J_0 \cdot \partial_t \bar{F}_s \right) \right\} \end{aligned} \tag{4.15}$$

[8] Due to the fact that $\dot{\mathbf{z}} \cdot \nabla_{\mathbf{z}} \bar{H}_s = 0$, $\int \mathrm{d}\tau^* \bar{H}_s \frac{1}{B_{\parallel s}^*} \nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B_{\parallel s}^* \bar{F}_s \right) = \int \mathrm{d}\tau^* \frac{1}{B_{\parallel s}^*} \nabla_{\mathbf{z}} \cdot \left(\bar{H}_s \dot{\mathbf{z}} B_{\parallel s}^* \bar{F}_s \right) = \partial_\chi \left[\int_\chi^{\chi+\delta\chi} \int \mathrm{d}\tau^* \frac{1}{B_{\parallel s}^*} \nabla_{\mathbf{z}} \cdot \left(\bar{H}_s \dot{\mathbf{z}} B_{\parallel s}^* \bar{F}_s \right) \right] = \partial_\chi \left[\int \mathrm{d}\tau^* \left(\bar{H}_s \dot{\mathbf{z}} \bar{F}_s \right) \cdot \nabla \chi \right]$.

As detailed in Appendix M.2 the first term in Eq.(M.5) can be expressed in function of the potential energy E_p as

$$\sum_{species} q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \partial_t \bar{F}_s\right) \equiv \partial_t E_p \quad \text{with} \quad E_p \equiv \frac{1}{2} \sum_{species} q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \bar{F}_s\right)$$

Besides, the second term $\sum_s q_s \int \mathrm{d}\tau^* \left\{(J_0 \cdot U)\, \partial_t \bar{F}_s - U \left(J_0 \cdot \partial_t \bar{F}_s\right)\right\}$ in Eq.(M.5) corresponds to a polarization term, due to the difference between particles and gyro-center densities. As a remark, considering that the gyro-average operator J_0 is a self-adjoint operator, this term vanishes when integrating over the whole phase-space volume. It is indeed the divergence of a flux in the local conservation equation. To express this term explicitly as a flux contribution, let us consider the low wavenumber approximation of the gyro-average operator used in the code, *i.e* the Padé approximation $J_0 \simeq 1 + \frac{1}{2}\nabla \cdot \left(\frac{m_s \mu_s}{q_s^2 B} \nabla_\perp\right)$ (see section 2.4.1 for more details). Using this approximation, $\mathcal{I}_E$ can be expressed for each species as

$$\mathcal{I}_E = \frac{m_s}{2q_s} \partial_\chi \left[\int \mathrm{d}\tau^*\, \partial_t \bar{F}_s\, \frac{\mu}{B} \nabla\chi \cdot \nabla_\perp U - \int \mathrm{d}\tau^*\, U \frac{\mu}{B} \nabla\chi \cdot \nabla_\perp (\partial_t \bar{F}_s)\right]$$

It can also be expressed in a more compact form using the gyrocenter perpendicular stress $P_{s,\perp} = \int \mathrm{d}^3\mathbf{v} \bar{F}_s \mu_s B$. This leads to the following conservation equation summed over all species

$$\partial_t \left(E_K + E_p\right) + \partial_\chi \left(Q + Q_{\mathrm{pot}} + Q_{\mathrm{pol}}\right) = 0 \tag{4.16}$$

where $E_K = \sum_s E_{K_s}$, $E_p = \sum_s E_{p_s}$, $Q = \sum_s Q_s$, $Q_{\mathrm{pot}} = \sum_s Q_{s,\mathrm{pot}}$ and $Q_{\mathrm{pol}} = \sum_s Q_{s,\mathrm{pol}}$ with

$$E_{K_s} = \int \mathrm{d}\tau^* \mathcal{E}_s \bar{F}_s \quad \text{with} \quad \mathcal{E}_s = \frac{1}{2} m_s v_{G\|}^2 + \mu_s B \tag{4.17}$$

$$E_{p_s} = \frac{q_s}{2} \int \mathrm{d}\tau^* U (J_0 \cdot \bar{F}_s) \tag{4.18}$$

$$Q_s = \int \mathrm{d}\tau^* \mathcal{E}_s (\dot{\mathbf{z}} \cdot \nabla\chi) \bar{F}_s \tag{4.19}$$

$$Q_{s,\mathrm{pot}} = \int \mathrm{d}\tau^* (J_0 \cdot U)(\dot{\mathbf{z}} \cdot \nabla\chi) \bar{F}_s \tag{4.20}$$

$$Q_{s,\mathrm{pol}} = \frac{m_s}{2q_s} \int \mathcal{J}_{\mathrm{x}}\, \mathrm{d}\theta\, \mathrm{d}\varphi \frac{1}{B^2} \left\{\partial_t P_{s,\perp} \nabla U \cdot \nabla\chi - U \nabla(\partial_t P_{s,\perp}) \cdot \nabla\chi\right\} \tag{4.21}$$

We recall that, although the polarization term is necessarily the divergence of a flux term, the expression for Q_{pol} given here is not exact as it relies on an approximation of the gyro-average operator. Q_s corresponds to the energy flux of species s while $Q_{s,\mathrm{pot}}$ corresponds to the flux due to the electric potential. For the numerical results presented in Figure 4.11 the term $Q_{s,\mathrm{pol}}$ is not taken into account. Indeed, it requires 3D values of U and $P_{s,\perp}$ which were not saved for this simulation because this possibility has been more recently implemented in the code. Analyzing recent non-linear flux-driven simulations where this computation is available shows that the assumption $Q_{s,\mathrm{pot}} \approx 0$ is justified. Even in regimes where turbulence is well developed the term $Q_{s,\mathrm{pot}}$ stays sufficiently small to have no impact on the local energy conservation law.

The numerical validation of equation (4.16) is performed on flux surface average quantities. The comparison between $\langle\, \partial_\chi \left(Q + Q_{\mathrm{pot}} + Q_{\mathrm{pol}}\right) \rangle_{\mathrm{FS}}$ and $\langle\, \left(E_K + E_p\right) \rangle_{\mathrm{FS}}$ seen in Figure 4.11 shows an agreement better than 15% at time $t = 20040\Omega_{C_0}^{-1}$. The separate contribution of each terms given by equations (4.17)-(4.21) are plotted at Figure 4.12 showing that this energy conservation derives from the compensation of different complex radial profiles.

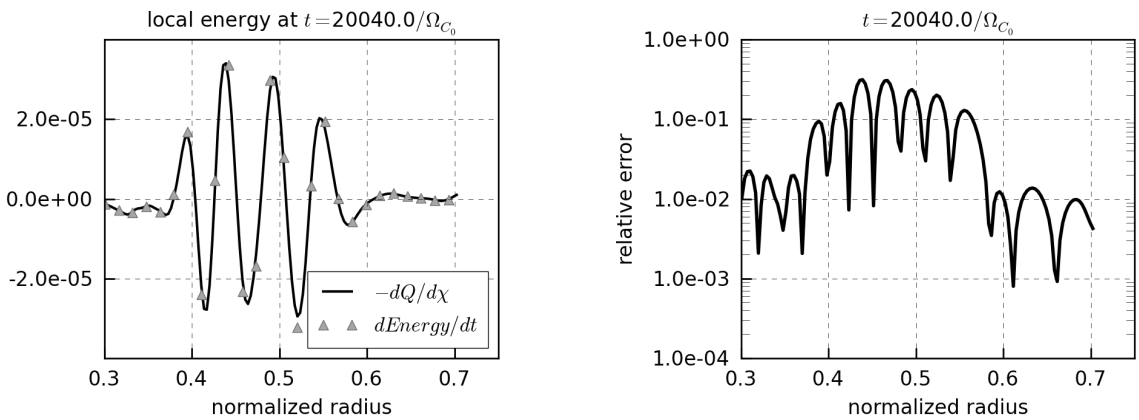


Figure 4.11: Numerical test for local energy conservation for simulation defined in Table 4.11. Left: Comparison between $-\langle\, \partial_\chi Q \,\rangle_{\rm FS}$ and $\mathrm{d}\langle\, E \,\rangle_{\rm FS}/\mathrm{d}t$ with $Q = Q_s + Q_{s,\,\rm pot} + Q_{s,\,\rm pol}$ and $E = E_{K_s} + E_{p_s}$ defined by eqs.(4.16)-(4.21). Right: Relative error (defined here as the difference normalized to the quadratic mean).

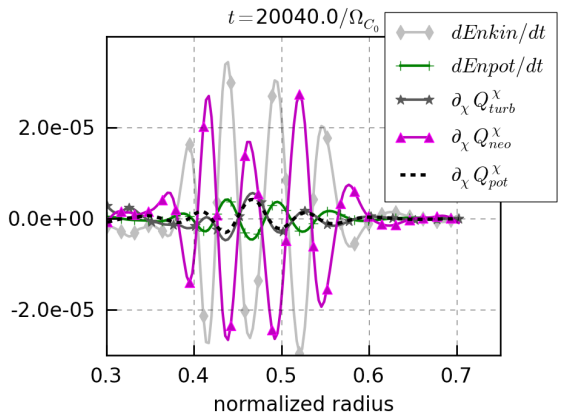


Figure 4.12: Detailed contribution of each terms in the local energy conservation for simulation defined in Table 4.11.

4.5.2 Local toroidal momentum conservation

Formally, the derivation of a conservation law for toroidal angular momentum is very similar to that for energy in the previous section. The general idea is to multiply the conservative gyrokinetic equation (4.2) by an invariant of motion. For the energy, this invariant was the gyro-center Hamiltonian $\bar{H}_s$ given by Eq. (4.8). For this conservation law, let us consider the gyrocenter toroidal canonical momentum P_φ defined as

$$P_\varphi = m_s u_\varphi - q_s \chi \tag{4.22}$$

with the notation $u_\varphi = (I/B) v_{G\|} = R^2 \mathbf{b}\cdot\nabla\varphi v_{G\|}$. P_φ is an exact invariant of the *unperturbed gyrocenter motion* described by the Hamiltonian $\bar{H}_{s,\rm eq} = \frac{1}{2} m_s v_{G\|}^2 + \mu_s B + \bar{U}_{\rm eq}$, which corresponds to collisionless motion in a fully axisymmetric system. $\bar{U}_{\rm eq}$ is the gyro-average of the equilibrium electric potential $U_{\rm eq}$ independent on the toroidal angle. When axisymmetry is broken, which can occur for instance due to turbulence or magnetic field ripple, P_φ is no longer a motion invariant. In particular, when the electric potential becomes non-axisymmetric, the evolution of P_φ is governed by the equation $\mathrm{d}_t P_\varphi = -q_s \partial_\varphi \bar{U}$. This result can be obtained by using the expression of the gyrokinetic Poisson brackets Eq. (1.38). Details of the calculation are presented in appendix M.3. From the definition of P_φ given by Eq. (4.22), we define the local toroidal angular momentum as

$$\mathcal{L}_\varphi = \sum_{species} m_s \int \mathrm{d}\tau^* u_\varphi \bar{F}_s \tag{4.23}$$

Note that $\mathcal{L}_\varphi$ is the momentum for *gyrocenter*, which differs from the particle momentum by terms of order $\mathcal{O}(\rho_s^2)$. As said before, in order to derive a local conservation equation for $\mathcal{L}_\varphi$, we multiply the conservative gyrokinetic equation (4.2) by P_φ and integrate over all variables other than χ, leading to

$$\int \mathrm{d}\tau^* \, P_\varphi \frac{\partial \bar{F}_s}{\partial t} + \int \mathrm{d}\tau^* \, P_\varphi \frac{1}{B^*_{\|s}} \nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B^*_{\|s} \bar{F}_s \right) = \int \mathrm{d}\tau^* \, P_\varphi \, \mathcal{C}(\bar{F}_s)$$

Using the conservation properties of the collision operator $\int \mathrm{d}\tau^* \, P_\varphi \, \mathcal{C}(\bar{F}_s) = 0$ and integrating by parts the second term, then

$$\partial_t \left(\int \mathrm{d}\tau^* \, P_\varphi \bar{F}_s \right) - \int \mathrm{d}\tau^* \, \bar{F}_s \frac{\partial P_\varphi}{\partial t} - \int \mathrm{d}\tau^* \, \bar{F}_s \frac{1}{B^*_{\|s}} \nabla_{\mathbf{z}} \cdot \left(\dot{\mathbf{z}} B^*_{\|s} P_\varphi \right) + \partial_\chi \int \mathrm{d}\tau^* (\dot{\mathbf{z}} \cdot \nabla \chi) \bar{F}_s \, P_\varphi = 0$$

Finally, using the fact that $\mathrm{d}_t P_\varphi = -q_s \partial \varphi \bar{U}$, we obtain for each species s,

$$m_s \partial_t \int \mathrm{d}\tau^* u_\varphi \bar{F}_s - q_s \int \mathrm{d}\tau^* \chi \partial_t \bar{F}_s + q_s \int \mathrm{d}\tau^* \bar{F}_s \partial_\varphi \bar{U} + \partial_\chi \int \mathrm{d}\tau^* (\dot{\mathbf{z}} \cdot \nabla \chi) \bar{F}_s \, P_\varphi = 0 \tag{4.24}$$

Summing over all species, this leads to

$$\partial_t \left(\sum_s m_s \int \mathrm{d}\tau^* u_\varphi \bar{F}_s \right) - \chi \sum_s q_s \int \mathrm{d}\tau^* \partial_t \bar{F}_s + \sum_s q_s \int \mathrm{d}\tau^* \bar{F}_s \partial_\varphi \bar{U} +$$
$$\partial_\chi \left(\sum_s m_s \int \mathrm{d}\tau^* (\dot{\mathbf{z}} \cdot \nabla \chi) \bar{F}_s u_\varphi \right) - \partial_\chi \left(q_s \sum_s \chi \int \mathrm{d}\tau^* (\dot{\mathbf{z}} \cdot \nabla \chi) \bar{F}_s \right) = 0$$

Using the local particle conservation Eq. (4.5), the second term can be written as $\chi \partial_\chi J^\chi$. We also identify the last term as $-\partial_\chi (\chi J^\chi)$. Then the conservation equation of the local toroidal momentum $\mathcal{L}_\varphi = \sum_s m_s \int \mathrm{d}\tau^* u_\varphi \bar{F}_s$ reads

$$\partial_t \mathcal{L}_\varphi + \partial_\chi \Pi^\chi_\varphi + T^\chi_{\mathrm{pol}} = J^\chi \tag{4.25}$$

where

$$\Pi^\chi_\varphi = \sum_{species} m_s \int \mathrm{d}\tau^* \bar{F}_s u_\varphi v^\chi_G \tag{4.26}$$

$$T^\chi_{\mathrm{pol}} = \sum_{species} q_s \int \mathrm{d}\tau^* \bar{F}_s \partial_\varphi \bar{U} \tag{4.27}$$

$$J^\chi = \sum_{species} q_s \int \mathrm{d}\tau^* v^\chi_G \bar{F}_s \tag{4.28}$$

Equation (4.25) is an *exact* equation for the transport of gyrocenter toroidal momentum, in the sense that it was obtained directly from the gyrokinetic model, with no additional assumptions of orderings. The term T^χ_{pol} corresponds to a polarization flux of momentum term. The numerical results obtained with the code are presented in Figure 4.13 (left). The relative error plotted in Figure 4.13 (right) shows an accuracy better than 10%. The contribution of the different terms of equation (4.25) averaged on the flux surface are shown in Figure 4.14.

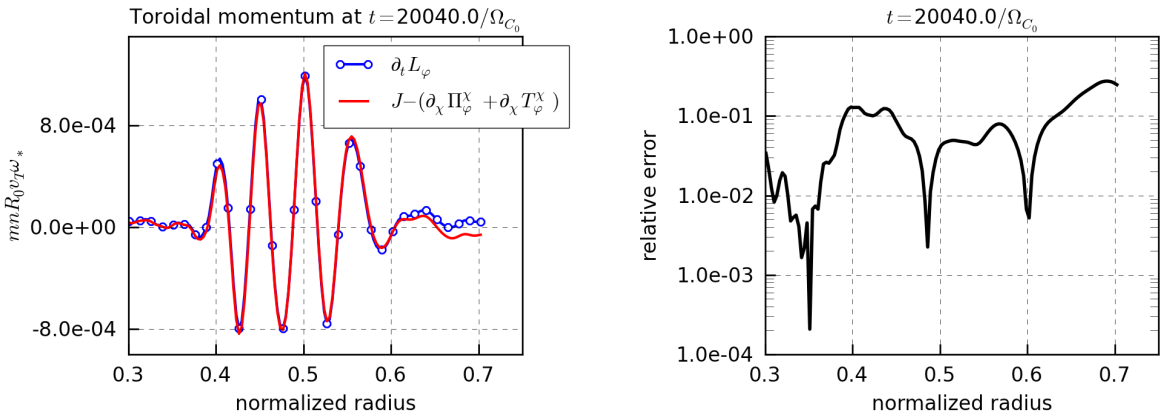


Figure 4.13: Numerical test for local toroidal momentum conservation for simulation detailed in Table 4.11. All the quantities are flux-surface averaged. Left: Comparison between $\partial_t \mathcal{L}_\varphi$ and $J^\chi - \partial_\chi \Pi^\chi_\varphi - T^\chi_{\text{pol}}$ which must be equal according to eq.(4.25). Right: Relative error (defined here as the difference normalized to the quadratic mean).

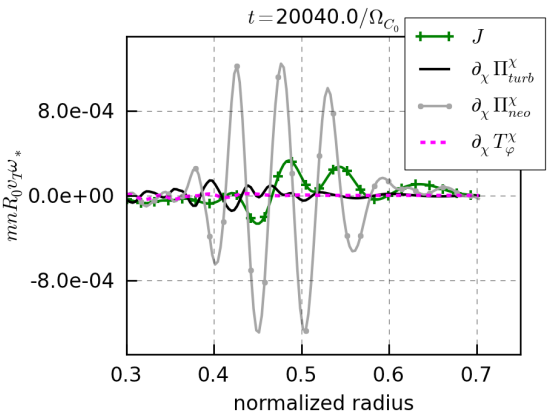


Figure 4.14: Detailed contributions of neoclassical and turbulent parts in local toroidal momentum conservation for simulation detailed in Table 4.11. All the quantities are flux-surface averaged.

Conclusion

All the work presented in this manuscript has been performed in strong collaboration with my CEA colleagues: Xavier Garbet, Philippe Ghendrih, Guilhem Dif-Pradalier, Chantal Passeron, Guillaume Latu and Yanick Sarazin and thanks to all the students which have been associated to the GYSELA project since its beginning. The list below attests that PhD and post-doc works are strongly linked to GYSELA story. It is certainly limiting to associate each student to one specific major step because it has always been a teamwork where each one has been involved in several multidisciplinary tasks. All of them have actively contributed to the fact that GYSELA is now the unique semi-Lagrangian 5D non-linear gyrokinetic code being global, full-f, flux-driven and able to tackle both neoclassical and turbulent ion transport in Tokamak plasma.

- Guilhem Dif-Pradalier (2005-2008) - *First-principle description of collisional gyrokinetic turbulence in tokamak plasmas*, Thèse Université de Provence Aix-Marseille I, oct. 2008.
- Paolo Angelino (Post-Doc: nov 2006-nov 2008) : *Linear and non-linear benchmarks between semi-lagrangian* GYSELA *code and Particle In Cell ORB5 code.*
- Antoine Strugarek (2009-2012) - *Turbulence, transport et confinement: des tokamaks au magnétisme des étoiles*, Thèse Université Paris-Diderot - Paris VII, dec 2012.
- David Zarzoso (2009-2012) - *Kinetic description of the interaction between energetic particles and waves in fusion plasmas*, Thèse Ecole Polytechnique, Palaisau, nov. 2012.
- Jérémie Abiteboul (2009-2012) - *Aspect multi-échelle du transport cinétique des électrons et des ions dans la turbulence plasma d'ITER*, Thèse Université de Provence Aix-Marseille I, novembre 2012.
- Simon Allfrey (Post-Doc: may 2009-may 2011) : *Multi-species version of the* GYSELA *code.*
- Thomas Cartier-Michaud (2011-2014) - *Réduction fluide sous-jacente lors de l'évolution cinétique de la turbulence plasma dans ITER*, Thèse Université de Provence Aix-Marseille , oct. 2015.
- Claudia Norscini (2012-2015) - *Transport turbulent non-local dans les plasmas d'ITER.*
- Fabien Rozar (2012-2015) - *Peta and exascale algorithms for turbulence simulation of fusion plasmas*, Thèse Université de Bordeaux, nov. 2015.
- Damien Estève (2012-2015) - *Interaction gyrocinétique entre deux espèces de particules en turbulence de plasma de fusion*, Thèse Université de Provence Aix-Marseille I, dec. 2015.

This highly parallel code has already shown good efficiency up to hundreds of thousands cores. Non-linear simulations currently run on several thousands of cores during several dozens of days. The capacities of the code to simulate ion plasma turbulence is already large. Let us mention, in this wide spectrum, some crucial topics for ITER tokamak where GYSELA results have already contributed, as:

- Transport barriers,
- Interaction between energetic particles and turbulence, namely the role of energetic geodesic accoustic modes (EGAMs),

- Impurity transport,

- Synergy between neoclassical and turbulent transport.

The two main weakness of the code are that: (i) electrons are assumed adiabatic that does not permit to study particle transport and (ii) the toroidal geometry takes only into account simplified circular magnetic configurations. New developments are under progress to relax these two limitations. Concerning more complex magnetic configurations, the development of a new numerical approach based on an hybrid semi-Lagrangian scheme is still under tests in the SELALIB plateform. The next step, which will be the implementation in GYSELA is foreseen for 2017 via a new post-doc project. The implementation of full kinetic electrons has been recently achieved in the code and is now under verification and validation. First results on collisionless Rosenbluth-Hinton test are promising. First non-linear simulations with both kinetic ions and electrons have been also performed but for a large $\rho_* = 1/100$ on a small corona (from $r_{\min} = 0.4\rho_*$ to $r_{\max} = 0.7\rho_*$) with a non-realistic mass aspect ratio $m_i/m_e = 100$. Even for such constrained parameters, simulations are already very demanding both in terms of CPU time and memory consumption. Knowing that, for equal temperatures, electrons have a much larger parallel speed of a ratio $(m_i/m_e)^{1/2}$ compared to ions, simulations with realistic mass aspect ratio ($m_i/m_e \sim 3600$) will be rapidly extremely prohibitive. Indeed, to accurately capture electron dynamics, the numerical time step has to be reduced by the same factor $(m_i/m_e)^{1/2}$ as compared to simulations with adiabatic electrons while the number of mesh points should be increased by a factor $(\sqrt{m_i/m_e})^3 \sim 60^3$. Several ideas are currently being explored simultaneously to limit these strong constraints and to improve kinetic electron simulations:

(i) The possibility to filter between trapped and passing electrons by considering kinetically (deeply) trapped electrons only, while continuing to assume an adiabatic response for other electrons (passing and barely trapped). This will reduce the parallel velocity of the electrons kinetically treated.

(ii) The using of aligned coordinates to reduce the number of points in toroidal direction. First results on non-linear simulations with adiabatic electrons have already proved a possible gain of a factor 4 which will be even better for kinetic electrons.

(iii) The present simplified version of the multi-species collision operator is only valid for trace impurities. A new collision operator will be necessary to take correctly into account ion-electron collisions. One of the main difficulty is that the simplification based on a linearized operator in parallel velocity direction, which had been designed to optimize the code parallelization, is no more possible. The development of a $(v_\parallel, v_\perp)$ collision operator is under progress and will be first validated for heavy impurities.

(iv) The optimization of the global parallelization of the code is actively pursued to prepare the next future exascale simulations.

All these improvements are again strongly associated to PhD works:

- Laura Mendoza (at IPP Garching/Germany) (2012-2016) - *Realistic magnetic equilibrium and field-aligned coordinates for high performance computing of gyrokinetic turbulence in tokamaks.*

- Charles Ehrlacher (nov 2014-nov 2017) - *Contribution des électrons cinétiques au transport turbulent dans les plasmas de fusion.*

- Peter Donnel (nov 2015-nov 2018) - *Transports néoclassique et turbulent des impuretés dans les plasmas de fusion: compétition et synergie.*

- Nicolas Bouzat (nov 2015-nov 2018) - *Parallélisation de l'opérateur de gyromoyenne, optimisation et parallélisation de la méthode alignée dans GYSELA.*

To conclude, let's say that GYSELA adventure has been the opportunity of plenty of collaborations with national laboratories (IRMA from Strasbourg, Maison de la Simulation from Saclay, LPII from

Nancy, LPIIM and CPT from Marseille, LPP from Ecole Polytechnique, LABRI from Bordeaux, ...) but also with IPP Garching (Germany) and Swiss Plasma Center from Lausanne. GYSELA project has been supported by several national and European grants. It has been the main basis of two consecutive ANR (National Research Agency): ANR EGYPT (2008-2011) followed by ANR GYPSI (2011-2015) and part of ANR Nufuse G8@exascale (2012-2016). The code is involved in two INRIA Project Labs as well: the first one dedicated to fusion (IPL FRATRES) and the second one to exascale (IPL C2S@Exa). It has also been one of the basis for the development of the SELALIB test platform for numerical methods applied to Vlasov equation solving which was supported by an INRIA ADT (Technological Development Action) from 2011 to 2015. At the European level, benchmark activity and improvement of numerical schemes are actually granted by the European Fusion project "Verification and development of new algorithms for gyrokinetic codes" (2013-2017) leaded by E. Sonnendrücker at Garching. Finally, research and development to prepare the code to exascale architectures are supported by the EoCoE (Energy Oriented Center of Excellence) H2020 project leaded by Maison de la simulation at Paris (2015-2018).

Quand on voyage vers un objectif, il est très important de prêter attention au chemin. C'est toujours le chemin qui nous enseigne la meilleure façon d'y parvenir, et il nous enrichit à mesure que nous le parcourons. - Paul Coelho

Appendix A

Relationship between $F_s(\mathbf{x}, \mathbf{v}, t)$ and $\bar{F}_s(\mathbf{x}_G, \mathbf{v}_G, t)$

The old canonical coordinates $(\mathbf{x}, \mathbf{p})$ – that of the particle – and the new ones $(\mathbf{x}', \mathbf{p}')$ – those of the guiding-center, only differ by infinitesimal quantities of order $\epsilon \sim \rho_*$. In such a case, they can be shown to be related, at first order in the small parameter ϵ, by the generating function $S(\mathbf{x}, \mathbf{p})$[1] (which remains to be determined) as follows [GPS02] :

$$\begin{aligned} \mathbf{x} - \mathbf{x}' &= \partial_{\mathbf{p}'} S \\ \mathbf{p} - \mathbf{p}' &= -\partial_{\mathbf{x}'} S \end{aligned}$$

Any field f evaluated at the position $(\mathbf{x}, \mathbf{p})$ can then be expressed in terms of its value at $(\mathbf{x}', \mathbf{p}')$, namely:

$$f(\mathbf{x}, \mathbf{p}) = f(\mathbf{x}', \mathbf{p}') + [f, S]_{\mathbf{x}', \mathbf{p}'} + o(\epsilon) \tag{A.1}$$

This is especially the case for the Hamiltonian H:

$$H(\mathbf{x}, \mathbf{p}) = H(\mathbf{x}', \mathbf{p}') + [H, S]_{\mathbf{x}', \mathbf{p}'} + o(\epsilon) \tag{A.2}$$

Since the generating function does not explicitly depend on time, $H(\mathbf{x}', \mathbf{p}')$ also stands for the new Hamilonian with respect to which the new coordinates are canonically conjugated. To avoid any confusion with H, we will denote it $\bar{H}(\mathbf{x}', \mathbf{p}')$ hereafter. As far as the gyrokinetic theory is concerned, the transformation acts on the first pair of conjugate coordinates, namely (φ_c, J_1), with $J_1 = -m_s \mu_s / q_s$. From eq. (A.2), it comes:

$$\begin{aligned} H(\mathbf{x}, \mathbf{p}) &= \bar{H}(\mathbf{x}_G, \mathbf{p}_G) + [\bar{H}, S]_{\varphi_c, J_1} \\ &= \bar{H}(\mathbf{x}_G, \mathbf{p}_G) + \Omega_s \, \partial_{\varphi_c} S + \partial_{\varphi_c} \bar{H} \, \partial_\mu S \end{aligned} \tag{A.3}$$

with $\Omega_s = (q_s/m_s)\partial_\mu \bar{H}$ the cyclotron frequency. The additional imposed constraint is that $\bar{H}$ should not depend on the gyro-angle φ_c. In this case, the last term on the right hand side of eq. (A.3) vanishes. This allows one to express the generating function S as function of the old and new Hamiltonians:

$$\begin{aligned} S(\mathbf{x}, \mathbf{p}) &= \int \frac{\mathrm{d}\varphi_c}{\Omega_s} \left\{ H(\mathbf{x}, \mathbf{p}) - \bar{H}(\mathbf{x}_G, \mathbf{p}_G) \right\} \\ &= \int \frac{m_s \, \mathrm{d}\varphi_c}{B} \left\{ U(\mathbf{x}) - \bar{U}(\mathbf{x}_G, \mathbf{p}_G) \right\} \end{aligned} \tag{A.4}$$

Injecting the expression of S, eq. (A.4), in equation (A.1), the distribution function of particles F_s can then be related to the one of the gyro-centers $\bar{F}_s$:

$$F_s(\mathbf{x}, \mathbf{v}, t) = \bar{F}_s(\mathbf{x}_G, \mathbf{v}_G, t) + \frac{q_s}{B} \left\{ U(\mathbf{x}, t) - \bar{U}(\mathbf{x}_G, \mathbf{v}_G, t) \right\} \, \partial_\mu \bar{F}_{s,\mathrm{eq}}(\mathbf{x}_G, \mathbf{v}_G)$$

Since the computation is performed at order one in ϵ, the equilibrium distribution function $\bar{F}_{s,\mathrm{eq}}$ only is retained in the last term.

[1] Rigorously speaking, the generating function G is $G = \mathbf{x}.\mathbf{p}' + \epsilon S$, where $\mathbf{x}.\mathbf{p}'$ can be shown to be the identity transform.

Appendix B

Some useful linear algebra expressions

This appendix is just a reminder of some simple but useful linear algebra used all along the manuscript.

B.1 Some useful expressions of the main vectorial operators for a general metric tensor

The Einstein notation will be used in this section. Let us consider a set of coordinates labelled $\{x^i\}$, the metric tensor $\{g_{ij}\}$ is defined via the length element

$$\mathrm{d}s^2 = g_{ij}\,\mathrm{d}x^i\,\mathrm{d}x^j$$

It is the product of the transpose Jacobian matrix J^T and the Jacobian matrix J, i.e $\{g_{ij}\} = J^T J$. Let g represents the determinant of the metric tensor (i.e $g = \det\{g_{ij}\}$), then the jacobian in space $\mathcal{J}_\mathrm{x}$ is defined as $\mathcal{J}_\mathrm{x} = \sqrt{g}$ and so equal to $\mathcal{J}_\mathrm{x} = \left[\left(\boldsymbol{\nabla}x^1 \times \boldsymbol{\nabla}x^2\right)\cdot\boldsymbol{\nabla}x^3\right]^{-1}$. The tensor $\{g^{ij}\}$ is the inverse of the tensor $\{g_{ij}\}$. The element of the contravariant metric tensor verify the relation $g^{ij} = \boldsymbol{\nabla}x^i\cdot\boldsymbol{\nabla}x^j$, i.e:

$$\{g^{ij}\} = \begin{pmatrix} g^{11} & g^{12} & g^{13} \\ g^{21} & g^{22} & g^{23} \\ g^{31} & g^{32} & g^{33} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\nabla}x^1\cdot\boldsymbol{\nabla}x^1 & \boldsymbol{\nabla}x^1\cdot\boldsymbol{\nabla}x^2 & \boldsymbol{\nabla}x^1\cdot\boldsymbol{\nabla}x^3 \\ \boldsymbol{\nabla}x^2\cdot\boldsymbol{\nabla}x^1 & \boldsymbol{\nabla}x^2\cdot\boldsymbol{\nabla}x^2 & \boldsymbol{\nabla}x^2\cdot\boldsymbol{\nabla}x^3 \\ \boldsymbol{\nabla}x^3\cdot\boldsymbol{\nabla}x^1 & \boldsymbol{\nabla}x^3\cdot\boldsymbol{\nabla}x^2 & \boldsymbol{\nabla}x^3\cdot\boldsymbol{\nabla}x^3 \end{pmatrix} \tag{B.1}$$

With these notations, any vector $\mathbf{A}$ is defined by:

$$\mathbf{A} = A_i\boldsymbol{\nabla}x^i$$

and the equivalent norm is given by:

$$\|\mathbf{A}\| = \sqrt{(A_1)^2 g^{11} + (A_2)^2 g^{22} + (A_3)^2 g^{33}}$$

The covariant coordinates are expressed as

$$A_i = \mathcal{J}_\mathrm{x}\frac{\epsilon_{ijk}}{2}\left(\boldsymbol{\nabla}x^j \times \boldsymbol{\nabla}x^k\right)\cdot\mathbf{A}$$

or alternatively as

$$\mathbf{A} = \mathcal{J}_\mathrm{x}\frac{\epsilon_{ijk}}{2}A^i\left(\boldsymbol{\nabla}x^j \times \boldsymbol{\nabla}x^k\right)$$

where the contravariant coordinates are

$$A^i = \mathbf{A}\cdot\boldsymbol{\nabla}x^i$$

Therefore, the contra and covariant coordinates are related by the relations

$$A^i = g^{ij}A_j \quad \text{and} \quad A_i = g_{ij}A^j$$

The vector divergence and the laplacian can be defined by

$$\nabla\cdot\mathbf{A} = \frac{1}{\mathcal{J}_\mathrm{x}}\partial_{x^i}\left(\mathcal{J}_\mathrm{x}A^i\right) \quad \text{and} \quad \nabla^2\phi = \frac{1}{\mathcal{J}_\mathrm{x}}\partial_{x^i}(\mathcal{J}_\mathrm{x}g^{ij}\partial_{x^j}\phi)$$

The gradient can be written

$$\boldsymbol{\nabla}\phi = \partial_{x^i}\phi\nabla x^i$$

from which one deduces the components of a rotational

$$(\boldsymbol{\nabla}\times\mathbf{B})^i = \frac{1}{\mathcal{J}_{\mathrm{x}}}\epsilon^{ijk}\partial_{x^j}B_k$$

B.2 Poisson brackets defined as $[F,G] = \mathbf{b}\cdot(\boldsymbol{\nabla}F\times\boldsymbol{\nabla}G)$

For the following, let us defined the Poisson bracket by $[F,G] = \mathbf{b}\cdot(\boldsymbol{\nabla}F\times\boldsymbol{\nabla}G)$ with $\mathbf{b} = \mathbf{B}/\|B\|$ the unitary magnetic field. Let b_k represent the covariant components of $\mathbf{b}$ and $\mathcal{J}_{\mathrm{x}}$ be the jacobian of the system, then it can be easily checked that:

$$[F,G] = \mathcal{J}_{\mathrm{x}}^{-1}\epsilon^{ijk}\partial_{x^i}F\partial_{x^j}Gb_k \tag{B.2}$$

with ϵ^{ijk} the Levi-Civita symbol. Using the previous expression (B.2), a more explicit form of the Poisson bracket $[F,G]$ reads:

$$\begin{aligned}
[F,G] &= \frac{b_1}{\mathcal{J}_{\mathrm{x}}}\left(\partial_{x^2}F\,\partial_{x^3}G - \partial_{x^3}F\,\partial_{x^2}G\right) + \frac{b_2}{\mathcal{J}_{\mathrm{x}}}\left(\partial_{x^3}F\,\partial_{x^1}G - \partial_{x^1}F\,\partial_{x^3}G\right)\\
&\quad + \frac{b_3}{\mathcal{J}_{\mathrm{x}}}\left(\partial_{x^1}F\,\partial_{x^2}G - \partial_{x^2}F\,\partial_{x^1}G\right)\\
[F,G] &= \frac{B_1}{\mathcal{J}_{\mathrm{x}}B}\left(\partial_{x^2}F\,\partial_{x^3}G - \partial_{x^3}F\,\partial_{x^2}G\right) + \frac{B_2}{\mathcal{J}_{\mathrm{x}}B}\left(\partial_{x^3}F\,\partial_{x^1}G - \partial_{x^1}F\,\partial_{x^3}G\right)\\
&\quad + \frac{B_3}{\mathcal{J}_{\mathrm{x}}B}\left(\partial_{x^1}F\,\partial_{x^2}G - \partial_{x^2}F\,\partial_{x^1}G\right)
\end{aligned} \tag{B.3}$$

Appendix C

Quasi-neutrality solver

C.1 How to overcome the difficulty due to $\langle\, U \,\rangle_{\mathrm{FS}}$ term ?

In this appendix, we describe the numerical solving of the quasi-neutrality equation (2.13). In particular, we explain what is done to overcome the problem of the term $\langle\ \rangle_{\text{Flux Surf}}$ in Fourier space.
For the following, let us first notice that equation (2.13) can be re-written as

$$\mathcal{L}U + \frac{e}{T_e(r)}\left[U - \lambda\langle\, U \,\rangle_{\text{Flux Surf}}\right] = \rho(r,\theta,\varphi) \tag{C.1}$$

where the differential operator of second order $\mathcal{L}$ is defined as:

$$\mathcal{L} = -\frac{1}{n_{e_0}(r)}\sum_s Z_s \nabla_\perp \cdot \left(\frac{n_{s,\mathrm{eq}}(r)}{B\Omega_s}\nabla_\perp\right)$$

In the present version of the code, two assumptions are done for this operator: (i) $B(r,\theta)$ is assumed constant equal to B_0 the magnetic field on the magnetic axis and (ii) $n_{s,\mathrm{eq}} = n_{s_0}$ where n_{s_0} is the initial radial density profile. Therefore, the differential operator reads

$$\begin{aligned}
\mathcal{L} &= -\frac{1}{n_{e_0}(r)}\sum_s \frac{Z_s}{B_0\Omega_s}\nabla_\perp \cdot (n_{s_0}(r)\nabla_\perp) \\
&= -\frac{1}{n_{e_0}(r)}\sum_s \frac{Z_s}{B_0\Omega_s} n_{s_0}(r)\left\{\frac{\partial^2}{\partial r^2} + \left[\frac{1}{r} + \frac{1}{n_{s_0}(r)}\frac{\mathrm{d}n_{s_0}(r)}{\mathrm{d}r}\right]\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial\theta^2}\right\}
\end{aligned}$$

The right hand side reads

$$\rho(r,\theta,\varphi) = \frac{1}{n_{e_0}(r)}\sum_s Z_s\left[n_{G_s}(r,\theta,\varphi) - n_{G_s,\mathrm{eq}}(r,\theta)\right]$$

where n_{G_s} and $n_{G_s,\mathrm{eq}}$ are respectively defined by equations (2.14) and (2.15). The constant λ can be chosen equal to 1 (by default) or equal to 0.
Let for all function g, $\langle\, g \,\rangle_{\theta,\,\varphi}$ being the radial function equal to

$$\langle\, g(r) \,\rangle_{\theta,\,\varphi} = \frac{1}{L_\theta L_\varphi}\int\int g(r,\theta,\varphi)\,\mathrm{d}\theta\,\mathrm{d}\varphi$$

By applying the integration $\frac{1}{L_\theta L_\varphi}\int\int\cdot\,\mathrm{d}\theta\,\mathrm{d}\varphi$ to the previous equation (C.1) and by using the fact that $\langle\,\langle\, U \,\rangle_{\theta,\,\varphi}\,\rangle_{\text{Flux Surf}} = \langle\, U \,\rangle_{\text{Flux Surf}}$ then:

$$\mathcal{L}\langle\, U \,\rangle_{\theta,\,\varphi} + \frac{e}{T_e}\left[\langle\, U \,\rangle_{\theta,\,\varphi} - \lambda\langle\, U \,\rangle_{\text{Flux Surf}}\right] = \langle\, \rho \,\rangle_{\theta,\,\varphi} \tag{C.2}$$

Let $\mathcal{U}$ being $\mathcal{U} = U - \langle\, U \,\rangle_{\theta,\,\varphi}$ then, by subtracting (C.2) to (C.1), and by using Dirichlet boundary conditions we obtain $\forall\lambda \in \mathbb{R}$, $\forall\theta \in [0, L_\theta]$ and $\forall\varphi \in [0, L_\varphi]$:

$$\begin{cases} \left(\mathcal{L} + \frac{e}{T_e}\right)\mathcal{U}(r,\theta,\varphi) = \varrho(r,\theta,\varphi) \quad \text{with} \quad \varrho = \rho - \langle\, \rho \,\rangle_{\theta,\,\varphi} \qquad \forall r \in [r_{\min}, r_{\max}] \\ \mathcal{U}(r_{\min},\theta,\varphi) = \mathcal{U}(r_{\max},\theta,\varphi) = 0 \end{cases} \tag{C.3}$$

Besides, (C.2) can be written as

$$\begin{aligned}&\mathcal{L}\left(\langle U\rangle_{\theta,\varphi}-\langle U\rangle_{\text{Flux Surf}}\right)+\mathcal{L}\langle U\rangle_{\text{Flux Surf}}\\&+\frac{e}{T_e}\left[\left(\langle U\rangle_{\theta,\varphi}-\langle U\rangle_{\text{Flux Surf}}\right)+\langle U\rangle_{\text{Flux Surf}}-\lambda\langle U\rangle_{\text{Flux Surf}}\right]=\langle\rho\rangle_{\theta,\varphi}\end{aligned}$$

Then using the fact that $\langle\mathcal{U}\rangle_{\text{Flux Surf}}=\langle U\rangle_{\text{Flux Surf}}-\langle\langle U\rangle_{\theta,\varphi}\rangle_{\text{Flux Surf}}$ and using Dirichlet boundary conditions, the previous equation leads to the following system:

$$\begin{cases}\left(\mathcal{L}+(1-\lambda)\frac{e}{T_e}\right)\langle U\rangle_{\text{Flux Surf}}=\langle\rho\rangle_{\theta,\varphi}+\left(\mathcal{L}+\frac{e}{T_e}\right)\langle\mathcal{U}\rangle_{\text{Flux Surf}}\\ \langle U\rangle_{\text{Flux Surf}}(r_{\min})=\langle U\rangle_{\text{Flux Surf}}(r_{\max})=0\end{cases}\tag{C.4}$$

Using the definition of $\mathcal{U}$, then

$$\begin{aligned}\langle\mathcal{U}\rangle_{\text{Flux Surf}}&=\langle U\rangle_{\text{Flux Surf}}-\langle\langle U\rangle_{\theta,\varphi}\rangle_{\text{Flux Surf}}\\&=\langle U\rangle_{\text{Flux Surf}}-\langle U\rangle_{\theta,\varphi}\qquad(\text{because }\langle\langle U\rangle_{\theta,\varphi}\rangle_{\text{Flux Surf}}=\langle U\rangle_{\theta,\varphi})\\\text{so }\langle U\rangle_{\theta,\varphi}&=\langle U\rangle_{\text{Flux Surf}}-\langle\mathcal{U}\rangle_{\text{Flux Surf}}\end{aligned}$$

and using the fact that $U=\mathcal{U}+\langle U\rangle_{\theta,\varphi}$, we obtain the expression of the electric potential U as:

$$U(r,\theta,\varphi)=\mathcal{U}(r,\theta,\varphi)-\langle\mathcal{U}\rangle_{\text{Flux Surf}}(r)+\langle U\rangle_{\text{Flux Surf}}(r)\tag{C.5}$$

To summarize, the solving of the equation (2.13) can be replaced by the solving of two simpler equations (C.3) and (C.4). Indeed, the equation (C.4) is a differential equation only depending on the radial direction. Besides, in (C.3) the variable φ plays the role of a parameter, then the discretization of the equation can be performed by projecting in Fourier space in θ direction and by using finite differences in the radial direction as described in the following paragraph.

However, it is important to realize that the boundary conditions are not directly applied on U but on $\mathcal{U}=U-\langle U\rangle_{\theta,\varphi}$. So the fact to impose $\mathcal{U}(r_{\min})=0$ does not imply $U(r_{\min})=0$ but $U(r_{\min})=\langle U\rangle_{\theta,\varphi}(r_{\min})$ (same remark can be done at $r=r_{\max}$). Another treatment is available when $r_{\min}$ is sufficiently close to 0 ($r_{\min}<10^{-2}$ in the code). Indeed, let us assume that in this case $\mathcal{J}_{\text{x}}(r_{\min},\theta)$ is equal to a constant. Then, for all function g,

$$\langle g\rangle_{\text{FS}}(r_{\min})=\frac{\int g\mathcal{J}_{\text{x}}(r_{\min},\theta)\,\mathrm{d}\theta\,\mathrm{d}\varphi}{\int\mathcal{J}_{\text{x}}(r_{\min},\theta)\,\mathrm{d}\theta\,\mathrm{d}\varphi}=\frac{1}{4\pi}\langle g\rangle_{\theta,\varphi}\quad\forall|r_{\min}|\ll1$$

In this case employing a Neumann boundary condition on the $(0,0)$ mode at the axis (i.e $\partial_r\langle U\rangle_{\theta,\varphi}(r_{\min})=0$) is equivalent to applying $\partial_r\langle U\rangle_{\text{FS}}(r_{\min})=0$ in the matrix system (C.4).

C.2 Finite differences in radial direction and Fourier projections in θ and ϕ

C.2.1 Solving of the equation system (C.3)

Let $\mathcal{U}$ and ϱ be represented in terms of the Fourier expansion as

$$\begin{cases}\mathcal{U}(r,\theta,\varphi)=\sum_m\mathcal{U}^m(r,\varphi)\exp(im\theta)\\ \varrho(r,\theta,\varphi)=\sum_m\varrho^m(\hat{r},\varphi)\exp(im\theta)\end{cases}$$

then the equation (C.3) can be rewritten in the wave number representation, for each poloidal mode m and for each independent value of φ, as the following differential equation:

$$\left(\mathcal{L}^m+\frac{e}{T_e(r)}\right)\mathcal{U}^m(r,\varphi)=\varrho^m(\hat{r},\varphi)\tag{C.6}$$

with the operator $\mathcal{L}^m$ defined as

$$\mathcal{L}^m = -\frac{1}{n_{e_0}(r)} \sum_s \frac{Z_s}{B_0 \Omega_s} n_{s_0}(r) \left\{ \frac{\partial^2}{\partial r^2} + \left[\frac{1}{r} + \frac{1}{n_{s_0}(r)} \frac{\mathrm{d}n_{s_0}(r)}{\mathrm{d}r} \right] \frac{\partial}{\partial r} - \frac{m^2}{r^2} \right\} \tag{C.7}$$

and where $\mathcal{U}^m$ (resp. ϱ^m) is the Fourier transform in θ of $\mathcal{U}$ (resp. ϱ). Let N_r be the number of radial points and let assumes that the radial domain is defined inside $[r_1, r_{N_r}]$ (i.e $r_{\min} = r_1$ and $r_{\max} = r_{N_r}$), then up to second order in Δr, the system of equations (C.3) leads to the tridiagonal $(N_r - 2) \times (N_r - 2)$ system:

$$\begin{pmatrix} d^m_{r_2} & u_{r_2} & 0 & & \\ l_{r_3} & d^m_{r_3} & u_{r_3} & 0 & \\ & \ddots & \ddots & \ddots & \\ & 0 & l_{r_{N_r-2}} & d^m_{r_{N_r-2}} & u_{r_{N_r-2}} \\ & & 0 & l_{r_{N_r-1}} & d^m_{r_{N_r-1}} \end{pmatrix} \begin{pmatrix} \mathcal{U}^m_2 \\ \mathcal{U}^m_3 \\ \vdots \\ \mathcal{U}^m_{N_r-2} \\ \mathcal{U}^m_{N_r-1} \end{pmatrix} = \begin{pmatrix} \varrho^m_2 \\ \varrho^m_3 \\ \vdots \\ \varrho^m_{N_r-2} \\ \varrho^m_{N_r-1} \end{pmatrix} \tag{C.8}$$

with for each $r_i \in [r_1, r_{N_r}]$

$$\begin{cases} l_{r_i} &= -\left(\frac{\beta(r_i)}{\Delta r^2} - \frac{\alpha(r_i)}{2\Delta r} \right) \\ d^m_{r_i} &= \beta(r_i) \left(\frac{2}{\Delta r^2} + \frac{m^2}{r_i^2} \right) + \frac{e}{T_e(r_i)} \\ u_{r_i} &= -\left(\frac{\beta(r_i)}{\Delta r^2} + \frac{\alpha(r_i)}{2\Delta r} \right) \\ \varrho^m &= \varrho^m(r_i) \end{cases} \tag{C.9}$$

with

$$\begin{cases} \alpha(r_i) &= \sum_s \frac{Z_s}{B_0 \Omega_s} \frac{n_{s_0}(r_i)}{n_{e_0}(r_i)} \left(\frac{1}{r_i} + \frac{1}{n_{s_0}(r_i)} \frac{\mathrm{d}n_{s_0}(r_i)}{\mathrm{d}r} \right) \\ \beta(r_i) &= \sum_s \frac{Z_s}{B_0 \Omega_s} \frac{n_{s_0}(r_i)}{n_{e_0}(r_i)} \end{cases} \tag{C.10}$$

and where $\mathcal{U}^m_1 = \mathcal{U}^m_{N_r} = 0$.
Solving the previous matrix system (C.8) is equivalent to solve a matrix system of the form $A\mathbf{x} = \mathbf{b}$ where A is of the form (E.1) where the three diagonals are defined by:

$$\begin{aligned} [l_1, \cdots, l_N] &= [0, l_{r_3}, \cdots, l_{r_{N_r-1}}] \\ [d_1, \cdots, d_N] &= [d_{r_2}, d_{r_3}, \cdots, d_{r_{N_r-1}}] \\ [u_1, \cdots, u_N] &= [u_{r_2}, \cdots, u_{r_{N_r-2}}, 0] \end{aligned}$$

and the right side vector $\mathbf{b}$ corresponds to

$$[b_1, \cdots, b_N] = [\varrho^m(r_2), \cdots, \varrho^m(r_{N_r-1})]$$

while the result vector $\mathcal{U}^m$ is given by:

$$[\mathcal{U}^m_1, \cdots, \mathcal{U}^m_{N_r}] = [0, x_1, x_2, \cdots, x_N, 0]$$

C.2.2 Solving of the equation system (C.4)

In the following $\langle \cdot \rangle_{\text{Flux Surf}}$ is replaced by $\langle \cdot \rangle_{\text{FS}}$ for more readability. The system (C.4) can be rewritten as:

$$\begin{cases} \left(\mathcal{L} + (1-\lambda) \frac{e}{T_e(r_i)} \right) \langle U \rangle_{\text{FS}}(r_i) = \Gamma(r_i) \qquad \text{for each } r_i \in [r_1, r_{N_r}] \\ \langle U \rangle_{\text{FS}}(r_{\min}) = \langle U \rangle_{\text{FS}}(r_{\max}) = 0 \end{cases}$$

with $\Gamma(r_i) = \langle\, \rho \,\rangle_{\theta,\,\varphi}(r_i) + \left(\mathcal{L} + \frac{e}{T_e(r_i)}\right) \langle\, \mathcal{U} \,\rangle_{\mathrm{FS}}$ where $\langle\, \rho \,\rangle_{\theta,\,\varphi}(r_i) = \frac{1}{L_\theta L_\varphi} \int\int \rho(r_i, \theta, \varphi)\, \mathrm{d}\theta\, \mathrm{d}\varphi$ which is equivalent (by using the same notation than for the previous matrix system (C.8)) to:

$$\begin{pmatrix} d_{r_2} & u_{r_2} & 0 & & \\ l_{r_3} & d_{r_3} & u_{r_3} & 0 & \\ & \ddots & \ddots & \ddots & \\ & 0 & l_{r_{N_r-2}} & d_{r_{N_r-2}} & u_{r_{N_r-2}} \\ & & 0 & l_{r_{N_r-1}} & d_{r_{N_r-1}} \end{pmatrix} \begin{pmatrix} \langle\, U \,\rangle_{\mathrm{FS}}(r_2) \\ \langle\, U \,\rangle_{\mathrm{FS}}(r_3) \\ \vdots \\ \langle\, U \,\rangle_{\mathrm{FS}}(r_{N_r-2}) \\ \langle\, U \,\rangle_{\mathrm{FS}}(r_{N_r-1}) \end{pmatrix} = \begin{pmatrix} \Gamma(r_2) \\ \Gamma(r_3) \\ \vdots \\ \Gamma(r_{N_r-2}) \\ \Gamma(r_{N_r-1}) \end{pmatrix} \tag{C.11}$$

with for each $r_i \in [r_2, r_{N_r-1}]$

$$\begin{cases} l_{r_i} &= -\left(\frac{\beta(r_i)}{\Delta r^2} - \frac{\alpha(r_i)}{2\Delta r}\right) \\ d_{r_i} &= \frac{2}{\Delta r^2}\beta(r_i) + (1-\lambda)\,\frac{e}{T_e(r_i)} \\ u_{r_i} &= -\left(\frac{\beta(r_i)}{\Delta r^2} + \frac{\alpha(r_i)}{2\Delta r}\right) \\ \Gamma(r_i) &= \langle\, \rho \,\rangle_{\theta,\,\varphi}(r_i) + \left(\mathcal{L} + \frac{e}{T_e(r_i)}\right) \end{cases}$$

where $\alpha(r_i)$ and $\beta(r_i)$ are defined by (C.10). Let us remark that the super-diagonal $(u_{r_i})_{i=2,\cdots,N_r-2}$ and the lower-diagonal $(l_{r_i})_{i=3,\cdots,N_r-1}$ are the same than the ones in the previous matrix system (C.8), while the diagonal can be deduced from the previous diagonal of the poloidal mode $m = 0$ (i.e $\left(d^m_{r_i}\right)_{i=2,\cdots,N_r-1}$ for $m = 0$) by the relation

$$d_{r_i} = d^0_{r_i} - \lambda \frac{e}{T_e(r_i)}$$

C.2.3 Global algorithm for the quasi-neutrality solver

Then the different steps for solving (C.1) and obtaining U are the following:

1. Compute and save ρ,
2. Solve (C.3) to obtain $\mathcal{U}$ and save the 3D array $\mathcal{U}$,
3. Compute $\langle\, \mathcal{U} \,\rangle_{\mathrm{Flux\,Surf}}$ and save this 1D array,
4. Compute the RHS of (C.4), i.e $\langle\, \rho \,\rangle_{\theta,\,\varphi} + \left(\mathcal{L} + \frac{e}{T_e}\right) \langle\, \mathcal{U} \,\rangle_{\mathrm{Flux\,Surf}}$ and save this 1D array,
5. Solve (C.4) to obtain $\langle\, U \,\rangle_{\mathrm{Flux\,Surf}}$ and store it,
6. Compute $U(r, \theta, \varphi)$ by using (C.5)

where the matrix systems (C.3) and (C.4) are both solved by using finite differences in radial direction and Fourier projection in θ direction. Fourier space is also used in ϕ direction for (C.4).

Appendix D

Padé approximation for gyro-average operator

This appendix focuses on the implementation of the gyro-average operator in the code when the choice of a Padé approximation is done. This option is still currently the default choice in the code. The implementation detailed below is directly accessible in GYSELA source files. The second choice based on an integration on the gyro-circles by using Hermite interpolation will not be presented in the following. This option is still for the moment compiled as an external library. All details of computation can be founded in Steiner's paper [SMC+15] and Rozar's paper [RSM+15]. Let us first recall that this gyro-average operator take the form

$$\bar{g}(\mathbf{x}_G, v_\perp) = \int_{-\infty}^{+\infty} \frac{\mathrm{d}^3\mathbf{k}}{(2\pi)^3}\, J_0(k_\perp \rho_s)\hat{g}(\mathbf{k})\mathrm{e}^{i\mathbf{k}\cdot\mathbf{x}_G} \tag{D.1}$$

where ρ_s is the Larmor radius ($\rho_s = v_\perp/\Omega_s$ with $\Omega_s = q_s B_0/m_s$). We can first notice that φ plays the role of a parameter. The Fourier projection can be used along the periodic direction θ but not in the radial one. This is one of the drawback of global codes. A possibility is to use finite differences in the radial direction. However, such a representation does not provide $\vec{k}_\perp$ involved in gyro-average definition (D.1). To overcome this difficulty, the Bessel function is replaced by the following Padé approximation:

$$J_0(k_\perp \rho_s) \sim \frac{1}{1 + \frac{(k_\perp \rho_s)^2}{4}} \tag{D.2}$$

This approximation gives the right limit at vanishing $k_\perp \rho_s$, while keeping J_0 finite in the opposite limit $k_\perp \rho_s \to \infty$. Using the equivalence $i\vec{k}_\perp \leftrightarrow \boldsymbol{\nabla}_\perp$, the gyro-average operation of any g function then leads to the following implicit equation:

$$\left(1 - \frac{\rho_s^2}{4}\nabla_\perp^2\right)\bar{g}(r,\theta,\varphi) = g(r,\theta,\varphi) \tag{D.3}$$

where we recall that $\nabla_\perp^2 = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2}{\partial\theta^2} = \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial\theta^2}$.
So the equation (D.3) expressed in terms of μ_s ($v_\perp = \sqrt{2B(r,\theta)\mu_s/m_s}$) reads:

$$\left(1 - \frac{1}{2\Omega_s^2}\frac{B(r,\theta)}{m_s}\mu_s\nabla_\perp^2\right)\bar{g}(r,\theta,\varphi) = g(r,\theta,\varphi) \tag{D.4}$$

At first approximation and to be consistent with the solving of the quasi-neutrality equation $B(r,\theta)$ is replaced by B_0, such that each m Fourier mode of $\bar{g}$ is the solution of the equation:

$$\left[1 - \frac{1}{2\Omega_s^2}\frac{B_0}{m_s}\mu_s\left(\frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} - \frac{m^2}{r^2}\right)\right]\bar{g}^m(r,\varphi) = g^m(r,\varphi)$$

For the discrete problem, we consider (N_r+1) grid points in the radial direction, such that $r_i \in [r_0, r_{N_r}]$. Then, using a Taylor expansion of second order for the first and second derivatives gives the following

discrete problem, $\forall i = 1, \cdots, N_r - 1$:

$$
\begin{aligned}
&- K_s \frac{\mu_s}{2} \left(\frac{1}{\Delta r^2} - \frac{1}{2 r_i \Delta r} \right) \bar{g}_{i-1}^m(\varphi) \\
&\qquad + \left[1 + K_s \frac{\mu_s}{2} \left(\frac{2}{\Delta r^2} + \frac{m^2}{r_i^2} \right) \right] \bar{g}_i^m(\varphi) - K_s \frac{\mu_s}{2} \left(\frac{1}{\Delta r^2} + \frac{1}{2 r_i \Delta r} \right) \bar{g}_{i+1}^m(\varphi) = g_i^m(\varphi)
\end{aligned}
$$

where the constant K_s is defined as $K_s = B_0/(m_s \Omega_s^2)$. This is equivalent to the matrix system

$$
a_{r_i}\, \bar{g}_{i-1}^m(\varphi) + b_{r_i}\, \bar{g}_i^m(\varphi) + c_{r_i}\, \bar{g}_{i+1}^m(\varphi) = g_i^m(\varphi) \qquad \forall i = 1, \cdots, N_r - 1
$$

with

$$
\begin{cases}
a_{r_i} &= -K_s \frac{\mu_s}{2} \left(\frac{1}{\Delta r^2} - \frac{1}{2 r_i \Delta r} \right) \\
b_{r_i} &= 1 + K_s \frac{\mu_s}{2} \left(\frac{2}{\Delta r^2} + \frac{m^2}{r_i^2} \right) \\
c_{r_i} &= -K_s \frac{\mu_s}{2} \left(\frac{1}{\Delta r^2} + \frac{1}{2 r_i \Delta r} \right)
\end{cases}
$$

Since *Neumann boundary conditions are assumed* (i.e $\partial_r \bar{g}(r_0) = \partial_r \bar{g}(r_{N_r}) = 0$) solving the equation requires, for each φ, the inversion of the following tridiagonal $(N_r - 1) \times (N_r - 1)$ system:

$$
\begin{pmatrix}
a_{r_1} + b_{r_1} & c_{r_1} & 0 & & \\
a_{r_2} & b_{r_2} & c_{r_2} & 0 & \\
 & \ddots & \ddots & \ddots & \\
 & 0 & a_{r_{N_r-2}} & b_{r_{N_r-2}} & c_{r_{N_r-2}} \\
 & & 0 & a_{r_{N_r-1}} & b_{r_{N_r-1}} + c_{r_{N_r-1}}
\end{pmatrix}
\begin{pmatrix}
\bar{g}_1^m(\varphi) \\ \bar{g}_2^m(\varphi) \\ \vdots \\ \bar{g}_{N_r-2}^m(\varphi) \\ \bar{g}_{N_r-1}^m(\varphi)
\end{pmatrix}
=
\begin{pmatrix}
g_1^m(\varphi) \\ g_2^m(\varphi) \\ \vdots \\ g_{N_r-2}^m(\varphi) \\ g_{N_r-1}^m(\varphi)
\end{pmatrix}
\tag{D.5}
$$

then $\bar{g}_0^m = \bar{g}_1^m$ and $\bar{g}_{N_r}^m = \bar{g}_{N_r-1}^m$. The projection in Fourier space in the poloidal direction is performed by FFT. And for each point φ_k in the toroidal direction the previous tridiagonal $(N_r - 1) \times (N_r - 1)$ system is solved with a modified Thomas algorithm (see Appendix E for details of the implementation in the code).

Appendix E

LU factorization with a modified Thomas algorithm

The choice which has been done to perform the LU factorization (which appears in the quasi-neutrality equation solving (see Appendix C) and in the gyro-average operator (see Appendix D) is to use a modified version of Thomas algorithm. Details of this widespread numerical algorithm are the subject of this appendix. Let us consider a $N \times N$ linear system of the form:

$$A\mathbf{x} = \mathbf{b}$$

where we denote by $A = (a_{ij}) \in \mathbb{R}^{N\times N}$ the coefficient matrix, by $\mathbf{b} = (b_i) \in \mathbb{R}^N$ the right side vector and by $\mathbf{x} = (b_i) \in \mathbb{R}^N$ the unknown vector, respectively.

E.1 Tridiagonal Matrices

Consider the particular case of a linear system with $N \times N$ nonsingular tridiagonal matrix A given by

$$A = \begin{bmatrix} d_1 & u_2 & & 0 \\ l_2 & d_2 & \ddots & \\ & \ddots & \ddots & u_{N-1} \\ 0 & & l_N & d_N \end{bmatrix} \tag{E.1}$$

In such an event, the matrices L and U of the LU factorization of A are bidiagonal matrices of the form:

$$L = \begin{bmatrix} 1 & & & 0 \\ \beta_2 & 1 & & \\ & \ddots & \ddots & \\ 0 & & \beta_N & 1 \end{bmatrix} \qquad U = \begin{bmatrix} \alpha_1 & u_2 & & 0 \\ & \alpha_2 & \ddots & \\ & & \ddots & u_{N-1} \\ 0 & & & \alpha_N \end{bmatrix}$$

E.1.1 Thomas algorithm

The coefficients α_i and β_i can be easily be computed by the following relations

$$\alpha_1 = l_1 \ , \ \beta_i = \frac{l_i}{\alpha_{i-1}} \ , \ \alpha_i = d_i - \beta_i u_{i-1} \ , \ i = 2, \cdots, N \tag{E.2}$$

This is known as the *Thomas algorithm* and can be regarded as a particular instance of the Doolittle factorization, without pivoting. When one is not interested in storing the coefficients of the original matrix, the entries α_i and β_i can be overwritten on A. The Thomas algorithm can also be extended to solve the whole tridiagonal system $A\mathbf{x} = \mathbf{b}$. This amounts to solving two bidiagonal systems $L\mathbf{y} = \mathbf{b}$

and $U\mathbf{x} = \mathbf{y}$, for which the following formulae hold:

$$(L\mathbf{y} = \mathbf{b}): \quad y_1 = b_1,\ y_i = b_i - \beta_i\, y_{i-1},\ i = 2, \cdots, N \tag{E.3}$$

$$(U\mathbf{x} = \mathbf{y}): \quad x_n = \frac{y_n}{\alpha_n},\ x_i = (y_i - u_i\, x_{i+1})/\alpha_i,\ i = N-1, \cdots, 1 \tag{E.4}$$

The algorithm requires only $8N - 7$ flops: precisely, $3(N-1)$ flops for the factorization (E.2) and $5N - 4$ flops for the substitution procedure (E.3)-(E.4).

E.1.2 Modified version of Thomas algorithm

The Thomas algorithm can be implemented in several ways. In particular, when implementing it on computers where divisions are more costly than multiplications, it is possible (and convenient) to devise a version of the algorithm without divisions in (E.4), by resorting to the following form of the factorization

$$A = LDM^T = \begin{bmatrix} \gamma_1^{-1} & 0 & & 0 \\ l_2 & \gamma_2^{-1} & \ddots & \\ & \ddots & \ddots & 0 \\ 0 & & l_N & \gamma_N^{-1} \end{bmatrix} \begin{bmatrix} \gamma_1 & & & 0 \\ & \gamma_2 & & \\ & & \ddots & \\ 0 & & & \gamma_N \end{bmatrix} \begin{bmatrix} \gamma_1^{-1} & u_2 & & 0 \\ 0 & \gamma_2^{-1} & \ddots & \\ & \ddots & \ddots & u_{N-1} \\ 0 & & 0 & \gamma_N^{-1} \end{bmatrix}$$

The coefficients γ_i can be recursively computed by the formulae

$$\gamma_i = (d_i - l_i\, \gamma_{i-1}\, u_{i-1})^{-1},\ i = 1, \cdots, N \tag{E.5}$$

where $\gamma_0 = 0$, $l_1 = 0$ and $u_N = 0$ have been assumed. The forward and backward substitutions algorithms respectively read:

$$(L\mathbf{y} = \mathbf{b}): \quad y_1 = \gamma_1\, b_1,\ y_i = \gamma_i(b_i - l_i\, y_{i-1}),\ i = 2, \cdots, N \tag{E.6}$$

$$(U\mathbf{x} = \mathbf{y}): \quad x_n = y_n,\ x_i = y_i - \gamma_i\, u_i\, x_{i+1},\ i = N-1, \cdots, 1 \tag{E.7}$$

Appendix F

Numerical implementation of the collision operator in Gysela

In this appendix, the simplified expression of the Lorentz-type operator which is used in GYSELA is detailed. The expression of this collision operator (including the perpendicular direction, which is not yet implemented in GYSELA) is

$$\mathcal{C}(\bar{F}_s) = \frac{1}{B^*_{\|s}}\partial_{v_{G\|}}\left\{B^*_{\|s}\mathcal{D}_{\|s}\bar{F}_{M_s}\partial_{v_{G\|}}\left(\frac{\bar{F}_s}{\bar{F}_{M_s}}\right)\right\} + \frac{1}{B^*_{\|s}}\partial_\mu\left\{B^*_{\|s}\mathcal{D}_{\perp s}\bar{F}_{M_s}\frac{1}{B^2}\partial_\mu\left(\frac{\bar{F}_s}{\bar{F}_{M_s}}\right)\right\} \tag{F.1}$$

where $B^*_{\|s}(r,\theta,v_{G\|}) = B(r,\theta) + v_{G\|}\mathbf{b}\cdot\nabla\times\mathbf{b}$ is the Jacobian of the guiding-center coordinates, and $\bar{F}_{M_s}$ is the following shifted Maxwellian distribution

$$\bar{F}_{M_s} = \frac{n_{s_0}}{(2\pi T_{s,\text{coll}}/m_s)^{3/2}}\exp\left(-\frac{m_s\left(v_{G\|}-V_{\|s,\text{coll}}\right)^2}{2T_{s,\text{coll}}} - \frac{\mu_s B}{T_{s,\text{coll}}}\right) \tag{F.2}$$

with the mean temperature $T_{s,\text{coll}} = T_{s,\text{coll}}(r,\theta,\varphi)$ and the mean velocity $V_{\|s,\text{coll}} = V_{\|s,\text{coll}}(r,\theta,\varphi)$. $\bar{F}_{M_s}$ is such that $\mathcal{C}(\bar{F}_s) = 0$. The collision term $\mathcal{D}_{\|s}$ is defined by the equations (2.27)-(2.29). The expression of $T_{s,\text{coll}}$ and $V_{\|s,\text{coll}}$ are constrained such that the collision operator is momentum and energy preserving. These calculations and expressions are detailed in a first paragraph F.1 while the numerical implementation based on a semi-implicit second order Crank-Nicolson scheme is described in the second one F.2.

F.1 Expressions of the mean temperature $T_{s,\text{coll}}$ and mean velocity $V_{\|s,\text{coll}}$ for the collision operator

While this simplified collision operator obviously conserves the number of particles, the profiles $T_{s,\text{coll}}(r,\theta,\varphi)$ and $V_{\|s,\text{coll}}(r,\theta,\varphi)$ must be chosen so that the operator is also compatible with the conservations of momentum and energy

$$\int \mathcal{J}_\text{v}\,\mathrm{d}\mu_s\,\mathrm{d}v_{G\|}\,m_s v_{G\|}\mathcal{C}(\bar{F}_s) = 0 \tag{F.3}$$

$$\int \mathcal{J}_\text{v}\,\mathrm{d}\mu_s\,\mathrm{d}v_{G\|}\left(\mu_s B + \frac{1}{2}m_s v_{G\|}^2\right)\mathcal{C}(\bar{F}_s) = 0 \tag{F.4}$$

where $\mathcal{J}_\text{v} = 2\pi B^*_{\|s}/m_s$ is the jacobian in velocity space. Let us first consider only the contribution of collisions in the parallel direction. We use the expression of the collision operator (F.1) and integrate

Eq. (F.3) by parts twice in the variable $v_{G\|}$

$$
\begin{aligned}
(\text{F.3})_{\|} &= \int \frac{2\pi B^*_{\|s}}{m_s}\, \mathrm{d}\mu_s\, \mathrm{d}v_{G\|}\, m_s v_{G\|} \frac{1}{B^*_{\|s}} \partial_{v_{G\|}} \left\{ B^*_{\|s} \mathcal{D}_{\|s} \bar{F}_{M_s} \partial_{v_{G\|}} \left(\frac{\bar{F}_s}{\bar{F}_{M_s}} \right) \right\} \\
&= -2\pi \int \mathrm{d}\mu_s\, \mathrm{d}v_{G\|}\, B^*_{\|s} \mathcal{D}_{\|s} \bar{F}_{M_s} \partial_{v_{G\|}} \left(\frac{\bar{F}_s}{\bar{F}_{M_s}} \right) \\
&= 2\pi \int \mathrm{d}\mu_s\, \mathrm{d}v_{G\|}\, \frac{\bar{F}_s}{\bar{F}_{M_s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s} \bar{F}_{M_s}) \\
&= 2\pi \int \mathrm{d}\mu_s\, \mathrm{d}v_{G\|}\, \left\{ \frac{\bar{F}_s}{\bar{F}_{M_s}} B^*_{\|s} \mathcal{D}_{\|s} \partial_{v_{G\|}} \bar{F}_{M_s} + \bar{F}_s \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s}) \right\}
\end{aligned}
$$

Using the fact that $\partial_{v_{G\|}} \bar{F}_{M_s} = -\bar{F}_{M_s} \frac{m_s (v_{G\|} - V_{\|s,\mathrm{coll}})}{T_{s,\mathrm{coll}}}$, we obtain

$$
\begin{aligned}
(\text{F.3})_{\|} &= \frac{2\pi}{T_{s,\mathrm{coll}}} \int B^*_{\|s}\, \mathrm{d}\mu_s\, \mathrm{d}v_{G\|}\, \left\{ \frac{T_{s,\mathrm{coll}} \bar{F}_s}{B^*_{\|s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s}) - \mathcal{D}_{\|s} \bar{F}_s m_s (v_{G\|} - V_{\|s,\mathrm{coll}}) \right\} \\
&= \frac{m_s}{T_{s,\mathrm{coll}}} \left\{ V_{\|s,\mathrm{coll}} \left\langle m_s \mathcal{D}_{\|s} \right\rangle - \left\langle m_s \mathcal{D}_{\|s} v_{G\|} \right\rangle + T_{s,\mathrm{coll}} \left\langle \frac{1}{B^*_{\|s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s}) \right\rangle \right\}
\end{aligned} \tag{F.5}
$$

where

$$
\langle ... \rangle = \int \mathcal{J}_{\mathrm{v}}\, \mathrm{d}\mu_s\, \mathrm{d}v_{G\|}\, \bar{F}_s ... \tag{F.6}
$$

We perform similar operations on the contribution to Eq. (F.4) of collisions in the parallel direction:

$$
\begin{aligned}
(\text{F.4})_{\|} &= \int \mathcal{J}_{\mathrm{v}}\, \mathrm{d}\mu_s\, \mathrm{d}v_{G\|} (\mu_s B + \frac{1}{2} m_s v_{G\|}^2) \frac{1}{B^*_{\|s}} \partial_{v_{G\|}} \left\{ B^*_{\|s} \mathcal{D}_{\|s} \bar{F}_{M_s} \partial_{v_{G\|}} \left(\frac{\bar{F}_s}{\bar{F}_{M_s}} \right) \right\} \\
&= -2\pi \int \mathrm{d}\mu_s\, \mathrm{d}v_{G\|}\, B^*_{\|s} \mathcal{D}_{\|s} \bar{F}_{M_s} v_{G\|} \partial_{v_{G\|}} \left(\frac{\bar{F}_s}{\bar{F}_{M_s}} \right) \\
&= 2\pi \int \mathrm{d}\mu_s\, \mathrm{d}v_{G\|} \frac{\bar{F}_s}{\bar{F}_{M_s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s} v_{G\|} \bar{F}_{M_s}) \\
&= \frac{2\pi}{T_{s,\mathrm{coll}}} \int B^*_{\|s}\, \mathrm{d}\mu_s\, \mathrm{d}v_{G\|} \left\{ -v_{G\|} \mathcal{D}_{\|s} \bar{F}_s m_s (v_{G\|} - V_{\|s,\mathrm{coll}}) + \frac{T_{s,\mathrm{coll}}}{B^*_{\|s}} \bar{F}_s \partial_{v_{G\|}} (B^*_{\|s} v_{G\|} \mathcal{D}_{\|s}) \right\} \\
&= \frac{m_s}{T_{s,\mathrm{coll}}} \left\{ V_{\|s,\mathrm{coll}} \left\langle m_s \mathcal{D}_{\|s} v_{G\|} \right\rangle - \left\langle m_s \mathcal{D}_{\|s} v_{G\|}^2 \right\rangle + T_{s,\mathrm{coll}} \left\langle \frac{1}{B^*_{\|s}} \partial_{v_{G\|}} (B^*_{\|s} \mathcal{D}_{\|s} v_{G\|}) \right\rangle \right\}
\end{aligned} \tag{F.7}
$$

For collisions in the perpendicular direction, Eq. (F.3) is trivially verified. We perform two integrations by parts in the variable μ_s for Eq. (F.4)

$$
\begin{aligned}
(\text{F.4})_{\perp} &= \int \mathcal{J}_{\mathrm{v}}\, \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s\, (\mu_s B + \frac{1}{2} m_s v_{G\|}^2) \frac{1}{B^*_{\|s}} \partial_{\mu} \left\{ B^*_{\|s} \mathcal{D}_{\perp s} \bar{F}_{M_s} \frac{1}{B^2} \partial_{\mu} \left(\frac{\bar{F}_s}{\bar{F}_{M_s}} \right) \right\} \\
&= -\frac{2\pi}{m_s} \int \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s\, \frac{B^*_{\|s}}{B} \mathcal{D}_{\perp s} \bar{F}_{M_s} \partial_{\mu} \left(\frac{\bar{F}_s}{\bar{F}_{M_s}} \right) \\
&= \frac{2\pi}{m_s} \int \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s \frac{\bar{F}_s}{\bar{F}_{M_s}} \frac{1}{B} \partial_{\mu} (B^*_{\|s} \mathcal{D}_{\perp s} \bar{F}_{M_s})
\end{aligned}
$$

Using the fact that $\partial_{\mu} \bar{F}_{M_s} = -\bar{F}_{M_s} \frac{B}{T_{s,\mathrm{coll}}}$, we obtain

$$
\begin{aligned}
(\text{F.4})_{\perp} &= \frac{2\pi}{m_s} \int \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s\, \left\{ \frac{\bar{F}_s}{B} \partial_{\mu} (B^*_{\|s} \mathcal{D}_{\perp s}) - B^*_{\|s} \mathcal{D}_{\perp s} \frac{1}{T_{s,\mathrm{coll}}} \bar{F}_s \right\} \\
&= \frac{1}{T_{s,\mathrm{coll}}} \left\{ \left\langle \frac{1}{B B^*_{\|s}} \partial_{\mu} (B^*_{\|s} \mathcal{D}_{\perp s}) \right\rangle T_{s,\mathrm{coll}} - \langle \mathcal{D}_{\perp s} \rangle \right\}
\end{aligned} \tag{F.8}
$$

Using Equations (F.5), (F.7) and (F.8), the conservation equations (F.3) and (F.4) form a linear system in $T_{s,\mathrm{coll}}(r,\theta,\varphi)$ and $V_{\|s,\mathrm{coll}}(r,\theta,\varphi)$ as follows

$$\begin{aligned} V_{\|s,\mathrm{coll}}\left\langle m_s\mathcal{D}_{\|s}\right\rangle + T_{s,\mathrm{coll}}\left\langle \frac{1}{B^*_{\|s}}\partial_{v_{G\|}}(B^*_{\|s}\mathcal{D}_{\|s})\right\rangle &= \left\langle m_s\mathcal{D}_{\|s}v_{G\|}\right\rangle \\ V_{\|s,\mathrm{coll}}\left\langle m_s^2\mathcal{D}_{\|s}v_{G\|}\right\rangle + T_{s,\mathrm{coll}}\left\langle \frac{m_s}{B^*_{\|s}}\partial_{v_{G\|}}(B^*_{\|s}\mathcal{D}_{\|s}v_{G\|}) + \frac{1}{BB^*_{\|s}}\partial_\mu(B^*_{\|s}\mathcal{D}_{\perp s})\right\rangle &= \left\langle m_s^2\mathcal{D}_{\|s}v_{G\|}^2 + \mathcal{D}_{\perp s}\right\rangle \end{aligned}$$

Solving this system, we find that the conservation constraints are verified if the profiles $V_{\|s,\mathrm{coll}}(r,\theta,\varphi)$ and $T_{s,\mathrm{coll}}(r,\theta,\varphi)$ for the collision operator are defined as follows

$$\begin{aligned} m_sPV_{\|s,\mathrm{coll}} &= \left\langle \frac{m_s}{B^*_{\|s}}\partial_{v_{G\|}}(B^*_{\|s}\mathcal{D}_{\|s}v_{G\|}) + \frac{1}{BB^*_{\|s}}\partial_\mu(B^*_{\|s}\mathcal{D}_{\perp s})\right\rangle\left\langle m_s\mathcal{D}_{\|s}v_{G\|}\right\rangle \\ &\quad - \left\langle \frac{1}{B^*_{\|s}}\partial_{v_{G\|}}(B^*_{\|s}\mathcal{D}_{\|s})\right\rangle\left\langle m_s^2\mathcal{D}_{\|s}v_{G\|}^2 + \mathcal{D}_{\perp s}\right\rangle \\ PT_{s,\mathrm{coll}} &= \left\langle \mathcal{D}_{\|s}\right\rangle\left\langle m_s^2\mathcal{D}_{\|s}v_{G\|}^2 + \mathcal{D}_{\perp s}\right\rangle - \left\langle m_s\mathcal{D}_{\|s}v_{G\|}\right\rangle^2 \end{aligned}$$

where

$$P = \left\langle \mathcal{D}_{\|s}\right\rangle\left\langle \frac{m_s}{B^*_{\|s}}\partial_{v_{G\|}}(B^*_{\|s}\mathcal{D}_{\|s}v_{G\|}) + \frac{1}{BB^*_{\|s}}\partial_\mu(B^*_{\|s}\mathcal{D}_{\perp s})\right\rangle - \left\langle m_s\mathcal{D}_{\|s}v_{G\|}\right\rangle\left\langle \frac{1}{B^*_{\|s}}\partial_{v_{G\|}}(B^*_{\|s}\mathcal{D}_{\|s})\right\rangle$$

We recall that

$$\langle ...\rangle = \int \mathcal{J}_\mathrm{v}\,\mathrm{d}\mu_s\,\mathrm{d}v_{G\|}\bar{F}_s... \quad \text{with} \quad \mathcal{J}_\mathrm{v} = 2\pi B^*_{\|s}/m_s$$

Let us remind that only collisions in the parallel direction are taken into account at the moment in the code, i.e $\mathcal{D}_{\perp s} = 0$ in above expressions. Then, let us define the 5 following integrals:

$$\begin{aligned} &I_0 = \langle\mathcal{D}_{\|s}\rangle \quad ; \quad I_1 = \langle m_s\mathcal{D}_{\|s}v_{G\|}\rangle \quad ; \quad I_2 = \langle m_s^2\mathcal{D}_{\|s}v_{G\|}^2\rangle \\ &I_3 = \langle\frac{1}{B^*_{\|s}}\,\partial_{v_{G\|}}\left(B^*_{\|s}\,\mathcal{D}_{\|s}\right)\rangle \quad \text{and} \quad I_4 = \langle\frac{m_s}{B^*_{\|s}}\,\partial_{v_{G\|}}\left(B^*_{\|s}\,v_{G\|}\,\mathcal{D}_{\|s}\right)\rangle \end{aligned}$$

Therefore the mean velocity and mean temperature can be simply expressed as:

$$\begin{aligned} m_sV_{\|s,\mathrm{coll}} &= P^{-1}\left(I_4\times I_1 - I_2\times I_3\right) \\ T_{s,\mathrm{coll}} &= P^{-1}\left(I_0\times I_2 - I_1^2\right) \\ P &= I_0\times I_4 - I_1\times I_3 \end{aligned}$$

F.2 Crank-Nicolson scheme for collision operator solving

In the following, the semi-implicit second-order Crank-Nicolson scheme which is implemented in GYSELA to take into account the collisional effects as

$$\partial_t\bar{F}_s = \mathcal{C}_{ss}(\bar{F}_s) = \frac{1}{B^*_{\|s}}\partial_{v_{G\|}}\left\{B^*_{\|s}\mathcal{D}\partial_{v_{G\|}}\bar{F}_s - B^*_{\|s}\mathcal{V}\bar{F}_s\right\} \tag{F.9}$$

is detailed. The diffusion term $\mathcal{D}$ is defined as $\mathcal{D} = \mathcal{D}_{\|s}(r,v) = \mathcal{A}(r)\left(\frac{\Phi(v)-G(v)}{2v}\right)$ with $\mathcal{A}(r) = 3\frac{\sqrt{\pi}}{2}\frac{v_{T_{s0}}^3\epsilon^{3/2}}{qR_0}\nu_*$ while the expression of the drag term $\mathcal{V}$ is given by $\mathcal{V} = \mathcal{V}_{\|s}(r,v) = -\frac{\left(v_{G\|}-V_{\|s,\mathrm{coll}}\right)}{v_{T_{s,\mathrm{coll}}}^2}\mathcal{D}_{\|s}(r,v)$.

Classically, let us write: $f_j^n \equiv \bar{F}_s(r,\theta,\varphi,v_{G\|,j},t_n)$, where subscript $j \in \{0,\dots,N\}$ refers to the discretised index in parallel velocity space and superscript n refers to the time. For each value of μ_s, each term in Eq. (F.9) reads:

$$B^*_{\|s}\partial_t f \to B^*_{\|s,j}\frac{f_j^{n+1}-f_j^n}{\Delta t} \tag{F.10}$$

$$\partial_{v_{G\|}}\left(B^*_{\|s}\mathcal{D}\partial_{v_{G\|}}f\right) \to \frac{1}{2\Delta v_{G\|}^2}\left\{B^*_{\|s,j+\frac{1}{2}}\mathcal{D}_{j+\frac{1}{2}}\left[f_{j+1}^{n+1}-f_j^{n+1}+f_{j+1}^n-f_j^n\right]\right.$$
$$\left.-B^*_{\|s,j-\frac{1}{2}}\mathcal{D}_{j-\frac{1}{2}}\left[f_j^{n+1}-f_{j-1}^{n+1}+f_j^n-f_{j-1}^n\right]\right\} \tag{F.11}$$

$$\partial_{v_{G\|}}\left(B^*_{\|s}\mathcal{V}f\right) \to \frac{B^*_{\|s,j+1}\mathcal{V}_{j+1}\left[f_{j+1}^{n+1}+f_{j+1}^n\right]-B^*_{\|s,j-1}\mathcal{V}_{j-1}\left[f_{j-1}^{n+1}+f_{j-1}^n\right]}{4\Delta v_{G\|}} \tag{F.12}$$

where we use the fact that $B^*_{\|s}$ is linear in $v_{G\|}$, i.e. the expression $B^*_{\|s,j\pm\frac{1}{2}} = \frac{1}{2}\left(B^*_{\|s,j}+B^*_{\|s,j\pm1}\right)$ is exact. Then, using expressions (F.10), (F.11) and (F.12) in the equation (F.9) gives, for all $j = 0,\cdots,N$:

$$A_j f_{j-1}^{n+1} + B_j f_j^{n+1} + C_j f_{j+1}^{n+1} = -A_j f_{j-1}^n + \left(\frac{2B^*_{\|s,j}}{\Delta t} - B_j\right) f_j^n - C_j f_{j+1}^n \tag{F.13}$$

where the coefficients A_j, B_j and C_j are defined as:

$$A_j = -\frac{\alpha_{j-\frac{1}{2}}}{2\Delta v_{G\|}^2} - \frac{\beta_{j-1}}{4\Delta v_{G\|}} \tag{F.14}$$

$$B_j = \frac{B^*_{\|s,j}}{\Delta t} + \frac{\alpha_{j+\frac{1}{2}}+\alpha_{j-\frac{1}{2}}}{2\Delta v_{G\|}^2} \tag{F.15}$$

$$C_j = -\frac{\alpha_{j+\frac{1}{2}}}{2\Delta v_{G\|}^2} + \frac{\beta_{j+1}}{4\Delta v_{G\|}} \tag{F.16}$$

with

$$\alpha_j = B^*_{\|s,j}\mathcal{D}_j \quad \text{and} \quad \beta_j = B^*_{\|s,j}\mathcal{V}_j \tag{F.17}$$

To compute the terms $\alpha_{-1/2}$, β_{-1}, $\alpha_{N+1/2}$ and β_{N+1} required for A_0, B_0, B_N and C_N calculations, the boundary conditions $\partial^2\mathcal{D}/\partial v_{G\|}^2 = 0$ and $\partial^2\mathcal{V}/\partial v_{G\|}^2 = 0$ are imposed, i.e :

$$\mathcal{D}_{-\frac{1}{2}} = 2\mathcal{D}_0 - \mathcal{D}_{\frac{1}{2}} \quad \text{and} \quad \mathcal{V}_{-1} = 2\mathcal{V}_0 - \mathcal{V}_1 \tag{F.18}$$

$$\mathcal{D}_{N+\frac{1}{2}} = 2\mathcal{D}_N - \mathcal{D}_{N-\frac{1}{2}} \text{ and } \mathcal{V}_{N+1} = 2\mathcal{V}_N - \mathcal{V}_{N-1} \tag{F.19}$$

and the fact that $B^*_{\|}$ is linear in $v_{G\|}$ is used, i.e [1]:

$$B^*_{\|s,-\frac{1}{2}} = \frac{1}{2}\left(3B^*_{\|s,0} - B^*_{\|s,\frac{1}{2}}\right) \quad ; \quad B^*_{\|s,-1} = 2B^*_{\|s,0} - B^*_{\|s,1} \tag{F.20}$$

$$B^*_{\|s,N+\frac{1}{2}} = \frac{1}{2}\left(3B^*_{\|s,N} - B^*_{\|s,N-\frac{1}{2}}\right) \quad ; \quad B^*_{\|s,N+1} = 2B^*_{\|s,N} - B^*_{\|s,N-1} \tag{F.21}$$

Finally, let us consider that the second derivative of f is vanishing at domain boundaries, i.e

$$f_{-1} = 2f_0 - f_1 \quad \text{and} \quad f_{N+1} = 2f_N - f_{N-1}$$

then

$$(B_0+2A_0)f_0^{n+1} + (C_0-A_0)f_1^{n+1} = \left(\frac{2B^*_{\|s,0}}{\Delta t} - B_0 - 2A_0\right) f_0^n - (C_0-A_0)f_1^n$$

$$(A_N-C_N)f_{N-1}^{n+1} + (B_N+2A_N)f_N^{n+1} = -(A_N-C_N)f_{N-1}^n + \left(\frac{2B^*_{\|s,N}}{\Delta t} - B_N - 2C_N\right) f_N^n$$

[1]

$$B^*_{\|s,-\frac{1}{2}} = 2B^*_{\|s,0} - B^*_{\|s,\frac{1}{2}} = 2B^*_{\|s,0} - \frac{1}{2}\left(B^*_{\|s,0}+B^*_{\|s,1}\right) = \frac{3}{2}B^*_{\|s,0} - \frac{1}{2}B^*_{\|s,1}$$

The system can be rewritten in the compact tridiagonal form:

$$\begin{pmatrix} B_0 + 2A_0 & C_0 - A_0 & & \\ A_1 & B_1 & \ddots & \\ & \ddots & \ddots & C_{N-1} \\ & & A_N - C_N & B_N + 2A_N \end{pmatrix} \begin{pmatrix} f_0^{n+1} \\ \vdots \\ \vdots \\ f_N^{n+1} \end{pmatrix} = \begin{pmatrix} R_0^n \\ \vdots \\ \vdots \\ R_N^n \end{pmatrix} \tag{F.22}$$

and $(R_0^n, \cdots, R_N^n)^t = \mathcal{R}\,(f_0^n, \cdots, f_N^n)^t$ with the matrix $\mathcal{R}$ defined as

$$\mathcal{R} = \begin{pmatrix} -2A_0 - B_0 + \frac{2B^*_{\|s,0}}{\Delta t} & -(C_0 - A_0) & & \\ -A_1 & -B_1 + \frac{2B^*_{\|s,1}}{\Delta t} & \ddots & \\ & \ddots & \ddots & -C_{N-1} \\ & & -(A_N - C_N) & -2C_N - B_N + \frac{2B^*_{\|s,N}}{\Delta t} \end{pmatrix} \tag{F.23}$$

The tridiagonal system (F.22)-(F.23) is solved by using a modified Thomas algorithm (see Appendix E).

Appendix G

Expressions for simplified collisional transfer between two species

In previous appendix F, we have described the intra-species collision operator. With the capability to consider not only an unique species but also impurities, it was primordial to take into account inter-species collisions. We describe in this appendix the first inter-species collision operator implemented in the code. It corresponds to a highly simplified version which ensures both conservative momentum and energy transfers between species by the sum of two linearized operators described below.

G.1 Conservation properties of collisional energy transfer

Let us consider the energy exchange between two species defined by equations (2.36)-(2.41), *i.e.* $\mathrm{d}\bar{F}_s/\,\mathrm{d}t = C^{\mathcal{E}}_{ss'}(\mathcal{E}_{ss'})$ and $\mathrm{d}\bar{F}_{s'}/\,\mathrm{d}t = C^{\mathcal{E}}_{s's}(\mathcal{E}_{s's})$ with

$$C^{\mathcal{E}}_{ss'}(\mathcal{E}_{ss'}) \equiv -\frac{\eta^{\mathcal{E}}_{ss'}}{(2\pi T_{\mathrm{mean}}/m_s)^{3/2}}\frac{\Delta T_{ss'}}{T_{\mathrm{mean}}}\left(\frac{\mathcal{E}_{ss'}}{T_{\mathrm{mean}}}-\frac{3}{2}\right)\exp\left(-\frac{\mathcal{E}_{ss'}}{T_{\mathrm{mean}}}\right)$$
$$C^{\mathcal{E}}_{s's}(\mathcal{E}_{s's}) \equiv -\frac{\eta^{\mathcal{E}}_{ss'}}{(2\pi T_{\mathrm{mean}}/m_{s'})^{3/2}}\frac{\Delta T_{s's}}{T_{\mathrm{mean}}}\left(\frac{\mathcal{E}_{s's}}{T_{\mathrm{mean}}}-\frac{3}{2}\right)\exp\left(-\frac{\mathcal{E}_{s's}}{T_{\mathrm{mean}}}\right)$$

with $\mathcal{E}_{ss'}$, $\mathcal{E}_{s's}$ defined by equation (2.40) and T_{mean}, $\Delta T_{ss'}$ defined by equation (2.38). Let us consider the brackets $\langle\cdot\rangle_s$ defined as

$$\langle\,\cdot\,\rangle_{v,T} = \frac{1}{(2\pi T/m_s)^{3/2}}\int\int\cdot\exp\left(-\frac{v^2}{2T}-\frac{\mu B}{T}\right)\frac{2\pi}{m_s}B^*_{\parallel s}\,\mathrm{d}\mu_s\,\mathrm{d}v \tag{G.1}$$

Let us define $\bar{v}_\parallel = v_{G\parallel} - V_{\mathrm{mean}}$. Then, using the fact that $\langle\,1\,\rangle_{\bar{v}_\parallel,T_{\mathrm{mean}}} = 1$ and $\left\langle\, m_s\bar{v}_\parallel^2\,\right\rangle_{\bar{v}_\parallel,T_{\mathrm{mean}}} = \langle\,\mu_s B\,\rangle_{\bar{v}_\parallel,T_{\mathrm{mean}}} = T_{\mathrm{mean}}$, it is straightforward to show that such a collision operator conserves both particles and parallel momentum, *i.e* that

$$\left\langle C^{\mathcal{E}}_{ss'}(\mathcal{E}_s)\right\rangle = \left\langle C^{\mathcal{E}}_{s's}(\mathcal{E}_{s'})\right\rangle = 0 \quad\text{and}\quad \left\langle v_{G\parallel}C^{\mathcal{E}}_{ss'}(\mathcal{E}_s)\right\rangle + \left\langle v_{G\parallel}C^{\mathcal{E}}_{s's}(\mathcal{E}_{s'})\right\rangle = 0 \quad\text{where}\ \langle\cdot\rangle = \int\cdot\,\mathrm{d}^3\mathbf{v}$$

The parameters $\eta^{\mathcal{E}}_{ss'}$ and $\eta^{\mathcal{E}}_{s's}$ are designed such that the collisional energy exchange between species satisfy $\left\langle\mathcal{E}_s\,C^{\mathcal{E}}_{ss'}(\mathcal{E}_s)\right\rangle = -\left\langle\mathcal{E}_{s'}\,C^{\mathcal{E}}_{s's}(\mathcal{E}_{s'})\right\rangle$. So let us compute the following integral in velocity space,

$$\left\langle\mathcal{E}_s\,C^{\mathcal{E}}_{ss'}\right\rangle = \int\mathcal{E}_s\,C^{\mathcal{E}}_{ss'}\,\mathrm{d}^3\mathbf{v} = -\eta^{\mathcal{E}}_{ss'}\frac{\Delta T_{ss'}}{T_{\mathrm{mean}}}\left\langle\,\mathcal{E}_s\left(\frac{\mathcal{E}_{ss'}}{T_{\mathrm{mean}}}-\frac{3}{2}\right)\right\rangle_{\bar{v}_\parallel,T_{\mathrm{mean}}}$$

which is equivalent to $\langle \mathcal{E}_s\, C^{\mathcal{E}}_{ss'} \rangle = -\eta^{\mathcal{E}}_{ss'}\,(\Delta T_{ss'}/T_{\text{mean}})\,\mathcal{I}$, where

$$\begin{aligned}
\mathcal{I} &= \left\langle \left(m_s \frac{\bar{v}_\parallel^2}{2} + m_s \frac{\left(V_{\parallel s}^2 - V_{\text{mean}}^2\right)}{2} - m_s\left(\bar{v}_\parallel + V_{\text{mean}}\right)\frac{\Delta V_{ss'}}{2} + \mu_s B \right) \left(\frac{1}{2} m_s \frac{\bar{v}_\parallel^2}{T_{\text{mean}}} + \frac{\mu_s B}{T_{\text{mean}}} - \frac{3}{2} \right) \right\rangle_{\bar{v}_\parallel, T_{\text{mean}}} \\
&= \left\langle \left(m_s \frac{\bar{v}_\parallel^2}{2} - m_s \frac{\Delta V_{ss'}}{2}\bar{v}_\parallel + m_s \frac{\Delta V_{ss'}^2}{8} + \mu_s B \right) \left(\frac{1}{2} m_s \frac{\bar{v}_\parallel^2}{T_{\text{mean}}} + \frac{\mu_s B}{T_{\text{mean}}} - \frac{3}{2} \right) \right\rangle_{\bar{v}_\parallel, T_{\text{mean}}} \\
&= \left\langle \frac{m_s^2 \bar{v}_\parallel^4}{4T_{\text{mean}}} + \left(m_s \frac{\bar{v}_\parallel^2}{2} + \mu_s B \right) \left(m_s \frac{\Delta V_{s's}^2}{8T_{\text{mean}}} - \frac{3}{2} \right) + \frac{m_s \bar{v}_\parallel^2 \mu_s B}{T_{\text{mean}}} + \frac{\mu_s^2 B^2}{T_{\text{mean}}} - m_s \Delta V_{ss'} \frac{3}{16} \right\rangle_{\bar{v}_\parallel, T_{\text{mean}}}
\end{aligned}$$

Then, using the properties, $\left\langle m_s^2 \bar{v}_\parallel^4 \right\rangle_{\bar{v}_\parallel, T_{\text{mean}}} = 3T_{\text{mean}}^2$ and $\left\langle \mu_s^2 B^2 \right\rangle_{\bar{v}_\parallel, T_{\text{mean}}} = 2T_{\text{mean}}^2$ and $\left\langle m_s \bar{v}_\parallel^2 \mu_s B \right\rangle_{\bar{v}_\parallel, T_{\text{mean}}} = T_{\text{mean}}^2$, the collisional energy exchange between species occurs at the following rate

$$\langle \mathcal{E}_s\, C^{\mathcal{E}}_{ss'} \rangle = -\eta^{\mathcal{E}}_{ss'}\, \frac{3\Delta T_{ss'}}{2} \left(1 - \frac{\Delta V_{ss'}{}^2}{8T_{\text{mean}}} \right) \tag{G.2}$$

Let us use the property that the energy exchange term between two species is of the form (cf. [HM03], p.184)

$$W_{ss'} = -\frac{4}{\sqrt{\pi}} \frac{\gamma_{ss'}}{m_{s'}} \frac{n_s n_{s'}(T_s - T_{s'})}{\left(v_{Ts}^2 + v_{Ts'}^2\right)^{3/2}} \equiv \int \mathrm{d}^3\mathbf{v} \frac{1}{2} m_s v^2 C_{ss'} \quad \text{with} \quad \gamma_{ss'} \sim \frac{m_s v_{T>} v_{T_s}^2}{n_{s'}} \nu_{ss'}$$

where $v_{T>} = \max(v_{T_s}, v_{T'_s})$ and the collision frequency $\nu_{ss'}$ is deduced from the ion-ion collision frequency ν_{ss} as

$$\nu_{ss'} = \left(\frac{Z_{s'}}{Z_s} \right)^2 \frac{n_{s'}}{n_s} \frac{v_{T_s}}{v_{T>}} \nu_{ss} \quad \text{and} \quad \nu_{ss} = \frac{\varepsilon^{3/2}}{qR_0} \left(\frac{T_{s,\text{coll}}}{m_s} \right)^{1/2} \nu_{*s} \tag{G.3}$$

Therefore

$$W_{ss'} = -\frac{4}{\sqrt{\pi}} \nu_{ss'} \frac{v_{T>}}{v_{T_s}} n_s \frac{m_s}{m_{s'}} \Delta T_{ss'} \left(1 + \frac{v_{T'_s}^2}{v_{T_s}^2} \right)^{-3/2} \tag{G.4}$$

Finally, by analogy between (G.2) and eq. (G.4) (considering $\Delta V_{ss'}^2/8T_{\text{mean}} \ll 1$), this provides for the effective collision frequency $\eta^{\mathcal{E}}_{ss'}$ the following expression,

$$\eta^{\mathcal{E}}_{ss'} = \frac{8\,\varepsilon^{3/2}}{3\sqrt{\pi}}\, n_{s'}\, \frac{m_s}{m_{s'}} \left(\frac{Z_{s'}}{Z_s} \right)^2 \left(1 + \frac{v_{T'_s}^2}{v_{T_s}^2} \right)^{-3/2} \frac{\sqrt{T_{s,\text{coll}}/m_s}}{qR_0} \nu_{*s}$$

which is equivalent to the one given by equation (2.41).

G.2 Conservation properties of collisional parallel momentum transfer

Let us consider the parallel momentum transfer defined by equations (2.42)-(2.44), *i.e* $\mathrm{d}\bar{F}_s/\,\mathrm{d}t = C^{v_{G\parallel}}_{ss'}(E_s)$ and $\mathrm{d}\bar{F}_{s'}/\,\mathrm{d}t = C^{v_{G\parallel}}_{s's}(E_{s'})$ with

$$C^{v_{G\parallel}}_{ss'}(E_s) \equiv -\eta^{v_{G\parallel}}_{ss'} \Delta V_{ss'} v_{G\parallel} \exp\left(-\frac{E_s}{T_{\text{mean}}} \right) \quad \text{and} \quad C^{v_{G\parallel}}_{s's}(E_{s'}) \equiv -\eta^{v_{G\parallel}}_{s's} \Delta V_{s's} v_{G\parallel} \exp\left(-\frac{E_{s'}}{T_{\text{mean}}} \right)$$

Considering this approximation, the momentum exchanges between two species reads [1]

$$\left\langle m_s v_{G\parallel} C^{v_{G\parallel}}_{ss'}(E_s) \right\rangle = \int \mathrm{d}^3\mathbf{v}\; m_s v_{G\parallel} \frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = -\eta^{v_{G\parallel}}_{ss'} \Delta V_{ss'} \left(\frac{2\pi}{m_s} \right)^{3/2} T_{\text{mean}}^{5/2} \tag{G.5}$$

$$\left\langle m_s v_{G\parallel} C^{v_{G\parallel}}_{s's}(E_{s'}) \right\rangle = \int \mathrm{d}^3\mathbf{v}\; m_{s'} v_{G\parallel} \frac{\mathrm{d}\bar{F}_{s'}}{\mathrm{d}t} - \eta^{v_{G\parallel}}_{s's} \Delta V_{s's} \left(\frac{2\pi}{m_{s'}} \right)^{3/2} T_{\text{mean}}^{5/2} \tag{G.6}$$

[1] $\left\langle m_s v_{G\parallel} C^{v_{G\parallel}}_{ss'} \right\rangle = -\eta^{v_{G\parallel}}_{ss'} \Delta V_{ss'} \left(\frac{2\pi T_{\text{mean}}}{m_s} \right)^{3/2} \left\langle m_s v_{G\parallel}^2 \right\rangle_{v_{G\parallel}, T_{\text{mean}}} = -\eta^{v_{G\parallel}}_{ss'} \Delta V_{ss'} \left(\frac{2\pi T_{\text{mean}}}{m_s} \right)^{3/2} T_{\text{mean}}$

Therefore, the action-reaction principle $\left\langle m_s v_{G\parallel} C_{ss'}^{v_{G\parallel}} \right\rangle = -\left\langle m_s v_{G\parallel} C_{s's}^{v_{G\parallel}} \right\rangle$, leads to the first constraint $\eta_{s's}^{v_{G\parallel}} = (m_{s'}/m_s)^{3/2} \eta_{ss'}^{v_{G\parallel}}$. The second constraint comes from the neoclassical friction relation,

$$-\int \mathrm{d}^3\mathbf{v}\; m_{s'} v_{G\parallel} \frac{\mathrm{d}\bar{F}_{s'}}{\mathrm{d}t} = m_s n_s \nu_{ss'} \Delta V_{ss'} = -m_{s'} n_{s'} \nu_{s's} \Delta V_{s's}$$

then using equation (G.5) leads to

$$\eta_{ss'}^{v_{G\parallel}} = m_s n_s \left(\frac{2\pi}{m_s}\right)^{-3/2} T_{\mathrm{mean}}^{-5/2} \nu_{ss'}$$

Finally, using the relation (G.3) for $\nu_{ss'}$ expression, $\eta_{ss'}^{v_{G\parallel}}$ can be expressed as

$$\eta_{ss'}^{v_{G\parallel}} = n_{s'} m_s^2 \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{qR_0} \frac{\sqrt{T_{s,\mathrm{coll}}}}{T_{\mathrm{mean}}^{5/2}} \left(\frac{Z_{s'}}{Z_s}\right)^2 \frac{v_{T_s}}{v_{T>}} \nu_{*s}$$

which is equivalent to equation (2.44).

Appendix H

Source terms

In this appendix, we first detail the construction of the prescribed source developed in GYSELA to drive the system. As described in the following this source consists of the sum of the product of Hermite and Laguerre polynomials in $v_{G\|}$ and μ. It is versatile enough to allow for separate injection of heat, parallel momentum and vorticity. In a second part, we present the numerical time scheme used in the global algorithm of the code. Let us focus on the time evolution of the distribution function $\bar{F}_s$ due to the source term S, i.e

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = S \tag{H.1}$$

This source is defined as the product of two terms as

$$S \equiv S_E(r, \theta, v_{G\|}, \mu_s) S_r(r) \tag{H.2}$$

The prescribed radial source profile $S_r(r)$ is defined as the sum of two hyperbolic tangents

$$S_r(r) = -\frac{1}{2}\left[\tanh\left(\frac{\rho - \rho_S - 3L_S}{L_S}\right) + \tanh\left(\frac{\rho_S - 3L_S - \rho}{L_S}\right)\right] \tag{H.3}$$

where ρ_S and L_S are input data and $\rho = (r - r_{\min})/L_r$ where L_r is the length of the radial box. The energy dependent part of the source is decomposed on the basis of orthogonal Hermite and Laguerre poynomials (cf. next section H.1 as a reminder):

$$S_E(r, \theta, v_{G\|}, \mu_s) = \sum_{\ell=0}^{+\infty}\sum_{h=0}^{+\infty} c_{h\ell} H_h(\bar{v}_{G_\| s}) L_\ell(\bar{\mu}_s) \mathrm{e}^{-\bar{v}^2_{G_\| s} - \bar{\mu}_s} \tag{H.4}$$

where the $c_{h\ell}$ coefficients depend on the space coordinates only. The following definitions have been introduced:

$$\bar{\mu}_s \equiv \frac{\mu_s B}{T_{s,\mathrm{srce}}} \quad ; \quad \bar{v}_{G_\| s} \equiv \frac{v_{G\|}}{\sqrt{2T_{s,\mathrm{srce}}/m_s}} \tag{H.5}$$

with $T_{s,\mathrm{srce}}$ the normalized source temperature.

H.1 Hermite and Laguerre polynomials

The Hermite and Laguerre poynomials form the set of orthogonal basis with respect to the following scalar products:

$$\text{Laguerre } L_\ell(x): \quad \int_0^{+\infty} L_\ell L_{\ell'} \mathrm{e}^{-x}\, \mathrm{d}x = \delta_{\ell\ell'} |L_\ell|^2 \tag{H.6}$$

$$\text{Hermite } H_h(x): \quad \int_{-\infty}^{+\infty} H_h H_{h'} \mathrm{e}^{-x^2}\, \mathrm{d}x = \delta_{hh'} |H_h|^2 \tag{H.7}$$

The Laguerre polynomials are normalized: $|L_\ell|^2 = 1$. The norm of the Hermite polynomials is:

$$|H_h|^2 \equiv \int_{-\infty}^{+\infty} H_h^2 \, \mathrm{e}^{-x^2} \, \mathrm{d}x = \sqrt{\pi} \, 2^h h! \tag{H.8}$$

The five first Laguerre and Hermite polynomials are:

$$\begin{aligned} L_0(x) &= 1 \\ L_1(x) &= 1 - x \\ L_2(x) &= \tfrac{1}{2}(2 - 4x + x^2) \\ L_3(x) &= \tfrac{1}{6}(6 - 18x + 9x^2 - x^3) \\ L_4(x) &= \tfrac{1}{24}(24 - 96x + 72x^2 - 16x^3 + x^4) \end{aligned} \tag{H.9}$$

and

$$\begin{aligned} H_0(x) = 1 \quad &\rightarrow \quad |H_0|^2 = \sqrt{\pi} \\ H_1(x) = 2x \quad &\rightarrow \quad |H_1|^2 = 2\sqrt{\pi} \\ H_2(x) = -2 + 4x^2 \quad &\rightarrow \quad |H_2|^2 = 8\sqrt{\pi} \\ H_3(x) = -12x + 8x^3 \quad &\rightarrow \quad |H_3|^2 = 48\sqrt{\pi} \\ H_4(x) = 12 - 48x^2 + 16x^4 \quad &\rightarrow \quad |H_4|^2 = 384\sqrt{\pi} \end{aligned} \tag{H.10}$$

H.2 Corresponding sources for fluid moments

Let's derive the corresponding source of matter $\mathbb{S}_n$, of parallel momentum $\mathbb{S}_{v_{G\|}}$, of energy $\mathbb{S}_\mathcal{E}$ and of vorticity $\mathbb{S}_\Omega$. With the adopted definitions,

$$B^*_{\|s} = B(1 + J_{\|B}\bar{v}_{G_\| s}) \quad \text{and} \quad J_{\|B} \equiv \frac{\sqrt{m_s}}{q_s}\frac{\sqrt{2T_{s,\mathrm{srce}}}}{B^2}\,\mu_0 J_\| \tag{H.11}$$

The parallel current $\mu_0 J_\| = \mu_0 \mathbf{b}.\mathbf{J}$ is prescribed in GYSELA. Then, the integral over the velocity space reads as follows:

$$\begin{aligned} \int \mathrm{d}^3\mathbf{v} \quad &\equiv \quad \int_{-\infty}^{+\infty} \mathrm{d}v_{G\|} \int_0^{+\infty} \frac{2\pi B^*_{\|s}}{m_s} \, \mathrm{d}\mu_s \\ &= \quad \frac{1}{\sqrt{\pi}}\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} \int_{-\infty}^{+\infty} (1 + J_{\|B}\bar{v}_{G_\| s}) \; \mathrm{d}\bar{v}_{G_\| s} \int_0^{+\infty} \mathrm{d}\bar{\mu}_s \end{aligned}$$

Notice that $(1 + J_{\|B}\bar{v}_{G_\| s}) = H_0(\bar{v}_{G_\| s}) + \frac{J_{\|B}}{2} H_1(\bar{v}_{G_\| s})$ and $L_0(\bar{\mu}_s) = 1$.

H.2.1 Source of matter (guiding-centers)

The fluid source of matter $\mathbb{S}_n$ is simply $\mathbb{S}_n \equiv \int \mathrm{d}^3\mathbf{v} \; S_E S_r$. Using the decomposition of S_E on the basis of orthogonal polynomials (H.4), the source of matter becomes [1] :

$$\mathbb{S}_n = \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} \left(c_{00} + J_{\|B} \; c_{10}\right) S_r \tag{H.12}$$

[1] Let us define $\beta \equiv \frac{1}{\sqrt{\pi}}\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2}$, then the integral of $S_E S_r$ in velocity space is equal to

$$\int \mathrm{d}^3\mathbf{v} \; S_E S_r \quad = \quad \beta \int_{-\infty}^{+\infty} \mathrm{d}\bar{v}_{G_\| s} \int_0^{+\infty} \mathrm{d}\bar{\mu}_s \left[L_0(\bar{\mu}_s)\left(H_0(\bar{v}_{G_\| s}) + \frac{J_{\|B}}{2} H_1(\bar{v}_{G_\| s})\right) \sum_{\ell=0}^{+\infty}\sum_{h=0}^{+\infty} c_{h\ell} H_h(\bar{v}_{G_\| s}) L_\ell(\bar{\mu}_s) \mathrm{e}^{-\bar{v}^2_{G_\| s} - \bar{\mu}_s} \right]$$

which gives, according to Laguerre and Hermite properties

$$\begin{aligned} \int \mathrm{d}^3\mathbf{v} \; S_E S_r \quad &= \quad \beta \, L_0^2(\bar{\mu}_s) \left(c_{00} H_0^2(\bar{v}_{G_\| s}) + c_{10}\frac{J_{\|B}}{2} H_1^2(\bar{v}_{G_\| s})\right) \; S_r \\ &= \quad \beta \left(c_{00}\sqrt{\pi} + c_{10}\frac{J_{\|B}}{2}(2\sqrt{\pi})\right) = \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} \left(c_{00} + c_{10}\frac{J_{\|B}}{2}\right) \; S_r \end{aligned}$$

H.2.2 Source of parallel momentum

The fluid source of parallel momentum $\mathbb{S}_{v_{G\|}}$ reads as follows: $\mathbb{S}_{v_{G\|}} \equiv \int \mathrm{d}^3\mathbf{v}\ v_{G\|} S_E S_r$. Following the same procedure than for the density source, one finally obtains:

$$\mathbb{S}_{v_{G\|}} = 2\pi^{3/2} \left(\frac{T_{s,\mathrm{srce}}}{m_s}\right)^2 \left[2c_{10} + J_{\|B}(c_{00} + 4c_{20})\right] S_r \tag{H.13}$$

Source of energy – heating

The fluid source of energy $\mathbb{S}_{\mathcal{E}}$ is defined as follows: $\mathbb{S}_{\mathcal{E}} \equiv \int \mathrm{d}^3\mathbf{v}\ \left(m_s \frac{v_{G\|}^2}{2} + \mu_s B\right) S_E S_r$. Notice that $(m_s v_{G\|}^2/2 + \mu_s B) = T_{s,\mathrm{srce}}(\bar{v}_{G_\| s}^2 + \bar{\mu}_s)$. Again, the energy source can also be recast in terms of the $c_{h\ell}$ coefficients:

$$\mathbb{S}_{\mathcal{E}} = \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}} \left[2\ c_{20} + \frac{3}{2}c_{00} - c_{01} + \frac{5}{2}J_{\|B}\ c_{10} + 6J_{\|B}\ c_{30} - J_{\|B}\ c_{11}\right] S_r \tag{H.14}$$

Source of vorticity

The fluid source of vorticity $\mathbb{S}_\Omega$ is simply: $\mathbb{S}_\Omega \equiv \int \mathrm{d}^3\mathbf{v}\ J_{0_s}.(S_E S_r)$, where J_{0_s} is the gyro-average operator. We use the Padé approximation:

$$J_{0_s} \approx 1 + \frac{m_s}{q_s^2}\frac{\mu_s}{2B}\nabla_\perp^2 = 1 + \alpha\,\bar{\mu}_s\,\nabla_\perp^2 \quad \text{with} \quad \alpha = \frac{m_s}{q_s^2}\frac{T_{s,\mathrm{srce}}}{2B^2}$$

Again, the vorticity source can be recast in terms of the $c_{h\ell}$ coefficients [2] :

$$\mathbb{S}_\Omega = \mathbb{S}_n + \alpha \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} \left[\nabla_\perp^2\left((c_{00} - c_{01})\,S_r\right) + J_{\|B}\nabla_\perp^2\left((c_{10} - c_{11})\,S_r\right)\right] \tag{H.15}$$

H.2.3 Pure sources of momentum, energy and vorticity

The expressions of $\mathbb{S}_n$ (eq. (H.12)), $\mathbb{S}_{v_{G\|}}$ (eq. H.13), $\mathbb{S}_{\mathcal{E}}$ (eq. (H.14)) and $\mathbb{S}_\Omega$ (eq. (H.15)) provide the constraints on the $c_{h\ell}$ coefficients in order to impose independently zero source of density, of momentum, of energy or of vorticity. Let's consider three cases:

- Non vanishing source of energy, with no injection of particles nor of momentum.
- Non vanishing source of momentum, with no injection of particles nor of energy.
- Non vanishing source of vorticity, with no injection of particles, of momentum nor of energy.

[2] Let us define $\beta \equiv \frac{1}{\sqrt{\pi}}\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2}$, then

$$\begin{aligned}
\mathbb{S}_\Omega &= \beta \sum_{h,l} \int_0^{+\infty} \mathrm{d}\bar{\mu}_s \int_{-\infty}^{+\infty} \mathrm{d}\bar{v}_{G_\| s} H_h(\bar{v}_{G_\| s}) L_l(\bar{\mu}_s) e^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} (1 + J_{\|B}\bar{v}_{G_\| s})\left(1 + \alpha\,\bar{\mu}_s\,\nabla_\perp^2\right) c_{h\ell} S_r \\
&= \mathbb{S}_n + \beta \sum_{h,l} \int_0^{+\infty} \mathrm{d}\bar{\mu}_s \int_{-\infty}^{+\infty} \mathrm{d}\bar{v}_{G_\| s} H_h\, L_l\, e^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \alpha \left(H_0 + \frac{J_{\|B}}{2}H_1\right)(L_0 - L_1)\,\nabla_\perp^2(c_{h\ell} S_r) \\
&= \mathbb{S}_n + \beta\,\alpha \left[|H_0|^2\nabla_\perp^2(c_{00}S_r) - |H_0|^2\nabla_\perp^2(c_{01}S_r) + \frac{J_{\|B}}{2}|H_1|^2\nabla_\perp^2(c_{10}S_r) - \frac{J_{\|B}}{2}|H_1|^2\nabla_\perp^2(c_{11}S_r)\right] \\
&= \mathbb{S}_n + \alpha \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} \left[\nabla_\perp^2(c_{00}S_r) - \nabla_\perp^2(c_{01}S_r) + \frac{J_{\|B}}{2}\left(\nabla_\perp^2(c_{10}S_r) - \nabla_\perp^2(c_{11}S_r)\right)\right]
\end{aligned}$$

These three cases are considered hereafter. Imposing zero source of matter, the fluid sources of parallel momentum, of energy and of vorticity are proportional to:

$$\mathbb{S}_n = 0 \quad \Rightarrow \quad c_{00} + J_{\|B}\, c_{10} = 0 \tag{H.16}$$

$$\mathbb{S}_{v_{G\|}} \propto \left(2 - J_{\|B}^2\right) c_{10} + 4 J_{\|B}\, c_{20} \tag{H.17}$$

$$\mathbb{S}_{\mathcal{E}} \propto 2c_{20} - c_{00} + 6 J_{\|B}\, c_{30} - c_{01} - J_{\|B}\, c_{11} \tag{H.18}$$

$$\mathbb{S}_{\Omega} \propto -\nabla_\perp^2 \left(J_{\|B}\, c_{10} S_r\right) + J_{\|B} \nabla_\perp^2 \left(c_{10} S_r\right) - \nabla_\perp^2 \left(c_{01} S_r\right) - J_{\|B} \nabla_\perp^2 \left(c_{11} S_r\right) \tag{H.19}$$

Pure source of energy

Killing the fluid sources of particles, momentum and vorticity, while keeping finite the fluid source of energy, imposes eq. (H.17) and eq (H.19) to vanish. Several solutions can be envisaged. Let's choose $c_{30} = 0$ and $(c_{01} + J_{\|B}\, c_{11}) = 0$, with $(2c_{20} - c_{00}) \neq 0$. Then, the fluid source of vorticitiy trivially vanishes for $c_{10} = c_{11}$ [3] . To summarize, we propose the following set of coefficients for a pure source of energy:

$$\begin{cases} c_{11} = c_{10} = -\frac{4 J_{\|B}}{2 - J_{\|B}^2}\, c_{20} \\ c_{00} = c_{01} = \frac{4 J_{\|B}^2}{2 - J_{\|B}^2}\, c_{20} \\ c_{30} = 0 \end{cases} \tag{H.20}$$

with $c_{20} \neq 0$.

In order to inject solely energy into the system, the source term that should appear in the right hand side of the gyrokinetic equation would then take the following form:

$$\begin{aligned} S_{\mathcal{E}} &= S_r \left(c_{00} + c_{01} L_1 + c_{10} H_1 + c_{11} H_1 L_1 + c_{20} H_2\right) \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \\ &= 4\, c_{20} S_r \left[\bar{v}_{G_\| s}^2 - \frac{1}{2} + \frac{J_{\|B}}{2 - J_{\|B}^2} (2 - \bar{\mu}_s)(J_{\|B} - 2\bar{v}_{G_\| s})\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \quad \text{due to (H.20)} \end{aligned}$$

while the fluid source of energy $\mathbb{S}_{\mathcal{E}}$ would have the following magnitude:

$$\begin{aligned} \mathbb{S}_{\mathcal{E}} &= \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}} (2\, c_{20} - c_{00}) S_r \quad \text{according to (H.14) and (H.20)} \\ &= 2 \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}} \left(1 - \frac{2 J_{\|B}^2}{2 - J_{\|B}^2}\right) c_{20} S_r \end{aligned}$$

Let us introduce the normalized intensity:

$$S_0^{\mathcal{E}} \equiv 2\, c_{20} \left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}} \tag{H.21}$$

the previous equality becomes,

$$\mathbb{S}_{\mathcal{E}} = \left(1 - \frac{2 J_{\|B}^2}{2 - J_{\|B}^2}\right) S_0^{\mathcal{E}}\, S_r \tag{H.22}$$

Then, up to small terms proportional to $J_{\|B}$, S_r provides the radial shape of the energy source, while $S_0^{\mathcal{E}}$ gives its magnitude[4]. Finally, $S_{\mathcal{E}}$ can be expressed as:

$$S_{\mathcal{E}} = \frac{S_0^{\mathcal{E}}\, S_r}{\sqrt{2} \left(\frac{\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}}} \left[\bar{v}_{G_\| s}^2 - \frac{1}{2} - \frac{J_{\|B}}{2 - J_{\|B}^2} (2 - \bar{\mu}_s) \left(2\bar{v}_{G_\| s} - J_{\|B}\right)\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \tag{H.23}$$

with $J_{\|B}$ defined by (H.11).

[3] Using the choice $(c_{01} + J_{\|B}\, c_{11}) = 0$, the equation (H.19) is equivalent to

$$\mathbb{S}_{\Omega} \propto -c_{10} S_r \nabla_\perp^2 J_{\|B} + \nabla_\perp^2 \left(J_{\|B}\, c_{11} S_r\right) - J_{\|B} \nabla_\perp^2 (c_{11} S_r) = (-c_{10} S_r + c_{11} S_r) \nabla_\perp^2 J_{\|B}$$

[4] Provided S_r is properly normalized, i.e. such that its volume integral is equal to unity.

Pure parallel momentum

Killing the fluid sources of particles, energy and vorticity, while keeping finite the fluid source of parallel momentum, imposes equations (H.18)-(H.19) to vanish. Again, several options could be considered. Following the same strategy as for the energy, namely:

$$\begin{cases} c_{11} = c_{10} \\ c_{00} = c_{01} = -J_{\|B}\, c_{10} \end{cases}$$

then the source of energy vanishes if $c_{30} = 0$ and $2\, c_{20} = c_{00}$.

Consistently, in order to inject only parallel momentum, the following source is proposed [5] :

$$S_{v_{G\|}} = \frac{S_0^{v_{G\|}}\, S_r}{4\pi^{3/2}\left(\frac{T_{s,\mathrm{srce}}}{m_s}\right)^2} \left[2\bar{v}_{G_\| s}(2-\bar{\mu}_s) - J_{\|B}\left(1 + 2\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right)\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \tag{H.24}$$

where the the normalized intensity $S_0^{v_{G\|}}$ is defined as

$$S_0^{v_{G\|}} \equiv 4\pi^{3/2}\left(\frac{T_{s,\mathrm{srce}}}{m_s}\right)^2 c_{10} \tag{H.25}$$

The corresponding fluid source of momentum is (according to (H.13)):

$$\mathbb{S}_{v_{G\|}} = \left(1 - \frac{3J_{\|B}{}^2}{2}\right) S_0^{v_{G\|}}\, S_r \tag{H.26}$$

Then, at leading order in $J_{\|B}$, S_r corresponds to the radial shape of the momentum source, and $S_0^{v_{G\|}}$ to its magnitude.

Pure source of vorticity

So as to inject vorticity only, the simplest choice appears to be: $c_{00} = c_{10} = c_{11} = 0$ and $c_{20} = 0$. Then the source of vorticitiy is governed by the c_{01} coefficient only: $\mathbb{S}_\Omega \propto -\nabla_\perp^2(c_{01}S_r)$, while that of momentum eq. (H.17) is set to zero. The source of energy eq. (H.18) vanishes provided that $c_{30} = (1/6J_{\|B})\, c_{01}$. Obviously, such a constraint is invalid for those simulations performed at vanishing parallel current.

Alternatively, one decides to allow for some parallel momentum injection by taking $c_{20} = c_{01}/2 \neq 0$ and $c_{30} = 0$. Then, the source term to be considered is the following [6] :

$$S_\Omega = -\frac{S_0^\Omega\, S_r}{\alpha\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2}} \left[2\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \quad \text{with } \alpha = \frac{m_s}{q_s^2}\frac{T_{s,\mathrm{srce}}}{2B^2} \tag{H.27}$$

where the normalized intensity S_0^Ω is defined by:

$$S_0^\Omega \equiv -c_{01}\alpha\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} \tag{H.28}$$

[5] According to expression (H.4),

$$\begin{aligned} S_{v_{G\|}} &= S_r\left[c_{00} + c_{01}L_1 + c_{20}H_2 + (c_{10} + c_{11}L_1)H_1\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \\ &= c_{10}S_r\left[2\bar{v}_{G_\| s}(2-\bar{\mu}_s) - J_{\|B}(1 + 2\bar{v}_{G_\| s} - \bar{\mu}_s\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} \end{aligned}$$

[6] Taking into account the constraints $c_{00} = c_{10} = c_{11} = c_{30} = 0$ and $c_{20} = c_{01}/2$,

$$S_\Omega = S_r(c_{01}L_1 + c_{20}H_2)\mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} = c_{01}S_r\left[1 - \bar{\mu}_s + (2\bar{v}_{G_\| s}^2 - 1)\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s} = c_{01}S_r\left[2\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s}$$

which is equivalent to (H.27) by expressing c_{01} with definition (H.28).

and the resulting fluid source of vorticity is:

$$\mathbb{S}_\Omega = S_0^\Omega \nabla_\perp^2 \left(S_r\right) \tag{H.29}$$

We recall that such a source does inject some momentum as well. However, its magnitude remains small, and equal to $(-\frac{1}{\alpha}\sqrt{\frac{2m_s}{T_s}} J_{\|B}\, S_0^\Omega S_r)$ [7] .

Another possibility for the heating source

There exists a simplified version of the heating source possible in the code which is not exactly a pure source of heating. In this case the energy dependent part of the source is defined as:

$$S_\mathcal{E}(r,\theta,v_{G\|},\mu_s) = \frac{S_0^\mathcal{E}}{3\sqrt{2}\left(\frac{\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}}} \left(\frac{E}{T_{s,\mathrm{srce}}} - \frac{3}{2}\right) \exp\left(-\frac{E}{T_{s,\mathrm{srce}}}\right) \tag{H.30}$$

where E the energy is equal to $E = \frac{1}{2} m_s v_{G\|} + \mu_s B$. Therefore,

$$\int_{-\infty}^{\infty} \mathrm{d}v_{G\|} \int_0^\infty \frac{2\pi B_{\|s}^*}{m_s}\, \mathrm{d}\mu_s\, E\, S_E = S_0^\mathcal{E}$$

H.2.4 A source of impurities

According to equation (H.15), it is clear that in this source formalism it is complicate to generate a source of matter without injecting a source of vorticity. So as first choice, we take $c_{00} = c_{01}$ and $c_{10} = c_{11} = 0$, such that $\mathbb{S}_\Omega = \mathbb{S}_n$. The injection of momentum and energy can be avoided by imposing in equations (H.13) and (H.14), $c_{00} + 4c_{20} = 0$ and $2c_{20} + (3/2)c_{00} - c_{01} + 6J_{\|B}c_{30} = 0$. Due to the previous assumptions, this leads to $c_{30} = 0$. Therefore, using the fact that $c_{01} = c_{00}$ and $c_{20} = -(1/4)c_{00}$, the matter source term take the form [8]

$$S_n = S_r c_{00} \left(\frac{5}{2} - \bar{\mu}_s - \bar{v}_{G_{\|}s}^2\right) \mathrm{e}^{-\bar{v}_{G_{\|}s}^2 - \bar{\mu}_s}$$

Then, according to equation (H.12) and considering the normalized matter source $S_0^n = (2\pi T_{s,\mathrm{srce}}/m_s)^{3/2} c_{00}$,

$$S_n = \frac{S_0^n S_r}{\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2}} \left(\frac{5}{2} - \bar{\mu}_s - \bar{v}_{G_{\|}s}^2\right) \mathrm{e}^{-\bar{v}_{G_{\|}s}^2 - \bar{\mu}_s} \tag{H.31}$$

Let us denote by S_{n_s} (resp. $S_{n_{s'}}$) the source of matter associated to the majority species s (resp. to the impurity species s'). The injection of impurity must be compensated by the injection of the majority species such that

$$Z_s \int S_{n_s} \mathcal{J}_{\mathrm{v}_s}\, \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s + Z_{s'} \int S_{n_{s'}} \mathcal{J}_{\mathrm{v}_{s'}}\, \mathrm{d}v_{G\|}\, \mathrm{d}\mu_{s'} = 0$$

[7] Considering the constraints $c_{10} = c_{00} = 0$ and $c_{20} = c_{01}/2$ in $\mathbb{S}_{v_{G\|}}$ expression's (H.13)

$$\mathbb{S}_{v_{G\|}} = 4\pi^{3/2}\left(\frac{T_{s,\mathrm{srce}}}{m_s}\right)^2 J_{\|B}\, c_{01} = -4\pi^{3/2}\left(\frac{T_{s,\mathrm{srce}}}{m_s}\right)^2 J_{\|B} S_0^\Omega \frac{1}{\alpha}\left(\frac{m_s}{2\pi T_{s,\mathrm{srce}}}\right)^{3/2} S_r = -\frac{1}{\alpha}\sqrt{\frac{2m_s}{T_{s,\mathrm{srce}}}} J_{\|B} S_0^\Omega S_r$$

[8] The projection of the matter source on the Hermite and Laguerre polynomials gives

$$S_n = S_r c_{00}\left(1 + L_1(\bar{\mu}_s) - \frac{1}{4}H_2(\bar{v}_{G_{\|}s})\right) \mathrm{e}^{-\bar{v}_{G_{\|}s}^2 - \bar{\mu}_s} = S_r c_{00}\left[1 + 1 - \bar{\mu}_s - \frac{1}{4}\left(-2 + 4\bar{v}_{G_{\|}s}^2\right)\right] \mathrm{e}^{-\bar{v}_{G_{\|}s}^2 - \bar{\mu}_s}$$

H.2.5 Numerical treatment of the source terms

The source terms are taken into account by solving $\mathrm{d}_t\bar{F}_s = S_{\mathcal{E}} + S_{v_{G\|}} + S_\Omega + S_n$ with $S_{\mathcal{E}}$ defined by eq. (H.23), $S_{v_{G\|}}$ by eq. (H.24), S_Ω by eq. (H.27) and S_n by eq. (H.31). For the following, let us use the fact that each source is of the form $S_E(r(t), \theta(t), v_{G\|}(t), \mu_s)S_r(r(t))$. For more readability let us consider one unique source knowing that the numerical method described below can be generalized to a sum of sources. Let us integrate in time between t and $t + \Delta t$ the equation

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t}(r, \theta, \varphi, v_{G\|}, \mu) = S(t) \quad \text{with} \quad S(t) = S_E(r(t), \theta(t), v_{G\|}(t), \mu_s)S_r(r(t))$$

then

$$\bar{F}_s(t + \Delta t) - \bar{F}_s(t) = \int_t^{t+\Delta t} S(t')\,\mathrm{d}t'$$

Let us consider a Taylor expansion of second order for S_r as

$$\begin{aligned}
S_r(r(t')) &= S_r(r(t + \Delta t)) + \left[r(t') - r(t + \Delta t)\right] \frac{\partial S_r}{\partial r}(r(t + \Delta t)) \\
&= S_r(r(t + \Delta t)) + \left[r(t + \Delta t) + (t' - t - \Delta t)\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t} - r(t + \Delta t)\right] \frac{\partial S_r}{\partial r}(r(t + \Delta t)) \\
&= S_r(r(t + \Delta t)) + (t' - t - \Delta t)\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_r}{\partial r}(r(t + \Delta t)) + \mathcal{O}(\Delta t^2)
\end{aligned}$$

Besides, let us define $S_E(t') = S_E(r(t'), \theta(t'), v_{G\|}(t'), \mu_s)$, then

$$\begin{aligned}
S_E(t') &= S_E(r(t + \Delta t), \theta(t'), v_{G\|}(t'), \mu_s) + \\
&\quad (t' - t - \Delta t)\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial r}(r(t + \Delta t), \theta(t'), v_{G\|}(t'), \mu_s) + \mathcal{O}(\Delta t^2) \\
&= S_E\left(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t'), \mu_s\right) + \\
&\quad (t' - t - \Delta t)\left[\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial r}(r(t + \Delta t), \theta(t'), v_{G\|}(t'), \mu_s) + \right. \\
&\qquad \left. \frac{\mathrm{d}\theta}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial \theta}(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t'), \mu_s)\right] + \mathcal{O}(\Delta t^2) \\
&= S_E(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t + \Delta t), \mu_s) + \\
&\quad (t' - t - \Delta t)\left[\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial r}(r(t + \Delta t), \theta(t'), v_{G\|}(t'), \mu_s) + \right. \\
&\qquad \frac{\mathrm{d}\theta}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial \theta}(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t'), \mu_s) + \\
&\qquad \left. \frac{\mathrm{d}v_{G\|}}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial v_{G\|}}(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t + \Delta t), \mu_s)\right] + \mathcal{O}(\Delta t^2) \\
&\approx S_E(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t + \Delta t), \mu_s) + \\
&\quad (t' - t - \Delta t)\left[\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial r}(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t + \Delta t), \mu_s) + \right. \\
&\qquad \frac{\mathrm{d}\theta}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial \theta}(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t + \Delta t), \mu_s) + \\
&\qquad \left. \frac{\mathrm{d}v_{G\|}}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial v_{G\|}}(r(t + \Delta t), \theta(t + \Delta t), v_{G\|}(t + \Delta t), \mu_s)\right] + \mathcal{O}(\Delta t^2)
\end{aligned}$$

Therefore, the distribution function $\bar{F}_s$ at time $t+\Delta t$ is given at second order in time by

$$\begin{aligned}
\bar{F}_s(t+\Delta t) &= \bar{F}_s(t) + \int_t^{t+\Delta t} S_E(t') \left[S_r(t+\Delta t) + (t'-t-\Delta t)\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_r}{\partial r}|_{t+\Delta t} \right] \mathrm{d}t' \\
&= \bar{F}_s(t) + \int_t^{t+\Delta t} S_E(t+\Delta t) S_r(t+\Delta t)\, \mathrm{d}t' + \\
&\quad \int_t^{t+\Delta t} (t'-t-\Delta t) \left\{ S_E(t+\Delta t)\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_r}{\partial r}|_{t+\Delta t} + S_r(t+\Delta t) \right. \\
&\quad \left. \left[\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial r}|_{t+\Delta t} + \frac{\mathrm{d}\theta}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial \theta}|_{t+\Delta t} + \frac{\mathrm{d}v_{G\|}}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial v_{G\|}}|_{t+\Delta t} \right] \right\} \mathrm{d}t'
\end{aligned}$$

so

$$\begin{aligned}
\bar{F}_s(t+\Delta t) &= \bar{F}_s(t) + \Delta t S_E(t+\Delta t) S_r(t+\Delta t) - \\
&\quad \frac{\Delta t^2}{2} \left[S_E(t+\Delta t)\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_r}{\partial r}|_{t+\Delta t} + S_r(t+\Delta t) \left(\frac{\mathrm{d}r}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial r}|_{t+\Delta t} + \right. \right. \\
&\quad \left. \left. \frac{\mathrm{d}\theta}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial \theta}|_{t+\Delta t} + \frac{\mathrm{d}v_{G\|}}{\mathrm{d}t}|_{t+\Delta t}\frac{\partial S_E}{\partial v_{G\|}}|_{t+\Delta t} \right) \right]
\end{aligned}$$

In the code, the radial derivatives of the sources of: (i) energy $\partial_r S_{\mathcal{E}}$, (ii) momentum $\partial_r S_{v_{G\|}}$, (iii) vorticity $\partial_r S_\Omega$ and (iv) density $\partial_r S_n$ are computed numerically as well as the poloidal derivatives. On the other hand, the derivatives in $v_{G\|}$ direction are expressed analytically for:

(i) the pure source of energy [9] :

$$\begin{aligned}
\frac{\partial S_{\mathcal{E}}}{\partial v_{G\|}} &= \frac{S_0^{\mathcal{E}}\, S_r}{\sqrt{2}\left(\frac{\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}}} \frac{1}{\sqrt{2T_{s,\mathrm{srce}}}} \exp\left(-\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right) \times \\
&\quad \left[-2K_h(2-\bar{\mu}_s) - 2\bar{v}_{G_\| s} \left\{ \bar{v}_{G_\| s}^2 - \frac{3}{2} - K_h(2-\bar{\mu}_s)\left(2\bar{v}_{G_\| s} - J_{\| B}\right) \right\} \right]
\end{aligned}$$

(ii) the pure source of momentum [10] :

$$\begin{aligned}
\frac{\partial S_{v_{G\|}}}{\partial v_{G\|}} &= \frac{S_0^{v_{G\|}}\, S_r}{4\pi^{3/2}\left(\frac{T_{s,\mathrm{srce}}}{m_s}\right)^2} \frac{1}{\sqrt{2T_{s,\mathrm{srce}}}} \exp\left(-\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right) \times \\
&\quad \left[2(2-\bar{\mu}_s)\left(1 - 2\bar{v}_{G_\| s}^2\right) + 2J_{\| B}\bar{v}_{G_\| s}\left(-1 + 2\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right) \right]
\end{aligned}$$

(iii) the pure source of vorticity (according to eq. (H.27)):

$$\frac{\partial S_\Omega}{\partial v_{G\|}} = -\frac{S_0^{\Omega}\, S_r}{\alpha\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2}} \frac{1}{\sqrt{2T_{s,\mathrm{srce}}}} \exp\left(-\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right) \left[4\bar{v}_{G_\| s} - 2\bar{v}_{G_\| s}\left(2\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right) \right]$$

[9] According to eq. (H.23), the derivative of $S_{\mathcal{E}}$ reads

$$\frac{\partial S_{\mathcal{E}}}{\partial v_{G\|}} = g_1(r)\frac{1}{\sqrt{2T_{s,\mathrm{srce}}}} \left[2\bar{v}_{G_\| s} - 2K_h(2-\bar{\mu}_s)2\bar{v}_{G_\| s} \left\{ \bar{v}_{G_\| s}^2 - \frac{1}{2} - K_h(2-\bar{\mu}_s)\left(2\bar{v}_{G_\| s} - J_{\| B}\right) \right\} \right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s}$$

with the radial function $g_1(r) = S_0^{\mathcal{E}}\, S_r / \left(\sqrt{2}\left(\frac{\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2} T_{s,\mathrm{srce}} \right)$ and $K_h = J_{\| B} / \left(2 - J_{\| B}^2\right)$.

[10] According to eq. (H.24), the derivative of $S_{v_{G\|}}$ reads

$$\frac{\partial S_{v_{G\|}}}{\partial v_{G\|}} = g_2(r)\frac{1}{\sqrt{2T_{s,\mathrm{srce}}}} \left[2(2-\bar{\mu}_s) - 4\bar{v}_{G_\| s}J_{\| B} - 2\bar{v}_{G_\| s}\left(2\bar{v}_{G_\| s}(2-\bar{\mu}_s) - J_{\| B}\left(1 + 2\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right)\right) \right] \mathrm{e}^{-\bar{v}_{G_\| s}^2 - \bar{\mu}_s}$$

with the radial function $g_2(r)$ defined as $g_2(r) = S_0^{v_{G\|}}\, S_r / \left(4\pi^{3/2}\left(\frac{T_{s,\mathrm{srce}}}{m_s}\right)^2 \right)$.

(iv) the source of impurity (according to eq. (H.31)):

$$\frac{\partial S_n}{\partial v_{G\|}} = \frac{S_0^n S_r}{\left(\frac{2\pi T_{s,\mathrm{srce}}}{m_s}\right)^{3/2}} \frac{1}{\sqrt{2T_{s,\mathrm{srce}}}} \exp\left(-\bar{v}_{G_\| s}^2 - \bar{\mu}_s\right) \left[-2\bar{v}_{G_\| s}\left(\frac{7}{2} - \bar{\mu}_s - \bar{v}_{G_\| s}^2\right)\right]$$

Appendix I

A Crank-Nicolson scheme for diffusion terms

In this appendix the semi-implicit Crank-Nicolson scheme, which is implemented in GYSELA to take into account a diffusion term of the form

$$\partial_t \bar{F}_s = \frac{1}{r}\partial_r \left(r\chi(r)\partial_r \bar{F}_s \right) \tag{I.1}$$

is detailed. Classically, let us write: $f_i^n \equiv \bar{F}_s(r_i, \theta, \varphi, v_{G\|}, t_n)$, where subscript $j \in \{0, \ldots, N\}$ refers to the discretised index in radial direction and superscript n refers to the time. Let us also define $\mathcal{D} \equiv r\chi(r)$ with $\chi(r) = \chi_0 H_{\text{buff}}(r)$. The diffusion amplitude χ_0 is modulated by a radial profile H_{buff} which is equal to 1 in the buffer region and equal to 0 elsewhere. For each value of μ_s, each term in Eq. (I.1) reads [1] for all $i = 0, \cdots, N$:

$$\begin{aligned} f_i^{n+1} &= f_i^n + \frac{\Delta t}{2\,\Delta r^2\, r_i} \Big\{ \mathcal{D}_{i+\frac{1}{2}} \left[\left(f_{i+1}^{n+1} - f_i^{n+1} \right) + \left(f_{i+1}^n - f_i^n \right) \right] \\ &\qquad - \mathcal{D}_{i-\frac{1}{2}} \left[\left(f_i^{n+1} - f_{i-1}^{n+1} \right) + \left(f_i^n - f_{i-1}^n \right) \right] \Big\} \end{aligned} \tag{I.2}$$

Let us introduce $\alpha_i = \frac{\Delta t}{2\,\Delta r^2\, r_i}$, then

$$-\alpha_i \mathcal{D}_{i-\frac{1}{2}} f_{i-1}^{n+1} + \left[1 + \alpha_i \left(\mathcal{D}_{i-\frac{1}{2}} + \mathcal{D}_{i+\frac{1}{2}} \right) \right] f_i^{n+1} - \alpha_i \mathcal{D}_{i+\frac{1}{2}} f_{i+1}^{n+1} = R_i^n$$

with

$$R_i^n = \alpha_i \mathcal{D}_{i-\frac{1}{2}} f_{i-1}^n + \left[1 - \alpha_i \left(\mathcal{D}_{i-\frac{1}{2}} + \mathcal{D}_{i+\frac{1}{2}} \right) \right] f_i^n + \alpha_i \mathcal{D}_{i+\frac{1}{2}} f_{i+1}^n$$

Therefore, for all $i = 0, \cdots, N$,

$$A_i f_{i-1}^{n+1} + B_i f_i^{n+1} + C_i f_{i+1}^{n+1} = -A_i f_{i-1}^n + (2 - B_i)\, f_i^n - C_i f_{i+1}^n \tag{I.3}$$

[1] Let us defined the temporary variable $\Upsilon = r\chi(r)\partial_r \bar{F}_s$, then applying a Crank-Nicolson scheme is equivalent to write:

$$\begin{aligned} \frac{f_i^{n+1} - f_i^n}{\Delta t} &= \frac{1}{2r_i} \left\{ \left(\frac{\Upsilon_{i-\frac{1}{2}}^{n+1} - \Upsilon_{i+\frac{1}{2}}^{n+1}}{\Delta r} \right) + \left(\frac{\Upsilon_{i-\frac{1}{2}}^{n} - \Upsilon_{i+\frac{1}{2}}^{n}}{\Delta r} \right) \right\} \\ &= \frac{1}{2r_i} \left\{ \left(\frac{\mathcal{D}_{i-\frac{1}{2}} \partial_r f_{i-\frac{1}{2}}^{n+1} - \mathcal{D}_{i+\frac{1}{2}} f_{i+\frac{1}{2}}^{n+1}}{\Delta r} \right) + \left(\frac{\mathcal{D}_{i-\frac{1}{2}} \partial_r f_{i-\frac{1}{2}}^{n} - \mathcal{D}_{i+\frac{1}{2}} f_{i+\frac{1}{2}}^{n}}{\Delta r} \right) \right\} \\ &= \frac{1}{2r_i \Delta r} \left\{ \mathcal{D}_{i-\frac{1}{2}} \left(\frac{f_{i-1}^{n+1} - f_i^{n+1}}{\Delta r} \right) - \mathcal{D}_{i+\frac{1}{2}} \left(\frac{f_i^{n+1} - f_{i+1}^{n+1}}{\Delta r} \right) \right. \\ &\qquad \left. + \mathcal{D}_{i-\frac{1}{2}} \left(\frac{f_{i-1}^{n} - f_i^{n}}{\Delta r} \right) - \mathcal{D}_{i+\frac{1}{2}} \left(\frac{f_i^{n} - f_{i+1}^{n}}{\Delta r} \right) \right\} \\ &= \frac{1}{2\, r_i\, \Delta r^2} \left\{ \mathcal{D}_{i+\frac{1}{2}} \left[\left(f_{i+1}^{n+1} - f_i^{n+1} \right) + \left(f_{i+1}^n - f_i^n \right) \right] - \mathcal{D}_{i-\frac{1}{2}} \left[\left(f_i^{n+1} - f_{i-1}^{n+1} \right) + \left(f_i^n - f_{i-1}^n \right) \right] \right\} \end{aligned}$$

which is equivalent to (I.2).

where the coefficients A_i, B_i and C_i are defined as:

$$A_i = -\alpha_i \mathcal{D}_{i-\frac{1}{2}} \tag{I.4}$$

$$B_i = 1 + \alpha_i \left(\mathcal{D}_{i-\frac{1}{2}} + \mathcal{D}_{i+\frac{1}{2}} \right) \tag{I.5}$$

$$C_i = -\alpha_i \mathcal{D}_{i+\frac{1}{2}} \tag{I.6}$$

Using the fact that $r_{i\pm 1/2} = 0.5\,(r_i + r_{i\pm 1})$,

$$\mathcal{D}_{i\pm\frac{1}{2}} = \frac{1}{4}\left(r_i\chi_i + r_{i\pm 1}\chi_i + r_i\chi_{i\pm 1} + r_{i\pm 1}\chi_{i\pm 1}\right) \tag{I.7}$$

Concerning the boundary conditions, let us consider that the second derivative of the radial profile χ is equal to 0 at the boundaries (i.e $\frac{\partial^2\chi}{\partial r^2}|r_{\min} = \frac{\partial^2\chi}{\partial r^2}|r_{\max} = 0$) , then [2]

$$\mathcal{D}_{-\frac{1}{2}} = \frac{1}{2}\left(r_0 - \frac{\Delta r}{2}\right)(3\chi_0 - \chi_1) \quad \text{and} \quad \mathcal{D}_{N+\frac{1}{2}} = \frac{1}{2}\left(r_N + \frac{\Delta r}{2}\right)(3\chi_N - \chi_{N-1})$$

Besides, a non-homogeneous Dirichlet boundary condition is applied to the outer radial boundary such that $\bar{F}_s(r = r_{\max}, \theta, \varphi, v_{G\|}, \mu_s) = \bar{F}_{s,\text{eq}}(r = r_{\max}, \theta, v_{G\|}, \mu_s)$. Concerning the inner radial boundary, there are two possibility: (i) non-homogeneous Dirichlet boundary condition $\bar{F}_s(r = r_{\min}, \theta, \varphi, v_{G\|}, \mu_s) = \bar{F}_{s,\text{eq}}(r = r_{\min}, \theta, v_{G\|}, \mu_s)$ in the case of thermal bath or (ii) Neumann boundary conditions (i.e $f_{-1} = f_1$) in the case of flux-driven simulation. Let us consider the boolean κ which is equal to 1 if Neumann boundary conditions are applied (i.e in the case of flux driven boundary conditions) and is equal to 0 otherwise. The system can be rewritten in the compact tridiagonal form:

$$\begin{pmatrix} (1-\kappa)+\kappa B_0 & \kappa(A_0+C_0) & & \\ A_1 & B_1 & \ddots & \\ & \ddots & \ddots & C_{N-2} \\ & A_{N-1} & B_{N-1} & C_{N-1} \\ & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_0^{n+1} \\ \vdots \\ f_N^{n+1} \end{pmatrix} = \begin{pmatrix} R_0^n \\ \vdots \\ R_N^n \end{pmatrix} \tag{I.8}$$

with

$$\begin{pmatrix} R_0^n \\ \vdots \\ R_N^n \end{pmatrix} = \begin{pmatrix} (1-\kappa)+\kappa(2-B_0) & -\kappa(A_0+C_0) & & \\ -A_1 & (2-B_1) & \ddots & \\ & \ddots & \ddots & -C_{N-2} \\ & -A_{N-1} & (2-B_{N-1}) & -C_{N-1} \\ & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f_0^n \\ \vdots \\ f_N^n \end{pmatrix}$$

The tridiagonal system (I.8) is solved by using a modified Thomas algorithm (see Appendix E).

Let us notice that it is also possible in the code to apply diffusion in the poloidal direction. The same strategy than above is adopted. The resulting matrix system differs from (I.8) by first and last lines due to boundary conditions which are periodic in θ.

[2] Using the fact that $\chi_{-1} = \chi_0$ in (I.7),

$$\mathcal{D}_{-\frac{1}{2}} = \frac{1}{4}\left(r_0 + r_{-1}\right)\left(\chi_0 + \chi_{-1}\right) = \frac{1}{4}\left(r_0 + (r_0 - \Delta r)\right)\left(\chi_0 + (2\chi_0 - \chi_1)\right) = \frac{1}{2}\left(r_0 - \frac{\Delta r}{2}\right)(3\chi_0 - \chi_1)$$

Appendix J

Interpolation by cubic splines

One of the important time consuming step of the semi-Lagrangian scheme is the interpolation step. In the code all the interpolations are performed with cubic splines, in 1D in φ and $v_{G\|}$ directions and in 2D in (r, θ) planes. The numerical implementation of these cubic splines is detailed in the first section of this appendix. However, the drawback of such an interpolation is the fact that computing one interpolated value requires to know the function on all the nodes of the mesh. This global dependency is not adapted at all to a competitive parallelization. The new approach presented in the second part of this appendix, has been explicitly developed to overcome this problem (cf [CLS09]). This approach is based on a domain decomposition, where cubic splines are locally applied. The gain in terms of CPU time and memory size is tackled in the last part.

J.1 Interpolation with classical cubic splines

J.1.1 Cubic spline interpolation in 1D

Let $g(x)$ be a function defined in the x direction with $x \in [x_0, x_{N_x}]$ where N_x represents the number of points in x (the step h being constant). Using a cubic spline for the interpolation of g consists in representing this function in terms of piecewise cubic polynomials Λ_α, twice continuously differentiable [DeB01] as:

$$g(x) \simeq s(x) = \sum_{\alpha=-1}^{N_x+1} c_\alpha \Lambda_\alpha(x)$$

where

$$\Lambda_\alpha(x) = \frac{1}{6h^3} \begin{cases} (x - x_{\alpha-2})^3 & \text{if } x_{\alpha-2} \leq x \leq x_{\alpha-1} \\ h^3 + 3h^2(x - x_{\alpha-1}) + 3h(x - x_{\alpha-1})^2 & \\ \quad -3(x - x_{\alpha-1})^3 & \text{if } x_{\alpha-1} \leq x \leq x_\alpha \\ h^3 + 3h^2(x_{\alpha+1} - x) + 3h(x_{\alpha+1} - x)^2 & \\ \quad -3(x_{\alpha+1} - x)^3 & \text{if } x_\alpha \leq x \leq x_{\alpha+1} \\ (x_{\alpha+2} - x)^3 & \text{if } x_{\alpha+1} \leq x \leq x_{\alpha+2} \\ 0 & \text{otherwise} \end{cases} \tag{J.1}$$

with $h = |x_{N_x} - x_0| / N_x$. The useful properties of these piecewise cubic polynomials (summarized in table J.1) show that the only cubic B-spline which are not vanishing at point x_i are $\Lambda_{i\pm1}(x_i) = 1/6$ and $\Lambda_i(x_i) = 2/3$. So the interpolating spline s is uniquely determined by the $(N_x + 1)$ interpolating conditions

$$g(x_i) = s(x_i) = \frac{1}{6}c_{i-1} + \frac{2}{3}c_i + \frac{1}{6}c_{i+1} \qquad i = 0, \cdots, N_x \tag{J.2}$$

and by 2 equations given by boundary conditions. In the case of periodic boundary conditions, these two equations are given by:

$$\begin{cases} g'(x_0) = g'(x_{N_x}) \\ g''(x_0) = g''(x_{N_x}) \end{cases} \Rightarrow \begin{cases} \frac{-1}{2h}c_{-1} + \frac{1}{2h}c_1 + \frac{1}{2h}c_{N_x-1} - \frac{1}{2h}c_{N_x+1} = 0 \\ \frac{1}{h^2}c_{-1} - \frac{2}{h^2}c_0 + \frac{1}{h^2}c_1 - \frac{1}{h^2}c_{N_x-1} + \frac{2}{h^2}c_{N_x} - \frac{1}{h^2}c_{N_x+1} = 0 \end{cases}$$

x	$x_{\alpha-2}$	$x_{\alpha-1}$	x_α	$x_{\alpha+1}$	$x_{\alpha+2}$
$\Lambda_\alpha(x)$	0	$1/6$	$2/3$	$1/6$	0
$\Lambda'_\alpha(x)$	0	$1/2h$	0	$-1/2h$	0
$\Lambda''_\alpha(x)$	0	$1/h^2$	$-2/h^2$	$1/h^2$	0

Table J.1: Values of the cubic spline function $\Lambda_\alpha(x)$ and its first and second derivative.

i.e,

$$\begin{cases} -c_{-1} + c_1 + c_{N_x-1} - c_{N_x+1} = 0 \\ c_{-1} - 2c_0 + c_1 - c_{N_x-1} + 2c_{N_x} - c_{N_x+1} = 0 \end{cases} \tag{J.3}$$

In the case of non-periodic boundary conditions, these two equations can be given by Hermite boundary conditions, as follow:

$$\begin{cases} g'(x_0) = s'(x_0) = -\frac{1}{2h}c_{-1} + \frac{1}{2h}c_1 \\ g'(x_{N_x}) = s'(x_{N_x}) = -\frac{1}{2h}c_{N_x-1} + \frac{1}{2h}c_{N_x+1} \end{cases} \tag{J.4}$$

or by boundary conditions based on the second derivatives:

$$\begin{cases} g''(x_0) = s''(x_0) = \frac{1}{h^2}c_{-1} - \frac{2}{h^2}c_0 + \frac{1}{h^2}c_1 \\ g''(x_{N_x}) = s''(x_{N_x}) = \frac{1}{h^2}c_{N_x-1} - \frac{2}{h^2}c_{N_x} + \frac{1}{h^2}c_{N_x+1} \end{cases}$$

The choice made in the code is the one based on Hermite boundary conditions. In the case of Neumann boundary conditions (first derivative equal to 0) $g'(x_0) = g'(x_{N_x}) = 0$ but in the general case (which is the most common in the code) there is no information on these derivatives. In our case these derivatives are approximated by a cubic Lagrange polynomial fit of g, i.e:

$$\lambda_{(3)}(x) = \sum_{i=0}^{3} g(x_i)L_i(x) \qquad \text{with} \qquad L_i(x) = \prod_{j=0, j\neq i}^{3} \left(\frac{x - x_i}{x_i - x_j}\right)$$

and then $g'(x_0) = \lambda'_{(3)}(x_0)$ and $g'(x_{N_x}) = \lambda'_{(3)}(x_{N_x})$ which gives the two following conditions:

$$\begin{cases} -\frac{1}{2h}c_{-1} + \frac{1}{2h}c_1 = -\frac{11}{6h}g(x_0) + \frac{3}{h}g(x_1) - \frac{3}{2h}g(x_2) + \frac{1}{3h}g(x_3) \\ -\frac{1}{2h}c_{N_x-1} + \frac{1}{2h}c_{N_x+1} = -\frac{1}{3h}g(x_{N_x-3}) + \frac{3}{2h}g(x_{N_x-2}) - \frac{3}{h}g(x_{N_x-1}) + \frac{11}{6h}g(x_{N_x}) \end{cases}$$

i.e

$$\begin{cases} -c_{-1} + c_1 = -\frac{11}{3}g(x_0) + 6g(x_1) - 3g(x_2) + \frac{2}{3}g(x_3) \\ -c_{N_x-1} + c_{N_x+1} = -\frac{2}{3}g(x_{N_x-3}) + 3g(x_{N_x-2}) - 6g(x_{N_x-1}) + \frac{11}{3}g(x_{N_x}) \end{cases} \tag{J.5}$$

In the both cases (J.2)+(J.3) or (J.2)+(J.5), the spline coefficients c_α are determined by solving the (N_x+3, N_x+3) matrix system

$$A\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} b \\ c \end{pmatrix} \qquad \text{with} \begin{cases} u = (c_0, \cdots, c_{N_x})^t \\ v = (c_{N_x+1}, c_{-1})^t \\ b = (g(x_0), \cdots, g(x_{N_x}))^t \\ c = (\sigma_1, \sigma_2)^t \end{cases} \qquad \text{and } A = \left(\begin{array}{c|c} A_{\text{tridiag}} & \gamma \\ \hline \lambda & \delta \end{array}\right) \tag{J.6}$$

where

$$\begin{cases} \cdot\ A_{\text{tridiag}} \text{ is the } (N_x+1)\times(N_x+1) \text{ tridiagonal symmetric matrix : } \begin{pmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ & & & 1 & 4 \end{pmatrix}, \\ \cdot\ \gamma \text{ is equal to the } (N_x+1)\times 2 \text{ matrix : } \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix}^t \end{cases}$$

and in the case of periodic boundary conditions (according to (J.3)):

$$\begin{cases} \cdot\, c = (\sigma_1, \sigma_2)^t = (0,0)^t, \\ \cdot\, \lambda = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 1 & 0 \\ -2 & 1 & 0 & \cdots & 0 & -1 & 2 \end{pmatrix} \text{ and} \\ \cdot\, \delta = \begin{pmatrix} \xi_1 & \xi_2 \\ \xi_3 & \xi_4 \end{pmatrix} = \begin{pmatrix} -1 & -1 \\ -1 & 1 \end{pmatrix} \end{cases}$$

while in the case of non-periodic boundary conditions (according to (J.4)):

$$\begin{cases} \cdot\, \sigma_1 = -\frac{2}{3} g(x_{N_x-3}) + 3g(x_{N_x-2}) - 6g(x_{N_x-1}) + \frac{11}{3} g(x_{N_x}), \\ \cdot\, \sigma_2 = -\frac{11}{3} g(x_0) + 6g(x_1) - 3g(x_2) + \frac{2}{3} g(x_3), \\ \cdot\, \lambda = \begin{pmatrix} 0 & \cdots & 0 & -1 & 0 \\ 0 & 1 & 0 & \cdots & 0 \end{pmatrix} \text{ and} \\ \cdot\, \delta = \begin{pmatrix} \xi_1 & \xi_2 \\ \xi_3 & \xi_4 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \end{cases}$$

The solving of the matrix system (J.6) is performed by taking advantage of the fact that A can be factorized in a LU form, like:

$$A = \begin{pmatrix} A_{\text{tridiag}} & 0 \\ \lambda & \bar{\delta} \end{pmatrix} \times \begin{pmatrix} I & A^{-1}_{\text{tridiag}}\gamma \\ 0 & I \end{pmatrix} \qquad \text{with} \qquad \bar{\delta} = \delta - \lambda A^{-1}_{\text{tridiag}}\gamma$$

With this LU factorization the matrix system is solved by forward and backward substitutions with the two successive sequences:

$$\begin{pmatrix} A_{\text{tridiag}} & 0 \\ \lambda & \bar{\delta} \end{pmatrix} \times \begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} b \\ c \end{pmatrix} \text{ and then } \begin{pmatrix} I & A^{-1}_{\text{tridiag}}\gamma \\ 0 & I \end{pmatrix} \times \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u' \\ v' \end{pmatrix}$$

So computation of the interpolation coefficients $\vec{c} = [c_{-1},\, c_0,\, \cdots,\, c_{N_x},\, c_{N_x+1}]$ can be summarized in the following steps:

1. Initialization:
   (a) Factorize and store A_{tridiag} in a LDL^t form,
   (b) Compute and store $A^{-1}_{\text{tridiag}}\gamma$ using the previous factorization and
   (c) Assemble the (2×2) matrix $\bar{\delta} = \delta - \lambda A^{-1}_{\text{tridiag}}\gamma$.
2. Time loop:
   (a) Compute and store $u' = A^{-1}_{\text{tridiag}} b$ using the stored factorization of A_{tridiag},
   (b) Assemble $c - \lambda A^{-1}_{\text{tridiag}} b$,
   (c) Solve the (2×2) system $\bar{\delta} v' = c - \lambda A^{-1}_{\text{tridiag}} b$ using the Cramer formula for $\bar{\delta}$ inverse computation $\bar{\delta}^{-1} = \frac{1}{\det(\bar{\delta})} \begin{pmatrix} \bar{\xi}_4 & -\bar{\xi}_2 \\ -\bar{\xi}_3 & \bar{\xi}_1 \end{pmatrix}$ and
   (d) Compute u using the previous storage of $A^{-1}_{\text{tridiag}}\gamma$ by $u = u' - A^{-1}_{\text{tridiag}}\gamma v$, where v is trivially equal to v'.

J.1.2 Cubic spline interpolation in 2D

Let $g(x, y)$ be a function defined in the x and y directions with $x \in [x_0, x_{N_x}]$ and $y \in [y_0, y_{N_y}]$ where N_x and N_y represent respectively the number of points in x and y directions. As for the previous 1D case, the function $g(x, y)$ can be interpolated with cubic B-splines as:

$$g(x, y) \simeq s(x, y) = \sum_{\alpha=-1}^{N_x+1} \sum_{\beta=-1}^{N_y+1} c_{\alpha,\beta} \Lambda_\alpha(x) \Lambda_\beta(y)$$

where the piecewise cubic polynomials Λ_α and Λ_β are defined by (J.1). Let assume g non-periodic in x direction and periodic in y direction. The interpolation coefficient $c_{\alpha,\beta}$ are computed by first solving the (N_y+1) unidimensional non-periodic systems (for $j=0,\cdots,N_y$) with Hermite boundary conditions ($\partial_x g(x_0,y)=\partial_x s(x_0,y)$ and $\partial_x g(x_{N_x},y)=\partial_x s(x_{N_x},y)$):

$$\begin{aligned}
\sum_{\alpha=-1}^{N_x+1}\gamma_\alpha(y_j)\Lambda_\alpha(x_i) &= g(x_i,y_j) \qquad \forall i=0,\cdots,N_x\\
-\gamma_{-1}(y_j)+\gamma_1(y_j) &= -\frac{11}{3}g(x_0,y_j)+6g(x_1,y_j)-3g(x_2,y_j)+\frac{2}{3}g(x_3,y_j)\\
-\gamma_{N_x-1}(y_j)+\gamma_{N_x+1}(y_j) &= -\frac{2}{3}g(x_{N_x-3},y_j)+3g(x_{N_x-2},y_j)-6g(x_{N_x+1},y_j)+\frac{11}{3}g(x_{N_x},y_j)
\end{aligned} \tag{J.7}$$

The second step consists in solving the (N_x+3) following unidimensional periodic systems (for $\alpha=-1,\cdots,N_x+1$):

$$\begin{aligned}
\gamma_\alpha(y_j) &= \sum_{\beta=-1}^{N_y+1}c_{\alpha,\beta}\Lambda_\beta(y_j) \qquad \forall j=0,\cdots,N_y\\
-\gamma_\alpha(y_{-1})+\gamma_\alpha(y_1)+\gamma_\alpha(y_{N_y-1})-\gamma_\alpha(y_{N_y+1}) &= 0\\
\gamma_\alpha(y_{-1})-2\gamma_\alpha(y_0)+\gamma_\alpha(y_1)-\gamma_\alpha(y_{N_y-1})+2\gamma_\alpha(y_{N_y})-\gamma_\alpha(y_{N_y+1}) &= 0
\end{aligned} \tag{J.8}$$

J.2 A new interpolation approach for Semi-Lagrangian scheme

Interpolation represents one of the steps of semi-Lagrangian numerical schemes which require much CPU time. It is most often performed using cubic splines. The drawback of splines is that the knowledge of the function on all the mesh nodes is required to interpolate this function on any single point. This global dependency conflicts with any tentative of competitive parallelization. The original approach presented in this appendix has been explicitly developed to overcome this problem (cf [CLS09]). The basic idea of this so-called *local spline* approach is to split the global domain in separated sub-domains, where standard cubic splines are applied ([ANW67, DeB01]). The difficulty then relies on the choice of appropriated boundary conditions for these sub-domains.

Domain decomposition in 1D

Let $g(x)$ be any function defined on the global domain $[x_0,x_{N_x}]\subset\mathbb{R}$. Let us split this domain into several sub-domains, each being associated to one processor. The sub-domain associated to processor p is generically called $[x_{p0},x_{pK}]$ with the notation $x_{pi}=x_{p0}+ih$, where h is the cell size $h=(x_{pK}-x_{p0})/K$ and $K\in\mathbb{N}$ the number of cells in the sub-domain. Let us now restrict the study of g to $g:x\mapsto g(x)$ on the interval $[x_{p0},x_{pK}]$. Boundary conditions are non-periodic in this sub-domain. The projection s_{loc} of g on the basis of cubic splines reads:

$$g(x)\simeq s_{\text{loc}}(x)=\sum_{\alpha=-1}^{K+1}c_\alpha\Lambda_\alpha(x)$$

with Λ_α the piecewise cubic polynomials defined as follows, for any integer α in the interval $-1\le\alpha\le K+1$:

$$\Lambda_\alpha(x)=\frac{1}{6h^3}\begin{cases}
(x-x_{\alpha-2})^3 & \text{if } x_{\alpha-2}\le x\le x_{\alpha-1}\\
h^3+3h^2(x-x_{\alpha-1})+3h(x-x_{\alpha-1})^2 & \\
\quad -3(x-x_{\alpha-1})^3 & \text{if } x_{\alpha-1}\le x\le x_\alpha\\
h^3+3h^2(x_{\alpha+1}-x)+3h(x_{\alpha+1}-x)^2 & \\
\quad -3(x_{\alpha+1}-x)^3 & \text{if } x_\alpha\le x\le x_{\alpha+1}\\
(x_{\alpha+2}-x)^3 & \text{if } x_{\alpha+1}\le x\le x_{\alpha+2}\\
0 & \text{otherwise}
\end{cases} \tag{J.9}$$

Using Hermite boundary conditions, the interpolated quantity s_{loc} is uniquely determined by the $(K+3)$ following equations:

$$g(x_{pi}) = s_{\text{loc}}(x_{pi}) = \frac{1}{6}c_{i-1} + \frac{2}{3}c_i + \frac{1}{6}c_{i+1} \qquad \forall i = 0, \cdots, K \tag{J.10}$$

$$g'(x_{pi}) = s'_{\text{loc}}(x_{pi}) = -\frac{1}{2h}c_{i-1} + \frac{1}{2h}c_{i+1} \qquad \text{for } i = 0 \text{ and } i = K \tag{J.11}$$

Finally, the spline coefficient vector $\mathbf{c} = [c_{-1}, \cdots, c_{K+1}]^T$ fulfills the $(K+3) \times (K+3)$ linear system $A\mathbf{c} = \mathbf{g}$, where $\mathbf{g}$ is the vector $\mathbf{g} = [g'(x_{p0}), g(x_{p0}), \cdots, g(x_{pK}), g'(x_{pK})]$ and A the matrix:

$$A = \frac{1}{6}\begin{pmatrix} -3/h & 0 & 3/h & 0 & \cdots & 0 \\ 1 & 4 & 1 & & & \\ & \ddots & \ddots & \ddots & & \\ & & & 1 & 4 & 1 \\ 0 & \cdots & 0 & -3/h & 0 & 3/h \end{pmatrix}$$

Notice that the A matrix can easily be factorized in LU form, where L is a lower triangular matrix with all diagonal terms equal to 1, and U is an upper triangular matrix. The standard algorithm leading to LU factorization is based on Gauss elimination [1]:

$$L = \begin{pmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ -h/3 & 1 & 0 & \ddots & & \vdots \\ 0 & l_1 & 1 & \ddots & & \\ 0 & 0 & \ddots & \ddots & 0 & \vdots \\ \vdots & \ddots & \ddots & l_K & 1 & 0 \\ 0 & \cdots & 0 & -(3l_K)/h & (3l_{K+1})/h & 1 \end{pmatrix} \quad \text{and} \quad U = \frac{1}{6}\begin{pmatrix} -3/h & 0 & 3/h & 0 & \cdots & 0 \\ 0 & d_1 & 2 & 0 & & \vdots \\ 0 & 0 & d_2 & 1 & \ddots & \vdots \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & d_{K+1} & 1 \\ 0 & \cdots & 0 & 0 & 0 & (3d_{K+2})/h \end{pmatrix}$$

where l_i and d_i are obtained as follows:
$d_1 = 4,\ l_1 = 1/4,\ d_2 = 4 - 2l_1 = 7/2,$
for $i = 2, K$
. $l_i = 1/d_i,$
. $d_{i+1} = 4 - l_i,$
end for
$l_{K+1} = 1/d_{K+1},\ d_{K+2} = 1 - l_{K+1}$
The LU decomposition of A needs only be computed ones, at the beginning of the numerical code.

[1] Let L and U be defined by

$$L = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ l_{2,1} & 1 & 0 & \cdots & 0 \\ l_{3,1} & l_{3,2} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ l_{n,1} & l_{n,2} & \cdots & l_{n,n} & 1 \end{pmatrix} \quad \text{and} \quad U = \begin{pmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n} \\ 0 & u_{2,2} & \cdots & u_{2,n} \\ 0 & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & u_{n,n} \end{pmatrix}$$

The algorithm of LU factorization reads as follows:
$u_{1,1} = a_{1,1}$
for $j = 2, \cdots, n$
. $u_{1,j} = a_{1,j}$
. $l_{j,1} = a_{j,1}/a_{1,1}$
end for
for $i = 2, \cdots, n-1$
. $u_{i,i} = a_{i,i} - \sum_{k=1}^{i-1} l_{i,k}\, u_{k,i}$
. for $j = i+1, \cdots, n$
. $\quad u_{i,j} = a_{i,j} - \sum_{k=1}^{i-1} l_{i,k}\, u_{k,j}$
. $\quad l_{i,j} = \frac{1}{u_{i,i}}\left[a_{j,i} - \sum_{k=1}^{i-1} l_{j,k}\, u_{k,i}\right]$
. end for
end for
$u_{n,n} = a_{n,n} - \sum_{k=1}^{n-1} l_{n,k}\, u_{k,n}$

At each time step, the matrix system $A\mathbf{c} = \mathbf{g}$ is then solved in two steps. The lower triangular matrix system $L\mathbf{x} = \mathbf{g}$ is first solved. The second step then consists in solving the upper triangular matrix system $U\mathbf{c} = \mathbf{x}$.

Approximation of the interface derivatives

One of the most delicate points in the local cubic splines approach is to provide "good" approximations of the interface derivatives $g'(x_{p0})$ and $g'(x_{pK})$. Good approximations correspond to minimal error between the local and the global splines. Several classical solutions, like finite differences of various orders, or ad-hoc cubic spline approximations, have been explored in reference [CLS09]. These authors show that the most robust approximation which remains valid with a relatively small number of grid points employs special combinations of cubic spline coefficients. Let us detail this method. According to Eq. (J.11), $c_{i-1} = \frac{3}{2}g_{i-1} - \frac{1}{4}c_{i-2} - \frac{1}{4}c_i$ and $c_{i+1} = \frac{3}{2}g_{i+1} - \frac{1}{4}c_i - \frac{1}{4}c_{i+2}$ where g_{i+1} (resp. g_{i-1}) represents the value of g at point $x_{p(i+1)}$ (resp. $x_{p(i-1)}$). Replacing c_{i-1} and c_{i+1} by these expressions in Eq. (J.10) leads to the iterative equality:

$$s'_i = \frac{3}{4h}(g_{i+1} - g_{i-1}) - \frac{1}{4}(s'_{i+1} + s'_{i-1}) \tag{J.12}$$

with the notation $s'_i = s'_{\text{loc}}(x_{pi})$. s'_{i+1} and s'_{i-1} can be computed using the same procedure. These derivatives involve s'_{i+2} and s'_{i-2}. Injecting these expressions in Eq. (J.12) then yields:

$$s'_i = \frac{6}{7h}(g_{i+1} - g_{i-1}) - \frac{3}{14h}(g_{i+2} - g_{i-2}) + \frac{1}{14}(s'_{i+2} + s'_{i-2}) \tag{J.13}$$

Again, s'_{i+2} and s'_{i-2} can be evaluated from Eq. (J.13), provided s'_{i+4} and s'_{i-4} are introduced. Incorporating these expressions in Eq. (J.13) then leads to the following approximation of s'_i:

$$\begin{aligned} s'_i = &\frac{39}{49\alpha h}(g_{i+1} - g_{i-1}) - \frac{3}{14\alpha h}(g_{i+2} - g_{i-2}) + \frac{3}{49\alpha h}(g_{i+3} - g_{i-3}) \\ &- \frac{3\gamma}{h}(g_{i+4} - g_{i-4}) + \gamma(s'_{i+4} + s'_{i-4}) \end{aligned}$$

where $\alpha = 1 - 2/14^2$ and $\gamma = 1/(14^2\alpha)$. The procedure is iterated once more, leading to the final retained approximation for s'_i, namely:

$$\begin{aligned} \beta h\, s'_i = &\frac{1}{49\alpha}(39 - 3\gamma)(g_{i+1} - g_{i-1}) - \frac{3}{14\alpha}(1 - \gamma)(g_{i+2} - g_{i-2}) \\ &+ \frac{1}{49\alpha}(3 - 39\gamma)(g_{i+3} - g_{i-3}) - 3\gamma(g_{i+4} - g_{i-4}) \\ &+ \frac{39\gamma}{49\alpha}(g_{i+5} - g_{i-5}) - \frac{3\gamma}{14\alpha}(g_{i+6} - g_{i-6}) + \frac{3\gamma}{49\alpha}(g_{i+7} - g_{i-7}) \\ &+ 3\gamma^2(g_{i+8} - g_{i-8}) - \gamma^2 h(s'_{i+8} + s'_{i-8}) \end{aligned}$$

with $\beta = 1 - 2\gamma^2$. So as to close the system, s'_{i+8} and s'_{i-8} are evaluated using 4th order finite differences:

$$\begin{cases} s'_{i+8} = (g_{i+6} - 8g_{i+7} + 8g_{i+9} - g_{i+10})/12h \\ s'_{i-8} = (g_{i-10} - 8g_{i-9} + 8g_{i-7} - g_{i-6})/12h \end{cases}$$

For instance, when using 20 values of g, the approximation of $s'_{\text{loc}}(x_{pi})$ is given by (notice that $w_0 = 0$):

$$s'_{\text{loc}}(x_{pi}) = \sum_{j=1}^{10} w_j \frac{(g_{i+j} - g_{i-j})}{h}$$

where the w_j coefficients are given in table J.2 This approach can be viewed as the generalization of the standard cubic splines. Indeed, should the number of sub-domains be equal to one, corresponding actually to the full domain, this technique turns out to be equivalent to cubic splines.

w_1	w_2	w_3	w_4
$15126/18817$ 0.8038475846	$-4053/18817$ -0.2153903385	$1086/18817$ 0.05771376946	$-291/18817$ -0.01546473933
w_5	w_6	w_7	w_8
$78/18817$ $4.145187862E-3$	$-503/451608$ $-1.113797807E-3$	$17/56451$ $3.011461267E-4$	$-3/37634$ $-7.971515119E-5$
w_9	w_{10}		
$1/56451$ $1.771447804E-5$	$-1/451608$ $-2.214309755E-6$		

Table J.2: Coefficients for the approximation of the interface derivatives

Appendix K

Gysela normalized equations

This appendix contains expressions of all the normalized quantities and normalized equations implemented in the code GYSELA . The second part corresponds to the detailed proofs of the normalized equations. They are just calculation exercises. Therefore, part K.2 can be skipped by all readers not interested in the precise form of each terms involved in GYSELA equations.

K.1 Gysela normalization

Let us first briefly remind the normalization used in GYSELA code. For this let us consider $m_0 = A_0 m_p$ (Kilogram) a reference ion mass, $q_0 = Z_0 e$ (Coulomb) a reference ion charge, B_0 (Tesla) a reference magnetic induction and T_0 (eV) a reference thermal velocity. A_0 and Z_0 are the (dimensionless) mass number and charge state of the main ion species and e the modulus of the electron charge. These quantities are used to define the reference ion cyclotron frequency Ω_{c_0}, the reference thermal speed v_{T_0} and the reference Larmor-radius ρ_0 as

$$\Omega_{c_0} = \frac{Z_0\, e\, B_0}{m_0} \quad ; \quad v_{T_0} = \sqrt{\frac{T_0}{m_0}} \quad ; \quad \rho_0 = \frac{v_{T_0}}{\Omega_{c_0}} = \frac{\sqrt{T_0 m_0}}{Z_0\, e\, B_0} \tag{K.1}$$

Actually, the seven normalizing quantities $(A_0, Z_0, n_0, T_0, B_0, a, R_0)$ are not completely free. Three dimensionless parameters are further specified in the input data, which provide relationships between these seven parameters. These are the aspect ratio $A = R_0/a$, rhostar $\rho_{*0} = \sqrt{A_0 m_p T_0}/Z_0 e B_0$ and nustar $\nu_{*0} = \varepsilon^{-3/2}(qR_0/v_{T0})\,\nu_{coll.0} \sim Z_0^4 n_0 R_0/T_0^2$ (where ε and q are taken at $r_{\rm peak} = r_{\rm min} + 0.5(r_{\rm max} - r_{\rm min})$). It follows that, given A, ρ_{*0} and ν_{*0}, it remains only four free normalizing quantities. Hereafter, $\hat{X}$ refers to the dimensionless expression of the X quantity. The normalizations used in the code are summarized in the table K.1. Notice that the parallel velocity is normalized to $v_{T_{s0}}$ while all drift

$$\boxed{\begin{array}{rcl} m_s &=& m_0 \hat{A}_s \\ q_s &=& Z_0\, e \hat{Z}_s \\ l &=& \rho_0 \hat{l} \\ t &=& \hat{t}/\Omega_{c_0} \\ n_s &=& n_0\, \hat{n}_s \\ T_s &=& T_0 \hat{T}_s \\ B &=& B_0 \hat{B} \end{array}} \quad \Rightarrow \quad \boxed{\begin{array}{rcl} v_{G\|} &=& v_{T_{s0}} \hat{v}_{G\|s} \\ &=& (v_{T_0}/\sqrt{\hat{A}_s}) \hat{v}_{G\|s} \\ v_{E\times B_s} &=& v_{T_0} \hat{v}_{E\times B_s} \\ v_{D_s} &=& v_{T_0} \hat{v}_{D_s} \\ \mu_s &=& (T_0/B_0) \hat{\mu}_s \\ U &=& [T_0/(Z_0\, e)] \hat{\phi} \\ E &=& (v_{T_0} B_0) \hat{E} \\ F_s &=& (n_0/v_{T_{s0}}^3) \hat{F}_s \end{array}} \quad \text{and} \quad \boxed{\begin{array}{rcl} \chi &=& \rho_0^2\, \Omega_{c_0}\, \hat{\chi} \\ \nu &=& \Omega_{c_0}\, \hat{\nu} \\ \mathcal{D}_{\|s} &=& v_{T_{s0}}^2\, \Omega_{c_0}\, \hat{\mathcal{D}}_{\|s} \\ \mathcal{V}_{\|s} &=& v_{T_{s0}}\, \Omega_{c_0}\, \hat{\mathcal{V}}_{\|s} \\ S_0^{\mathcal{E}} &=& n_0 \Omega_{C_0} T_0 \hat{S}_0^{\mathcal{E}} \\ S_0^{v_{G\|}} &=& n_0 \Omega_{C_0} v_{T_{s0}} \hat{S}_0^{v_{G\|}} \\ S_0^{\Omega} &=& \frac{n_0 \Omega_{C_0}}{\rho_0^2} \hat{S}_0^{\Omega} \end{array}}$$

Table K.1: Links between physical and normalized quantities.

velocities are normalized to v_{T_0}. This appendix is divided in two parts. The first part contains a summarize of all the normalized equations used in GYSELA . The second part contains the details of

the computation for obtaining these normalized expressions.
The normalized system of equations is made of the 5D gyrokinetic equation (including source terms and collisions) self-consistently coupled to the 3D quasi-neutrality. The gyrokinetic equation involves Poisson brackets $[\cdot,\cdot]$. Introducing the unit vector $\mathbf{b} = \mathbf{B}/\|B\|$ along the magnetic field, its covariant components b_k, and the jacobien $\mathcal{J}_{\mathrm{x}}$ of the configuration space, these brackets read as follows:

$$[F,G] = \mathbf{b}\cdot(\boldsymbol{\nabla}F\times\boldsymbol{\nabla}G) = \mathcal{J}_{\mathrm{x}}^{-1}\epsilon^{ijk}\partial_i F\partial_j G b_k \tag{K.2}$$

K.1.1 Normalized Boltzmann equation

The evolution of the guiding-center distribution function $\hat{\bar{F}}_s$ is governed by the following normalized Boltzmann equation:

$$\frac{\partial\hat{\bar{F}}_s}{\partial\hat{t}} + \frac{1}{\hat{B}^*_{\|s}}\hat{\boldsymbol{\nabla}}\cdot\left(\hat{B}^*_{\|s}\frac{\mathrm{d}\hat{\mathbf{x}}_G}{\mathrm{d}\hat{t}}\hat{\bar{F}}_s\right) + \frac{1}{\hat{B}^*_{\|s}}\frac{\partial}{\partial\hat{v}_{G\|s}}\left(\hat{B}^*_{\|s}\frac{\mathrm{d}\hat{v}_{G\|s}}{\mathrm{d}\hat{t}}\hat{\bar{F}}_s\right) = \hat{\mathcal{D}}_r(\hat{\bar{F}}_s) + \hat{\mathcal{K}}(\hat{\bar{F}}_s) + \hat{\mathcal{C}}(\hat{\bar{F}}_s) + \hat{\mathcal{S}} \tag{K.3}$$

where $\hat{\mathcal{D}}_r$ and $\hat{\mathcal{K}}$ are respectively a diffusion term and a Krook operator applied on a radial buffer region (see section K.1.3), while $\hat{\mathcal{C}}$ refers to a collision operator (see section K.1.4) and $\hat{\mathcal{S}}$ corresponds to source terms (see section K.1.6 for detailed expressions). The evolution of the gyro-center coordinates $(\mathbf{x}_G, v_{G\|}, \mu_s)$ of species s is given by:

$$\frac{\mathrm{d}\hat{x}^i_G}{\mathrm{d}\hat{t}} = \frac{1}{\sqrt{\hat{A}_s}}\hat{v}_{G\|s}\hat{\mathbf{b}}^*_s\cdot\hat{\boldsymbol{\nabla}}x^i + \hat{\mathbf{v}}_{E\times B_s}\cdot\hat{\boldsymbol{\nabla}}x^i + \hat{\mathbf{v}}_{D_s}\cdot\hat{\boldsymbol{\nabla}}x^i \tag{K.4}$$

$$\frac{\mathrm{d}\hat{v}_{G\|s}}{\mathrm{d}\hat{t}} = -\frac{1}{\sqrt{\hat{A}_s}}\hat{\mu}_s\hat{\mathbf{b}}^*_s\cdot\hat{\boldsymbol{\nabla}}\hat{B} - \frac{\hat{Z}_s}{\sqrt{\hat{A}_s}}\hat{\mathbf{b}}^*_s\cdot\hat{\boldsymbol{\nabla}}\hat{\bar{\phi}} + K_{\nabla B}\,\hat{v}_{G\|s}\hat{\mathbf{v}}_{E\times Bs}\cdot\frac{\hat{\boldsymbol{\nabla}}\hat{B}}{\hat{B}} \tag{K.5}$$

The i-th covariant coordinates of the normalized drift velocities are given by:

$$\hat{\mathbf{v}}_{E\times B_s}\cdot\hat{\boldsymbol{\nabla}}x^i = \hat{\mathbf{v}}^i_{E\times B_s} = \frac{1}{\hat{B}^*_{\|s}}\left[\hat{\bar{\phi}},\hat{x}^i\right] \tag{K.6}$$

and

$$\hat{\mathbf{v}}_{D_s}\cdot\hat{\boldsymbol{\nabla}}x^i = \hat{\mathbf{v}}^i_{D_s} = K_{\nabla B}\left(\frac{\hat{v}^2_{G\|s} + \hat{\mu}_s\hat{B}}{\hat{Z}_s\hat{B}^*_{\|s}\hat{B}}\right)\left[\hat{B},\hat{x}^i\right] \tag{K.7}$$

while $\hat{\mathbf{b}}^*_s$ and $\hat{B}^*_{\|s}$ are defined as:

$$\hat{\mathbf{b}}^*_s = \frac{1}{\hat{B}^*_{\|s}}\left(\hat{\mathbf{B}} + \frac{\sqrt{\hat{A}_s}}{\hat{Z}_s}\frac{\hat{v}_{G\|s}}{\hat{B}}\hat{\mathbf{J}}\right) \quad \text{and} \quad \hat{B}^*_{\|s} = \hat{B} + \frac{\sqrt{\hat{A}_s}}{\hat{Z}_s}\frac{\hat{v}_{G\|s}}{\hat{B}}\mathbf{b}\cdot\hat{\mathbf{J}} \tag{K.8}$$

The normalized magnetic field $\hat{B}$ is defined as

$$\hat{B} = \frac{\hat{R}_0}{\hat{R}(r,\theta)}\left[\zeta(\hat{r})\mathbf{e}_\theta + \mathbf{e}_\varphi\right] \quad \text{with} \quad \zeta(\hat{r}) = \frac{\hat{r}}{q\hat{R}_0} \tag{K.9}$$

while the normalized current reads

$$\hat{\mathbf{J}} = \hat{J}_T\hat{R}\hat{\boldsymbol{\nabla}}\varphi \quad \text{with} \quad \hat{J}_T = \frac{\hat{R}_0}{\hat{R}}\frac{\zeta}{\hat{r}}\left(1 + \frac{\hat{r}}{\zeta}\frac{\mathrm{d}\zeta}{\mathrm{d}\hat{\mathrm{r}}} - \frac{\hat{r}}{\hat{R}}\cos\theta\right) \tag{K.10}$$

The constant $K_{\nabla B}$ has been added for tests in the code. $K_{\nabla B}$ is equal to 1 if the curvature of the magnetic field is taken into account and 0 otherwise. In equations (K.4) and (K.5) the parallel projection $\hat{\boldsymbol{\nabla}}^*_{\|}F$ is computed as $\hat{\mathbf{b}}^*_s\cdot\hat{\boldsymbol{\nabla}}F = \hat{b}^{*i}\hat{\partial}_i F$. In equations (K.6) and (K.7), the Poisson brackets are defined as $[F,G] = \hat{\mathbf{b}}\cdot\left(\hat{\boldsymbol{\nabla}}F\times\hat{\boldsymbol{\nabla}}G\right)$ (detailed expressions are given by (K.11)-(K.13) and (K.14)-(K.16)).

Expression of the Poisson Bracket in GYSELA code

For readability, the hat symbols are omitted for this paragraph. Using the Poisson bracket definition (B.2) and its explicit form (B.3), then the three components of $[\bar{\phi}, x^i]$ are given by:

$$\left[\bar{\phi}, x^1\right] = \frac{1}{\mathcal{J}_\mathrm{x}} \left\{ b_2\, \partial_{x^3} \bar{\phi} - b_3\, \partial_{x^2} \bar{\phi} \right\} = \frac{1}{\mathcal{J}_\mathrm{x} B} \left\{ B_2\, \partial_{x^3} \bar{\phi} - B_3\, \partial_{x^2} \bar{\phi} \right\} \tag{K.11}$$

$$\left[\bar{\phi}, x^2\right] = \frac{1}{\mathcal{J}_\mathrm{x}} \left\{ -b_1\, \partial_{x^3} \bar{\phi} + b_3\, \partial_{x^1} \bar{\phi} \right\} = \frac{1}{\mathcal{J}_\mathrm{x} B} \left\{ -B_1\, \partial_{x^3} \bar{\phi} + B_3\, \partial_{x^1} \bar{\phi} \right\} \tag{K.12}$$

$$\left[\bar{\phi}, x^3\right] = \frac{1}{\mathcal{J}_\mathrm{x}} \left\{ b_1\, \partial_{x^2} \bar{\phi} - b_2\, \partial_{x^1} \bar{\phi} \right\} = \frac{1}{\mathcal{J}_\mathrm{x} B} \left\{ B_1\, \partial_{x^2} \bar{\phi} - B_2\, \partial_{x^1} \bar{\phi} \right\} \tag{K.13}$$

Besides, in the case where B only depends on x^1 and x^2, $[B, x^i]$ is defined as:

$$\begin{aligned}\left[B, x^i\right] &= \frac{1}{\mathcal{J}_\mathrm{x}} \left\{ b_1\, \partial_{x^2} B\, \partial_{x^3} x^i - b_2\, \partial_{x^1} B\, \partial_{x^3} x^i + b_3 \left(\partial_{x^1} B\, \partial_{x^2} x^i - \partial_{x^2} B\, \partial_{x^1} x^i \right) \right\} \\ &= \frac{1}{\mathcal{J}_\mathrm{x} B} \left\{ B_1\, \partial_{x^2} B\, \partial_{x^3} x^i - B_2\, \partial_{x^1} B\, \partial_{x^3} x^i + B_3 \left(\partial_{x^1} B\, \partial_{x^2} x^i - \partial_{x^2} B\, \partial_{x^1} x^i \right) \right\}\end{aligned}$$

which corresponds for each components to:

$$\left[B, x^1\right] = -\frac{1}{\mathcal{J}_\mathrm{x}} \times b_3\, \partial_{x^2} B = -\frac{1}{\mathcal{J}_\mathrm{x} B} \times B_3\, \partial_{x^2} B \tag{K.14}$$

$$\left[B, x^2\right] = \frac{1}{\mathcal{J}_\mathrm{x}} \times b_3\, \partial_{x^1} B = \frac{1}{\mathcal{J}_\mathrm{x} B} \times B_3\, \partial_{x^1} B \tag{K.15}$$

$$\left[B, x^3\right] = \frac{1}{\mathcal{J}_\mathrm{x}} \left(b_1\, \partial_{x^2} B - b_2\, \partial_{x^1} B \right) = \frac{1}{\mathcal{J}_\mathrm{x} B} \left(B_1\, \partial_{x^2} B - B_2\, \partial_{x^1} B \right) \tag{K.16}$$

In the case where B depends on the three coordinates x^1, x^2 and x^3 (i.e the ripple is taken into account), the Poisson bracket $\left[B, x^i\right]$ is defined by (K.11) to (K.13) by replacing $\bar{\phi}$ by B. And finally, $\left[B, \bar{\phi}\right]$ is given by (B.3) by replacing F by B and G by $\bar{\phi}$.

K.1.2 Normalized quasi-neutrality equation

The normalized quasi-neutrality equation is

$$-\frac{1}{\hat{n}_{e_0}} \sum_s \hat{A}_s \hat{\nabla}_\perp \cdot \left(\frac{\hat{n}_{s_0}}{\hat{B}_0} \hat{\nabla}_\perp \hat{\phi} \right) + \frac{1}{Z_0^2 \hat{T}_e} \left[\hat{\phi} - \lambda \langle\, \hat{\phi}\, \rangle_\mathrm{FS} \right] = \frac{1}{\hat{n}_{e_0}} \sum_s \hat{Z}_s \left[\hat{n}_{G_s} - \hat{n}_{G_s,\mathrm{eq}} \right] \tag{K.17}$$

where the normalized electron density $\hat{n}_{e_0}$ is defined as $\hat{n}_{e_0} = \sum_s Z_0 \hat{Z}_s \hat{n}_{s_0}$. Notice that, in the polarization term (first term of eq. (K.17)), $\hat{B}$ has been replaced by $\hat{B}_0 = 1$. The integral $\langle\, \hat{\phi}\, \rangle_\mathrm{FS} = \int \hat{\phi} \hat{\mathcal{J}}_\mathrm{x}\ \mathrm{d}\theta\, \mathrm{d}\varphi / \int \hat{\mathcal{J}}_\mathrm{x}\ \mathrm{d}\theta\, \mathrm{d}\varphi$ represents the flux surface average of $\hat{\phi}$ (with $\hat{\mathcal{J}}_\mathrm{x} = 1/(\hat{B} \cdot \hat{\nabla}\theta)$ being the normalized jacobian space). The parameter λ has been added for tests. It can be chosen equal to 1 or 0. The normalized guiding-center density $\hat{n}_{G_s}$ of species s is given by:

$$\hat{n}_{G_s} = \int \mathrm{d}\hat{\mu}_s \int \hat{\mathcal{J}}_\mathrm{v}\, \mathrm{d}\hat{v}_{G\|s}\, \hat{J}_{0_s} \cdot \hat{\bar{F}}_s \tag{K.18}$$

with the normalized jacobian in velocity equal to $\hat{\mathcal{J}}_\mathrm{v} = 2\pi \hat{B}^*_{\|s}$. The correction term $\hat{n}_{G_s,\mathrm{eq}}$ in the right hand side is defined as follows:

$$\hat{n}_{G_s,\mathrm{eq}} = \int \mathrm{d}\hat{\mu}_s \int \hat{\mathcal{J}}_\mathrm{v}\, \mathrm{d}\hat{v}_{G\|s}\, \hat{J}_{0_s} \cdot \hat{\bar{F}}_{s,\mathrm{eq}} \tag{K.19}$$

where the normalized equilibrium distribution function is defined as

$$\hat{\bar{F}}_{s,\mathrm{eq}} = c_s \frac{\hat{n}_{s_0}}{(2\pi \hat{T}_s)^{3/2}} \exp\left[-\frac{\left(\hat{v}^2_{G\|s}/2 + \hat{\mu}_s \hat{B} \right)}{\hat{T}_s} \right] \tag{K.20}$$

represents the equilibrium part of the distribution function. The concentration c_s is such that $\sum_s c_s Z_0 \hat{Z}_s = 1$. The normalized gyro-average operator $\hat{J}_{0_s}$ approximated by Padé corresponds to:

$$\hat{J}_{0_s} \approx 1 + \frac{1}{2}\frac{\hat{A}_s}{\hat{Z}_s^2}\frac{\hat{\mu}_s}{\hat{B}}\hat{\nabla}_\perp^2 \tag{K.21}$$

where, as in the quasi-neutrality equation, $\hat{B}$ is replaced by $\hat{B}_0 = 1$ in the code. Let us notice that in the code, to avoid the expensive gyro-average operation for each value of $v_{G\|}$ –which occur for the computation of the RHS of the quasi-neutrality equation (K.17)– we use the definition (K.8) of $B^*_{\|s}$ and therefore the fact that the term $\hat{n}_{G_s} - \hat{n}_{G_s,\mathrm{eq}}$ can be expressed as:

$$\hat{n}_{G_s} - \hat{n}_{G_s,\mathrm{eq}} = 2\pi \int \mathrm{d}\hat{\mu}_s \left(\hat{B}\, \hat{J}_{0_s} \cdot I_0(\hat{r},\theta,\varphi,\hat{\mu}_s) + \frac{\sqrt{\hat{A}_s}}{\hat{Z}_s}\frac{\hat{v}_{G\|s}}{\hat{B}}\mathbf{b}\cdot\hat{\mathbf{J}}\, \hat{J}_{0_s} \cdot I_1(\hat{r},\theta,\varphi,\hat{\mu}_s)\right)$$

where the integrals I_0 and I_1 are defined by:

$$I_0(\hat{r},\theta,\varphi,\hat{\mu}_s) = \int \left(\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\mathrm{eq}}\right) \mathrm{d}\hat{v}_{G\|s} \quad \text{and} \quad I_1(\hat{r},\theta,\varphi,\hat{\mu}_s) = \int \hat{v}_{G\|s}\left(\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\mathrm{eq}}\right) \mathrm{d}\hat{v}_{G\|s}$$

K.1.3 Normalized diffusion terms and Krook operator in buffer regions

A radial diffusion and an artificial damping are applied in buffer regions. These buffer regions are defined at each boundaries of the radial domain $\hat{r} \in [\hat{r}_{\min}, \hat{r}_{\max}]$ as a sum of hyperbolic tangents:

$$\hat{H}_{\mathrm{buff}}(\hat{r}) = 1 + \frac{1}{2}\left[\tanh\left(\frac{\hat{r} - \hat{r}_{\max} + B_L\,\hat{L}_r}{B_S\,\hat{L}_r}\right) - \tanh\left(\frac{\hat{r} - \hat{r}_{\min} - B_L\,\hat{L}_r}{B_S\,\hat{L}_r}\right)\right] \tag{K.22}$$

where $\hat{L}_r$ is the normalized length of the radial domain. B_L and B_S are respectively the location and the stiffness of the buffer regions. The function $\hat{H}_{\mathrm{buff}}$ plays the role of a mask which is equal to 1 in the buffer regions and equal to 0 elsewhere.
A radial diffusion is applied on this buffer region by solving the following equation

$$\frac{\partial \hat{\bar{F}}_s}{\partial \hat{t}} = \hat{\mathcal{D}}_r(\hat{\bar{F}}_s) \quad \text{with} \quad \hat{\mathcal{D}}_r(\hat{\bar{F}}_s) = \frac{1}{\hat{r}\hat{B}^*_{\|s}}\frac{\partial}{\partial \hat{r}}\left(\hat{r}\hat{\chi}(\hat{r})\frac{\partial}{\partial \hat{r}}\left(\hat{B}^*_{\|s}\hat{\bar{F}}_s\right)\right) \tag{K.23}$$

where $\hat{\chi}(\hat{r}) = \hat{\chi}_0 \hat{H}_{\mathrm{buff}}(\hat{r})$, where the normalized magnitude $\hat{\chi}_0$ of the diffusion coefficient in the buffer region is $\hat{\chi}_0 = \chi_0/\rho_0^2\Omega_{c_0}$.
A Krook operator is applied by solving

$$\frac{\partial \hat{\bar{F}}_s}{\partial \hat{t}} = \hat{\mathcal{K}}_r(\hat{\bar{F}}_s) \quad \text{with} \quad \hat{\mathcal{K}}_r(\hat{\bar{F}}_s) = -\hat{\nu}(\hat{r})(\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\mathrm{eq}}) \tag{K.24}$$

where $\hat{\nu}(\hat{r}) = \hat{\nu}_0 \hat{H}_{\mathrm{buff}}(\hat{r})$ such that $\hat{\nu}(\hat{r}) = \nu(r)/\Omega_{c_0}$.

K.1.4 Normalized collision operator

A collision operator $\mathcal{C}_s$ is present on the RHS of the normalized Vlasov equation (K.3), such that the normalized Boltzmann equation reads:

$$\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} = \hat{\mathcal{C}}_s(\hat{\bar{F}}_s) \quad \text{with} \quad \hat{\mathcal{C}}_s(\hat{\bar{F}}_s) = \frac{1}{\hat{B}^*_{\|s}}\frac{\partial}{\partial \hat{v}_{G\|s}}\left[\hat{B}^*_{\|s}\left(\hat{\mathcal{D}}_{\|s}\frac{\partial \hat{\bar{F}}_s}{\partial \hat{v}_{G\|s}} - \hat{\mathcal{V}}_{\|s}\hat{\bar{F}}_s\right)\right] \tag{K.25}$$

where the diffusion term $\hat{\mathcal{D}}_{\|s}$ and the drag term $\hat{\mathcal{V}}_{\|s}$ in the parallel direction are defined as:

$$\hat{\mathcal{D}}_{\|s}(\hat{r},\hat{v}) = \hat{\mathcal{A}}_s(\hat{r})\left(\frac{\Phi(\hat{v}) - G(\hat{v})}{2\hat{v}}\right) \quad ; \quad \hat{\mathcal{V}}_{\|s}(\hat{r},\hat{v}) = -\frac{\left(\hat{v}_{G\|s} - \hat{V}_{\|s,\mathrm{coll}}\right)}{\hat{v}^2_{T_{s,\mathrm{coll}}}}\hat{\mathcal{D}}_{\|s}(\hat{r},\hat{v}) \tag{K.26}$$

where

$$\hat{v} = \sqrt{\frac{1}{\hat{T}_{s,\mathrm{coll}}}\left(\frac{1}{2}\hat{v}_{G\|s}^2 + \hat{\mu}_s \hat{B}_{\mathrm{axis}}\right)} \tag{K.27}$$

while the Chandrasekhar function G is defined as follows:

$$G(\hat{v}) = \frac{\Phi(\hat{v}) - \hat{v}\Phi'(\hat{v})}{2\hat{v}^2} \quad ; \quad \Phi(\hat{v}) = \frac{2}{\sqrt{\pi}}\int_0^{\hat{v}} e^{-x^2}\mathrm{d}x \quad ; \quad \Phi'(\hat{v}) = \frac{2}{\sqrt{\pi}}e^{-\hat{v}^2} \tag{K.28}$$

The radial profile $\hat{\mathcal{A}}_s$ is given the form

$$\hat{\mathcal{A}}_s(\hat{r}) = \frac{1}{\sqrt{\hat{A}_s}}\left(3\frac{\sqrt{\pi}}{2}\frac{\hat{v}_{T_{s,\mathrm{coll}}}^3 \epsilon^{3/2}}{q(\hat{r})\hat{R}_0}\right)\hat{\nu}_{*s} \tag{K.29}$$

The normalized collisionality $\hat{\nu}_{*s}$ is an input data. The normalized collisional frequency is then obtained as follows:

$$\hat{\nu}_{ss} = \frac{1}{\sqrt{\hat{A}_s}}\left(\frac{\hat{v}_{T_{s,\mathrm{coll}}}\epsilon^{3/2}}{q(\hat{r})\hat{R}_0}\right)\hat{\nu}_{*s} \tag{K.30}$$

Considering that the species s is the major ion species, the collisionality $\hat{\nu}_{*s'}$ of each minority ion species s' is determined by

$$\hat{\nu}_{*s'} = \left(\frac{\hat{n}_{s'}}{\hat{n}_s}\right)\left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^4\left(\frac{\hat{T}_{s,\mathrm{coll}}}{\hat{T}_{s',\mathrm{coll}}}\right)^2\hat{\nu}_{*s} \tag{K.31}$$

In practice, $\hat{T}_{s,\mathrm{coll}}/\hat{T}_{s',\mathrm{coll}}$ is approximated by $\hat{T}_s/\hat{T}_{s'}$. The ratio $\hat{n}_{s'}/\hat{n}_s$ and $\hat{T}_{s,\mathrm{coll}}/\hat{T}_{s',\mathrm{coll}}$ are computed at the radial point $\hat{r}_p$ which usually corresponds to the middle of the simulation radial domain. The normalized mean velocity $\hat{V}_{\|s,\mathrm{coll}}$ and the normalized mean temperature $\hat{T}_{s,\mathrm{coll}}$ can be calculated as follows:

$$\hat{V}_{\|s,\mathrm{coll}} = \hat{P}^{-1}\left(\hat{I}_4 \times \hat{I}_1 - \hat{I}_2 \times \hat{I}_3\right) \quad ; \quad \hat{T}_{s,\mathrm{coll}} = \hat{P}^{-1}\left(\hat{I}_0 \times \hat{I}_2 - \hat{I}_1^2\right) \tag{K.32}$$

with the normalized mean pressure defined by

$$\hat{P} = \hat{I}_0 \times \hat{I}_4 - \hat{I}_1 \times \hat{I}_3 \tag{K.33}$$

and where the five integrals $\hat{I}_0$, $\hat{I}_1$, $\hat{I}_2$, $\hat{I}_3$ and $\hat{I}_4$ are

$$\hat{I}_0 = \langle\hat{\mathcal{D}}_{\|s}\rangle \quad ; \quad \hat{I}_1 = \langle\hat{\mathcal{D}}_{\|s}\hat{v}_{G\|s}\rangle \quad ; \quad \hat{I}_2 = \langle\hat{\mathcal{D}}_{\|s}\hat{v}_{G\|s}^2\rangle \tag{K.34}$$

$$\hat{I}_3 = \langle\frac{1}{\hat{B}_{\|s}^*}\partial_{\hat{v}_{G\|s}}\left(\hat{B}_{\|s}^*\hat{\mathcal{D}}_{\|s}\right)\rangle \quad \text{and} \quad \hat{I}_4 = \langle\frac{1}{\hat{B}_{\|s}^*}\partial_{\hat{v}_{G\|s}}\left(\hat{B}_{\|s}^*\,\hat{v}_{G\|s}\,\hat{\mathcal{D}}_{\|s}\right)\rangle \tag{K.35}$$

with the brackets $\langle\cdot\rangle = \int \cdot\, \hat{\bar{F}}_s\hat{\mathcal{J}}_{\mathrm{v}}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s$.

K.1.5 Normalized collisional transfers between two species

Normalized collisional energy transfer Energy exchange between two species can be expressed by the following reduced collision operator:

$$\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} = -\frac{\hat{\eta}_{ss'}^{\mathcal{E}}}{\left(2\pi\hat{T}_{\mathrm{mean}}\right)^{3/2}}\frac{\Delta\hat{T}_{ss'}}{\hat{T}_{\mathrm{mean}}}\left(\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}} - \frac{3}{2}\right)\exp\left(-\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}}\right) \tag{K.36}$$

$$\frac{\mathrm{d}\hat{\bar{F}}_{s'}}{\mathrm{d}t} = -\frac{\hat{\eta}_{ss'}^{\mathcal{E}}}{\left(2\pi\hat{T}_{\mathrm{mean}}\right)^{3/2}}\frac{\Delta\hat{T}_{s's}}{\hat{T}_{\mathrm{mean}}}\left(\frac{\hat{\mathcal{E}}_{s's}}{\hat{T}_{\mathrm{mean}}} - \frac{3}{2}\right)\exp\left(-\frac{\hat{\mathcal{E}}_{s's}}{\hat{T}_{\mathrm{mean}}}\right) \tag{K.37}$$

with

$$\hat{T}_{\text{mean}} = \frac{\hat{T}_s + \hat{T}_{s'}}{2} \quad ; \quad \Delta\hat{T}_{ss'} = \hat{T}_s - \hat{T}_{s'} = -\Delta\hat{T}_{s's} \quad ; \quad \hat{V}_{\text{mean}} = \frac{\hat{V}_{\|s} + \hat{V}_{\|s'}}{2}$$

$$\hat{\mathcal{E}}_{ss'} = \frac{\left(\hat{v}_{G\|s} - \hat{V}_{\text{mean}}\right)^2}{2} + \hat{\mu}_s\hat{B} \quad \text{and} \quad \hat{\mathcal{E}}_{s's} = \frac{\left(\hat{v}_{G\|s'} - \hat{V}_{\text{mean}}\right)^2}{2} + \hat{\mu}_s\hat{B}$$

and

$$\hat{\eta}^{\mathcal{E}}_{ss'}(\hat{r}) = \frac{8\,\hat{\varepsilon}^{3/2}}{3\sqrt{\pi}}\,\frac{\sqrt{\hat{A}_s}}{\hat{A}_{s'}}\left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2 \frac{\hat{n}_{s'}\,\sqrt{\langle\,\hat{T}_{s,\text{coll}}\,\rangle_{\text{FS}}}}{q\hat{R}_0}\left(1 + \frac{\hat{A}_s}{\hat{A}_{s'}}\frac{\hat{T}^2_{s'}}{\hat{T}^2_s}\right)^{-3/2}\hat{\nu}_{*s} \tag{K.38}$$

Let us notice that in practice, the flux surface average of $\hat{T}_{s,\text{coll}}$ is used (compare to the general expression given by (2.44)), to consider only the radial dependency of $\hat{\eta}^{\mathcal{E}}_{ss'}$. The normalized fluid moments $\hat{V}_{\|s}$ and $\hat{T}_s$ are computed as $3\hat{T}_s(r)/2 = \langle \int \hat{\mathcal{E}}_s\,\hat{\bar{F}}_s\,\mathrm{d}^3\hat{\mathbf{v}}\,\rangle_{\text{FS}}/\hat{N}_s(r)$ and $\hat{V}_{\|s}(r) = \langle \int \hat{v}_{G\|s}\,\hat{\bar{F}}_s\,\mathrm{d}^3\hat{\mathbf{v}}\,\rangle_{\text{FS}}/\hat{N}_s(r)$ where $\hat{\mathcal{E}}_s$ is defined as $\hat{\mathcal{E}}_s = \frac{1}{2}\left(\hat{v}_{G\|s} - \hat{V}_{\|s}\right)^2 + \hat{\mu}_s\hat{B}$ while $\hat{N}_s(r) = \langle \int \hat{\bar{F}}_s\,\mathrm{d}^3\hat{\mathbf{v}}\,\rangle_{\text{FS}}$.

Normalized collisional momentum transfer Momentum exchange between two species can be expressed by the following reduced collision operator

$$\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} = -\hat{\eta}^{v_{G\|}}\hat{A}_s^{-1/2}\Delta\hat{V}_{ss'}\hat{v}_{G\|s}\exp\left(-\frac{\hat{E}_s}{\hat{T}_{\text{mean}}}\right) \tag{K.39}$$

$$\frac{\mathrm{d}\hat{\bar{F}}_{s'}}{\mathrm{d}\hat{t}} = -\hat{\eta}^{v_{G\|}}\hat{A}_{s'}^{-1/2}\Delta\hat{V}_{s's}\hat{v}_{G\|s'}\exp\left(-\frac{\hat{E}_{s'}}{\hat{T}_{\text{mean}}}\right) \tag{K.40}$$

with

$$\Delta\hat{V}_{ss'} = \hat{V}_{\|s}\hat{A}_s^{-1/2} - \hat{V}_{\|s'}\hat{A}_{s'}^{-1/2} \tag{K.41}$$

The averages $\hat{T}_{\text{mean}}$ and $\hat{V}_{\text{mean}}$ have already been introduced. The energy $\hat{E}_s$ is equal to $\frac{1}{2}\hat{v}^2_{G\|s} + \hat{\mu}_s\hat{B}$. The radial profile $\hat{\eta}^{v_{G\|}}$ reads:

$$\hat{\eta}^{v_{G\|}}(r) = \left(\frac{\varepsilon}{2\pi}\right)^{3/2}\frac{1}{q\hat{R}_0}\frac{\hat{n}_{s'}}{\hat{v}_{T_>}}\frac{\sqrt{\hat{T}_s}\sqrt{\langle\,\hat{T}_{s,\text{coll}}\,\rangle_{\text{FS}}}}{\hat{T}^{5/2}_{\text{mean}}}\left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2\hat{\nu}_{*s} \tag{K.42}$$

where the velocity $\hat{v}_{T_>}$ is calculated as $\hat{v}_{T_>} = \max(\hat{A}_s^{-1/2}\hat{v}_{T_s}, \hat{A}_{s'}^{-1/2}\hat{v}_{T_{s'}})$ where $\hat{v}_{T_s} = \sqrt{\hat{T}_s}$ and $\hat{v}_{T_{s'}} = \sqrt{\hat{T}_{s'}}$ and the collision frequencies $\hat{\nu}_{*s}$, $\hat{\nu}_{*s'}$ are linked by the relation (K.31).

K.1.6 Normalized source terms

Focusing on the source term, the gyrokinetic equation is:

$$\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} = \hat{S}_{\text{heat}}(\hat{r},\theta,\hat{v}_{G\|s},\hat{\mu}_s) + \hat{S}_{\text{moment}}(\hat{r},\theta,\hat{v}_{G\|s},\hat{\mu}_s) + \hat{S}_{\text{vorticity}}(\hat{r},\theta,\hat{v}_{G\|s},\hat{\mu}_s) \tag{K.43}$$

where the heat source can be defined as

$$\hat{S}_{\text{heat}} = \frac{\hat{S}^{\mathcal{E}}_0}{3\sqrt{2}\left(\pi\hat{T}_{s,\text{srce}}\right)^{3/2}\hat{T}_{s,\text{srce}}}\left(\frac{\hat{E}_s}{\hat{T}_{s,\text{srce}}} - \frac{3}{2}\right)\exp\left(-\frac{\hat{E}_s}{\hat{T}_{s,\text{srce}}}\right)\hat{S}^{\mathcal{E}}_r \tag{K.44}$$

with $\hat{E}_s = \frac{1}{2}\hat{v}_{G\|s} + \hat{\mu}_s\hat{B}$ or as a pure source of heating

$$\hat{S}_{\text{heat}} = \left\{\left[\hat{\bar{v}}^2_{G_\|s} - \frac{1}{2} - \frac{\hat{J}_{\|B}}{2 - \hat{J}^2_{\|B}}\,(2 - \hat{\bar{\mu}}_s)\left(2\hat{\bar{v}}_{G_\|s} - \hat{J}_{\|B}\right)\right]\frac{2\hat{S}^{\mathcal{E}}_0}{\left(2\pi\hat{T}_{s,\text{srce}}\right)^{3/2}\hat{T}_{s,\text{srce}}}\,\hat{S}^{\mathcal{E}}_r\right\}\,\mathrm{e}^{-\hat{\bar{v}}^2_{G_\|s} - \hat{\bar{\mu}}_s} \tag{K.45}$$

with $\hat{\bar{\mu}}_s = \frac{\hat{\mu}_s \hat{B}}{\hat{T}_{s,\mathrm{srce}}}$, $\hat{\bar{v}}_{G\|s} = \frac{\hat{v}_{G\|s}}{\sqrt{2\hat{T}_{s,\mathrm{srce}}}}$ and $\hat{J}_{\|B} \equiv \frac{\sqrt{\hat{A}_s}}{\hat{Z}_s}\frac{\sqrt{2\hat{T}_{s,\mathrm{srce}}}}{\hat{B}^2}\mu_0 \hat{J}_{\|}$. The source $\hat{S}_{\mathrm{moment}}$ is a pure source of momentum expressed as

$$\hat{S}_{\mathrm{moment}} = \left\{ \left[2\hat{\bar{v}}_{G\|s}(2-\hat{\bar{\mu}}_s) - \hat{J}_{\|B}\left(1 + 2\hat{\bar{v}}_{G\|s} - \hat{\bar{\mu}}_s\right)\right] \frac{\hat{S}_0^{v_{G\|}}}{4\pi^{3/2}\hat{T}_{s,\mathrm{srce}}^2}\, \hat{S}_r^{v_{G\|}} \right\} \mathrm{e}^{-\hat{\bar{v}}_{G\|s}^2 - \hat{\bar{\mu}}_s} \tag{K.46}$$

and the $\hat{S}_{\mathrm{vorticity}}$ is a pure source of vorticity is defined as

$$\hat{S}_{\mathrm{vorticity}} = -\left\{ \left[2\hat{\bar{v}}_{G\|s}^2 - \hat{\bar{\mu}}_s\right] \frac{\hat{S}_0^{\Omega}}{\alpha\left(2\pi\hat{T}_{s,\mathrm{srce}}\right)^{3/2}}\, \hat{S}_r^{\Omega} \right\} \mathrm{e}^{-\hat{\bar{v}}_{G\|s}^2 - \hat{\bar{\mu}}_s} \quad \text{with } \hat{\alpha} = \frac{\hat{A}_s}{\hat{Z}_s^2}\frac{\hat{T}_{s,\mathrm{srce}}}{2\hat{B}^2} \tag{K.47}$$

The radial components of the sources (i.e $\hat{S}_r^{\mathcal{E}}$, $\hat{S}_r^{v_{G\|}}$ and $\hat{S}_r^{\Omega}$) are defined as:

$$\hat{S}_r^x(\hat{r}) = -\frac{1}{2}\left[\tanh\left(\frac{\hat{\rho} - (\hat{\rho}_S^x + 3\hat{L}_S^x)}{\hat{L}_S^x}\right) + \tanh\left(-\frac{\hat{\rho} - (\hat{\rho}_S^x - 3\hat{L}_S^x)}{\hat{L}_S^x}\right)\right] \tag{K.48}$$

where $\hat{\rho} = (\hat{r} - \hat{r}_{\mathrm{min}})/\hat{L}_r$. $\hat{\rho}_S^x$ and $\hat{L}_S^x$ are input data corresponding to the radial position and the normalized width of the different sources. These radial parts are normalized such that

$$\int_{\hat{r}_{\mathrm{min}}}^{\hat{r}_{\mathrm{max}}} \hat{r}\, \mathrm{d}\hat{r}\, \hat{S}_r^x(\hat{r}) = 1$$

K.1.7 Normalized source of impurity

Let us consider the species s as the major species and the species s' as an impurity species. Then, it is possible to add a source of impurities, by adding to the right hand side of equation (K.43) a source of matter. The source $\hat{S}_{n_{s'}}(\hat{r}, \theta, \hat{v}_{G\|s}, \hat{\mu}_s)$ of impurity s' is defined by

$$\hat{S}_{n_{s'}} = \frac{\hat{S}_0^{n'_s}\hat{S}_r}{\left(2\pi\hat{T}_{s',\mathrm{srce}}\right)^{3/2}}\left(\frac{5}{2} - \hat{\bar{\mu}}_{s'} - \hat{\bar{v}}_{G\|s'}^2\right)\exp\left(-\hat{\bar{v}}_{G\|s'}^2 - \hat{\bar{\mu}}_{s'}\right) \tag{K.49}$$

where $\hat{\bar{v}}_{G\|s'} = \hat{v}_{G\|s'}/\sqrt{2\hat{T}_{s',\mathrm{srce}}}$ and $\hat{\bar{\mu}}_{s'} = \hat{\mu}_{s'}\hat{B}/\hat{T}_{s',\mathrm{srce}}$. $\hat{S}_0^{n'_s}$ corresponds to the normalized intensity of the source. To avoid any injection of charges, the injection of impurities must be compensated by the injection of source of matter for the majority species such that

$$Z_s \int S_{n_s}\mathcal{J}_{\mathrm{v}_s}\, \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s + Z_{s'}\int S_{n_{s'}}\mathcal{J}_{\mathrm{v}_{s'}}\, \mathrm{d}v_{G\|}\, \mathrm{d}\mu_{s'} = 0$$

with $\mathcal{J}_{\mathrm{v}_s}$ (resp. $\mathcal{J}_{\mathrm{v}_{s'}}$) the jacobian in velocity space for species s (resp. species s'). The normalized radial profiles of the sources are assumed equal to $\hat{S}_r$. Therefore,

$$\hat{S}_{n_s} = \frac{\hat{S}_0^{n_s}\hat{S}_r}{\left(2\pi\hat{T}_{s,\mathrm{srce}}\right)^{3/2}}\left(\frac{5}{2} - \hat{\bar{\mu}}_s - \hat{\bar{v}}_{G\|s}^2\right)\mathrm{e}^{-\hat{\bar{v}}_{G\|s}^2 - \hat{\bar{\mu}}_s} \quad \text{with } \hat{S}_0^{n_s} = -\frac{\hat{Z}_{s'}}{\hat{Z}_s}\left(\frac{\hat{T}_{s',\mathrm{srce}}}{\hat{T}_{s,\mathrm{srce}}}\right)^{3/2}\hat{S}_0^{n_{s'}} \tag{K.50}$$

In the code, the temperature profile of the density sources are taken equal, i.e $\hat{T}_{s,\mathrm{srce}} = \hat{T}_{s',\mathrm{srce}}$.

K.2 Proofs for Gysela normalized equations

In this section, the derivation from the general equations to the normalized equations used in the Gysela code is detailed. This derivation is performed by using the normalization defined in table K.1. The following notes play the role of archives.

K.2.1 Proof for gyro-average operator normalization (eq.(K.21))

$$\begin{aligned} J_{0_s} &\approx 1+\frac{1}{4}\rho_s^2\nabla_\perp^2 \quad \text{with} \quad \rho_s^2=\left(\frac{m_s v_\perp}{Z_s\, e\, B}\right)^2=\frac{m_s}{(Z_s\, e)^2}\frac{2\mu_s}{B} \\ &\approx 1+\frac{m_s}{(Z_s\, e)^2}\frac{\mu_s}{2B}\nabla_\perp^2 = 1+\frac{m_0\hat{A}_s}{\left(Z_0\hat{Z}_s e\right)^2}\frac{T_0}{B_0}\hat{\mu}_s\frac{1}{2B_0\hat{B}}\frac{1}{\rho_0^2}\hat{\nabla}_\perp^2 = 1+\frac{\hat{A}_s}{\hat{Z}_s^2}\frac{\hat{\mu}_s}{2\hat{B}}\hat{\nabla}_\perp^2 \quad \blacksquare \end{aligned}$$

K.2.2 Proof for velocity space integral normalization

The velocity space integral is defined as $\int \mathrm{d}^3v = \int \mathcal{J}_\mathrm{v}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_s$ where $\mathcal{J}_\mathrm{v}$ the jacobian in velocity space is equal to $\mathcal{J}_\mathrm{v} = 2\pi B^*_{\|s}/m_s$. Therefore,

$$\int \mathcal{J}_\mathrm{v}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_s = \int \frac{2\pi B_0\hat{B}^*_{\|s}}{m_0\hat{A}_s}\times v_{T_{s0}}\,\mathrm{d}\hat{v}_{G\|s}\times\frac{T_0}{B_0}\,\mathrm{d}\hat{\mu}_s = \frac{B_0}{m_0}\times\frac{T_0}{B_0}\times\frac{v_{T_{s0}}}{\hat{A}_s}\int 2\pi\hat{B}^*_{\|s}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s$$

then

$$\int \mathcal{J}_\mathrm{v}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_s = v_{T_{s0}}^3\int \hat{\mathcal{J}}_\mathrm{v}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s \qquad \text{with} \qquad \hat{\mathcal{J}}_\mathrm{v} = 2\pi\hat{B}^*_{\|s} \quad \blacksquare \tag{K.51}$$

K.2.3 Proof for normalized equilibrium function (eq.(K.20))

$$\begin{aligned} F_{s,\mathrm{eq}} &= \frac{n_0}{(2\pi T_s/m_s)^{3/2}}\exp\left(-\left[\frac{m_s v_{G\|}^2}{2T_s}+\frac{\mu_s B}{T_s}\right]\right) \\ &= \frac{n_0}{(2\pi\hat{T}_s)^{3/2}}\left(\frac{\hat{A}_s m_0}{T_0}\right)^{3/2}\exp\left(-\frac{1}{T_0\hat{T}_s}\left[\frac{\hat{v}_{G\|s}^2 v_{T_{s0}}^2\hat{A}_s m_0}{2}+\hat{\mu}_s\hat{B}\frac{T_0}{B_0}B_0\right]\right) \\ &= \frac{n_0}{(2\pi\hat{T}_s)^{3/2}}\frac{1}{v_{T_{s0}}^3}\exp\left(-\frac{1}{\hat{T}_s}\left[\frac{\hat{v}_{G\|s}^2}{2}+\hat{\mu}_s\hat{B}\right]\right) \end{aligned}$$

then

$$F_{s,\mathrm{eq}} = \frac{n_0}{v_{T_{s0}}^3}\hat{F}_{s,\mathrm{eq}} \quad \text{with} \quad \hat{F}_{s,\mathrm{eq}} = \frac{1}{(2\pi\hat{T}_s)^{3/2}}\exp\left(-\frac{1}{\hat{T}_s}\left[\frac{\hat{v}_{G\|s}^2}{2}+\hat{\mu}_s\hat{B}\right]\right) \tag{K.52}$$

K.2.4 Proof for normalization of motion equations (eq.(K.4)-(K.10))

Let remind that the motion equations for the guiding-centers in the 3D real space read (cf eq. (1.48)),

$$\frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t} = v_{G\|}\mathbf{b}_s^* + \mathbf{v}_{E\times B_s} + \mathbf{v}_{D_s} \tag{K.53}$$

with

$$\mathbf{b}_s^* \equiv \frac{\mathbf{B}}{B^*_{\|s}} + \frac{m_s v_{G\|}}{q_s B^*_{\|s} B}\boldsymbol{\nabla}\times\mathbf{B} \quad \text{where} \quad B^*_{\|s} \equiv B + \frac{m_s v_{G\|}}{q_s B}\mathbf{b}\cdot(\boldsymbol{\nabla}\times\mathbf{B})$$

The drift-velocities are defined as:

$$\mathbf{v}_{E\times B_s} = \frac{1}{B^*_{\|s}}\mathbf{b}\times\boldsymbol{\nabla}\bar{U} \quad ; \quad \mathbf{v}_{D_s} = \left(\frac{m_s v_{G\|}^2+\mu_s B}{q_s B^*_{\|s}}\right)\mathbf{b}\times\frac{\boldsymbol{\nabla}B}{B}$$

First of all, let express $B^*_{\|s}$ in function of the normalized quantities,

$$\begin{aligned} B^*_{\|s} &= B + m_s\frac{v_{G\|}}{q_s B}\mu_0\mathbf{b}\cdot\mathbf{J} \\ &= B_0\hat{B} + \frac{m_0\hat{A}_s v_{T_{s0}}\hat{v}_{G\|s}}{eZ_0\hat{Z}_s B_0\hat{B}}\mathbf{b}\cdot\hat{\mathbf{J}}\frac{B_0}{\rho_0} = B_0\left[\hat{B}+\sqrt{\hat{A}_s}\frac{\hat{v}_{G\|s}}{\hat{Z}_s\hat{B}}\mathbf{b}\cdot\hat{\mathbf{J}}\right] \end{aligned} \tag{K.54}$$

then

$$B^*_{\|s} = B_0 \hat{B}^*_{\|s} \quad \text{with} \quad \hat{B}^*_{\|s} = \hat{B} + \frac{\sqrt{\hat{A}_s}}{\hat{Z}_s} \frac{\hat{v}_{G\|s}}{\hat{B}} \mathbf{b} \cdot \hat{\mathbf{J}} \tag{K.55}$$

and

$$\begin{aligned} \mathbf{b}^*_s &\equiv \frac{1}{B^*_{\|s}} \left(\mathbf{B} + \frac{m_s v_{G\|}}{q_s B} \mu_0 \mathbf{J} \right) = \frac{1}{B_0 \hat{B}^*_{\|s}} \left(B_0 \hat{\mathbf{B}} + \frac{\hat{A}_s m_0 \, v_{T_{s0}} \hat{v}_{G\|s}}{Z_0 \, e \, \hat{Z}_s \, B_0 \hat{B}} \hat{\mathbf{J}} \frac{B_0}{\rho_0} \right) \\ &= \frac{1}{\hat{B}^*_{\|s}} \left(\hat{\mathbf{B}} + K_s \frac{\hat{v}_{G\|s}}{\hat{Z}_s \hat{B}} \hat{\mathbf{J}} \right) \end{aligned}$$

with the constant K_s defined as

$$K_s = \frac{\hat{A}_s m_0 v_{T_{s0}}}{e Z_0 B_0 \rho_0} = \hat{A}_s \frac{v_{T_{s0}}}{\rho_0} \frac{m_0}{e B_0 Z_0} = \hat{A}_s \frac{v_{T_{s0}}}{v_{T_0}} = \sqrt{\hat{A}_s}$$

then the expression of $\mathbf{b}^*_s$ is equivalent to equation (K.8). The first term of the right hand side of equation (K.53) expressed with the normalized quantities gives :

$$v_{G\|} \mathbf{b}^*_s = v_{T_{s0}} \hat{v}_{G\|s} \hat{\mathbf{b}}^*_s$$

The second term of the right hand side of equation (K.53) reads

$$\mathbf{v}_{E\times B_s} = \frac{1}{B_0 \hat{B}^*_{\|s}} \left(\hat{\mathbf{b}} \times \hat{\boldsymbol{\nabla}} \left(\frac{T_0}{Z_0 \, e} \hat{\bar{\phi}} \right) \right) \frac{1}{\rho_0} = v_{T_0} \frac{1}{\hat{B}^*_{\|s}} \left(\hat{\mathbf{b}} \times \hat{\boldsymbol{\nabla}} \hat{\bar{\phi}} \right)$$

The third term of the right hand side of equation (K.53) gives

$$\mathbf{v}_{D_s} = \left(\frac{\hat{A}_s m_0 v_{T_0}^2 \hat{v}_{G\|s}^2 + \hat{\mu}_s T_0 \hat{B}}{Z_0 \hat{Z}_s e B_0^2 \hat{B}^*_{\|s} \hat{B}} \right) \left(\hat{\mathbf{b}} \times \hat{\boldsymbol{\nabla}} \hat{B} \right) \frac{B_0}{\rho_0}$$

and

$$\frac{\hat{A}_s m_0 v_{T_{s0}}^2}{Z_0 e B_0 \rho_0} = \frac{\hat{A}_s v_{T_{s0}}^2}{\Omega_{c_0} \rho_0} = \frac{\hat{A}_s v_{T_{s0}}^2}{v_{T_0}} = v_{T_0} \quad ; \quad \frac{T_0}{Z_0 e B_0 \rho_0} = \frac{T_0}{m_0 \Omega_{c_0} \rho_0} = v_{T_0}$$

so

$$\mathbf{v}_{D_s} = v_{T_0} \left(\frac{\hat{v}_{G\|s}^2 + \hat{\mu}_s \hat{B}}{\hat{Z}_s \hat{B}^*_{\|s} \hat{B}} \right) \left(\hat{\mathbf{b}} \times \hat{\boldsymbol{\nabla}} \hat{\mathbf{B}} \right)$$

Therefore, using the fact that the parallel velocity is normalized to $v_{T_{s0}}$ while the drift velocities are normalized to v_{T_0},

$$\frac{\mathrm{d}\hat{\mathbf{x}}_G}{\mathrm{d}t} = \frac{1}{\sqrt{\hat{A}_s}} \hat{v}_{G\|s} \hat{\mathbf{b}}^*_s + \hat{\mathbf{v}}_{E\times B_s} + \hat{\mathbf{v}}_{D_s}$$

with

$$\hat{\mathbf{v}}_{E\times B_s} = \frac{1}{\hat{B}^*_{\|s}} \left(\hat{\mathbf{b}} \times \hat{\boldsymbol{\nabla}} \hat{\phi} \right) \quad \text{and} \quad \hat{\mathbf{v}}_{D_s} = \left(\frac{\hat{v}_{G\|s}^2 + \hat{\mu}_s \hat{B}}{\hat{Z}_s \hat{B}^*_{\|s} \hat{B}} \right) \left(\hat{\mathbf{b}} \times \hat{\boldsymbol{\nabla}} \hat{\mathbf{B}} \right)$$

which is equivalent to equation (K.4). ∎

According to equation (1.49), the time derivative of $v_{G\|}$ reads:

$$m_s \frac{\mathrm{d}v_{G\|}}{\mathrm{d}t} = -\mu_s \mathbf{b}^*_s \cdot \boldsymbol{\nabla} B - q_s \mathbf{b}^*_s \cdot \boldsymbol{\nabla} \bar{\phi} + m_s v_{G\|} \mathbf{v}_{E\times Bs} \cdot \frac{\boldsymbol{\nabla} B}{B}$$

then

$$\frac{\mathrm{d}\hat{v}_{G\|s}}{\mathrm{d}t} = \frac{1}{K} \left(-\mu_s \mathbf{b}^*_s \cdot \boldsymbol{\nabla} B - q_s \mathbf{b}^*_s \cdot \boldsymbol{\nabla} \bar{\phi} + m_s v_{G\|} \mathbf{v}_{E\times Bs} \cdot \frac{\boldsymbol{\nabla} B}{B} \right)$$

with $K = \hat{A}_s m_0 v_{T_{s0}} \Omega_0$. Therefore, the first term of the right hand side of previous equation reads

$$\frac{1}{K}\mu_s \mathbf{b}_s^* \cdot \boldsymbol{\nabla} B = \frac{1}{\hat{A}_s m_0 v_{T_{s0}} \Omega_0} \frac{T_0}{B_0} \hat{\mu}_s \mathbf{b}_s^* \cdot \hat{\boldsymbol{\nabla}} \hat{B} \frac{B_0}{\rho_0} = \frac{1}{\hat{A}_s} \frac{v_{T_0}^2}{v_{T_{s0}}} \frac{1}{v_{T_0}} \hat{\mu}_s \mathbf{b}_s^* \cdot \hat{\boldsymbol{\nabla}} \hat{B} = \frac{1}{\sqrt{\hat{A}_s}} \hat{\mu}_s \mathbf{b}_s^* \cdot \hat{\boldsymbol{\nabla}} \hat{B}$$

The second term reads

$$\frac{1}{K}\left(q_s \mathbf{b}_s^* \cdot \boldsymbol{\nabla} \bar{\phi}\right) = Z_0 \hat{Z}_s e \mathbf{b}_s^* \frac{1}{\rho_0} \frac{T_0}{Z_0\, e} \hat{\boldsymbol{\nabla}} \hat{\bar{\phi}} \frac{1}{\hat{A}_s m_0 v_{T_{s0}} \Omega_0} = \frac{T_0}{\hat{A}_s m_0 v_{T_{s0}} v_{T_0}} \hat{Z}_s \mathbf{b}_s^* \cdot \hat{\boldsymbol{\nabla}} \hat{\bar{\phi}} = \frac{1}{\sqrt{\hat{A}_s}} \hat{Z}_s \mathbf{b}_s^* \cdot \hat{\boldsymbol{\nabla}} \hat{\bar{\phi}}$$

and finally the last term of the right hand side reads:

$$\frac{1}{K}\left(m_s v_{G\|} \mathbf{v}_{E\times Bs} \cdot \frac{\boldsymbol{\nabla} B}{B}\right) = \frac{1}{\hat{A}_s m_0 v_{T_{s0}} \Omega_0} m_0 \hat{A}_s v_{T_{s0}} \hat{v}_{G\|s} (v_{T_0} \hat{\mathbf{v}}_{E\times Bs}) \cdot \frac{\hat{\boldsymbol{\nabla}} \ln \hat{B}}{\rho_0} = \hat{v}_{G\|s} \hat{\mathbf{v}}_{E\times Bs} \cdot \hat{\boldsymbol{\nabla}} \ln \hat{B}$$

Therefore, the time derivative of the normalized velocity $\hat{v}_{G\|s}$ is given by:

$$\frac{\mathrm{d}\hat{v}_{G\|s}}{\mathrm{d}t} = -\frac{1}{\sqrt{\hat{A}_s}} \hat{\mu}_s \mathbf{b}_s^* \cdot \hat{\boldsymbol{\nabla}} \hat{B} - \frac{\hat{Z}_s}{\sqrt{\hat{A}_s}} \mathbf{b}_s^* \cdot \hat{\boldsymbol{\nabla}} \hat{\bar{\phi}} + \hat{v}_{G\|s} \hat{\mathbf{v}}_{E\times Bs} \cdot \hat{\boldsymbol{\nabla}} \ln \hat{B}$$

which is equivalent to the expression (K.5). ∎

K.2.5 Proof for quasi-neutrality equation normalization (eq.(K.17))

As seen in section 1.4.4, the Poisson equation reads:

$$-\epsilon_0 \nabla^2 U - \sum_s Z_s e \nabla_\perp \cdot \left(\frac{n_{s,\mathrm{eq}}}{B\Omega_s} \nabla_\perp U(\mathbf{x}, t)\right) + e^2 n_{e_0} \left[\frac{U - \langle U \rangle_{\mathrm{FS}}}{T_e}\right] = \sum_s q_s n_{G_s} - e n_{e_0}$$

Due to the expression of cyclotron frequency $\Omega_s = q_s B_0 / m_s$, the previous equation is equivalent to:

$$-\epsilon_0 \nabla^2 U - \sum_s m_s \nabla_\perp \cdot \left(\frac{n_{s,\mathrm{eq}}}{B B_0} \nabla_\perp U(\mathbf{x}, t)\right) + e^2 n_{e_0} \left[\frac{U - \langle U \rangle_{\mathrm{FS}}}{T_e}\right] = \sum_s q_s n_{G_s} - e n_{e_0}$$

Then using the fact that $\lambda_0 = [\epsilon_0 T_0 / (Z_0^2 e^2 n_0)]^{1/2}$ and the fact that $n_{G_s} = n_0 \hat{n}_{G_s}$ [1] with $\hat{n}_{G_s}$ defined by (K.18),

$$\begin{aligned} - \left(\frac{\lambda_0}{\rho_0}\right)^2 \frac{Z_0^2 e^2 n_0}{T_0} \hat{\nabla}^2 \left(\frac{T_0}{Z_0\, e} \hat{\phi}\right) - \sum_s \hat{\nabla}_\perp \left(\frac{m_0 \hat{A}_s n_0 \hat{n}_{s,\mathrm{eq}}}{\rho_0^2 B_0^2 \hat{B}}\right) \hat{\nabla}_\perp \left(\frac{T_0}{Z_0\, e} \hat{\phi}\right) \\ + \frac{e^2 n_0 \hat{n}_{e_0} T_0}{Z_0\, e\, T_0} \left[\frac{\hat{\phi} - \langle \hat{\phi} \rangle_{\mathrm{FS}}}{\hat{T}_e}\right] = \sum_s Z_0 \hat{Z}_s e n_0 \hat{n}_{G_s} - e n_0 \hat{n}_{e_0} \end{aligned}$$

Therefore, assuming that $\hat{n}_{s,\mathrm{eq}} = \hat{n}_{G_s,\mathrm{eq}}$ (with $\hat{n}_{G_s,\mathrm{eq}}$ defined by equation (K.19)) and using the fact that $\hat{n}_{e_0} = \sum_s Z_0 \hat{Z}_s \hat{n}_{s,\mathrm{eq}}$, the RHS of the previous equation is equivalent to $\sum_s \hat{Z}_s \left(\hat{n}_{G_s} - \hat{n}_{G_s,\mathrm{eq}}\right)$ and the previous equation reads

$$\left(\frac{\lambda_0}{\rho_0}\right)^2 \hat{\nabla}^2 \hat{\phi} - \sum_s \hat{\nabla}_\perp \left(\frac{\hat{A}_s \hat{n}_{s,\mathrm{eq}}}{\hat{B}} \hat{\nabla}_\perp \hat{\phi}\right) + \frac{\hat{n}_{e_0}}{Z_0^2} \left[\frac{\hat{\phi} - \langle \hat{\phi} \rangle_{\mathrm{FS}}}{\hat{T}_e}\right] = \sum_s \hat{Z}_s \left(\hat{n}_{G_s} - \hat{n}_{G_s,\mathrm{eq}}\right)$$

Besides, in the quasi-neutral limit, i.e $\lambda_0 \ll \rho_0$, the first term $\frac{1}{\hat{n}_{e_0}} \left(\frac{\lambda_0}{\rho_0}\right)^2 \hat{\nabla}^2 \hat{\phi}$ can be dropped out and the previous equation is then equivalent to the normalized quasi-neutrality equation (K.17). ∎

[1] Due to the normalization of the velocity space integral given by (K.51),

$$n_{G_s} = \int \mathcal{J}_{\mathrm{v}}\, \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s\, J_{0_s} \cdot \bar{F}_s = v_{T_{s0}}^3 \int \hat{\mathcal{J}}_{\mathrm{v}}\, \mathrm{d}\hat{v}_{G\|s}\, \mathrm{d}\hat{\mu}_s\, \hat{J}_{0_s} \cdot \left(\hat{\bar{F}}_s \frac{n_0}{v_{T_{s0}}^3}\right) = n_0 \int \hat{\mathcal{J}}_{\mathrm{v}}\, \mathrm{d}\hat{v}_{G\|s}\, \mathrm{d}\hat{\mu}_s\, \hat{J}_{0_s} \cdot \hat{\bar{F}}_s$$

K.2.6 Proof for buffer region normalization (eq.(K.23)-(K.24))

As seen in section 2.8.4, a radial diffusion and a Krook operator are applied on a buffer region by solving an equation equivalent to

$$\frac{\partial \bar{F}_s}{\partial t} = \frac{1}{B^*_{\|s}}\left[\frac{1}{r}\frac{\partial}{\partial r}\left(\mathcal{D}_r B^*_{\|s}\frac{\partial}{\partial r}\bar{F}_s\right)\right] - \nu(r)(\bar{F}_s - \bar{F}_{s,\mathrm{eq}})$$

with $\mathcal{D}_r = r\chi_0 H_{\mathrm{buff}}(r)$ and $\nu(r) = \nu_0 H_{\mathrm{buff}}(r)$. Then, using GYSELA normalization, the previous equation becomes

$$\Omega_{c_0}\frac{\partial(n_0/v^3_{T_{s0}})\hat{\bar{F}}_s}{\partial \hat{t}} = \left(\frac{1}{B_0\hat{B}^*_{\|s}}\right)\left(\frac{1}{\rho_0}\frac{1}{\hat{r}}\right)\frac{1}{\rho_0}\frac{\partial}{\partial \hat{r}}\left(\rho_0\hat{r}\chi(r)\left(B_0\hat{B}^*_{\|s}\right)\frac{1}{\rho_0}\frac{\partial(n_0/v^3_{T_{s0}})\hat{\bar{F}}_s}{\partial \hat{r}}\right) - \nu(r)\frac{n_0}{v^3_{T_{s0}}}(\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\mathrm{eq}})$$

which is equivalent to

$$\frac{\partial \hat{\bar{F}}_s}{\partial \hat{t}} = \frac{1}{\hat{B}^*_{\|s}}\left[\frac{1}{\hat{r}}\frac{\partial}{\partial \hat{r}}\left(\hat{r}\hat{B}^*_{\|s}\hat{\chi}\frac{\partial \hat{\bar{F}}_s}{\partial \hat{r}}\right)\right] - \hat{\nu}(\hat{r})(\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\mathrm{eq}})$$

with $\chi(r) = \rho_0^2\Omega_0\hat{\chi}(\hat{r})$ and $\nu(r) = \Omega_{c_0}\hat{\nu}(\hat{r})$. ∎

K.2.7 Proof for collision operator normalization (eqs.(K.25)-(K.35))

Let $v(r, v_{G\|})$ be defined as $v(r, v_{G\|}) = \sqrt{\frac{1}{T_{s,\mathrm{coll}}}\left(\frac{1}{2}m_s v^2_{G\|} + \mu_s B_{\mathrm{axis}}(r)\right)}$ then

$$\hat{v} = v(\hat{r}, \hat{v}_{G\|s}) = \sqrt{\frac{1}{T_0\hat{T}_{s,\mathrm{coll}}}\left(\frac{1}{2}m_0\hat{A}_s v^2_{T_{s0}}\hat{v}^2_{G\|s} + \frac{T_0}{B_0}\hat{\mu}_s B_0\hat{B}_{\mathrm{axis}}\right)} = \sqrt{\frac{1}{\hat{T}_{s,\mathrm{coll}}}\left(\frac{1}{2}\hat{v}^2_{G\|s} + \hat{\mu}_s\hat{B}_{\mathrm{axis}}\right)}$$

Let a function Φ and the Chandrasekhar function G be defined as

$$\Phi(\hat{v}) = \frac{2}{\sqrt{\pi}}\int_0^{\hat{v}} e^{-x^2}\mathrm{d}x \quad ; \quad \Phi'(\hat{v}) = \frac{2}{\sqrt{\pi}}e^{-\hat{v}^2} \quad \text{and} \quad G(\hat{v}) = \frac{\Phi(\hat{v}) - \hat{v}\Phi'(\hat{v})}{2\hat{v}^2}$$

The diffusion term $\mathcal{D}_{\|s}$ can be expressed in function of these functions as

$$\mathcal{D}_{\|s} = v^2_{T_{s0}}\Omega_{c_0}\hat{\mathcal{D}}_{\|s} \quad \text{with} \quad \hat{\mathcal{D}}_{\|s}(\hat{r}, \hat{v}) = \hat{\mathcal{A}}_s(\hat{r})\left(\frac{\Phi(\hat{v}) - G(\hat{v})}{2\hat{v}}\right) \tag{K.56}$$

where the normalised radial profile $\hat{\mathcal{A}}_s$ is given by [2] :

$$\hat{\mathcal{A}}_s(\hat{r}) = 3\frac{\sqrt{\pi}}{2}\frac{\hat{v}^3_{T_{s,\mathrm{coll}}}\hat{\epsilon}^{3/2}}{\sqrt{\hat{A}_s}q(\hat{r})\hat{R}_0}\hat{\nu}_{*s}$$

The collisionality $\hat{\nu}_{*s}$ being the scalar value given as input data in the code. The scalar values $\hat{\nu}_{*s'}$ for the minority ion species s' are deduced from $\hat{\nu}_{*s}$ as:

$$\hat{\nu}_{*s'} = \left(\frac{\hat{n}_{s'}}{\hat{n}_s}\right)\left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^4\left(\frac{\hat{T}_{s,\mathrm{coll}}}{\hat{T}_{s',\mathrm{coll}}}\right)^2\hat{\nu}_{*s}$$

[2] Because, $\mathcal{A}_s(r) = 3\frac{\sqrt{\pi}}{2}\frac{v^3_{T_{s,\mathrm{coll}}}\epsilon^{3/2}}{qR_0}\nu_{*s} = 3\frac{\sqrt{\pi}}{2}\frac{v^3_{T_{s0}}\hat{v}^3_{T_{s,\mathrm{coll}}}\hat{\epsilon}^{3/2}}{q(\hat{r})\rho_0\hat{R}_0}\hat{\nu}_{*s} = v^2_{T_{s0}}\frac{v_{T_0}}{\sqrt{\hat{A}_s}}\frac{1}{\rho_0}\left(3\frac{\sqrt{\pi}}{2}\frac{\hat{v}^3_{T_{s,\mathrm{coll}}}\hat{\epsilon}^{3/2}}{q(\hat{r})\hat{R}_0}\right)\hat{\nu}_{*s}$
$= v^2_{T_{s0}}\Omega_{c0}\frac{1}{\sqrt{\hat{A}_s}}\left(3\frac{\sqrt{\pi}}{2}\frac{\hat{v}^3_{T_{s,\mathrm{coll}}}\hat{\epsilon}^{3/2}}{q(\hat{r})\hat{R}_0}\right)\hat{\nu}_{*s}$

by using the equality $\nu_{*s'} = (n_{s'}/n_s)(Z_{s'}/Z_s)^4(T_{s,\mathrm{coll}}/T_{s',\mathrm{coll}})^2\nu_{*s}$. By using the previous equality $\mathcal{D}_{\|s} = v_{T_{s0}}^2\Omega_{c_0}\hat{\mathcal{D}}_{\|s}$, the parallel velocity drag $\mathcal{V}_{\|s}$ can be expressed as

$$\mathcal{V}_{\|s}(r,v) = -\frac{\left(v_{G\|} - V_{\|s,\mathrm{coll}}\right)}{v_{T_{s,\mathrm{coll}}}^2}\mathcal{D}_{\|s}(r,v) = -\frac{v_{T_{s0}}\left(\hat{v}_{G\|s} - \hat{V}_{\|s,\mathrm{coll}}\right)}{v_{T_{s0}}^2\hat{v}_{T_{s,\mathrm{coll}}}^2}\left(v_{T_{s0}}^2\Omega_{c_0}\hat{\mathcal{D}}_{\|s}\right)$$

i.e

$$\mathcal{V}_{\|s}(r,v) = v_{T_{s0}}\Omega_{c_0}\hat{\mathcal{V}}_{\|s}(\hat{r},\hat{v}) \quad \text{with} \quad \hat{\mathcal{V}}_{\|s}(\hat{r},\hat{v}) = -\frac{\left(\hat{v}_{G\|s} - \hat{V}_{\|s,\mathrm{coll}}\right)}{\hat{v}_{T_{s,\mathrm{coll}}}^2}\hat{\mathcal{D}}_{\|s}(\hat{r},\hat{v}) \tag{K.57}$$

Finally, the Boltzmann equation

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = \mathcal{C}_s(\bar{F}_s) \quad \text{with} \quad \mathcal{C}_s(\bar{F}_s) = \frac{1}{B_{\|s}^*}\partial_{v_{G\|}}\left\{B_{\|s}^*\mathcal{D}_{\|s}\partial_{v_{G\|}}\bar{F}_s - B_{\|s}^*\mathcal{V}_{\|s}\bar{F}_s\right\}$$

is equivalent to (by using the normalized quantities and the two equalities (K.56) and (K.57)):

$$\begin{aligned}\Omega_{c0}\left(\frac{n_0}{v_{T_{s0}}^3}\right)\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} \quad = \quad & \frac{1}{B_0\hat{B}_{\|s}^*}\frac{1}{v_{T_{s0}}}\frac{\partial}{\partial\hat{v}_{G\|s}}\left\{B_0\hat{B}_{\|s}^*\frac{n_0}{v_{T_{s0}}^3}\times\right.\\ & \left.\left[\left(v_{T_{s0}}^2\Omega_{c_0}\hat{\mathcal{D}}_{\|s}\right)\frac{1}{v_{T_{s0}}}\frac{\partial}{\partial\hat{v}_{G\|s}}\hat{\bar{F}}_s - v_{T_{s0}}\Omega_{c_0}\hat{\mathcal{V}}_{\|s}\hat{\bar{F}}_s\right]\right\}\end{aligned}$$

Therefore,

$$\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} \quad \text{with} \quad \hat{\mathcal{C}}_s(\bar{F}_s) = \frac{1}{\hat{B}_{\|s}^*}\frac{\partial}{\partial\hat{v}_{G\|s}}\left[\hat{B}_{\|s}^*\left(\hat{\mathcal{D}}_{\|s}\frac{\partial\hat{\bar{F}}_s}{\partial\hat{v}_{G\|s}} - \hat{\mathcal{V}}_{\|s}\hat{\bar{F}}_s\right)\right]$$

Concerning the computation of the mean velocity and the mean temperature, then the integrals involved read:

$$\begin{aligned}
I_0 &= \langle m_s\mathcal{D}_{\|s}\rangle = \int\frac{2\pi B_{\|s}^*}{m_s}m_s\mathcal{D}_{\|s}\bar{F}_s\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_s\\
&= \int 2\pi B_0\hat{B}_{\|s}^*(\Omega_{c_0}v_{T_{s0}}^2\hat{\mathcal{D}}_{\|s})\left(\frac{n_0}{v_{T_{s0}}^3}\hat{\bar{F}}_s\right)v_{T_{s0}}\,\mathrm{d}\hat{v}_{G\|s}\frac{T_0}{B_0}\,\mathrm{d}\hat{\mu}_s\\
&= (n_0\,\Omega_{c_0}\,T_0)\,\hat{I}_0 \quad \text{with} \quad \hat{I}_0 = \int\hat{\bar{F}}_s\hat{\mathcal{D}}_{\|s}\,\hat{\mathcal{J}}_{\mathrm{v}}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s = \langle\hat{\mathcal{D}}_{\|s}\rangle\\
I_1 &= \langle m_s\mathcal{D}_{\|s}v_{G\|}\rangle = (n_0\,\Omega_{c_0}\,T_0\,v_{T_{s0}})\,\hat{I}_1\\
&\quad \text{with} \quad \hat{I}_1 = \int\hat{v}_{G\|s}\hat{\mathcal{D}}_{\|s}\hat{\bar{F}}_s\,\hat{\mathcal{J}}_{\mathrm{v}}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s = \langle\hat{\mathcal{D}}_{\|s}\hat{v}_{G\|s}\rangle\\
I_2 &= \langle m_s^2\mathcal{D}_{\|s}v_{G\|}^2\rangle = \left(\hat{A}_s m_0 n_0\,\Omega_{c_0}\,T_0\,v_{T_{s0}}^2\right)\hat{I}_2\\
&\quad \text{with} \quad \hat{I}_2 = \int\hat{v}_{G\|s}^2\hat{\mathcal{D}}_{\|s}\hat{\bar{F}}_s\,\hat{\mathcal{J}}_{\mathrm{v}}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s = \langle\hat{\mathcal{D}}_{\|s}\hat{v}_{G\|s}^2\rangle\\
I_3 &= \langle\frac{1}{B_{\|s}^*}\,\partial_{v_{G\|}}\left(B_{\|s}^*\,\mathcal{D}_{\|s}\right)\rangle = \left(\frac{1}{\hat{A}_s m_0}\frac{n_0\,\Omega_{c_0}\,T_0}{v_{T_{s0}}}\right)\hat{I}_3 \quad \text{with} \quad \hat{I}_3 = \langle\frac{1}{\hat{B}_{\|s}^*}\,\partial_{\hat{v}_{G\|s}}\left(\hat{B}_{\|s}^*\,\hat{\mathcal{D}}_{\|s}\right)\rangle\\
I_4 &= \langle\frac{m_s}{B_{\|s}^*}\,\partial_{v_{G\|}}\left(B_{\|s}^*\,v_{G\|}\,\mathcal{D}_{\|s}\right)\rangle = (n_0\,\Omega_{c_0}\,T_0)\,\hat{I}_4 \quad \text{with} \quad \hat{I}_4 = \langle\frac{1}{\hat{B}_{\|s}^*}\,\partial_{\hat{v}_{G\|s}}\left(\hat{B}_{\|s}^*\,\hat{\mathcal{D}}_{\|s}\right)\rangle
\end{aligned}$$

where $\langle\cdot\rangle = \int\cdot\,\hat{\bar{F}}_s\hat{\mathcal{J}}_{\mathrm{v}}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s$. Finally,

$$\begin{aligned}
P &= I_0\times I_4 - m_s I_1\times I_3 = (n_0\,\Omega_{c_0}\,T_0)^2\,\hat{I}_0\hat{I}_4 - \hat{A}_s m_0\,(n_0\,\Omega_{c_0}\,T_0\,v_{T_{s0}})\,\hat{I}_1\frac{n_0\,\Omega_{c_0}\,T_0}{\hat{A}_s m_0 v_{T_{s0}}}\hat{I}_3\\
&= (n_0\,\Omega_{c_0}\,T_0)^2\,\hat{P} \quad \text{with} \quad \hat{P} = \left[\hat{I}_0\hat{I}_4 - \hat{I}_1\hat{I}_3\right]
\end{aligned}$$

and

$$\begin{aligned} V_{\|s,\mathrm{coll}} &= P^{-1}\left(I_4 \times I_1 - I_2 \times I_3\right) = \frac{\hat{P}^{-1}}{\left(n_0\,\Omega_{c_0}\,T_0\right)^2}\left[\left(n_0\,\Omega_{c_0}\,T_0\right)^2\left\{v_{T_{s0}}\hat{I}_4\hat{I}_1 - \frac{\hat{A}_s m_0 v_{T_{s0}}^2}{\hat{A}_s m_0 v_{T_{s0}}}\hat{I}_2\hat{I}_3\right\}\right] \\ &= v_{T_{s0}}\hat{V}_{\|s,\mathrm{coll}} \quad \text{with} \quad \hat{V}_{\|s,\mathrm{coll}} = \hat{P}^{-1}\left[\hat{I}_4\hat{I}_1 - \hat{I}_2\hat{I}_3\right] \end{aligned}$$

and

$$\begin{aligned} T_{s,\mathrm{coll}} &= P^{-1}\left(I_0 \times I_2 - m_s I_1^2\right) = \hat{P}^{-1}\left(\hat{A}_s m_0 v_{T_{s0}}^2 \hat{I}_0\hat{I}_2 - \hat{A}_s m_0 v_{T_{s0}}^2 \hat{I}_1^2\right) \\ &= \hat{A}_s m_0 \frac{T_0}{m_0}\left(\frac{1}{\sqrt{\hat{A}_s}}\right)^2 \hat{P}^{-1}\left(\hat{I}_0\hat{I}_2 - \hat{I}_1^2\right) \\ T_{s,\mathrm{coll}} &= T_0\hat{T}_{s,\mathrm{coll}} \quad \text{with} \quad \hat{T}_{s,\mathrm{coll}} = \hat{P}^{-1}\left(\hat{I}_0\hat{I}_2 - \hat{I}_1^2\right) \quad \blacksquare \end{aligned}$$

K.2.8 Proof for normalization of collisional energy transfers (eqs.(K.36)-(K.37))

According to (2.36) and (2.37) the energy exchange between two species is taken into account as:

$$\begin{aligned} \frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} &= -\frac{\eta^{\mathcal{E}}_{ss'}}{\left(2\pi T_{\mathrm{mean}}/m_s\right)^{3/2}}\frac{\Delta T_{ss'}}{T_{\mathrm{mean}}}\left(\frac{\mathcal{E}_{ss'}}{T_{\mathrm{mean}}} - \frac{3}{2}\right)\exp\left(-\frac{\mathcal{E}_{ss'}}{T_{\mathrm{mean}}}\right) \\ \frac{\mathrm{d}\bar{F}_{s'}}{\mathrm{d}t} &= -\frac{\eta^{\mathcal{E}}_{s's}}{\left(2\pi T_{\mathrm{mean}}/m_{s'}\right)^{3/2}}\frac{\Delta T_{s's}}{T_{\mathrm{mean}}}\left(\frac{\mathcal{E}_{s's}}{T_{\mathrm{mean}}} - \frac{3}{2}\right)\exp\left(-\frac{\mathcal{E}_{s's}}{T_{\mathrm{mean}}}\right) \end{aligned}$$

Therefore, using GYSELA normalizations, the previous first equation reads:

$$\begin{aligned} \Omega_{c_0}\left(\frac{n_0}{v_{T_{s0}}^3}\right)\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} &= -\frac{\eta^{\mathcal{E}}_{ss'}}{\left(2\pi T_0\hat{T}_{\mathrm{mean}}/(\hat{A}_s m_0)\right)^{3/2}}\frac{\Delta\hat{T}_{ss'}}{\hat{T}_{\mathrm{mean}}}\left(\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}} - \frac{3}{2}\right)\exp\left(-\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}}\right) \\ \frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} &= -\frac{1}{n_0\Omega_{c_0}}\frac{\eta^{\mathcal{E}}_{ss'}}{\left(2\pi\hat{T}_{\mathrm{mean}}\right)^{3/2}}\frac{\Delta\hat{T}_{ss'}}{\hat{T}_{\mathrm{mean}}}\left(\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}} - \frac{3}{2}\right)\exp\left(-\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}}\right) \end{aligned}$$

Besides,

$$\begin{aligned} \eta^{\mathcal{E}}_{ss'} &= \frac{8\,\varepsilon^{3/2}}{3\sqrt{\pi}}\frac{m_s}{m_{s'}}\left(\frac{Z_{s'}}{Z_s}\right)^2\frac{n_{s'}\sqrt{T_{s,\mathrm{coll}}/m_s}}{qR_0}\left(1 + \frac{v_{T_s'}^2}{v_{T_s}^2}\right)^{-3/2}\nu_{*s} \\ &= \frac{8\,\hat{\varepsilon}^{3/2}}{3\sqrt{\pi}}\frac{\hat{A}_s}{\hat{A}_{s'}}\left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2 n_0\hat{n}_{s'}\frac{\sqrt{T_0\hat{T}_{s,\mathrm{coll}}/(\hat{A}_s m_0)}}{\hat{q}\hat{R}_0\rho_0}\left(1 + \frac{\hat{A}_s}{\hat{A}_{s'}}\frac{\hat{T}_{s'}^2}{\hat{T}_s^2}\right)^{-3/2}\hat{\nu}_{*s} \end{aligned}$$

Then,

$$\hat{\eta}^{\mathcal{E}}_{ss'} = \left(\frac{n_0 v_{T_0}}{\rho_0}\right)\hat{\eta}^{\mathcal{E}}_{ss'} \quad \text{with } \hat{\eta}^{\mathcal{E}}_{ss'} = \frac{8\,\hat{\varepsilon}^{3/2}}{3\sqrt{\pi}}\frac{\sqrt{\hat{A}_s}}{\hat{A}_{s'}}\left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2\hat{n}_{s'}\frac{\sqrt{\hat{T}_{s,\mathrm{coll}}}}{\hat{q}\hat{R}_0}\left(1 + \frac{\hat{A}_s}{\hat{A}_{s'}}\frac{\hat{T}_{s'}^2}{\hat{T}_s^2}\right)^{-3/2}\hat{\nu}_{*s}$$

Finally, using the relation $\Omega_{c_0} = v_{T_0}/\rho_0$,

$$\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} = -\frac{\hat{\eta}^{\mathcal{E}}_{ss'}}{\left(2\pi\hat{T}_{\mathrm{mean}}\right)^{3/2}}\frac{\Delta\hat{T}_{ss'}}{\hat{T}_{\mathrm{mean}}}\left(\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}} - \frac{3}{2}\right)\exp\left(-\frac{\hat{\mathcal{E}}_{ss'}}{\hat{T}_{\mathrm{mean}}}\right) \quad \blacksquare$$

The same kind of proof can be done for $\frac{\mathrm{d}\hat{\bar{F}}_{s'}}{\mathrm{d}\hat{t}}$.

K.2.9 Proof for normalization of collisional momentum transfers (eqs.(K.39) - (K.40))

According to equation (2.42),

$$\begin{aligned}
\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} &= -\hat{\eta}_{ss'}^{v_{G\|}} \Delta V_{ss'} v_{G\|} \exp\left(-\frac{E_s}{T_{\text{mean}}}\right) \\
\frac{n_0\Omega_0}{v_{T_{s0}}^3}\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} &= -\hat{\eta}_{ss'}^{v_{G\|}} v_{T_0} \Delta \hat{V}_{ss'} v_{T_{s0}} \hat{v}_{G\|s} \exp\left(-\frac{\hat{E}_s}{\hat{T}_{\text{mean}}}\right) \\
\frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} &= -\left(\frac{v_{T_{s0}}^4 v_{T_0}}{n_0\Omega_0}\right) \hat{\eta}_{ss'}^{v_{G\|}} \Delta \hat{V}_{ss'} \hat{v}_{G\|s} \exp\left(-\frac{\hat{E}_s}{\hat{T}_{\text{mean}}}\right)
\end{aligned}$$

with $\Delta\hat{V}_{ss'} = \hat{V}_{\|s}\hat{A}_s^{-1/2} - \hat{V}_{\|s'}\hat{A}_{s'}^{-1/2}$. Besides, according to equation (2.44),

$$\eta_{ss'}^{v_{G\|}} = n_{s'} m_s^2 \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{qR_0} \frac{\sqrt{T_{s,\text{coll}}}}{T_{\text{mean}}^{5/2}} \left(\frac{Z_{s'}}{Z_s}\right)^2 \frac{v_{T_s}}{v_{T_>}} \nu_{*s}$$

where $v_{T_>} = \max(v_{T_s}, v_{T'_s}) = v_{T_0}\hat{v}_{T_>}$ with $\hat{v}_{T_>} = \max(\hat{A}_s^{-1/2}\hat{v}_{T_s}, \hat{A}_{s'}^{-1/2}\hat{v}_{T'_s})$. Therefore,

$$\begin{aligned}
\eta_{ss'}^{v_{G\|}} &= n_{s'} m_s^{3/2} \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{qR_0} \frac{\sqrt{T_{s,\text{coll}}}}{T_{\text{mean}}^{5/2}} \left(\frac{Z_{s'}}{Z_s}\right)^2 \frac{\sqrt{T_s}}{v_{T_>}} \nu_{*s} \\
&= n_0 \hat{n}_{s'} m_0^{3/2} \hat{A}_s^{3/2} \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{q\hat{R}_0\rho_0} \frac{\sqrt{T_0}\sqrt{\hat{T}_{s,\text{coll}}}}{T_0^{5/2}\hat{T}_{\text{mean}}^{5/2}} \left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2 \frac{\sqrt{T_0}\sqrt{\hat{T}_s}}{v_{T_0}\hat{v}_{T_>}} \nu_{*s}
\end{aligned}$$

which leads to

$$\begin{aligned}
\eta_{ss'}^{v_{G\|}} &= \frac{n_0 m_0^{3/2}}{\rho_0} \frac{T_0}{T_0^{5/2} v_{T_0}} \hat{A}_s^{3/2} \hat{n}_{s'} \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{q\hat{R}_0} \frac{\sqrt{\hat{T}_{s,\text{coll}}}}{\hat{T}_{\text{mean}}^{5/2}} \left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2 \frac{\sqrt{\hat{T}_s}}{\hat{v}_{T_>}} \nu_{*s} \\
&= \frac{n_0\Omega_0}{v_{T_0}^4} \frac{\hat{A}_s^{3/2}}{v_{T_0}} \hat{n}_{s'} \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{q\hat{R}_0} \frac{\sqrt{\hat{T}_{s,\text{coll}}}}{\hat{T}_{\text{mean}}^{5/2}} \left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2 \frac{\sqrt{\hat{T}_s}}{\hat{v}_{T_>}} \nu_{*s} \quad \text{because } \rho_0 = \frac{v_{T_0}}{\Omega_0} \\
&= \hat{A}_s^{-1/2} \frac{n_0\Omega_0}{v_{T_{s0}}^4 v_{T_0}} \hat{n}_{s'} \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{q\hat{R}_0} \frac{\sqrt{\hat{T}_{s,\text{coll}}}}{\hat{T}_{\text{mean}}^{5/2}} \left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2 \frac{\sqrt{\hat{T}_s}}{\hat{v}_{T_>}} \nu_{*s} \quad \text{because } v_{T_0} = \sqrt{\hat{A}_s}\, v_{T_{s0}}
\end{aligned}$$

Finally,

$$\eta_{ss'}^{v_{G\|}} = \frac{n_0\Omega_0}{v_{T_{s0}}^4 v_{T_0}} \hat{\eta}_{ss'}^{v_{G\|}} \quad \text{with} \quad \hat{\eta}_{ss'}^{v_{G\|}} = \hat{A}_s^{-1/2} \hat{n}_{s'} \left(\frac{\varepsilon}{2\pi}\right)^{3/2} \frac{1}{q\hat{R}_0} \frac{\sqrt{\hat{T}_s}\sqrt{\hat{T}_{s,\text{coll}}}}{\hat{v}_{T_>}\hat{T}_{\text{mean}}^{5/2}} \left(\frac{\hat{Z}_{s'}}{\hat{Z}_s}\right)^2 \nu_{*s}$$

Using previous $\eta_{ss'}^{v_{G\|}}$ expression gives equation (K.39). ∎
The same kind of computations can be done to prove equation (K.40).

K.2.10 Proof for source term normalization (eqs.(K.43) -(K.48))

As a reminder the source terms are defined as

$$\frac{\mathrm{d}\bar{F}_s}{\mathrm{d}t} = S_{\text{heat}}(r,\theta,v_{G\|},\mu_s) + S_{\text{moment}}(r,\theta,v_{G\|},\mu_s) + S_{\text{vorticity}}(r,\theta,v_{G\|},\mu_s) \tag{K.58}$$

with

$$S_{\text{heat}} = K_{\text{h}}(\bar{v}_{G_\parallel s}, \bar{\mu}_s, J_{\parallel B}) \frac{2S_0^{\mathcal{E}}}{\left(\frac{2\pi T_{s,\text{srce}}}{m_s}\right)^{3/2} T_{s,\text{srce}}} S_r^{\mathcal{E}} \, \mathrm{e}^{-\bar{v}_{G_\parallel s}^2 - \bar{\mu}_s} \tag{K.59}$$

$$S_{\text{moment}} = K_{\text{m}}(\bar{v}_{G_\parallel s}, \bar{\mu}_s, J_{\parallel B}) \frac{S_0^{v_{G\parallel}}}{4\pi^{3/2}\left(\frac{T_{s,\text{srce}}}{m_s}\right)^2} S_r^{v_{G\parallel}} \, \mathrm{e}^{-\bar{v}_{G_\parallel s}^2 - \bar{\mu}_s} \tag{K.60}$$

$$S_{\text{vorticity}} = -\left[2\bar{v}_{G_\parallel s}^2 - \bar{\mu}_s\right] \frac{S_0^{\Omega}}{\alpha\left(\frac{2\pi T_{s,\text{srce}}}{m_s}\right)^{3/2}} S_r^{\Omega} \, \mathrm{e}^{-\bar{v}_{G_\parallel s}^2 - \bar{\mu}_s} \tag{K.61}$$

where

$$K_{\text{h}}(\bar{v}_{G_\parallel s}, \bar{\mu}_s, J_{\parallel B}) = \left[\bar{v}_{G_\parallel s}^2 - \frac{1}{2} - \frac{J_{\parallel B}}{2 - J_{\parallel B}^2}(2 - \bar{\mu}_s)\left(2\bar{v}_{G_\parallel s} - J_{\parallel B}\right)\right]$$
$$K_{\text{m}}(\bar{v}_{G_\parallel s}, \bar{\mu}_s, J_{\parallel B}) = \left[2\bar{v}_{G_\parallel s}(2 - \bar{\mu}_s) - J_{\parallel B}\left(1 + 2\bar{v}_{G_\parallel s} - \bar{\mu}_s\right)\right]$$

and

$$\bar{\mu}_s = \mu_s B / T_{s,\text{srce}}, \; \bar{v}_{G_\parallel s} = \frac{v_{G\parallel}}{\sqrt{2T_{s,\text{srce}}/m_s}}, \; J_{\parallel B} \equiv \frac{\sqrt{m_s}}{q_s}\frac{\sqrt{2T_{s,\text{srce}}}}{B^2}\mu_0 J_\parallel \text{ and } \alpha = \frac{m_s}{q_s^2}\frac{T_{s,\text{srce}}}{2B^2} \tag{K.62}$$

Using the GYSELA normalization, the previous quantities defined by equations (K.62) reads

$$\bar{\mu}_s = \frac{\mu_s B}{T_{s,\text{srce}}} = \hat{\mu}_s \frac{T_0}{B_0}\frac{B_0\hat{B}}{T_0\hat{T}_{s,\text{srce}}} = \frac{\hat{\mu}_s\hat{B}}{\hat{T}_{s,\text{srce}}} = \hat{\bar{\mu}}_s$$
$$\bar{v}_{G_\parallel s} = \frac{v_{G\parallel}}{\sqrt{\frac{2T_{s,\text{srce}}}{m_s}}} = \frac{v_{T_{s0}}\hat{v}_{G\parallel s}}{\sqrt{\frac{2T_0\hat{T}_{s,\text{srce}}}{\hat{A}_s m_0}}} = \frac{v_{T_{s0}}}{v_{T_0}}\sqrt{\hat{A}_s}\frac{\hat{v}_{G\parallel s}}{\sqrt{2\hat{T}_{s,\text{srce}}}} = \frac{\hat{v}_{G\parallel s}}{\sqrt{2\hat{T}_{s,\text{srce}}}} = \hat{\bar{v}}_{G_\parallel s}$$

while

$$J_{\parallel B} = \frac{\sqrt{\hat{A}_s m_0}}{Z_0 e\hat{Z}_s}\frac{\sqrt{2T_0\hat{T}_{s,\text{srce}}}}{B_0^2\hat{B}^2}\mu_0\hat{J}_\parallel\frac{B_0}{\rho_0} = \frac{\sqrt{\hat{A}_s}}{\hat{Z}_s}\frac{\sqrt{m_0}}{Z_0 e B_0}\frac{\sqrt{T_0}}{\rho_0}\frac{\sqrt{2\hat{T}_{s,\text{srce}}}}{\hat{B}^2}\mu_0\hat{J}_\parallel$$
$$= \frac{\sqrt{\hat{A}_s}}{\hat{Z}_s}\frac{\sqrt{2\hat{T}_{s,\text{srce}}}}{\hat{B}^2}\mu_0\hat{J}_\parallel = \hat{J}_{\parallel B}$$

and

$$\alpha = \frac{m_s}{q_s^2}\frac{T_{s,\text{srce}}}{2B^2} = \frac{\hat{A}_s m_0}{Z_0^2 e^2\hat{Z}_s^2}\frac{T_0\hat{T}_{s,\text{srce}}}{2B_0^2\hat{B}^2} = \frac{m_0^2}{Z_0^2 e^2 B_0^2}\frac{T_0}{m_0}\frac{\hat{A}_s}{\hat{Z}_s^2}\frac{\hat{T}_{s,\text{srce}}}{2\hat{B}^2} = \frac{v_{T_0}^2}{\Omega_{c_0}^2}\frac{\hat{A}_s}{\hat{Z}_s^2}\frac{\hat{T}_{s,\text{srce}}}{2\hat{B}^2}$$
$$= \frac{1}{\rho_0^2}\frac{\hat{A}_s}{\hat{Z}_s^2}\frac{\hat{T}_{s,\text{srce}}}{2\hat{B}^2} = \frac{1}{\rho_0^2}\hat{\alpha} \quad \text{with } \hat{\alpha} = \frac{\hat{A}_s}{\hat{Z}_s^2}\frac{\hat{T}_{s,\text{srce}}}{2\hat{B}^2}$$

The prescribed radial source profiles ($S_r^{\mathcal{E}}$, $S_r^{v_{G\parallel}}$ and S_r^{Ω}) are defined as

$$S_r^x(r) = -\frac{1}{2}\left[\tanh\left(\frac{\rho - (\rho_S^x + 3L_S^x)}{L_S^x}\right) + \tanh\left(-\frac{\rho - (\rho_S^x - 3L_S^x)}{L_S^x}\right)\right]$$
$$= -\frac{1}{2}\left[\tanh\left(\frac{\hat{\rho} - (\hat{\rho}_S^x + 3\hat{L}_S^x)}{\hat{L}_S^x}\right) + \tanh\left(-\frac{\hat{\rho} - (\hat{\rho}_S^x - 3\hat{L}_S^x)}{\hat{L}_S^x}\right)\right] = \hat{S}_r^x(\hat{r})$$

According to definition (K.59)

$$S_{\text{heat}} = \hat{K}_{\text{h}} \frac{2S_0^{\mathcal{E}}}{\left(\frac{2\pi T_0 \hat{T}_{s,\text{srce}}}{\hat{A}_s m_0}\right)^{3/2} T_0 T_{s,\text{srce}}} \hat{S}_r^{\mathcal{E}} \exp\left(-\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right)$$
$$= \frac{1}{v_{T_{s0}}^3 T_0} \hat{K}_{\text{h}} \frac{2S_0^{\mathcal{E}}}{\left(2\pi \hat{T}_{s,\text{srce}}\right)^{3/2} T_{s,\text{srce}}} \hat{S}_r^{\mathcal{E}} \exp\left(-\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right)$$

with

$$\hat{K}_{\text{h}}(\hat{\bar{v}}_{G_\parallel s}, \hat{\bar{\mu}}_s, \hat{J}_{\parallel B}) = \left[\hat{\bar{v}}_{G_\parallel s}^2 - \frac{1}{2} - \frac{\hat{J}_{\parallel B}}{2 - \hat{J}_{\parallel B}^2}(2 - \hat{\bar{\mu}}_s)\left(2\hat{\bar{v}}_{G_\parallel s} - \hat{J}_{\parallel B}\right)\right]$$

According to definition (K.60)

$$S_{\text{moment}} = \hat{K}_{\text{m}} \frac{S_0^{v_{G\parallel}}}{4\pi^{3/2}\left(\frac{T_0 \hat{T}_{s,\text{srce}}}{\hat{A}_s m_0}\right)^2} \hat{S}_r^{v_{G\parallel}} \exp\left(-\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right)$$
$$= \frac{1}{v_{T_{s0}}^4} \hat{K}_{\text{m}} \frac{S_0^{v_{G\parallel}}}{4\pi^{3/2}\hat{T}_{s,\text{srce}}^2} \hat{S}_r^{v_{G\parallel}} \exp\left(-\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right)$$

with $\hat{K}_{\text{m}}(\hat{\bar{v}}_{G_\parallel s}, \hat{\bar{\mu}}_s, \hat{J}_{\parallel B}) = \left[2\hat{\bar{v}}_{G_\parallel s}(2 - \hat{\bar{\mu}}_s) - \hat{J}_{\parallel B}\left(1 + 2\hat{\bar{v}}_{G_\parallel s} - \hat{\bar{\mu}}_s\right)\right]$. Finally, according to (K.61)

$$S_{\text{vorticity}} = -\left[2\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right] \frac{S_0^{\Omega}}{\alpha\left(\frac{2\pi T_0 \hat{T}_{s,\text{srce}}}{\hat{A}_s m_0}\right)^{3/2}} \hat{S}_r^{\Omega} \exp\left(-\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right)$$
$$= -\frac{\rho_0^2}{v_{T_{s0}}^3}\left[2\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right] \frac{S_0^{\Omega}}{\hat{\alpha}\left(2\pi \hat{T}_{s,\text{srce}}\right)^{3/2}} \hat{S}_r^{\Omega} \exp\left(-\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right)$$

Replacing S_{heat}, S_{moment} and $S_{\text{vorticity}}$ by these new expressions in equation (K.58) leads to

$$\frac{n_0 \Omega_{C_0}}{v_{T_{s0}}^3} \frac{\mathrm{d}\hat{\bar{F}}_s}{\mathrm{d}\hat{t}} = \exp\left(-\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right) \left[\frac{1}{v_{T_{s0}}^3 T_0} \hat{K}_{\text{h}} \frac{2S_0^{\mathcal{E}}}{\left(2\pi \hat{T}_{s,\text{srce}}\right)^{3/2} T_{s,\text{srce}}} \hat{S}_r^{\mathcal{E}} + \right.$$
$$\left. \frac{1}{v_{T_{s0}}^4} \hat{K}_{\text{m}} \frac{S_0^{v_{G\parallel}}}{4\pi^{3/2}\hat{T}_{s,\text{srce}}^2} \hat{S}_r^{v_{G\parallel}} - \frac{\rho_0^2}{v_{T_{s0}}^3}\left[2\hat{\bar{v}}_{G_\parallel s}^2 - \hat{\bar{\mu}}_s\right] \frac{S_0^{\Omega}}{\hat{\alpha}\left(2\pi \hat{T}_{s,\text{srce}}\right)^{3/2}} \hat{S}_r^{\Omega}\right]$$

Then, using the following normalizations

$$S_0^{\mathcal{E}} = n_0 \Omega_{C_0} T_0 \hat{S}_0^{\mathcal{E}} \quad ; \quad S_0^{v_{G\parallel}} = n_0 \Omega_{C_0} v_{T_{s0}} \hat{S}_0^{v_{G\parallel}} \quad \text{and} \quad S_0^{\Omega} = \frac{n_0 \Omega_{C_0}}{\rho_0^2} \hat{S}_0^{\Omega}$$

permits to recover equations (K.43) to (K.48). ∎

K.2.11 Proof for normalization of source of impurity (eqs.(K.49)-(K.50))

Let us consider $S_{n_{s'}}$ the source of impurity s',

$$S_{n_{s'}} = \frac{S_0^{n_{s'}} S_r}{\left(\frac{2\pi T_{s',\text{srce}}}{m_{s'}}\right)^{3/2}} \left(\frac{5}{2} - \frac{\mu_{s'} B}{T_{s',\text{srce}}} - \frac{m_{s'} v_{G\parallel}^2}{2T_{s',\text{srce}}}\right) \exp\left(-\frac{m_{s'} v_{G\parallel}^2}{2T_{s',\text{srce}}} - \frac{\mu_{s'} B}{T_{s',\text{srce}}}\right)$$

then

$$
\begin{aligned}
S_{n_{s'}} &= \frac{\hat{S}_0^{n_{s'}}\hat{S}_r}{\left(\frac{2\pi T_0 \hat{T}_{s',\mathrm{srce}}}{\hat{A}_{s'} m_0}\right)^{3/2}}\left(\frac{5}{2} - \frac{T_0}{B_0}\frac{\hat{\mu}_{s'}B_0\hat{B}}{T_0\hat{T}_{s',\mathrm{srce}}} - \frac{m_0\hat{A}_{s'}v_{Ts0}^2\hat{v}_{G\|s'}^2}{2T_0\hat{T}_{s',\mathrm{srce}}}\right) \\
&\times \exp\left(-\frac{m_0\hat{A}_{s'}v_{Ts0}^2\hat{v}_{G\|s'}^2}{2T_0\hat{T}_{s',\mathrm{srce}}} - \frac{T_0\hat{\mu}_{s'}}{B_0}\frac{B_0\hat{B}}{T_0\hat{T}_{s',\mathrm{srce}}}\right) \\
&= \frac{1}{v_{Ts0}^3}\frac{\hat{S}_0^{n_{s'}}\hat{S}_r}{\left(2\pi\hat{T}_{s',\mathrm{srce}}\right)^{3/2}}\left(\frac{5}{2} - \frac{\hat{\mu}_{s'}\hat{B}}{\hat{T}_{s',\mathrm{srce}}} - \frac{\hat{v}_{G\|s'}^2}{2\hat{T}_{s',\mathrm{srce}}}\right)\exp\left(-\frac{\hat{v}_{G\|s'}^2}{2\hat{T}_{s',\mathrm{srce}}} - \frac{\hat{\mu}_{s'}\hat{B}}{\hat{T}_{s',\mathrm{srce}}}\right)
\end{aligned}
$$

Finally,

$$
S_{n_{s'}} = \frac{\hat{S}_{n_{s'}}}{v_{Ts0}^3} \quad \text{with } \hat{S}_{n_{s'}} = \frac{\hat{S}_0^{n_{s'}}\hat{S}_r}{\left(2\pi\hat{T}_{s',\mathrm{srce}}\right)^{3/2}}\left(\frac{5}{2} - \hat{\bar{\mu}}_{s'} - \hat{\bar{v}}_{G_\|s'}^2\right)\exp\left(-\hat{\bar{v}}_{G_\|s'}^2 - \hat{\bar{\mu}}_{s'}\right) \tag{K.63}
$$

where $\hat{\bar{v}}_{G_\|s'} = \hat{v}_{G\|s'}/\sqrt{2\hat{T}_{s',\mathrm{srce}}}$ and $\hat{\bar{\mu}}_{s'} = \hat{\mu}_{s'}\hat{B}/\hat{T}_{s',\mathrm{srce}}$. This source of impurity $S_{n_{s'}}$ is compensated by a negative source of major ion S_{n_s} such that

$$
Z_s\int S_{n_s}\mathcal{J}_{\mathrm{v}_s}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_s + Z_{s'}\int S_{n_{s'}}\mathcal{J}_{\mathrm{v}_{s'}}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_{s'} = 0
$$

which gives, due to equations (K.51) and (K.63), the following expression in normalized quantities,

$$
Z_0\hat{Z}_s\int\frac{\hat{S}_{n_s}}{v_{Ts0}^3}v_{Ts0}^3\hat{\mathcal{J}}_{\mathrm{v}_s}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s + Z_0\hat{Z}_{s'}\int\frac{\hat{S}_{n_{s'}}}{v_{Ts0}^3}v_{Ts0}^3\hat{\mathcal{J}}_{\mathrm{v}_{s'}}\,\mathrm{d}\hat{v}_{G\|s'}\,\mathrm{d}\hat{\mu}_{s'} = 0
$$

Let us define the function $H(v,\mu) = \left(\frac{5}{2} - \mu - v^2\right)\exp\left(-v^2 - \mu\right)$, then the previous equation reads,

$$
\begin{aligned}
&\hat{Z}_s\int\frac{\hat{S}_0^{n_s}\hat{S}_r}{\left(2\pi\hat{T}_{s,\mathrm{srce}}\right)^{3/2}}H(\hat{v}_{G\|s},\hat{\mu}_s)\hat{\mathcal{J}}_{\mathrm{v}_s}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s \\
&\qquad + \hat{Z}_{s'}\int\frac{\hat{S}_0^{n_{s'}}\hat{S}_r}{\left(2\pi\hat{T}_{s',\mathrm{srce}}\right)^{3/2}}H(\hat{v}_{G\|s'},\hat{\mu}_{s'})\hat{\mathcal{J}}_{\mathrm{v}_{s'}}\,\mathrm{d}\hat{v}_{G\|s'}\,\mathrm{d}\hat{\mu}_{s'} = 0
\end{aligned}
$$

$$
\left(\hat{Z}_s\frac{\hat{S}_0^{n_s}\hat{S}_r}{\left(2\pi\hat{T}_{s,\mathrm{srce}}\right)^{3/2}} + \hat{Z}_{s'}\frac{\hat{S}_0^{n_{s'}}\hat{S}_r}{\left(2\pi\hat{T}_{s',\mathrm{srce}}\right)^{3/2}}\right)\int H(\hat{v}_{G\|s},\hat{\mu}_s)\hat{\mathcal{J}}_{\mathrm{v}_s}\,\mathrm{d}\hat{v}_{G\|s}\,\mathrm{d}\hat{\mu}_s = 0
$$

This gives the expression of $\hat{S}_0^{n_s}$ in function of $\hat{S}_0^{n_{s'}}$ as, $\hat{S}_0^{n_s} = -\frac{\hat{Z}_{s'}}{\hat{Z}_s}\left(\frac{\hat{T}_{s',\mathrm{srce}}}{\hat{T}_{s,\mathrm{srce}}}\right)^{3/2}\hat{S}_0^{n_{s'}}$, therefore according to (K.63):

$$
\hat{S}_{n_s} = -\frac{\hat{Z}_{s'}}{\hat{Z}_s}\left(\frac{\hat{T}_{s',\mathrm{srce}}}{\hat{T}_{s,\mathrm{srce}}}\right)^{3/2}\frac{\hat{S}_0^{n_{s'}}\hat{S}_r}{\left(2\pi\hat{T}_{s,\mathrm{srce}}\right)^{3/2}}\left(\frac{5}{2} - \hat{\bar{\mu}}_s - \hat{\bar{v}}_{G_\|s}^2\right)\exp\left(-\hat{\bar{v}}_{G_\|s}^2 - \hat{\bar{\mu}}_s\right)
$$

In the code, the radial temperature profiles $\hat{T}_{s,\mathrm{srce}}$ and $\hat{T}_{s',\mathrm{srce}}$ are imposed equal.

Appendix L

Invariance

Let us consider a reference simulation without source terms. Let us also define a second simulation similar to the reference one but where the mass and charge have been respectively multiplied by a factor α_A and α_z. Then, it is shown in this appendix that 7 control parameters (α_x, α_t, α_{T_e}, α_ϕ, α_{coll}, α_{diff} and α_{Krook} that respectively rescale the space, the time, the electron temperature, the electrostatic potential, the collision operator, the diffusion term and the Krook operator) are sufficient to ensure that both simulations –reference and scaled– are equivalent. The constraints on the different control parameters are summarized in table L.1 and the proof follows.

	Charge	Mass	Length	Time	T_e	Elec. potential
Scaling	α_Z	α_A	$\sqrt{\alpha_A}/\alpha_Z$	α_A/α_Z	α_Z^{-1}	α_Z^{-1}

	Collision	Krook	Diffusion
Scaling	1	α_Z/α_A	α_Z^{-1}

Table L.1: Scaling which must be applied to the different characteristic quantities to obtain an equivalence between any reference simulation and a scaled simulation

For the proof let us first consider the following normalized reference Vlasov equation (deduced from eqs.(K.3) to (K.8)) for a species of charge $\hat{Z}_s$ and mass $\hat{A}_s$

$$\begin{aligned}
&\frac{\partial \hat{\bar{F}}_s}{\partial \hat{t}} + \left\{ \frac{1}{\sqrt{\hat{A}_s}} \hat{v}_{G\|} \hat{\mathbf{b}}_s^* + \left(\frac{\hat{v}_{G\|}^2 + \hat{\mu}_s \hat{B}}{\hat{Z}_s \hat{B}_{\|s}^* \hat{B}} \right) \left(\mathbf{b} \times \hat{\boldsymbol{\nabla}} \hat{B} \right) + \frac{1}{\hat{B}_{\|s}^*} \left(\mathbf{b} \times \hat{\boldsymbol{\nabla}} \hat{\bar{\phi}} \right) \right\} \cdot \hat{\boldsymbol{\nabla}} \hat{\bar{F}}_s \\
&\quad + \left(-\frac{1}{\sqrt{\hat{A}_s}} \hat{\mu}_s \hat{\mathbf{b}}_s^* \cdot \hat{\nabla} \hat{B} - \frac{\hat{Z}_s}{\sqrt{\hat{A}_s}} \hat{\mathbf{b}}_s^* \cdot \hat{\nabla} \hat{\bar{\phi}} + \hat{v}_{G\|} \hat{\mathbf{v}}_{E\times B_s} \cdot \hat{\nabla} \ln \hat{B} \right) \frac{\partial \hat{\bar{F}}_s}{\partial \hat{v}_{G\|}} \\
&\quad = \frac{1}{\hat{B}_{\|s}^*} \frac{\partial}{\partial \hat{v}_{G\|}} \left[\hat{B}_{\|s}^* \hat{\mathcal{D}}_{\|s} \left(\frac{\partial \hat{\bar{F}}_s}{\partial \hat{v}_{G\|}} + \frac{\left(\hat{v}_{G\|s} - \hat{V}_{\|s,\text{coll}} \right)}{\hat{v}_{T_{s,\text{coll}}}^2} \hat{\bar{F}}_s \right) \right] + \frac{1}{\hat{B}_{\|s}^*} \left[\frac{1}{\hat{r}} \frac{\partial}{\partial \hat{r}} \left(\hat{r} \hat{\chi} \hat{B}_{\|s}^* \frac{\partial \hat{\bar{F}}_s}{\partial \hat{r}} \right) \right] \\
&\quad - \hat{\nu}(\hat{r}) (\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\text{eq}})
\end{aligned}$$

where

$$\hat{\mathbf{b}}_s^* = \frac{1}{\hat{B}_{\|s}^*} \left(\hat{\mathbf{B}} + \sqrt{\hat{A}_s} \frac{\hat{v}_{G\|}}{\hat{Z}_s \hat{B}} \hat{\boldsymbol{\nabla}} \times \hat{\mathbf{B}} \right)$$

and

$$\hat{\mathcal{D}}_{\|s}(\hat{r}, \hat{v}) = \hat{\mathcal{A}}_s(\hat{r}) \left(\frac{\Phi(\hat{v}) - G(\hat{v})}{2\hat{v}} \right) \quad \text{with} \quad \hat{\mathcal{A}}_s(\hat{r}) = \frac{1}{\sqrt{\hat{A}_s}} \left(3 \frac{\sqrt{\pi}}{2} \frac{\hat{v}_{T_{s,\text{coll}}}^3 \hat{\epsilon}^{3/2}}{q(\hat{r}) \hat{R}_0} \right) \hat{\nu}_{*s}$$

The Chrandrasekhar function G and the function Φ are defined by equation (K.28). Let us apply a

scaling with the control parameters defined below, this leads to the following new Vlasov equation

$$\begin{aligned}
&\frac{1}{\alpha_t}\frac{\partial \hat{\bar{F}}_s}{\partial \hat{t}} + \left\{ \frac{1}{\sqrt{\alpha_A \hat{A}_s}} \hat{v}_{G\|} \hat{\mathbf{b}}_s^* + \left(\frac{\hat{v}_{G\|}^2 + \hat{\mu}_s \hat{B}}{\alpha_Z \hat{Z}_s \hat{B}_{\|s}^* \hat{B}} \right) \left(\mathbf{b} \times \frac{1}{\alpha_x} \hat{\boldsymbol{\nabla}} \hat{B} \right) + \frac{1}{\hat{B}_{\|s}^*} \left(\mathbf{b} \times \frac{1}{\alpha_x} \hat{\boldsymbol{\nabla}} (\alpha_\phi \hat{\bar{\phi}}) \right) \right\} \cdot \frac{\hat{\boldsymbol{\nabla}} \hat{\bar{F}}_s}{\alpha_x} \\
&\quad + \left(-\frac{1}{\sqrt{\alpha_A \hat{A}_s}} \hat{\mu}_s \hat{\mathbf{b}}_s^* \cdot \frac{\hat{\nabla} \hat{B}}{\alpha_x} - \frac{\alpha_Z \hat{Z}_s}{\sqrt{\alpha_A \hat{A}_s}} \hat{\mathbf{b}}_s^* \cdot \frac{\hat{\nabla}(\alpha_\phi \hat{\bar{\phi}})}{\alpha_x} + \hat{v}_{G\|} \frac{\alpha_\phi}{\alpha_x} \hat{\mathbf{v}}_{E\times B_s} \cdot \frac{1}{\alpha_x} \hat{\nabla} \ln \hat{B} \right) \frac{\partial \hat{\bar{F}}_s}{\partial \hat{v}_{G\|}} \\
&\quad = \frac{1}{\hat{B}_{\|s}^*} \frac{\partial}{\partial \hat{v}_{G\|}} \left[\hat{B}_{\|s}^* \frac{\alpha_{\text{coll}}}{\sqrt{\alpha_A} \alpha_x} \hat{\mathcal{D}}_{\|s} \left(\frac{\partial \hat{\bar{F}}_s}{\partial \hat{v}_{G\|}} + \frac{\left(\hat{v}_{G\|s} - \hat{V}_{\|s,\text{coll}} \right)}{\hat{v}_{T_{s,\text{coll}}}^2} \hat{\bar{F}}_s \right) \right] \\
&\quad + \frac{1}{\hat{B}_{\|s}^*} \left[\frac{1}{\alpha_x \hat{r}} \frac{\partial}{\partial \hat{r}} \left(\hat{r} \alpha_{\text{diff}} \hat{\chi} \hat{B}_{\|s}^* \frac{1}{\alpha_x} \frac{\partial \hat{\bar{F}}_s}{\partial \hat{r}} \right) \right] - \alpha_{\text{Krook}} \hat{\nu}(\hat{r}) (\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\text{eq}})
\end{aligned}$$

with

$$\hat{\mathbf{b}}_s^* = \frac{1}{\hat{B}_{\|s}^*} \left(\hat{\mathbf{B}} + \sqrt{\alpha_A \hat{A}_s} \frac{\hat{v}_{G\|}}{\alpha_Z \hat{Z}_s \hat{B}} \frac{\hat{\boldsymbol{\nabla}} \times \hat{\mathbf{B}}}{\alpha_x} \right)$$

Concerning the left hand side, this equation is equivalent to the reference one, if and only if:

$$\frac{1}{\alpha_t} = \frac{1}{\sqrt{\alpha_A} \alpha_x} = \frac{1}{\alpha_Z \alpha_x^2} = \frac{\alpha_\phi}{\alpha_x^2} \tag{L.1}$$

$$\frac{1}{\sqrt{\alpha_A}\ \alpha_x} = \frac{\alpha_Z\ \alpha_\phi}{\sqrt{\alpha_A} \alpha_x} = \frac{\alpha_\phi}{\alpha_x^2} \tag{L.2}$$

Therefore, according to the last equality of equation (L.2), $\alpha_x = \sqrt{\alpha_A}/\alpha_Z$. Besides, due to the last equality of equation (L.1), $\alpha_\phi = \alpha_Z^{-1}$. And finally, due to the first equality of equation (L.1), $\alpha_t = \sqrt{\alpha_A}/\alpha_x$, so $\alpha_t = \alpha_A/\alpha_Z$. Taking into account the collision operator $1/\alpha_t = \alpha_{\text{coll}}/(\sqrt{\alpha_A}\alpha_x)$ which gives $\alpha_{\text{coll}} = 1$. In addition, the equivalence condition for the Krook operator leads to $1/\alpha_t = \alpha_{\text{Krook}}$ thus $\alpha_{\text{Krook}} = \alpha_Z/\alpha_A$ while for the diffusion term the condition reads $1/\alpha_t = \alpha_{\text{diff}}/\alpha_x^2$, *i.e* $\alpha_{\text{diff}} = \alpha_Z^{-1}$. Concerning the equivalence for the quasi-neutrality let us consider normalized equation

$$-\sum_s \hat{A}_s \hat{\boldsymbol{\nabla}}_\perp \left(\frac{\hat{n}_{s,\text{eq}}}{\hat{B}} \hat{\boldsymbol{\nabla}}_\perp \hat{\phi} \right) + \frac{1}{Z_0 \hat{T}_e} \sum_s \hat{Z}_s \hat{n}_{s,\text{eq}} \left[\hat{\phi} - \langle \hat{\phi} \rangle_{\text{FS}} \right] = \sum_s \hat{Z}_s \int \hat{J}_0 \left(\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\text{eq}} \right) \ \mathrm{d}^3\hat{\mathbf{v}}$$

Therefore, the rescaled equation reads:

$$-\frac{\alpha_A \alpha_\phi}{\alpha_x^2} \sum_s \hat{A}_s \hat{\boldsymbol{\nabla}}_\perp \left(\frac{\hat{n}_{s,\text{eq}}}{\hat{B}} \hat{\boldsymbol{\nabla}}_\perp \hat{\phi} \right) + \frac{\alpha_Z \alpha_\phi}{\alpha_{T_e}} \frac{1}{Z_0 \hat{T}_e} \sum_s \hat{Z}_s \hat{n}_{s,\text{eq}} \left[\hat{\phi} - \langle \hat{\phi} \rangle_{\text{FS}} \right] = \sum_s \alpha_Z \hat{Z}_s \int \hat{J}_0 \left(\hat{\bar{F}}_s - \hat{\bar{F}}_{s,\text{eq}} \right) \ \mathrm{d}^3\hat{\mathbf{v}}$$

The equivalence between reference and rescaled QN equations is ensured if and only if:

$$\frac{\alpha_A \alpha_\phi}{\alpha_x^2} = \frac{\alpha_Z \alpha_\phi}{\alpha_{T_e}} \quad \Rightarrow \quad \alpha_{T_e} = \frac{\alpha_x^2 \alpha_Z}{\alpha_A} \qquad \text{and} \qquad \alpha_Z = \frac{\alpha_Z \alpha_\phi}{\alpha_{T_e}} \quad \Rightarrow \quad \alpha_\phi = \alpha_{T_e}$$

Therefore $\alpha_{T_e} = 1/\alpha_Z$ which confirms the previous equality $\alpha_\phi = \alpha_Z^{-1}$.
Finally, regarding the gyroaverage operator in the case of a Padé approximation, *i.e* : $\hat{J}_0 \sim 1 + \frac{1}{2} \frac{\hat{A}_s}{\hat{Z}_s^2} \frac{\hat{\mu}_s}{\hat{B}} \hat{\boldsymbol{\nabla}}_\perp^2$. its invariance is ensured provided the equality $\alpha_A = \alpha_Z^2 \alpha_x^2$ is fulfilled. This equality is already satisfied by the previous constraints.

Appendix M

Detailed computations for local conservation laws

In this appendix, we present several detailed calculations useful for the derivation of the gyrokinetic conservation laws in section 4.4.

M.1 Useful integrals

Let us consider two arbitrary fields X, Y and Z. The aim of this paragraph is to compute the general integral

$$\mathcal{I} \equiv \int \mathcal{J}_{\mathrm{x}}\, \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi X \nabla \cdot (Y \nabla_{\perp} Z) \tag{M.1}$$

where $\mathcal{J}_{\mathrm{x}}$ is the jacobian is space, *i.e* $\mathcal{J}_{\mathrm{x}} = \sqrt{g}$ with g representing the determinant of the metric tensor. in order to perform this integration it is useful to write the operator $\nabla \cdot (Y \nabla_{\perp} Z)$ by using Einstein notations. Given that $\nabla\varphi \cdot \nabla\theta = \nabla\varphi \cdot \nabla\chi = 0$, we can write this operator as

$$\nabla \cdot (Y \nabla_{\perp} Z) = \frac{1}{\sqrt{g}} \partial_i \left(\sqrt{g} Y g^{ij} \partial_j Z\right) + \frac{1}{\sqrt{g}} \partial_{\varphi} \left(\sqrt{g} Y g^{\varphi\varphi} \partial_{\varphi} Z\right)$$

where i and j correspond to χ or θ. With these covariant notations, the perpendicular Laplacian-type operator we consider is equivalent to [1]

$$\nabla \cdot (Y \nabla_{\perp} Z) = \nabla_{\perp} \cdot (Y \nabla_{\perp} Z) = \frac{1}{\sqrt{g}} \partial_i \left(\sqrt{g} Y g^{ij} \partial_j Z\right)$$

Then, the previous integral (M.1) can be written as

$$\mathcal{I} = \int \mathcal{J}_{\mathrm{x}}\, \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi\, X \left\{ \frac{1}{\mathcal{J}_{\mathrm{x}}} \partial_i \left(Y \mathcal{J}_{\mathrm{x}}\, g^{ij} \partial_j Z\right) \right\}$$

Then, using an integration by parts on the coordinate i

$$\mathcal{I} = -\int \mathrm{d}\chi \mathrm{d}\theta \mathrm{d}\varphi (\partial_i X) Y \mathcal{J}_{\mathrm{x}}\, g^{ij} (\partial_j Z) + \left[\int \mathrm{d}\theta\, \mathrm{d}\varphi\, X\, Y \mathcal{J}_{\mathrm{x}}\, g^{\chi j} (\partial_j Z)\right]_{\partial L_{\chi}} + \left[\int \mathrm{d}\theta\, \mathrm{d}\varphi\, X\, Y \mathcal{J}_{\mathrm{x}}\, g^{\theta j} (\partial_j Z)\right]_0^{2\pi}$$

where i and j are still θ or χ. Because of the periodicity in θ, the surface term resulting from the integration by parts on $i = \theta$ (*i.e* last term of previous equation) is equal to 0. Therefore,

$$\mathcal{I} = -\int \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi (\partial_i X) Y \mathcal{J}_{\mathrm{x}}\, g^{ij} (\partial_j Z) + \left[\int \mathrm{d}\theta\, \mathrm{d}\varphi\, X\, Y \mathcal{J}_{\mathrm{x}}\, g^{\chi j} (\partial_j Z)\right]_{\partial L_{\chi}} \tag{M.2}$$

[1] The direction $\perp$ corresponds to an approximation of the poloidal plane because it is actually perpendicular to $\nabla\varphi$ rather than $\mathbf{B}$.

Using the fact that $g^{ij} = \nabla x^i \cdot \nabla x^j$ and $\nabla X = (\partial_i X) \nabla x^i$ for all field X, then

$$\mathcal{I} = -\int \mathcal{J}_\mathrm{x}\, \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi \left[Y (\partial_i X) \nabla x^i \cdot \nabla x^j (\partial_j Z) \right] - \int \mathcal{J}_\mathrm{x}\, \mathrm{d}\theta\, \mathrm{d}\varphi X\, Y \nabla \chi \cdot \nabla x^j (\partial_j Z)$$

therefore,

$$\mathcal{I} = -\int \mathcal{J}_\mathrm{x}\, \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi \left[Y \nabla_\perp X \cdot \nabla_\perp Z \right] + \int \mathcal{J}_\mathrm{x}\, \mathrm{d}\theta\, \mathrm{d}\varphi X\, Y \nabla \chi \cdot \nabla_\perp Z \tag{M.3}$$

Then, using again an integration by parts on the coordinate j for equation (M.2) and the fact that the surface term $\left[\int \mathrm{d}\chi\, \mathrm{d}\varphi\, (\partial_i X) Y \mathcal{J}_\mathrm{x}\, g^{i\theta} Z \right]_0^{2\pi} = 0$ due to periodicity in θ,

$$\begin{aligned} \mathcal{I} = \int \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi\, \partial_j \left(Y \mathcal{J}_\mathrm{x}\, g^{ij} \partial_i X \right) Z - \left[\int \mathrm{d}\theta\, \mathrm{d}\varphi\, (\partial_i X) Y \mathcal{J}_\mathrm{x}\, g^{i\chi} Z \right]_{\partial L_\chi} \\ + \left[\int \mathrm{d}\theta\, \mathrm{d}\varphi\, X\, Y \mathcal{J}_\mathrm{x}\, g^{\chi j} (\partial_j Z) \right]_{\partial L_\chi} \end{aligned}$$

Finally, previous equation gives

$$\begin{aligned} \int \mathcal{J}_\mathrm{x}\, \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi X \nabla_\perp \cdot (Y \nabla_\perp Z) = \int \mathcal{J}_\mathrm{x}\, \mathrm{d}\chi\, \mathrm{d}\theta\, \mathrm{d}\varphi\, \nabla_\perp \cdot (Y \nabla_\perp X)\, Z \\ - \left[\int \mathcal{J}_\mathrm{x}\, \mathrm{d}\theta\, \mathrm{d}\varphi\, Y g^{\chi j} \left\{ (\partial_j X) Z - X (\partial_j Z) \right\} \right]_{\partial L_\chi} \end{aligned} \tag{M.4}$$

M.2 Expression of $\sum_s q_s \int \mathrm{d}\tau^* \bar{U} \partial_t \bar{F}_s$ for local energy conservation

The opposite of the total energy source $-W = \sum_s q_s \int \mathrm{d}\tau^* \bar{U} \partial_t \bar{F}_s$ can be divided into two parts

$$\sum_s q_s \int \mathrm{d}\tau^* \bar{U} \partial_t \bar{F}_s = \sum_s q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \partial_t \bar{F}_s \right) + \sum_s q_s \int \mathrm{d}\tau^* \left\{ (J_0 \cdot U)\, \partial_t \bar{F}_s - U \left(J_0 \cdot \partial_t \bar{F}_s \right) \right\} \tag{M.5}$$

Then, using the quasi-neutrality equation (2.13), the first term in eq.(M.5) can be expressed as

$$\sum_{species} q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \partial_t \bar{F}_s \right) = -\sum_{species} q_s \int \mathcal{J}_\mathrm{x}\, \mathrm{d}\theta\, \mathrm{d}\varphi\, U\, \nabla_\perp \cdot \left(\frac{n_{s,\mathrm{eq}}}{B \Omega_s} \nabla_\perp \partial_t U \right)$$

Besides, using the general following equality (M.3) the previous equation is equivalent to

$$\sum_{species} q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \partial_t \bar{F}_s \right) = -\frac{1}{2} \sum_{species} q_s\, \partial_t \int \mathcal{J}_\mathrm{x}\, \mathrm{d}\theta\, \mathrm{d}\varphi\, U\, \nabla_\perp \cdot \left(\frac{n_{s,\mathrm{eq}}}{B \Omega_s} \nabla_\perp U \right)$$

Let us define the potential energy as $E_p \equiv \frac{1}{2} \sum_{species} q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \bar{F}_s \right)$ this term can be identified to the time derivative of potential energy,

$$\sum_{species} q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \partial_t \bar{F}_s \right) = \partial_t \left[\frac{1}{2} \sum_{species} q_s \int \mathrm{d}\tau^* U \left(J_0 \cdot \bar{F}_s \right) \right] \equiv \partial_t E_p$$

Let us consider the low wavenumber approximation of the gyroaverage operator used in the code, *i.e* the Padé approximation $J_0 \simeq 1 + \frac{1}{2} \nabla \cdot \left(\frac{m_s \mu_s}{q_s^2 B} \nabla_\perp \right)$ (see section 2.4.1 for more details) then for each species

$$\begin{aligned} \mathcal{I}_E &= q_s \int \mathrm{d}\tau^* \left\{ (J_0 \cdot U)\, \partial_t \bar{F}_s - U \left(J_0 \cdot \partial_t \bar{F}_s \right) \right\} \\ &= \frac{m_s}{2 q_s} \int \mathrm{d}\tau^* \left\{ \partial_t \bar{F}_s\, \nabla \cdot \left(\frac{\mu_s}{B} \nabla_\perp U \right) - U\, \nabla \cdot \left(\frac{\mu_s}{B} \nabla_\perp \partial_t \bar{F}_s \right) \right\} \end{aligned}$$

and using the result (M.3) of Appendix M.

$$\begin{aligned}\mathcal{I}_E &= \frac{m_s}{2q_s}\partial_\chi\left[\int \mathrm{d}\tau\left\{\partial_t\bar{F}_s\,\nabla\cdot\left(\frac{\mu_s}{B}\nabla_\perp U\right) - U\,\nabla\cdot\left(\frac{\mu_s}{B}\nabla_\perp\partial_t\bar{F}_s\right)\right\}\right] \quad \text{with } \mathrm{d}\tau = \mathrm{d}\chi\,\mathrm{d}\tau^* \\ &= \frac{m_s}{2q_s}\partial_\chi\left[-\int \mathcal{J}_\mathrm{x}\,\mathrm{d}\chi\,\mathrm{d}\theta\,\mathrm{d}\varphi\,\mathrm{d}^3\mathbf{v}\,\frac{\mu}{B}\nabla_\perp(\partial_t\bar{F}_s)\cdot\nabla_\perp U + \int \mathcal{J}_\mathrm{x}\,\mathrm{d}\theta\,\mathrm{d}\varphi\,\mathrm{d}^3\mathbf{v}\,\partial_t\bar{F}_s\,\frac{\mu}{B}\nabla\chi\cdot\nabla_\perp U\right] \\ &\quad + \frac{m_s}{2q_s}\partial_\chi\left[\int \mathcal{J}_\mathrm{x}\,\mathrm{d}\chi\,\mathrm{d}\theta\,\mathrm{d}\varphi\,\mathrm{d}^3\mathbf{v}\,\frac{\mu}{B}\nabla_\perp U\cdot\nabla_\perp(\partial_t\bar{F}_s) - \int \mathcal{J}_\mathrm{x}\,\mathrm{d}\theta\,\mathrm{d}\varphi\,\mathrm{d}^3\mathbf{v}\,U\frac{\mu}{B}\nabla\chi\cdot\nabla_\perp(\partial_t\bar{F}_s)\right]\end{aligned}$$

The first and third terms cancel each other out. The remaining terms

$$\mathcal{I}_E = \frac{m_s}{2q_s}\partial_\chi\left[\int \mathrm{d}\tau^*\,\partial_t\bar{F}_s\,\frac{\mu}{B}\nabla\chi\cdot\nabla_\perp U - \int \mathrm{d}\tau^*\,U\frac{\mu}{B}\nabla\chi\cdot\nabla_\perp(\partial_t\bar{F}_s)\right]$$

So finally, using the gyrocenter perpendicular stress $P_{s,\perp} = \int \mathrm{d}^3v\bar{F}_s\mu_s B$,

$$\sum_s q_s\int \mathrm{d}\tau^*\bar{U}\partial_t\bar{F}_s = \partial_t E_p + \frac{m_s}{2q_s}\partial_\chi\left[\int \mathrm{d}\tau^*\,\partial_t\bar{F}_s\,\frac{\mu}{B}\nabla\chi\cdot\nabla_\perp U - \int \mathrm{d}\tau^*\,U\frac{\mu}{B}\nabla\chi\cdot\nabla_\perp(\partial_t\bar{F}_s)\right]$$

M.3 Effect of the electric potential on the toroidal canonical momentum

The objective of this section is to prove that

$$\mathrm{d}tP_\varphi = -q_s\partial_\varphi\bar{U} \quad \text{with} \quad P_\varphi = -q_s\chi + \frac{m_s I}{B}v_{G\|}$$

Due to the fact that P_φ is an invariant of the equilibrium motion, this result is equivalent to proving that

$$\left[q_s\bar{U}, P_\varphi\right]_\mathrm{GC} = q_s\partial_\varphi\bar{U} \tag{M.6}$$

where $[\cdot,\cdot]_\mathrm{GC}$ indicate the Poisson brackets in the gyro-center coordinates. In the gyrokinetic framework, we recall the expression of the Poisson brackets for two given fields X and Y

$$B^*_{\|s}\left[X,Y\right]_\mathrm{GC} = \frac{\mathbf{B}^*_s}{m_s}\cdot\left(\boldsymbol{\nabla}X\partial_{v_{G\|}}Y - \partial_{v_{G\|}}X\boldsymbol{\nabla}Y\right) - \frac{\mathbf{b}}{q_s}\cdot\left(\boldsymbol{\nabla}X\times\boldsymbol{\nabla}Y\right)$$

where $\mathbf{b} = \mathbf{B}/B$ is the unit vector parallel to the magnetic field. The quantities $\mathbf{B}^*_s$ and $B^*_{\|s}$ are respectively defined by eq.(1.39) and $B^*_{\|s} = \mathbf{B}^*_s\cdot\mathbf{B}/B$, *i.e* $\mathbf{B}^*_s \equiv \mathbf{B} + \frac{m_s}{q_s}v_{G\|}\boldsymbol{\nabla}\times\mathbf{b}$ and $B^*_{\|s} \equiv B + \frac{m_s}{q_s}v_{G\|}\mathbf{b}\cdot(\boldsymbol{\nabla}\times\mathbf{b})$.

$$-q_sB^*_{\|s}\left[\bar{U}, P_\varphi\right]_\mathrm{GC} = -q_s\boldsymbol{\nabla}\bar{U}\cdot\left(\frac{I}{B}\mathbf{B} - \mathbf{b}\times\boldsymbol{\nabla}\chi + \frac{m_s v_{G\|}}{q_s}\left[\frac{I}{B}\left(\boldsymbol{\nabla}\times\mathbf{b}\right) - \boldsymbol{\nabla}\left(\frac{I}{B}\right)\times\mathbf{b}\right]\right) \tag{M.7}$$

In the chosen coordinate system, we recall that the magnetic field $\mathbf{B}$ can be written as $\mathbf{B} = I(\chi)\nabla\varphi + \nabla\varphi\times\nabla\chi$ with I a flux function, then

$$\mathbf{B}\times\nabla\chi = I\mathbf{B} - I^2\nabla\varphi = I\mathbf{B} - B^2R^2\nabla\varphi \tag{M.8}$$

$$\nabla\times\mathbf{B} = -(\partial_\chi I)\mathbf{B} - R^2\partial_\chi P\nabla\varphi \tag{M.9}$$

Besides,

$$\begin{aligned}\boldsymbol{\nabla}\left(\frac{I}{B}\right)\times\mathbf{b} &= -\frac{I}{B^2}\left(\boldsymbol{\nabla}B\times\mathbf{b}\right) + \frac{1}{B}\left(\boldsymbol{\nabla}I\times\mathbf{b}\right) = -\frac{I}{B^2}\left(\boldsymbol{\nabla}B\times\mathbf{b}\right) + \frac{1}{B}\left(\partial_\chi I\,\nabla\chi\right)\times\mathbf{b} \\ &= \frac{I}{B^2}\left(\mathbf{b}\times\boldsymbol{\nabla}B\right) - \frac{1}{B}\left(\partial_\chi I\right)\mathbf{b}\times\nabla\chi\end{aligned} \tag{M.10}$$

Then according to (M.8) and (M.10), equation (M.7) becomes

$$-q_s B^*_{\|s}\left[\bar{U}, P_\varphi\right]_{\mathrm{GC}} = -q_s \boldsymbol{\nabla}\bar{U}\cdot\left[R^2 B\nabla\varphi + \frac{m_s v_{G\|}}{q_s}\left(\frac{I}{B}\left(\boldsymbol{\nabla}\times\mathbf{b}\right) - \frac{I}{B^2}\left(\mathbf{b}\times\boldsymbol{\nabla}B\right) + \frac{1}{B}\left(\partial_\chi I\right)\mathbf{b}\times\nabla\chi\right)\right]$$

Using the fact that $\nabla\times\mathbf{b} = \mathbf{b}\times\frac{\nabla B}{B} + \frac{1}{B}\nabla\times\mathbf{B}$, the previous equation reads

$$-q_s B^*_{\|s}\left[\bar{U}, P_\varphi\right]_{\mathrm{GC}} = -q_s \boldsymbol{\nabla}\bar{U}\cdot\left(R^2 B\nabla\varphi + \frac{m_s v_{G\|}}{q_s}\left[\frac{I}{B^2}\nabla\times\mathbf{B} + \frac{1}{B}\left(\partial_\chi I\right)\mathbf{b}\times\nabla\chi\right]\right) \tag{M.11}$$

Therefore,

$$\frac{I}{B^2}\nabla\times\mathbf{B} + \frac{1}{B}\left(\partial_\chi I\right)\mathbf{b}\times\nabla\chi = -R^2\nabla\varphi\left(\frac{I}{B^2}\partial_\chi P + \partial_\chi I\right)$$

So, using the fact that the volume element in guiding-center velocity space $B^*_{\|s}$ can be expressed as

$$B^*_{\|s} = B - \frac{m_s}{q_s}v_{G\|}\left(\partial_\chi I + \frac{I}{B^2}\partial_\chi P\right) \tag{M.12}$$

equation (M.11) becomes

$$-q_s B^*_{\|s}\left[\bar{U}, P_\varphi\right]_{\mathrm{GC}} = -q_s\boldsymbol{\nabla}\bar{U}\cdot\left(R^2 B^*_{\|s}\nabla\varphi\right) = -q_s\partial_\varphi\bar{U}\nabla\varphi\cdot\nabla\varphi R^2 B^*_{\|s} = -q_s B^*_{\|s}\partial_\varphi\bar{U}$$

which is equivalent to equation (M.6).

Appendix N

Diagnostics for local conservation laws in GYSELA

As shown in chapter 4, the local conservation (*i.e* with a radial dependency) of density, momentum and energy are parts of the validation of the global algorithm implemented in GYSELA. They are checked for each simulations. Good accuracy is a necessary (but not sufficient) proof for validation of the physical results. Main terms required for these local conservation laws are directly computed in the code and saved in HDF5 output files (for details see appendix O). In this appendix, we describe the exact equations which have been implemented as post-processing of the code via Python diagnostics. Let us remind that for the following the hat symbols refer to normalized quantities (see appendix K for more details on GYSELA normalization). Most of the quantities saved in the code are averaged on the magnetic flux surfaces. Let us define this average for all function f as

$$\langle\, f \,\rangle_{\mathrm{FS}} \equiv \frac{1}{V'} \int f \mathcal{J}_{\mathrm{x}} \,\mathrm{d}\theta \,\mathrm{d}\varphi \quad \text{where} \quad V' = \int \mathcal{J}_{\mathrm{x}} \,\mathrm{d}\theta \,\mathrm{d}\varphi$$

then in normalized coordinates $\langle\, f \,\rangle_{\mathrm{FS}} \equiv \frac{1}{\hat{V}'} \int f \hat{\mathcal{J}}_{\mathrm{x}} \,\mathrm{d}\theta \,\mathrm{d}\varphi$ with $\hat{V}' = \int \hat{\mathcal{J}}_{\mathrm{x}} \,\mathrm{d}\theta \,\mathrm{d}\varphi$ and $\hat{\mathcal{J}}_{\mathrm{x}}$ the normalized jacobian. For the following, we remind that,

$$\bar{F}_s = \frac{n_0}{v_{T_{s0}}^3} \hat{\bar{F}}_s \quad ; \quad U = \frac{T_0}{eZ_0} \hat{\phi} \quad ; \quad \int \mathcal{J}_{\mathrm{v}} \,\mathrm{d}v_{G\|} \,\mathrm{d}\mu_s = v_{T_{s0}}^3 \int \hat{\mathcal{J}}_{\mathrm{v}} \,\mathrm{d}\hat{v}_{G\|s} \,\mathrm{d}\hat{\mu}_s \tag{N.1}$$

and that the label of flux surfaces χ satisfy [1] $\chi = \rho_0^2 B_0 \hat{\chi}$ i.e

$$\partial_\chi = \frac{1}{\rho_0^2 B_0} \partial_{\hat{\chi}} \quad \text{with} \quad \partial_{\hat{\chi}} = \frac{q}{\hat{r}} \partial_{\hat{r}} \quad \text{and} \quad \nabla \chi = \rho_0 B_0 \hat{\nabla} \hat{\chi} \quad \text{and} \quad \hat{\nabla} \hat{\chi} = \frac{\hat{r}}{q} \hat{\nabla} \hat{r} \tag{N.2}$$

Therefore,

$$\mathbf{v}_{E\times B_s} \cdot \nabla \chi = \rho_0 B_0 v_{T_0} \left(\hat{\mathbf{v}}_{E\times B_s} \cdot \hat{\nabla} \hat{\chi} \right) \quad \text{with} \quad \hat{\mathbf{v}}_{E\times B_s} \cdot \hat{\nabla} \hat{\chi} = \frac{1}{\hat{B}^*_{\|s}} \left[\hat{\bar{\phi}}, \hat{\chi} \right] \tag{N.3}$$

$$\mathbf{v}_{D_s} \cdot \nabla \chi = \rho_0 B_0 v_{T_0} \left(\hat{\mathbf{v}}_{D_s} \cdot \hat{\nabla} \hat{\chi} \right) \quad \text{with} \quad \hat{\mathbf{v}}_{D_s} \cdot \hat{\nabla} \hat{\chi} = \left(\frac{\hat{v}_{G\|s}^2 + \hat{\mu}_s \hat{B}}{\hat{Z}_s \hat{B}^*_{\|s} \hat{B}} \right) \left[\hat{B}, \hat{\chi} \right] \tag{N.4}$$

$$\mathcal{E}_s = T_0 \hat{\mathcal{E}}_s \quad \text{with} \quad \hat{\mathcal{E}}_s = \frac{1}{2} \hat{v}_{G\|s}^2 + \hat{\mu}_s \hat{B} \tag{N.5}$$

[1] The label of flux surfaces χ is defined as

$$\chi = B_0 \int_0^r \frac{r'}{q(r')} \,\mathrm{d}r' = B_0 \int_0^{\hat{r}} \frac{\rho_0 \hat{r}'}{q(\hat{r}')} \rho_0 \,\mathrm{d}\hat{r}' = B_0 \rho_0^2 \hat{\chi} \quad \text{with } \hat{\chi} = \int_0^{\hat{r}} \frac{\hat{r}'}{q(\hat{r}')} \,\mathrm{d}\hat{r}'$$

which gives

$$\frac{\mathrm{d}\chi}{\mathrm{d}r} = B_0 \frac{r}{q} \quad \text{therefore} \quad \frac{\partial}{\partial \chi} = \frac{\partial r}{\partial \chi} \frac{\partial}{\partial r} = \frac{q}{r B_0} \frac{\partial}{\partial r} = \frac{1}{\rho_0^2 B_0} \frac{\partial}{\partial \hat{\chi}} \quad \text{with} \quad \frac{\partial}{\partial \hat{\chi}} = \frac{q}{\hat{r}} \frac{\partial}{\partial \hat{r}}$$

and

$$\nabla \chi = \frac{\mathrm{d}\chi}{\mathrm{d}r} \nabla r = \frac{B_0 r}{q} \nabla r = B_0 \rho_0 \hat{\nabla} \hat{\chi} \quad \text{with} \quad \hat{\nabla} \hat{\chi} = \frac{\hat{r}}{q} \hat{\nabla} \hat{r}$$

N.1 Local charge density conservation

Then, local charge density conservation (4.5) can be expressed with the flux surface average of the guiding-center density $\langle\, n_{G_s} \,\rangle_{\mathrm{FS}}$ and the flux surface average of the turbulent (resp. neoclassical) flux of the guiding-centers $\langle\, \Gamma_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}}$ (resp. $\langle\, \Gamma_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}}$) as

$$\sum_s q_s \left[V' \partial_t \langle\, n_{G_s} \,\rangle_{\mathrm{FS}} + \partial_\chi \left(V' \left\{ \langle\, \Gamma^\chi_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}} + \langle\, \Gamma^\chi_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}} \right\} \right) \right] = 0$$

where

$$\begin{aligned}
\langle\, n_{G_s} \,\rangle_{\mathrm{FS}} &= \langle \int \mathrm{d}^3\mathbf{v}\, \bar{F}_s \,\rangle_{\mathrm{FS}} \quad \text{with} \quad \mathrm{d}^3\mathbf{v} = \mathcal{J}_{\mathrm{v}}\, \mathrm{d}v_{G\|}\, \mathrm{d}\mu_s \\
\langle\, \Gamma^\chi_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}} &= \langle \int \mathrm{d}^3\mathbf{v}\, \left(\mathbf{v}_{E\times B_s} \cdot \nabla\chi \right) \bar{F}_s \,\rangle_{\mathrm{FS}} \\
\langle\, \Gamma^\chi_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}} &= \langle \int \mathrm{d}^3\mathbf{v}\, \left(\mathbf{v}_{D_s} \cdot \nabla\chi \right) \bar{F}_s \,\rangle_{\mathrm{FS}}
\end{aligned}$$

According to Eqs. (N.1)-(N.4), the previous equation leads to the normalized local density conservation

$$\sum_s \hat{Z}_s \left[\frac{\partial}{\partial \hat{t}} \langle\, \hat{n}_{G_s} \,\rangle_{\mathrm{FS}} + \frac{1}{\hat{V}'} \frac{q}{\hat{r}} \frac{\partial}{\partial \hat{r}} \left(\hat{V}' \frac{\hat{r}}{q} \left\{ \langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}} + \langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}} \right\} \right) \right] = 0 \tag{N.6}$$

with $\int \cdot\, \mathrm{d}^3\hat{\mathbf{v}} = \int \cdot\, \hat{\mathcal{J}}_{\mathrm{v}}\, \mathrm{d}\hat{v}_{G\|s}\, \mathrm{d}\hat{\mu}_s$ and

$$\langle\, \hat{n}_{G_s} \,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}}\, \hat{\bar{F}}_s \,\rangle_{\mathrm{FS}} \tag{N.7}$$

$$\langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}}\, \left(\hat{\mathbf{v}}_{E\times B_s} \cdot \hat{\nabla}\hat{r} \right) \hat{\bar{F}}_s \,\rangle_{\mathrm{FS}} \tag{N.8}$$

$$\langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}}\, \left(\hat{\mathbf{v}}_{D_s} \cdot \hat{\nabla}\hat{r} \right) \hat{\bar{F}}_s \,\rangle_{\mathrm{FS}} \tag{N.9}$$

where the normalization of the guiding-center density reads $\langle\, n_{G_s} \,\rangle_{\mathrm{FS}} = n_0 \langle\, \hat{n}_{G_s} \,\rangle_{\mathrm{FS}}$ while the guiding-center turbulent and neoclassical fluxes respectively satisfy $\langle\, \Gamma^r_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}} = n_0 B_0 \rho_0 v_{T_0} \langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}}$ and $\langle\, \Gamma^r_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}} = n_0 B_0 \rho_0 v_{T_0} \langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}}$.

Code comment: In the code, $\langle\, \hat{n}_{G_s} \,\rangle_{\mathrm{FS}} \equiv$ `densGC_FSavg`, $\langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}} \equiv$ `GammaGC_turb_r_FSavg` and $\langle\, \hat{\Gamma}^r_{\mathrm{GC}_s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}} \equiv$ `GammaGC_neo_r_FSavg`.

N.2 Local energy conservation

The local energy conservation (4.16) is expressed in function of the magnetic flux surface averages as

$$\begin{aligned}
\sum_s \Big[& V' \partial_t \left(\langle\, E_{K_s} \,\rangle_{\mathrm{FS}} + \langle\, E_{p_s} \,\rangle_{\mathrm{FS}} \right) + \partial_\chi \Big(V' \langle\, Q^\chi_{s,\,\mathrm{turb}} \,\rangle_{\mathrm{FS}} \Big) + \\
& \partial_\chi \Big(V' \langle\, Q^\chi_{s,\,\mathrm{neo}} \,\rangle_{\mathrm{FS}} \Big) + \partial_\chi \Big(V' \langle\, Q^\chi_{s,\,\mathrm{pot}} \,\rangle_{\mathrm{FS}} \Big) + \partial_\chi \Big(V' \langle\, Q^\chi_{s,\,\mathrm{pol}} \,\rangle_{\mathrm{FS}} \Big) \Big] = 0
\end{aligned}$$

with the flux surface average of kinetic and potential energies defined as

$$\begin{aligned}
\langle\, E_{K_s} \,\rangle_{\mathrm{FS}} &= \langle \int \mathrm{d}^3\mathbf{v}\, \mathcal{E}_s\, \bar{F}_s \,\rangle_{\mathrm{FS}} \quad \text{where} \quad \mathcal{E}_s = \frac{1}{2} m_s v_{G\|}^2 + \mu_s B \\
\langle\, E_{p_s} \,\rangle_{\mathrm{FS}} &= q_s \langle\, \frac{1}{2} \int \mathrm{d}^3\mathbf{v}\, U \left(J_0 \cdot \bar{F}_s \right) \,\rangle_{\mathrm{FS}} = \frac{q_s}{2} \langle\, U \int \mathrm{d}^3\mathbf{v}\, \left(J_0 \cdot \bar{F}_s \right) \,\rangle_{\mathrm{FS}}
\end{aligned}$$

and the different energy fluxes averaged on the flux surfaces given by

$$\langle\, Q^{\chi}_{s,\,\mathrm{turb}}\,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\mathbf{v}\, \mathcal{E}_s\, v^{\chi}_{E\times B_s}\, \bar{F}_s\,\rangle_{\mathrm{FS}}$$

$$\langle\, Q^{\chi}_{s,\,\mathrm{neo}}\,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\mathbf{v}\, \mathcal{E}_s\, v^{\chi}_{D_s}\, \bar{F}_s\,\rangle_{\mathrm{FS}}$$

$$\langle\, Q^{\chi}_{s,\,\mathrm{pot}}\,\rangle_{\mathrm{FS}} = q_s\langle \int \mathrm{d}^3\mathbf{v}\, (J_{0_s}\cdot U)\, \left(v^{\chi}_{E\times B_s} + v^{\chi}_{D_s}\right)\, \bar{F}_s\,\rangle_{\mathrm{FS}}$$

$$\langle\, Q^{\chi}_{s,\,\mathrm{pol}}\,\rangle_{\mathrm{FS}} = \frac{m_s}{2q_s}\langle\, \frac{1}{B^2}\,(\partial_t P_{s,\,\perp}\nabla U\cdot\nabla\chi - U\nabla(\partial_t P_{s,\,\perp})\cdot\nabla\chi)\,\rangle_{\mathrm{FS}}$$

where the integral in velocity space is defined as $\int\cdot\,\mathrm{d}^3\mathbf{v} = \int\cdot\,\mathcal{J}_{\mathrm{v}}\,\mathrm{d}v_{G\|}\,\mathrm{d}\mu_s$ and the contravariant components of the velocities are $v^{\chi}_{E\times B_s} = \mathbf{v}_{E\times B_s}\cdot\nabla\chi$ and $v^{\chi}_{D_s} = \mathbf{v}_{D_s}\cdot\nabla\chi$. The perpendicular stress $P_{s,\,\perp}$ which is equivalent to the perpendicular pressure $p_{s,\,\perp}$ is defined as $P_{s,\,\perp} = p_{s,\,\perp} = \int \mathrm{d}^3\mathbf{v}\bar{F}_s\mu_s B$. For the following we use the perpendicular pressure notation. Then, the normalized local energy conservation reads

$$\sum_s \Bigg[\hat{V}'\frac{\partial}{\partial\hat{t}}\Big(\langle\,\hat{E}_{K_s}\,\rangle_{\mathrm{FS}} + \langle\,\hat{E}_{p_s}\,\rangle_{\mathrm{FS}}\Big) + \frac{q}{\hat{r}}\frac{\partial}{\partial\hat{r}}\Big(\hat{V}'\,\frac{\hat{r}}{q}\Big\{\langle\,\hat{Q}^{r}_{s,\,\mathrm{turb}}\,\rangle_{\mathrm{FS}} + \langle\,\hat{Q}^{r}_{s,\,\mathrm{neo}}\,\rangle_{\mathrm{FS}} + \langle\,\hat{Q}^{r}_{s,\,\mathrm{pot}}\,\rangle_{\mathrm{FS}} + \langle\,\hat{Q}^{r}_{s,\,\mathrm{pol}}\,\rangle_{\mathrm{FS}}\Big\}\Big)\Bigg] = 0 \tag{N.10}$$

with

$$\langle\,\hat{E}_{K_s}\,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\hat{\mathcal{E}}_s\,\hat{\bar{F}}_s\,\rangle_{\mathrm{FS}} \quad ; \quad \langle\,\hat{E}_{p_s}\,\rangle_{\mathrm{FS}} = \hat{Z}_s\langle\,\frac{1}{2}\hat{\phi}\int \mathrm{d}^3\hat{\mathbf{v}}\,(J_0\cdot\hat{\bar{F}}_s)\,\rangle_{\mathrm{FS}} \tag{N.11}$$

$$\langle\,\hat{Q}^{r}_{s,\,\mathrm{turb}}\,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\hat{\mathcal{E}}_s\,\left(\hat{\mathbf{v}}_{E\times B_s}\cdot\hat{\nabla}\hat{r}\right)\,\hat{\bar{F}}_s\,\rangle_{\mathrm{FS}} \tag{N.12}$$

$$\langle\,\hat{Q}^{r}_{s,\,\mathrm{neo}}\,\rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\hat{\mathcal{E}}_s\,\left(\hat{\mathbf{v}}_{D_s}\cdot\hat{\nabla}\hat{r}\right)\,\hat{\bar{F}}_s\,\rangle_{\mathrm{FS}} \tag{N.13}$$

$$\langle\,\hat{Q}^{r}_{s,\,\mathrm{pot}}\,\rangle_{\mathrm{FS}} = \hat{Z}_s\langle \int \mathrm{d}^3\hat{\mathbf{v}}\,(J_0\cdot\hat{\phi})\,\left[\left(\hat{\mathbf{v}}_{E\times B_s}\cdot\hat{\nabla}\hat{r}\right) + \left(\hat{\mathbf{v}}_{D_s}\cdot\hat{\nabla}\hat{r}\right)\right]\,\hat{\bar{F}}_s\,\rangle_{\mathrm{FS}} \tag{N.14}$$

$$\langle\,\hat{Q}^{r}_{s,\,\mathrm{pol}}\,\rangle_{\mathrm{FS}} = \frac{1}{2}\frac{\hat{A}_s}{\hat{Z}_s}\langle\,\frac{1}{\hat{B}^2}\left(\frac{\partial\hat{p}_{s,\,\perp}}{\partial\hat{t}}\hat{\nabla}\hat{\phi}\cdot\hat{\nabla}\hat{r} - \hat{\phi}\hat{\nabla}\left(\frac{\partial\hat{p}_{s,\,\perp}}{\partial\hat{t}}\right)\cdot\hat{\nabla}\hat{r}\right)\,\rangle_{\mathrm{FS}} \tag{N.15}$$

where $\mathcal{E}_s = T_0\hat{\mathcal{E}}_s$, $\langle\, E_{K_s}\,\rangle_{\mathrm{FS}} = n_0T_0\langle\,\hat{E}_{K_s}\,\rangle_{\mathrm{FS}}$, $\langle\, E_{p_s}\,\rangle_{\mathrm{FS}} = n_0T_0\langle\,\hat{E}_{p_s}\,\rangle_{\mathrm{FS}}$ and where all the energy fluxes are normalized as [2] $\langle\, Q_s\,\rangle_{\mathrm{FS}} = n_0T_0v_{T_0}B_0\rho_0\langle\,\hat{Q}_s\,\rangle_{\mathrm{FS}}$. The normalized perpendicular pressure is given

[2] Let us take the example of the flux surface average of $Q^{\chi}_{s,\,\mathrm{pol}}$ which is given by

$$\begin{aligned}
\langle\, Q^{\chi}_{s,\,\mathrm{pol}}\,\rangle_{\mathrm{FS}} &= \frac{m_s}{2q_s}\langle\,\frac{1}{B^2}\,(\partial_t p_{s,\,\perp}\nabla U\cdot\nabla\chi - U\nabla(\partial_t p_{s,\,\perp}\cdot\nabla\chi)\,\rangle_{\mathrm{FS}} \\
&= \frac{\hat{A}_s m_0}{2Z_0\,e\hat{Z}_s}\Bigg[\langle\,\frac{1}{B_0^2\hat{B}^2}\left(\Omega_{C_0}\frac{\partial(n_0T_0\hat{p}_{s,\,\perp})}{\partial\hat{t}}\frac{1}{\rho_0}\hat{\nabla}\left(\frac{T_0}{eZ_0}\hat{\phi}\right)\cdot\left(\rho_0B_0\hat{\nabla}\hat{\chi}\right)\right)\,\rangle_{\mathrm{FS}} \\
&\qquad - \langle\,\frac{1}{B_0^2\hat{B}^2}\left(\frac{T_0}{eZ_0}\hat{\phi}\frac{1}{\rho_0}\hat{\nabla}\left(\Omega_{C_0}\frac{\partial(n_0T_0\hat{p}_{s,\,\perp})}{\partial\hat{t}}\cdot\left(\rho_0B_0\hat{\nabla}\hat{\chi}\right)\right)\right)\,\rangle_{\mathrm{FS}}\Bigg] \\
&= K_{Q,\,\mathrm{norm}}\frac{\hat{A}_s}{2\hat{Z}_s}\left[\langle\,\frac{1}{\hat{B}^2}\left(\frac{\partial\hat{p}_{s,\,\perp}}{\partial\hat{t}}\hat{\nabla}\hat{\phi}\cdot\hat{\nabla}\hat{\chi} - \hat{\phi}\hat{\nabla}\left(\frac{\partial\hat{p}_{s,\,\perp}}{\partial\hat{t}}\cdot\hat{\nabla}\hat{\chi}\right)\right)\,\rangle_{\mathrm{FS}}\right]
\end{aligned}$$

where due to the relations $v_{T_0} = \rho_0\Omega_{C_0}$ and $\Omega_{C_0} = Z_0\,e\,B_0/m_0$, the normalization coefficient reads

$$K_{Q,\,\mathrm{norm}} = \frac{m_0}{Z_0\,e}\frac{1}{B_0^2}\Omega_{C_0}\frac{n_0T_0}{\rho_0}\frac{T_0}{Z_0\,e}\rho_0B_0 = n_0T_0\frac{1}{\Omega_{C_0}^2}v_{T_0}^2\frac{\Omega_{C_0}}{\rho_0}\rho_0B_0 = n_0T_0v_{T_0}\rho_0B_0$$

Then, $\langle\, Q^{\chi}_{s,\,\mathrm{pol}}\,\rangle_{\mathrm{FS}} = n_0T_0v_{T_0}B_0\rho_0\langle\,\hat{Q}^{\chi}_{s,\,\mathrm{pol}}\,\rangle_{\mathrm{FS}}$ with

$$\langle\,\hat{Q}^{\chi}_{s,\,\mathrm{pol}}\,\rangle_{\mathrm{FS}} = \frac{\hat{A}_s}{2\hat{Z}_s}\left[\langle\,\frac{1}{\hat{B}^2}\left(\frac{\partial\hat{p}_{s,\,\perp}}{\partial\hat{t}}\hat{\nabla}\hat{\phi}\cdot\hat{\nabla}\hat{\chi} - \hat{\phi}\hat{\nabla}\left(\frac{\partial\hat{p}_{s,\,\perp}}{\partial\hat{t}}\cdot\hat{\nabla}\hat{\chi}\right)\right)\,\rangle_{\mathrm{FS}}\right]$$

by $\hat{p}_{s,\perp} = \int \mathrm{d}^3\hat{\mathbf{v}}\hat{\bar{F}}_s\hat{\mu}_s\hat{B}$ such that $p_{s,\perp} = n_0 T_0 \hat{p}_{s,\perp}$.

Code comment: In the code, the energies are labelled as $\langle \hat{E}_{K_s} \rangle_{\mathrm{FS}} \equiv$ `Enkin_FSavg` and $\langle \hat{E}_{p_s} \rangle_{\mathrm{FS}} \equiv$ `Enpot_FSavg`. The energy fluxes are called $\langle \hat{Q}^r_{s,\mathrm{x}} \rangle_{\mathrm{FS}} \equiv$ `QGC_x_r_FSavg` for $\mathrm{x} \in \{\mathrm{turb}, \mathrm{neo}, \mathrm{pot}, \mathrm{pol}\}$ while the perpendicular pressure is defined as $\hat{p}_{s,\perp} \equiv$ `press_perp_FSavg` and $\partial_t \hat{p}_{s,\perp} \equiv$ `dpress_perp_dt_FSavg`. The term $\langle \hat{Q}^r_{s,\mathrm{pol}} \rangle_{\mathrm{FS}}$ requires the time derivative of the 3D perpendicular pressure $\hat{p}_{s,\perp}(r,\theta,\varphi)$. Therefore, to avoid to save this 3D field, the energy flux term due to polarization is computed in post-processing by using the 3D diagnostic savings.

Other normalized quantities as the pressure and the heat fluxes (not required for the energy conservation laws but interesting as physical quantities) are saved. The pressure terms are called $\langle \hat{p}_s \rangle_{\mathrm{FS}} \equiv$ `pressGC_FSavg` and $\langle \hat{p}_{s,\parallel} \rangle_{\mathrm{FS}} \equiv$ `pressGC_par_FSavg` where

$$\langle \hat{p}_s \rangle_{\mathrm{FS}} = \langle \hat{p}_{s,\parallel} \rangle_{\mathrm{FS}} + \langle \hat{p}_{s,\perp} \rangle_{\mathrm{FS}} \quad \text{with} \quad \text{and}$$
$$\langle \hat{p}_{s,\parallel} \rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}}\, \frac{1}{2}\left(\hat{v}_{G\parallel} - \hat{V}_\parallel\right)^2 \hat{\bar{F}}_s \rangle_{\mathrm{FS}}$$

The normalized parallel velocity $\hat{V}_\parallel$ is defined as

$$\hat{V}_\parallel(r,\theta,\varphi) = \frac{\int \mathrm{d}^3\hat{\mathbf{v}}\, \hat{v}_{G\parallel}\hat{\bar{F}}_s}{\int \mathrm{d}^3\hat{\mathbf{v}}\hat{\bar{F}}_s} \equiv \frac{\mathtt{nVparGC}}{\mathtt{nGC}} \equiv \mathtt{VparGC}$$

The normalized heat fluxes are called $\langle \hat{q}_{s,\mathrm{turb}} \rangle_{\mathrm{FS}} \equiv$ `HfluxGC_turb_r_FSavg` and $\langle \hat{q}_{s,\mathrm{neo}} \rangle_{\mathrm{FS}} \equiv$ `HfluxGC_neo_r_FSavg` where

$$\langle \hat{q}_{s,\mathrm{turb}} \rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}} \left[\frac{1}{2}\left(\hat{v}_{G\parallel} - \hat{V}_\parallel\right)^2 + \hat{\mu}\hat{B}\right]\left(\hat{\mathbf{v}}_{E\times B_s}\cdot\hat{\nabla}\hat{r}\right)\hat{\bar{F}}_s \rangle_{\mathrm{FS}}$$
$$\langle \hat{q}_{s,\mathrm{neo}} \rangle_{\mathrm{FS}} = \langle \int \mathrm{d}^3\hat{\mathbf{v}} \left[\frac{1}{2}\left(\hat{v}_{G\parallel} - \hat{V}_\parallel\right)^2 + \hat{\mu}\hat{B}\right]\left(\hat{\mathbf{v}}_{D_s}\cdot\hat{\nabla}\hat{r}\right)\hat{\bar{F}}_s \rangle_{\mathrm{FS}}$$

N.3 Local conservation of the toroidal momentum

In the code, the local conservation of the toroidal momentum (4.25) is expressed as

$$\sum_s \left[\partial_t \langle \mathcal{L}_{\varphi,s} \rangle_{\mathrm{FS}} + \frac{1}{V'}\partial_\chi\left(V'\left\{\langle \Pi^\chi_{\varphi\, s,\mathrm{turb}} \rangle_{\mathrm{FS}} + \langle \Pi^\chi_{\varphi\, s,\mathrm{neo}} \rangle_{\mathrm{FS}}\right\}\right) + \langle T^\chi_{s,\mathrm{pol}} \rangle_{\mathrm{FS}}\right]$$
$$= \sum_s \left[\langle J^\chi_{s,\mathrm{turb}} \rangle_{\mathrm{FS}} + \langle J^\chi_{s,\mathrm{neo}} \rangle_{\mathrm{FS}}\right]$$

with $u_\varphi = (I/B)v_{G\parallel} = R^2(\mathbf{b}\cdot\nabla\varphi)v_{G\parallel}$ and

$$\langle \mathcal{L}_{\varphi,s} \rangle_{\mathrm{FS}} = m_s \langle \int \mathrm{d}^3\mathbf{v}\, u_\varphi \bar{F}_s \rangle_{\mathrm{FS}}$$
$$\langle \Pi^\chi_{\varphi\, s,\mathrm{turb}} \rangle_{\mathrm{FS}} = m_s \langle \int \mathrm{d}^3\mathbf{v}\, u_\varphi\, v^\chi_{E\times B_s}\, \bar{F}_s \rangle_{\mathrm{FS}}$$
$$\langle \Pi^\chi_{\varphi\, s,\mathrm{neo}} \rangle_{\mathrm{FS}} = m_s \langle \int \mathrm{d}^3\mathbf{v}\, u_\varphi\, v^\chi_{D_s}\, \bar{F}_s \rangle_{\mathrm{FS}}$$
$$\langle T^\chi_{s,\mathrm{pol}} \rangle_{\mathrm{FS}} = q_s \langle \int \mathrm{d}^3\mathbf{v}\, \bar{F}_s\, \partial_\varphi(J_0\cdot U) \rangle_{\mathrm{FS}}$$
$$\langle J^\chi_{s,\mathrm{turb}} \rangle_{\mathrm{FS}} = q_s \langle \int \mathrm{d}^3\mathbf{v}\, v^\chi_{E\times B_s}\, \bar{F}_s \rangle_{\mathrm{FS}} \quad \text{and} \quad \langle J^\chi_{s,\mathrm{neo}} \rangle_{\mathrm{FS}} = q_s \langle \int \mathrm{d}^3\mathbf{v}\, v^\chi_{D_s}\, \bar{F}_s \rangle_{\mathrm{FS}}$$

Then, using the following equivalences with the normalized quantities for the toroidal momentum [3] $\langle\, \mathcal{L}_{\varphi,\,s}\,\rangle_{\rm FS} = n_0 m_0 \rho_0 v_{T_{s0}} \langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS}$, for the polarization flux of momentum term $\langle\, T^{\chi}_{s,\,\rm pol}\,\rangle_{\rm FS} = n_0 T_0 \langle\, \hat{T}^{\chi}_{s,\,\rm pol}\,\rangle_{\rm FS}$, for the Reynold stress [4] $\langle\, \Pi^{\chi}_{\varphi_s}\,\rangle_{\rm FS} = n_0 m_0 v_{T_0} v_{T_{s0}} \rho_0^2 B_0 \langle\, \hat{\Pi}^{\chi}_{\varphi_s}\,\rangle_{\rm FS}$, and finally for the radial current [5] $\langle\, J^{\chi}_s\,\rangle_{\rm FS} = n_0 Z_0\, e\, v_{T_0} (\rho_0 B_0) \langle\, \hat{J}^{\chi}_s\,\rangle_{\rm FS}$, the normalized equation for the local conservation of the toroidal momentum reads [6] :

$$\sum_s \left[\frac{\partial}{\partial \hat{t}} \langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS} + \frac{1}{\hat{V}'} \frac{q}{\hat{r}} \frac{\partial}{\partial \hat{r}} \left(\frac{\hat{r}}{q} \hat{V}' \Big\{ \langle\, \hat{\Pi}^{r}_{\varphi\, s,\,\rm turb}\,\rangle_{\rm FS} + \langle\, \hat{\Pi}^{r}_{\varphi\, s,\,\rm neo}\,\rangle_{\rm FS} \Big\} \right) + \sqrt{\hat{A}_s} \langle\, \hat{T}^{\chi}_{s,\,\rm pol}\,\rangle_{\rm FS} \right] = \sum_s \sqrt{\hat{A}_s} \frac{\hat{r}}{q} \left[\langle\, \hat{J}^{r}_{s,\,\rm turb}\,\rangle_{\rm FS} + \langle\, \hat{J}^{r}_{s,\,\rm neo}\,\rangle_{\rm FS} \right] \tag{N.16}$$

[3] Using the following relation,

$$u_\varphi = R^2 (\mathbf{b}\cdot\nabla\varphi) v_{G\|} = \rho_0^2 \hat{R}^2 \left(\hat{\mathbf{b}} \cdot \frac{1}{\rho_0} \hat{\nabla}\varphi \right) v_{T_{s0}} \hat{v}_{G\|s} = \rho_0 v_{T_{s0}} \hat{u}_\varphi \quad \text{with } \hat{u}_\varphi = \hat{R}^2 \left(\hat{\mathbf{b}} \cdot \hat{\nabla}\varphi \right) \hat{v}_{G\|s}$$

the flux surface average of the toroidal momentum can be expressed as

$$\begin{aligned} \langle\, \mathcal{L}_{\varphi,\,s}\,\rangle_{\rm FS} &= m_s \langle \int \mathrm{d}^3\mathbf{v}\, u_\varphi \bar{F}_s \,\rangle_{\rm FS} = \hat{A}_s m_0 n_0 \langle \int \mathrm{d}^3\hat{\mathbf{v}} (\rho_0 v_{T_{s0}} \hat{u}_\varphi) \hat{\bar{F}}_s \,\rangle_{\rm FS} \\ &= n_0 m_0 \rho_0 v_{T_{s0}} \langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS} \quad \text{with } \hat{\mathcal{L}}_{\varphi,\,s} = \hat{A}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}} \hat{u}_\varphi \hat{\bar{F}}_s \,\rangle_{\rm FS} \end{aligned}$$

[4] The Reynold stress can be expressed as

$$\begin{aligned} \langle\, \Pi^{\chi}_{\varphi_s}\,\rangle_{\rm FS} &= m_s \langle \int \mathrm{d}^3\mathbf{v}\, u_\varphi \left(v^{\chi}_{E\times B_s} + v^{\chi}_{D_s} \right) \bar{F}_s \,\rangle_{\rm FS} \\ &= \hat{A}_s m_0 n_0 \langle \int \mathrm{d}^3\hat{\mathbf{v}} \left(\rho_0 v_{T_{s0}} \hat{u}_\varphi \right) \left(\rho_0 B_0 v_{T_0} \right) \left(\hat{v}_{E\times B_s} \cdot \hat{\nabla}\hat{\chi} + \hat{v}_{D_s} \cdot \hat{\nabla}\hat{\chi} \right) \hat{\bar{F}}_s \,\rangle_{\rm FS} \end{aligned}$$

Therefore, $\langle\, \Pi^{\chi}_{\varphi_s}\,\rangle_{\rm FS} = n_0 m_0 \rho_0^2 B_0 v_{T_{s0}} v_{T_0} \langle\, \hat{\Pi}^{\chi}_{\varphi_s}\,\rangle_{\rm FS}$ with $\langle\, \hat{\Pi}^{\chi}_{\varphi_s}\,\rangle_{\rm FS} = \hat{A}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}} \hat{u}_\varphi (\hat{v}_G \cdot \hat{\nabla}\hat{\chi}) \hat{\bar{F}}_s \,\rangle_{\rm FS}$.

[5] The radial current can be expressed as

$$\begin{aligned} \langle\, J^{\chi}_s\,\rangle_{\rm FS} &= q_s \langle \int \mathrm{d}^3\mathbf{v} \left(v^{\chi}_{E\times B_s} + v^{\chi}_{D_s} \right) \bar{F}_s \,\rangle_{\rm FS} \\ &= Z_0\, e \hat{Z}_s n_0 \langle \int \mathrm{d}^3\hat{\mathbf{v}}\, \rho_0 B_0 v_{T_0} \left(\hat{v}_{E\times B_s} \cdot \hat{\nabla}\hat{\chi} + \hat{v}_{D_s} \cdot \hat{\nabla}\hat{\chi} \right) \hat{\bar{F}}_s \,\rangle_{\rm FS} \end{aligned}$$

Therefore, $\langle\, J^{\chi}_s\,\rangle_{\rm FS} = n_0 Z_0\, e\, v_{T_0} (\rho_0 B_0) \langle\, \hat{J}^{\chi}_s\,\rangle_{\rm FS}$ with $\langle\, \hat{J}^{\chi}_s\,\rangle_{\rm FS} = \hat{Z}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}} \left(\hat{v}_G \cdot \hat{\nabla}\hat{\chi} \right) \hat{\bar{F}}_s \,\rangle_{\rm FS}$.

[6] According to the equivalence $\langle\, \mathcal{L}_{\varphi,\,s}\,\rangle_{\rm FS} = n_0 m_0 \rho_0 v_{T_{s0}} \langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS}$, $\langle\, T^{\chi}_{s,\,\rm pol}\,\rangle_{\rm FS} = n_0 T_0 \langle\, \hat{T}^{\chi}_{s,\,\rm pol}\,\rangle_{\rm FS}$, $\langle\, \Pi^{\chi}_{\varphi_s}\,\rangle_{\rm FS} = n_0 m_0 v_{T_0} v_{T_{s0}} \rho_0^2 B_0 \langle\, \hat{\Pi}^{\chi}_{\varphi_s}\,\rangle_{\rm FS}$ and $\langle\, J^{\chi}_s\,\rangle_{\rm FS} = n_0 Z_0\, e\, v_{T_0} (\rho_0 B_0) \langle\, \hat{J}^{\chi}_s\,\rangle_{\rm FS}$, the local conservation of the toroidal momentum reads:

$$\begin{aligned} &\sum_s \Big[\Omega_{C_0} n_0 m_0 \rho_0 v_{T_{s0}} \partial_{\hat{t}} \langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS} + \frac{1}{\hat{V}'} \frac{1}{\rho_0^2 B_0} \partial_{\hat{\chi}} \left(\hat{V}' \Big\{ n_0 m_0 v_{T_0} v_{T_{s0}} \rho_0^2 B_0 \langle\, \hat{\Pi}^{\chi}_{\varphi_s}\,\rangle_{\rm FS} \Big\} \right) \\ &\qquad + n_0 T_0 \langle\, T^{\chi}_{s,\,\rm pol}\,\rangle_{\rm FS} \Big] = \sum_s n_0 Z_0\, e\, \rho_0 B_0 v_{T_0} \langle\, J^{\chi}_s\,\rangle_{\rm FS} \\ &\sum_s \left[\partial_{\hat{t}} \langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS} + \frac{1}{\hat{V}'} \partial_{\hat{\chi}} \left(\hat{V}' \Big\{ \langle\, \hat{\Pi}^{\chi}_{\varphi_s}\,\rangle_{\rm FS} \Big\} \right) + K_{T,\,\rm norm} \langle\, T^{\chi}_{s,\,\rm pol}\,\rangle_{\rm FS} \right] = \sum_s K_{J,\,\rm norm} \langle\, J^{\chi}_s\,\rangle_{\rm FS} \end{aligned}$$

with $K_{T,\,\rm norm} = T_0 / \left(m_0 v_{T_0} v_{T_{s0}} \right) = v_{T_0} / v_{T_{s0}} = \sqrt{\hat{A}_s}$ and $K_{J,\,\rm norm} = \frac{Z_0\, e\, \rho_0 B_0 v_{T_0}}{m_0 v_{T_0} v_{T_{s0}}} = \frac{\Omega_{C_0} \rho_0}{v_{T_{s0}}} = \sqrt{\hat{A}_s}$.

where

$$\langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS} = \hat{A}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\hat{u}_\varphi \hat{\bar{F}}_s\,\rangle_{\rm FS} \quad \text{with} \quad \hat{u}_\varphi = \left(\hat{R}^2\hat{\mathbf{b}}\cdot\hat{\nabla}\varphi\right)\hat{v}_{G\|s} \tag{N.17}$$

$$\langle\, \hat{\Pi}^r_{\varphi\,s,\,\rm turb}\,\rangle_{\rm FS} = \hat{A}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\hat{u}_\varphi\,\left(\hat{v}_{E\times B_s}\cdot\hat{\nabla}\hat{r}\right)\,\hat{\bar{F}}_s\,\rangle_{\rm FS} \tag{N.18}$$

$$\langle\, \hat{\Pi}^r_{\varphi\,s,\,\rm neo}\,\rangle_{\rm FS} = \hat{A}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\hat{u}_\varphi\,\left(\hat{v}_{D_s}\cdot\hat{\nabla}\hat{r}\right)\,\hat{\bar{F}}_s\,\rangle_{\rm FS} \tag{N.19}$$

$$\langle\, \hat{T}^\chi_{s,\,\rm pol}\,\rangle_{\rm FS} = \hat{Z}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\partial_\varphi(J_0\cdot\hat{\phi})\,\hat{\bar{F}}_s\,\rangle_{\rm FS} \tag{N.20}$$

$$\langle\, \hat{J}^r_{s,\,\rm turb}\,\rangle_{\rm FS} = \hat{Z}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\left(\hat{v}_{E\times B_s}\cdot\hat{\nabla}\hat{r}\right)\,\hat{\bar{F}}_s\,\rangle_{\rm FS} \quad \text{and} \tag{N.21}$$

$$\langle\, \hat{J}^r_{s,\,\rm neo}\,\rangle_{\rm FS} = \hat{Z}_s \langle \int \mathrm{d}^3\hat{\mathbf{v}}\,\left(\hat{v}_{D_s}\cdot\hat{\nabla}\hat{r}\right)\,\bar{F}_s\,\rangle_{\rm FS} \tag{N.22}$$

Let us notice that the radial currents are related to the guiding-center fluxes as $\langle\, \hat{J}^r_{s,\,\rm turb}\,\rangle_{\rm FS} = \hat{Z}_s\langle\, \hat{\Gamma}^r_{{\rm GC}_s,\,\rm turb}\,\rangle_{\rm FS}$ with $\langle\, \hat{\Gamma}^r_{{\rm GC}_s,\,\rm turb}\,\rangle_{\rm FS}$ defined by (N.8) and $\langle\, \hat{J}^r_{s,\,\rm neo}\,\rangle_{\rm FS} = \hat{Z}_s\langle\, \hat{\Gamma}^r_{{\rm GC}_s,\,\rm neo}\,\rangle_{\rm FS}$ with $\langle\, \hat{\Gamma}^r_{{\rm GC}_s,\,\rm neo}\,\rangle_{\rm FS}$ defined by (N.9).

Code comment: In the code, $\langle\, \hat{\mathcal{L}}_{\varphi,\,s}\,\rangle_{\rm FS} \equiv$ `Lphi_GC_FSavg` while the Reynold stresses are defined as $\langle\, \hat{\Pi}^r_{\varphi\,s,\,\rm turb}\,\rangle_{\rm FS} \equiv$ `RSphiGC_turb_r_FSavg` and $\langle\, \hat{\Pi}^r_{\varphi\,s,\,\rm neo}\,\rangle_{\rm FS} \equiv$ `RSphiGC_neo_r_FSavg`. $\langle\, \hat{T}^\chi_{s,\,\rm pol}\,\rangle_{\rm FS} \equiv$ `TpolGC_chi_FSavg`, $\langle \hat{J}^r_{s,\,\rm turb}\rangle_{\rm FS} \equiv \hat{Z}_s\times$ `GammaGC_turb_r_FSavg` and finally $\langle \hat{J}^r_{s,\,\rm neo}\rangle_{\rm FS} \equiv \hat{Z}_s\times$ `GammaGC_neo_r_FSavg`.

N.4 Input data files

This appendix contains the input data files of simulations presented in this report.

```
15/01/2015                A150_FD_allwcoll2_semitore                    1
#*****************************************
# job parameters
#*****************************************
NB_RESTART   = 5
TIME         = 10:00:00
JOBNAME      = A75_A1Z1

#*****************************************
#  Variables used to specify if the
#   simulation is a restart from
#   another one
#*****************************************
SEARCH_SP0   = 0            ! 0 if no restart, 1 otherwise
DIR_SP0      =              ! if SEARCH_SP0=1, name of the directory
                            !   containing the major species restart
                            files

#*****************************************
#  Variables for parallelization
#*****************************************
NSPECIES = 1
NPROC_R  = 2
NPROC_TH = 4
NPROC_MU = 16
NTHREAD  = 4

#*****************************************
#  Variables for radial profile input files
#*****************************************
ns0_sp0_filename = ns0_sp0_test.dat
ns0_sp1_filename = ns0_sp1_test.dat
Ts0_sp0_filename = Ts0_sp0_test.dat
Ts0_sp1_filename = Ts0_sp1_test.dat
q_filename       = safety_factor_test.dat

#*****************************************
# gysela input datas
#*****************************************
&PARALLEL
  Nspecies       = $NSPECIES
  Nbproc_r       = $NPROC_R
  Nbproc_theta   = $NPROC_TH
  Nbproc_mu      = $NPROC_MU
  bloc_phi       = $NTHREAD
  Nbthread       = $NTHREAD
  large_platform = .true.
  distrib3D      = .true.,
/

&MESH
  CylindricalGeometry = .false.
  Nr                  = 255
  Ntheta              = 256
  Nphi                = 128
  Nvpar               = 127
  Nmu                 = $((NPROC_MU-1))
  a                   = 150.
  rhomin              = 0.15
  rhomax              = 0.85
```

```
15/01/2015                A150_FD_allwcoll2_semitore                    2
  Ltheta              = 6.283185307179586476925286766559005768394
  Lphi                = 3.14159265358979323846264338327950288419 7
  aspect_ratio        = 3.3
  nb_vth0             = 7.
  Lmu                 = 12.
  mumin               = 0.,
/

&EQUIL
  canonical_profile          = 0
  tau0                       = 1.
  A0                         = 1.
  Z0                         = 1.
  As                         = 1.
  Zs                         = 1.
  species_frac               = 1.
  read_ns0                   = .false.
  read_Ts0                   = .false.
  read_q                     = .false.
  read_Vpar0                 = .false.
  profile_choice             = 2
  rpeak                      = 0.5
  kappa_ns0                  = 2.2
  kappa_Ts0                  = 8.
  deltar_ns0                 = 0.04
  deltar_Ts0                 = 0.04
  q_profile                  = 2
  q_param1                   = 1.5
  q_param2                   = 1.7
  q_param3                   = 2.8
  q_param4                   = 1.
  magnetic_drift             = .true.
  perturb_amplitude          = 0.001
  perturb_choice             = 2
  m                          = 28
  n                          = 10
  FFTpotential_filter        = 0
  FFTpotential_filter_deltam = 0.
  !--> Gyroaverage tunable input parameters
  gyroaverage                = .true.
! gyro_strategy              = ""
! All possible choices for gyro_strategy (by default "PADE")
! ("PADE", "LIBGYROAVG_HERMITE", "LIBGYROAVG_HERMITE_C1",
!  "LIBGYROAVG_HERMITE_C1_PRE_COMP")
! gyro_Nbpoints              = 32,     ! usefull only if
gyro_strategy=LIBGYROAVG_*
  TF_ripple                  = .false.
  delta_ripple               = 0
  N_ripple                   = 0,
/

&COLLISION_OPERATOR
  collisions      = .true.
  version_coll    = 2
  energy_transfer = .false.
  moment_transfer = .false.
  nustar          = 0.28
  coll_FMrefresh  = 10
  coll_nbstep     = 1,
```

```
15/01/2015                A150_FD_allwcoll2_semitore                    3
/

&BUFFER_REGION
  coef_krook       = 0.01
  coef_diffr       = 0.015
  coef_diffth      = 0.0
  coef_nustar      = 20.
  buffer_asymmetry = .false.
  right_buffer_only = .true.
  buffer_location  = 0.06
  buffer_stiffness = 0.017635,
/

&ALGORITHM
  Vlasov_scheme = "BSL"
  limiter       = 0
  deltat        = 5.
  nbiter        = 250
  dt_diag       = 50.,
/

&SOURCE
  gradient_driven = .false.
  grad_version    = 1
  flux_driven     = .false.
  heat_source     = .false.
  heat_version    = 0
  Sce0            = 0.0217
  TS              = 1.5
  rhoS            = 0.
  LS              = 0.12
  dens_source     = .false.
  Sce_dens        = 0.01
  TS_dens         = 1.
  rhoS_dens       = 0.1
  LS_dens         = 0.01
  mom_source      = .false.
  Sce_mom         = -0.001
  TS_mom          = 1.5
  rhoS_mom        = 0.
  LS_mom          = 0.1
  vor_source      = .false.
  Sce_vor         = 0.03
  TS_vor          = 1.
  rhoS_vor        = 0.1
  LS_vor          = 0.01,
/

&TEST
  !--> Tests for general numerics
  integral_vperp     = .true.
  integration_scheme = 2
  !--> Tests for QN solver
  solve_QN             = .true.
  QN_version           = 5
  QN_comm_version      = 1
  QN_coef_polarization = 1.
  QN_coef_Phi00        = 1
  Phi00_BCrmin_Neumann = .false.
```

```
15/01/2015                A150_FD_allwcoll2_semitore                    4
  QN_species_impact    = 1.
  !--> Tests for Vlasov solver
  adaptive_deltat      = .false.
  advec2D_deltat       = 0.
  reduced_deltat       = 0.
  reduced_begin        = 800000.
  reduced_end          = 400000.
! hffilter_strategy    = "" ! ("DEFAULT", "NONE", "FFT", "WAVELET")
! hffilterfreq         = 0
! hffilterfref         = 0
  deltaF_in_diff       = .true.
  avoidtaylor          = .false.
  deltaF_in_advec      = .false.
  split_linear         = .false.
  clampv               = .false.
  cancel_nonlin        = .false.
  diagf                = .false.
  !--> Tests for RHS of Boltzmann equation
  RHS_only             = .false.
  !--> Tests for equilibrium definition
  feq_choice           = 1
  canonical_vpar0      = 4
  hvpar_in_fperturb    = .true.
  !--> Tests with simplified physics
  BstareqB             = .false.
  single_m             = .false.
  single_n             = .false.
  Rosenbluth_Hinton    = .false.
  !--> Tests for computer sciences
  ask_for_checksum     = 1
  modulo_restart_iter  = 150
! compact_rstfile = .false. !
  checkoverflow        = .false.,
/

&OUTPUT
  integration_CS       = .true.
  diag_level           = 7
  Phi3D_saving         = .false.
  Moments3D_saving     = .false.
  fluxes3D_GC_saving   = .false.
  f5D_saving           = .false.
  Phi3D_dt_diag        = 500.
  Moments3D_dt_diag    = 500.
  fluxes3D_GC_dt_diag  = 500.
  f5D_dt_diag          = 500.
  spectra_saving       = .false.
  spectra_dt_diag      = 10.
  r_diags              = 31
  doubleSP_m           = 2
  doubleSP_n           = 0
  iter_beg_write_trace = 1
  iter_end_write_trace = 3,
/
```

Figure N.1: Input data file used in the code for the first simulation for invariance tests (see section 4.2).

```
26/01/2015          test_RH_q1.9_Nr64Ntheta64Nvpar128_dt5                    1
#*****************************************
# job_parameters
#*****************************************
NB_RESTART   = 0
TIME         = 08:00:00
MACHINE_FILE =
JOBNAME      = gysela5D

#*****************************************
#  Variables used to specify if the
#   simulation is a restart from
#   another one
#*****************************************
SEARCH_SP0   = 0            ! 0 if no restart, 1 otherwise
DIR_SP0      =              ! if SEARCH_SP0=1, name of the directory
                            !   containing the major species restart
                            files

#*****************************************
#  Variables for parallelization
#*****************************************
NSPECIES = 1
NPROC_R  = 2
NPROC_TH = 1
NPROC_MU = 16
NTHREAD  = 4

#*****************************************
#  Variables for radial profile input files
#*****************************************
ns0_sp0_filename = ns0_sp0_test.dat
ns0_sp1_filename = ns0_sp1_test.dat
Ts0_sp0_filename = Ts0_sp0_test.dat
Ts0_sp1_filename = Ts0_sp1_test.dat
q_filename       = safety_factor_test.dat
Vpar0_filename   = Vpar0_test.dat
ripple_filename  = ripple_test.dat

#*****************************************
# gysela input datas
#*****************************************
&PARALLEL
  Nspecies        = $NSPECIES
  Nbproc_r        = $NPROC_R
  Nbproc_theta    = $NPROC_TH
  Nbproc_mu       = $NPROC_MU
  bloc_phi        = $NTHREAD
  Nbthread        = $NTHREAD
  large_platform = .true.
  distrib3D       = .false.,
/

&MESH
  CylindricalGeometry = .false.
  Nr                  = 63
  Ntheta              = 64
  Nphi                = 32
  Nvpar               = 127
  Nmu                 = $((NPROC_MU-1))
```

```
26/01/2015          test_RH_q1.9_Nr64Ntheta64Nvpar128_dt5                    2
  a                   = 100.
  rhomin              = 0.2
  rhomax              = 0.8
  Ltheta              = 6.283185307179586476925286766559005768394
  Lphi                = 6.283185307179586476925286766559005768394
  aspect_ratio        = 2.78
  nb_vth0             = 7.
  Lmu                 = 12.
  mumin               = 0.,
/

&EQUIL
  canonical_profile          = 1
  tau0                       = 1.
  A0                         = 1.
  Z0                         = 1.
  As                         = 1.
  Zs                         = 1.
  species_frac               = 1.
  read_ns0                   = .false.
  read_Ts0                   = .false.
  read_q                     = .false.
  read_Vpar0                 = .false.
  profile_choice             = 1
  rpeak                      = 0.5
  kappa_ns0                  = 0.0000001
  kappa_Ts0                  = 0.0000001
  deltar_ns0                 = 0.2
  deltar_Ts0                 = 0.1
  q_profile                  = 2
  q_param1                   = 1.9
  q_param2                   = 0
  q_param3                   = 1.9
  q_param4                   = 0.
  magnetic_drift             = .true.
  perturb_amplitude          = 0.001
  perturb_choice             = 1
  m                          = 0
  n                          = 0
  FFTpotential_filter        = 0
  FFTpotential_filter_deltam = 0.
  !--> Gyroaverage tunable input parameters
  gyroaverage                = .true.
! gyro_strategy               = ""
! All possible choices for gyro_strategy (by default "PADE")
! ("PADE", "LIBGYROAVG_HERMITE", "LIBGYROAVG_HERMITE_C1",
!  "LIBGYROAVG_HERMITE_C1_PRE_COMP")
! gyro_Nbpoints               = 32,     ! usefull only if
gyro_strategy=LIBGYROAVG_*
  TF_ripple                  = .false.
  read_ripple                = .false.
  delta_ripple               = 0
  N_ripple                   = 0,
/

&COLLISION_OPERATOR
  collisions      = .false.
  version_coll    = 2
  energy_transfer = .false.
```

```
26/01/2015          test_RH_q1.9_Nr64Ntheta64Nvpar128_dt5                    3
  moment_transfer = .false.
  nustar          = 0.1
  coll_FMrefresh  = 2
  coll_nbstep     = 1,
/

&BUFFER_REGION
  coef_krook       = 0.0
  coef_diffr       = 0.0
  coef_diffth      = 0.0
  coef_nustar      = 0.
  buffer_asymmetry = .false.
  buffer_location  = 0.1
  buffer_stiffness = 0.1,
/

&ALGORITHM
  Vlasov_scheme = "BSL"
  limiter       = 0
  deltat        = 5.
  nbiter        = 4000
  dt_diag       = 50.,
/

&SOURCE
  gradient_driven = .false.
  grad_version    = 1
  flux_driven     = .false.
  heat_source     = .false.
  heat_version    = 0
  Sce0            = 0.01
  TS              = 1.
  rhoS            = 0.1
  LS              = 0.01
  dens_source     = .false.
  Sce_dens        = 0.01
  TS_dens         = 1.
  rhoS_dens       = 0.1
  LS_dens         = 0.01
  mom_source      = .false.
  Sce_mom         = 0.02
  TS_mom          = 1.
  rhoS_mom        = 0.1
  LS_mom          = 0.02
  vor_source      = .false.
  Sce_vor         = 0.03
  TS_vor          = 1.
  rhoS_vor        = 0.1
  LS_vor          = 0.03
  fp_source       = .false.
  fast_part       = .false.
  Sce_fp          = 0.01
  Sce_fpplus      = 0.6
  TS_fp           = 0.5
  v0_fp           = 2.
  rhoS_fp         = 0.5
  LS_fp           = 0.015
  alpha_trapp     = 1.,
/
```

```
26/01/2015          test_RH_q1.9_Nr64Ntheta64Nvpar128_dt5                    4

&FAST_PARTICLES
  bump_on_tail  = .false.
  fp_rshape     = .false.
  fp_lambda     = .true.
  fp_mom        = .false.
  fp_energy     = .true.
  fp_momgauss   = .false.
  Eb_fp         = 50.
  Emin_fp       = 0.
  n_fp          = 0.
  dn_fp         = 0.
  E_critfp      = 5.
  rho_0fp       = 0.5
  deltarhofp    = 0.01
  Lambda_0fp    = 0.5
  deltaLambdafp = 0.2
  psi_0fp       = 0.5
  deltapsifp    = 0.01
  zetabar       = 4.
/

&TEST
  !--> Tests for general numerics
  integral_vperp       = .true.
  integration_scheme   = 2
  !--> Tests for QN solver
  solve_QN             = .true.
  QN_version           = 3
  QN_comm_version      = 1
  QN_coef_polarization = 1.
  QN_coef_Phi00        = 1
  Phi00_BCrmin_Neumann = .false.
  QN_species_impact    = 1.
  !--> Tests for Vlasov solver
  adaptive_deltat      = .false.
  advec2D_deltat       = 0.
  reduced_deltat       = 0.
  reduced_begin        = 800000.
  reduced_end          = 120000.
  hffilterfreq         = 0
! hffilter_strategy    = "" ! ("DEFAULT", "NONE", "FFT", "WAVELET")
  hffilterfref         = 0
  deltaF_in_diff       = .false.
  avoidtaylor          = .false.
  deltaF_in_advec      = .false.
  split_linear         = .false.
  clampv               = .false.
  cancel_nonlin        = .false.
  diagf                = .false.
  !--> Tests for RHS of Boltzmann equation
  RHS_only             = .false.
  !--> Tests for equilibrium definition
  feq_choice           = 1
  canonical_vpar0      = 3
  hvpar_in_fperturb    = .true.
  !--> Tests with simplified physics
  BstareqB             = .false.
  single_m             = .false.
```

Figure N.2: Example of input data file used for Rosenbluth-Hinton test (see section 4.3.1).

```
27/01/2015          A180_cyclone5D_NEWPROF_SINGLEN_m7n5                    1
#*********************************************
# job parameters
#*********************************************
NB_RESTART   = 8
TIME         = 08:00:00
MACHINE_FILE =
JOBNAME      = A180_m7n5

#*********************************************
#  Variables used to specify if the
#   simulation is a restart from
#   another one
#*********************************************
SEARCH_SP0   = 0            ! 0 if no restart, 1 otherwise
DIR_SP0      =              ! if SEARCH_SP0=1, name of the directory
                            !   containing the major species restart
                            files

#*********************************************
#  Variables for parallelization
#*********************************************
NSPECIES = 1
NPROC_R  = 2                !150:4
NPROC_TH = 4                !150:4
NPROC_MU = 16               !150:32
NTHREAD  = 4                !150:4

#*********************************************
#  Variables for radial profile input files
#*********************************************
ns0_sp0_filename = ns0_sp0_test.dat
ns0_sp1_filename = ns0_sp1_test.dat
Ts0_sp0_filename = Ts0_sp0_test.dat
Ts0_sp1_filename = Ts0_sp1_test.dat
Te0_filename     = Te0_test.dat
q_filename       = safety_factor_test.dat

#*********************************************
# gysela input datas
#*********************************************
&PARALLEL
  Nspecies        = $NSPECIES
  Nbproc_r        = $NPROC_R
  Nbproc_theta    = $NPROC_TH
  Nbproc_mu       = $NPROC_MU
  bloc_phi        = $NTHREAD
  Nbthread        = $NTHREAD
  transpose4D     = .true.
  large_platform = .true.
  distrib3D       = .false.,
/

&MESH
  CylindricalGeometry = .false.
  Nr                  = 255
  Ntheta              = 256
  Nphi                = 128
  Nvpar               = 64
  Nmu                 = $((NPROC_MU-1))
```

```
27/01/2015          A180_cyclone5D_NEWPROF_SINGLEN_m7n5                    2
  a                   = 180.
  rhomin              = 0.1
  rhomax              = 0.9
  Ltheta              = 6.2831853071795864769252867665590057683943
  Lphi                = 6.2831853071795864769252867665590057683943
  aspect_ratio        = 2.78
  nb_vth0             = 6.
  Lmu                 = 7.
  mumin               = 0.,
/

&EQUIL
  canonical_profile          = 0
  tau0                       = 1.
  A0                         = 1.
  Z0                         = 1.
  As                         = 1.
  Zs                         = 1.
  species_frac               = 1.
  read_ns0                   = .false.
  read_Ts0                   = .false.
  read_Te0                   = .false.
  read_q                     = .false.
  profile_choice             = 5
  rpeak                      = 0.5
  kappa_ns0                  = 2.23
  kappa_Ts0                  = 6.96
  kappa_Te0                  = 6.96
  deltar_ns0                 = 0.3
  deltar_Ts0                 = 0.3
  deltar_Te0                 = 0.3
  q_profile                  = 4
  q_param1                   = 0.86
  q_param2                   = -0.16
  q_param3                   = 2.52
  q_param4                   = 0.
  magnetic_drift             = .true.
  perturb_amplitude          = 0.001
  perturb_choice             = 2
  m                          = 7
  n                          = 5
  FFTpotential_filter        = 0
  FFTpotential_filter_deltam = 0.
  TF_ripple                  = .false.
  delta_ripple               = 0
  N_ripple                   = 0,
/

&COLLISION_OPERATOR
  collisions      = .false.
  version_coll    = 3
  energy_transfer = .false.
  moment_transfer = .false.
  nustar          = 1.
  coll_FMrefresh  = 15                    !150:10
  coll_nbstep     = 4,
/

&BUFFER_REGION
```

```
27/01/2015          A180_cyclone5D_NEWPROF_SINGLEN_m7n5                    3
  coef_krook        = 0.
  coef_diffr        = 0.0                 !150: 0.04
  coef_diffth       = 0.0                 !150: 0.04
  coef_nustar       = 20                  !75: 100.
  buffer_asymmetry = .false.              !.true.
  buffer_location  = 0.05                 !75: 0.1  !150: 0.08
  buffer_stiffness = 0.025,               !75: 0.02,    !150: 0.018
/

&ALGORITHM
  Vlasov_scheme = "BSL"
  limiter       = 0
  deltat        = 40.
  nbiter        = 200
  dt_diag       = 240.,
/

&SOURCE
  gradient_driven = .false.
  flux_driven     = .false.
  heat_source     = .false.
  heat_version    = 0
  Sce0            = 0.009
  TS              = 1.
  rhoS            = 0.1
  LS              = 0.01
  dens_source     = .false.
  Sce_dens        = 0.009
  TS_dens         = 1.
  rhoS_dens       = 0.1
  LS_dens         = 0.01
  mom_source      = .false.
  Sce_mom         = 0.018
  TS_mom          = 1.
  rhoS_mom        = 0.1
  LS_mom          = 0.02
  vor_source      = .false.
  Sce_vor         = 0.009
  TS_vor          = 1.
  rhoS_vor        = 0.1
  LS_vor          = 0.01,
/

&TEST
  gyroaverage          = .true.
  integral_vperp       = .true.
  BstareqB             = .false.
  Rosenbluth_Hinton    = .false.
  solve_QN             = .true.
  QN_version           = 4
  QN_comm_version      = 1
  QN_coef_polarization = 1.
  QN_coef_Phi00        = 0
  Phi00_BCrmin_Neumann = .false.
  single_m             = .false.
  single_n             = .true.
  RHS_only             = .false.
  feq_choice           = 1
  canonical_vpar0      = 3
```

```
27/01/2015          A180_cyclone5D_NEWPROF_SINGLEN_m7n5                    4
  hvpar_in_fperturb    = .true.
  adaptive_deltat      = .false.
  advec2D_deltat       = 0.
  reduced_deltat       = 0.
  reduced_begin        = 8000000.
  reduced_end          = 40000000.
  hffilterfreq         = 0
  hffiltertype         = 0
  hffilterfref         = 0
  deltaF_in_diff       = .true.
  asynchrone_writing   = .false.
  ask_for_checksum     = 1
  modulo_restart_iter  = 100,
  checkoverflow        = .false.
  avoidtaylor          = .false.
  deltaF_in_advec      = .false.
  split_linear         = .false.
  clampv               = .false.
  cancel_nonlin        = .false.
  diagf                = .false.,
/

&OUTPUT
  integration_CS       = .true.
  diag_level           = 6
  Phi3D_saving         = .false.
  Moments3D_saving     = .false.
  f5D_saving           = .false.
  Phi3D_dt_diag        = 1800.
  Moments3D_dt_diag    = 1800.
  f5D_dt_diag          = 1800.
  spectra_saving       = .false.
  spectra_dt_diag      = 10.
  r_diags              = 31
  doubleSP_m           = 2
  doubleSP_n           = 0
  rst_saving           = .true.
  iter_beg_write_trace = 1
  iter_end_write_trace = 3,
/
```

Figure N.3: Example of input data file used for the Cyclone Base Case benchmark with GENE code (see section 4.3.2).

Appendix O

GYSELA diagnostics

As imagined, one difficulty for gyrokinetic code is that they have to manipulate several 5D distribution functions. It is even more true for global codes as GYSELA which have to tackle huge 5D meshes. To get an idea, the memory size of a distribution function for meshes commonly used in GYSELA is of the order of several hundreds Gigabytes. Knowing that a flux-driven GYSELA simulation can contain several hundreds of thousands of iterations, you immediately see the issue linked to *big data* storage. It is absolutely not possible to envisage a frequent saving of these 5D distribution functions. One challenge for physicists is to find a good compromise between acceptable amounts of data to store and enough pertinent information to analyze plasma turbulence properties. We present here a summary, as complete as possible, of all the physical quantities which are saved all along a GYSELA simulation. In GYSELA, there are several level of diagnostics: (i) 3D quantities corresponding mainly to first fluid moments (see O.2 for details); (ii) 2D cross-sections of $\bar{F}_s$ and the electrostatic potential ϕ; (iii) 1D radial profiles generally related to flux-surface average (cf details in O.1) and (iv) scalars for quantities integrated on the whole phase space (*e.g.* number of ion species, number of electrons, ...).
Just a few 0D data are saved at each iterations. On the other hand 1D quantities are only saved every `<time_diag>` times. The frequency with which 3D diagnostics are computed is generally ten to hundred times lower.

O.1 Radial diagnostics

In this section all the quantities considered are normalized quantities but the hat symbols are omitted for more readability. In the following, $\langle \cdot \rangle_{\mathrm{FS}}$ corresponds to the flux surface average, i.e

$$\langle \cdot \rangle_{\mathrm{FS}} = \frac{1}{V'} \int \cdot \mathcal{J}_{\mathrm{x}} \, \mathrm{d}\theta \, \mathrm{d}\varphi \quad \text{where } V' = \int \mathcal{J}_{\mathrm{x}} \, \mathrm{d}\theta \, \mathrm{d}\varphi$$

with $\mathcal{J}_{\mathrm{x}}$ being the jacobian in space.

O.1.1 Diagnostics concerning guiding-centers

At the following, different colors are associated to the different values which are saved. The color coding is the following:

- For local charge density conservation
- For local conservation of the toroidal momentum
- For local energy conservation
- For parallel pressure balance
- For perpendicular pressure balance

Concerning notations $\bar{\mathbf{v}}_{E\times B_s}$ corresponds to the $E \times B$ drift associated to the guiding-center, i.e depends on $J_0\phi$.

	new name	old name	formula
1.	densGC_FSavg	niGCr	$\langle \int \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
2.	nvpoloGC_turb_FSavg	nvpoloGCr_turb	$\langle \int (\mathbf{v}_{E\times B_s} \cdot \mathbf{e}_\theta) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
3.	nvpoloGC_neo_FSavg	nvpoloGCr_neo	$\langle \int (\mathbf{v}_{D_s} \cdot \mathbf{e}_\theta) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
4.	nvpoloGC_vpar_FSavg	nvpoloGCr_vpar	$\langle \int (\mathbf{b}_s^* \cdot \mathbf{e}_\theta) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
5.	pressGC_perp_FSavg	pressGCr_perp	$p_{s,\perp} = \langle \int \mu B \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
6.	pressGC_par_FSavg		$p_{s,\parallel} = \langle \int \frac{1}{2} \mathcal{U}^2 \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$ with $\mathcal{U} = (v_{G\parallel} - V_{s,\parallel})$
7.	stressGC_par_FSavg	pressGCr_par	$P_{s,\parallel} = \langle \int v_{G\parallel}^2 \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
8.	LtorGC_FSavg	LphiGCr	$\mathcal{L}_{\varphi,s} = \langle \int u_\varphi \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$ with $u_\varphi = R^2(\mathbf{b}\cdot\nabla\varphi)v_{G\parallel}$
9.	TpolGC_chi_FSavg	TpolGCr	$\langle \int \partial_\varphi(J_{0_s} \cdot \phi) \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
10.	GammaGC_turb_r_FSavg	GammaGCr_turb	$\langle \int (\bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
11.	GammaGC_neo_r_FSavg	GammaGCr_neo	$\langle \int (\mathbf{v}_{D_s} \cdot \nabla r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
12.	RSphiGC_turb_r_FSavg	RSphiGCr_turb	$\langle \int u_\varphi \, (\bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$ with $u_\varphi = R^2(\mathbf{b}\cdot\nabla\varphi)v_{G\parallel}$
13.	RSphiGC_neo_r_FSavg	RSphiGCr_neo	$\langle \int u_\varphi \, (\mathbf{v}_{D_s} \cdot \nabla r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$ with $u_\varphi = R^2(\mathbf{b}\cdot\nabla\varphi)v_{G\parallel}$
14.	dLtorGC_dt_FSavg	dLphiGCr_dt	$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{L}_{\varphi,s}$
15.	QGC_perp_turb_r_FSavg	QGCr_perp_turb	$\langle \int \mu B \, (\bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
16.	QGC_perp_neo_r_FSavg	QGCr_perp_neo	$\langle \int \mu B \, (\mathbf{v}_{D_s} \cdot \nabla r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
17.	QGC_par_turb_r_FSavg	QGCr_par_turb	$\langle \int \frac{1}{2} v_{G\parallel}^2 \, (\bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$
18.	QGC_par_neo_r_FSavg	QGCr_par_neo	$\langle \int \frac{1}{2} v_{G\parallel}^2 \, (\mathbf{v}_{D_s} \cdot \nabla r) \; \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \rangle_{\mathrm{FS}}$

	new name	old name	formula
19.	QGC_dtvpar_FSavg	QGCr_dtvpar	$\langle \int v_{G\parallel} \frac{\mathrm{d}v_{G\parallel}}{\mathrm{d}t} \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$
20.	Qpot_r_FSavg		$\langle \int (J_{0_s} \cdot \phi) \, \bar{\mathbf{v}}_{\mathrm{drift}} \cdot \nabla r \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$ with $\bar{\mathbf{v}}_{\mathrm{drift}} = \bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r + \mathbf{v}_{D_s} \cdot \nabla r$
21.	dpressGC_dt_perp_FSavg	dpGCr_dt_perp	$\frac{\mathrm{d}}{\mathrm{d}t} p_{s,\perp}$
22.	dstressGC_dt_par_FSavg	dpGCr_dt_par	$\frac{\mathrm{d}}{\mathrm{d}t} P_{s,\parallel}$
23.	Enkin_FSavg		$\langle \int \mathcal{E} \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$ with $\mathcal{E} = \frac{1}{2} v_{G\parallel}^2 + \mu B$
24.	SGC_Pperp_par_FSavg	SGCr_Pperp_par	$\langle \int \mu B \left(v_{G\parallel} \cdot \frac{\nabla_\parallel B}{B} \right) \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$
25.	SGC_Pperp_turb_FSavg	SGCr_Pperp_turb	$\langle \int \mu B \left(\bar{\mathbf{v}}_{E\times B_s} \cdot \frac{\nabla_\parallel B}{B} \right) \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$
26.	heatexGC_turb_FSavg it should be zero ??	heatexGCr_turb	$-\langle \int (\bar{\mathbf{v}}_{E\times B_s} \cdot \nabla(J_{0_s}\phi)) \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$
27.	heatexGC_neo_FSavg	heatexGCr_neo	$-\langle \int (\mathbf{v}_{D_s} \cdot \nabla(J_{0_s} \cdot \phi)) \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$
28.	HfluxGC_par_turb_r_FSavg		$\langle \int \frac{1}{2} \mathcal{U}^2 \, (\bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r) \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$ with $\mathcal{U} = (v_{G\parallel} - V_{s,\parallel})$
29.	HfluxGC_par_neo_r_FSavg		$\langle \int \frac{1}{2} \mathcal{U}^2 \, (\mathbf{v}_{D_s} \cdot \nabla r) \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \rangle_{\mathrm{FS}}$ with $\mathcal{U} = (v_{G\parallel} - V_{s,\parallel})$
2D diagnostics in (r, θ)			
30.	QGC_rtheta	QGC_loc	$\frac{1}{L_\varphi} \int \mathcal{E} \, (\bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r) \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \mathrm{d}\varphi$ with $\mathcal{E} = \frac{1}{2} v_{G\parallel}^2 + \mu B$
31.	RSphiGC_rtheta	RSphiGC_loc	$\frac{1}{L_\varphi} \int u_\varphi \, (\bar{\mathbf{v}}_{E\times B_s} \cdot \boldsymbol{\nabla} r) \, \bar{F}_s \, \mathrm{d}^3\mathbf{v} \, \mathrm{d}\varphi$ with $u_\varphi = R^2 (\mathbf{b} \cdot \nabla\varphi) v_{G\parallel}$
0D diagnostics			
32.	entropy_tot	entropy	$\int \bar{F}_s \log \lvert \bar{F}_s \rvert \, \mathrm{d}^3\mathbf{x} \, \mathrm{d}^3\mathbf{v}$ with $\mathrm{d}^3\mathbf{x} = \mathcal{J}_{\mathrm{x}} \, \mathrm{d}r \, \mathrm{d}\theta \, \mathrm{d}\varphi$

	new name	old name	formula
33.	L2norm_tot	L2norm	$\int \bar{F}_s^2 \, \mathrm{d}^3\mathbf{x} \, \mathrm{d}^3\mathbf{v}$ with $\mathrm{d}^3\mathbf{x} = \mathcal{J}_\mathrm{x} \, \mathrm{d}r \, \mathrm{d}\theta \, \mathrm{d}\varphi$
34.	deltaEnkin_tot	Enkin	$\int \mathcal{E}(\bar{F}_s - \bar{F}_{s,\mathrm{eq}}) \, \mathrm{d}^3\mathbf{x} \, \mathrm{d}^3\mathbf{v}$ with $\mathrm{d}^3\mathbf{x} = \mathcal{J}_\mathrm{x} \, \mathrm{d}r \, \mathrm{d}\theta \, \mathrm{d}\varphi$ and with $\mathcal{E} = \frac{1}{2} v_{G\parallel}^2 + \mu B$

O.1.2 Diagnostics concerning test particles

	new name	old name	formula
1.	dens_FSavg	nir	$\langle \int (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$
2.	dens_trapped_FSavg	nir	$\langle \int_\mathrm{trapped} (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$ where particles are trapped if $v_{G\parallel}^2 < 2\mu(B(r,\theta=\pi) - B(r,\theta))$
3.	nupar_FSavg	nuparr	$\langle \int v_{G\parallel} (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$
4.	nvpolo_mag_FSavg	nvpolor_mag	$\langle \int \mu \, b_\varphi \, (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$ with $b_\varphi = B_\varphi / B$
5.	stress_FSavg	pressr	$\langle \int \mathcal{E} (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$ with $\mathcal{E} = \frac{1}{2} v_{G\parallel}^2 + \mu B$
6.	Q_turb_r_FSavg	Qr_turb	$\langle \int \mathcal{E} \, (\mathbf{v}_{E\times B_s} \cdot \nabla r) \, (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$ with $\mathcal{E} = \frac{1}{2} v_{G\parallel}^2 + \mu B$
7.	Q_neo_r_FSavg	Qr_neo	$\langle \int \mathcal{E} \, (\mathbf{v}_{D_s} \cdot \nabla r) \, (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$ with $\mathcal{E} = \frac{1}{2} v_{G\parallel}^2 + \mu B$
8.	Gamma_turb_r_FSavg	Gammar_turb	$\langle \int (\mathbf{v}_{E\times B_s} \cdot \nabla r) \, (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$
9.	Gamma_neo_r_FSavg	Gammar_neo	$\langle \int (\mathbf{v}_{D_s} \cdot \nabla r) \, (J_0 \cdot \bar{F}_s) \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$
10.	RStheta_r_FSavg	RSthetar	$\langle \int J_0 \cdot \left[(\mathbf{v}_{E\times B_s} \cdot \nabla r) \cdot \bar{F}_s \right] \, \mathrm{d}^3\mathbf{v} \, \rangle_\mathrm{FS}$

	new name	old name	formula
11.	Enpot_FSavg		$\frac{1}{2}\langle\, \phi \int (J_0 \cdot \bar{F}_s)\, \mathrm{d}^3\mathbf{v}\, \rangle_{\mathrm{FS}}$
0D diagnostics			
12.	deltaEnpot_tot	Enpot	$\langle \int \frac{1}{2} \left(J_0 \cdot \bar{F}_s - J_0 \cdot \bar{F}_{s,\mathrm{eq}}\right)\, \mathrm{d}^3\mathbf{x}\, \mathrm{d}^3\mathbf{v}\, \rangle_{\mathrm{FS}}$ with $\mathrm{d}^3\mathbf{x} = \mathcal{J}_{\mathrm{x}}\, \mathrm{d}r\, \mathrm{d}\theta\, \mathrm{d}\varphi$

O.2 3D diagnostics

All the 3D diagnostics are related to the guiding-centers.

- Densities	
densGC	$n_{G_s} = \int \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
densGC_trapped	$n_{G_s,\mathrm{trapped}} = \int_{\mathrm{trapped}} \bar{F}_s\, \mathrm{d}^3\mathbf{v}$ where guiding-centers are trapped if $v_{G\parallel}^2 < 2\mu(B(r,\theta=\pi) - B(r,\theta))$
- Parallel velocity	
VparGC	$V_{s,\parallel} = \frac{1}{n_{G_s}} \int v_{G\parallel} \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
- Parallel and perpendicular pressures	
pressGC_par	$p_{s,\parallel} = \int \left(v_{G\parallel} - V_{s,\parallel}\right)^2 \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
presssGC_perp	$p_{s,\perp} = \int \mu B\, \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
presssGC_perp_trapped	$p_{s,\perp\,\mathrm{trapped}} = \int_{\mathrm{trapped}} \mu B\, \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
- Turbulent flux	
GammmaGC_perp_r	$\Gamma_{s,\perp}^r = \int \left(\mathbf{v}_{E\times B_s} \cdot \boldsymbol{\nabla} r\right)\, \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
- Heat fluxes	
HfluxGC_turb_r_Epar	$q_{s,\mathrm{turb}\parallel}^r = \int \frac{1}{2} \left(v_{G\parallel} - V_{s,\parallel}\right)^2 \left(\mathbf{v}_{E\times B_s} \cdot \boldsymbol{\nabla} r\right)\, \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
HfluxGC_turb_r_Eperp	$q_{s,\mathrm{turb}\perp}^r = \int \mu B \left(\mathbf{v}_{E\times B_s} \cdot \boldsymbol{\nabla} r\right)\, \bar{F}_s\, \mathrm{d}^3\mathbf{v}$

HfluxGC_neo_r_Epar	$q^r_{s,\mathrm{neo}\parallel} = \int \frac{1}{2}\left(v_{G\parallel} - V_{s,\parallel}\right)^2 \left(\mathbf{v}_{D_s} \cdot \boldsymbol{\nabla} r\right)\ \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
HfluxGC_neo_r_Eperp	$q^r_{s,\mathrm{neo}\perp} = \int \mu B \left(\mathbf{v}_{D_s} \cdot \boldsymbol{\nabla} r\right)\ \bar{F}_s\, \mathrm{d}^3\mathbf{v}$
HfluxGC_par_Epar	$q_{s,\mathrm{par}\parallel} = \int \frac{1}{2}\left(v_{G\parallel} - V_{s,\parallel}\right)^3\ \bar{F}_s\, \mathrm{d}^3\mathbf{v}$

Bibliography

[ABe06] P. Angelino, A. Bottino, and R. Hatzky *et al.* On the definition of a kinetic equilibrium in global gyrokinetic simulations. *Phys. Plasmas*, 13:052304, 2006.

[Abi12] J. Abiteboul. *Turbulent and neoclassical toroidal momentum transport in tokamak plasmas.* Thesis, Aix-Marseille Université, Oct 2012.

[AGG+11] J. Abiteboul, X. Garbet, V. Grandgirard, S. J. Allfrey, Ph. Ghendrih, G. Latu, Y. Sarazin, and A. Strugarek. Conservation equations and calculation of mean flows in gyrokinetics. *Phys. Plasmas*, 18(8), 2011.

[AGG+13] J. Abiteboul, Ph. Ghendrih, V. Grandgirard, T. Cartier-Michaud, G. Dif-Pradalier, X. Garbet, G. Latu, Ch. Passeron, Y. Sarazin, A. Strugarek, O. Thomine, and D. Zarzoso. Turbulent momentum transport in core tokamak plasmas and penetration of scrape-off layer flows. *Plasma Physics and Controlled Fusion*, 55(7):074001, 2013.

[Amm90] H. Amman. *Ordinary differential equations.* De Gruyter, 1990.

[ANW67] J.H. Ahlberg, E.N Nilson, and J.L Walsh. *The Theory of Splines and Their Applications (1st edition).* Academic Press, new york edition, 1967.

[B0̈7] J. Büchner. Vlasov code simulation. *Advanced Methods for Space Simulations*, pages 23–46, 2007.

[BBLZ14] A. Biancalani, A. Bottino, Ph. Lauber, and D. Zarzoso. Numerical validation of the electromagnetic gyrokinetic code NEMORB on global axisymmetric modes. *Nuclear Fusion*, 54(10):104004, 2014.

[BCG+11] J-Ph. Braeunig, N. Crouseilles, V. Grandgirard, G. Latu, M. Mehrenberger, and E. Sonnendrücker. Some numerical aspects of the conservative PSM scheme in a 4D drift-kinetic code. Inria research report, Dec 2011.

[BCMG+15] J. Bigot, T Cartier-Michaud, V. Grandgirard, G. Latu, Ch. Passeron, and F. Rozar. An approach to increase reliability of HPC simulation, application to the GYSELA5D code. In *CEMRACS 2014*, volume submitted of *ESAIM: Proc.*, Luminy, France, 2015.

[BCMS12] J. P. Braeunig, N. Crouseilles, M. Mehrenberger, and E. Sonnendrücker. Guiding-center simulations on curvilinear meshes. *Discrete and Continuous Dynamical Systems Series S*, 5(3), 2012.

[BCS07] R. Barthelme, P. Ciarlet, and E. Sonnendrücker. Generalized formulations of Maxwell's equations for numerical Vlasov-Maxwell simulations. *Mathematical Models and Methods in Applied Science*, 17(5):657–680, 2007.

[BGL+13] J. Bigot, V. Grandgirard, G. Latu, Ch. Passeron, F. Rozar, and O. Thomine. Scaling gysela code beyond 32K-cores on bluegene/Q. In *CEMRACS 2012*, volume 43 of *ESAIM: Proc.*, pages 117–135, Luminy, France, 2013.

[BGS+04] M. Brunetti, V. Grandgirard, O. Sauter, J. Vaclavik, and L. Villard. A semi-lagrangian code for nonlinear global simulations of electrostatic drift-kinetic ITG modes. *Computer Physics Communications*, 163(1):1 – 21, 2004.

[BGTK+11] L. Bautista-Gomez, S. Tsuboi, D. Komatitsch, F. Cappello, N. Maruyama, and S. Matsuoka. FTI: high performance fault tolerance interface for hybrid systems. *Proceedings Int. Conf. High Performance Computing, Networking, Storage and Analysis (SC11)*, 17(5):1–32, 2011.

[BH07] A.J. Brizard and T.S. Hahm. Foundations of nonlinear gyrokinetic theory. *Rev. Mod. Phys.*, 79(2):421–468, Apr 2007.

[BL85] C.K. Birdsall and A.B. Langdon. *Plasma physics via computer simulation.* McGraw-Hill, 1985.

[BM08] N. Besse and M. Mehrenberger. Convergence of classes of high-order semi-Lagrangian schemes for the Vlasov-Poisson system. *Mathematics of Computation*, 77(61):93–123, 2008.

[BPH+07] A. Bottino, A. G. Peeters, R. Hatzky, S. Jolliet, B. F. McMillan, T. M. Tran, and L. Villard. Nonlinear low noise particle-in-cell simulations of electron temperature gradient driven turbulence. *Phys. Plasmas*, 14(1):010701, 2007.

[Bri10] A. J. Brizard. Noether derivation of exact conservation laws for dissipationless reduced-fluid models. *Phys. Plasmas*, 17(11):112503, 2010.

[BSB+10] A. Bottino, B. Scott, S. Brunner, B.F. McMillan, T.M. Tran, T. Vernay, L. Villard, S. Jolliet, R. Hatzky, and A.G. Peeters. Global nonlinear electromagnetic simulations of Tokamak turbulence. *Plasma Science, IEEE Transactions on*, 38(9):2129 –2135, 2010.

[BT11] A. J. Brizard and N. Tronko. Exact momentum conservation laws for the gyrokinetic Vlasov-Poisson equations. *Phys. Plasmas*, 18(8):082307, 2011.

[Cap09] F. Cappello. Fault tolerance in petascale/exascale systems: Current knowledge, challenges and research opportunities. *Int. J. High Perform. Comput. Appl.*, 23(3):212–226, August 2009.

[CK76] C.Z. Cheng and G. Knorr. The integration of the Vlasov equation in configuration spaces. *J. Comput. Phys.*, (22):330–351, 1976.

[CK08] C. S. Chang and S. Ku. Spontaneous rotation sources in a quiescent tokamak edge plasma. *Phys. Plasmas*, 15(6):062510, 2008.

[CKD+09] C. S. Chang, S. Ku, P. H. Diamond, Z. Lin, S. Parker, T. S. Hahm, and N. Samatova. Compressed ion temperature gradient turbulence in diverted tokamak edge. *Phys. Plasmas*, 056108(5):16, 2009.

[CLS07] N. Crouseilles, G. Latu, and E. Sonnendrücker. Hermite splines interpolation on patches for a parallel solving of the Vlasov-Poisson equation. *Internal Journal of Applied Mathematics and Computer Science*, 17(3):335–349, 2007.

[CLS09] Nicolas Crouseilles, Guillaume Latu, and Eric Sonnendrücker. A parallel vlasov solver based on local cubic spline interpolation on patches. *J. Comput. Phys.*, 228(5):1429 – 1446, 2009.

[CMS10a] N. Crouseilles, M. Mehrenberger, and Hocine Sellama. Numerical solution of the gyroaverage operator for the finite gyroradius guiding-center model. *CICP*, page 1, 2010.

[CMS10b] N. Crouseilles, M. Mehrenberger, and E. Sonnendrücker. Conservative semi-Lagrangian schemes for Vlasov equations. *J. Comput. Phys.*, 229(6):1927 – 1953, 2010.

[CRS09] Nicolas Crouseilles, Thomas Respaud, and Eric Sonnendrüker. A forward semi-Lagrangian method for the numerical solution of the Vlasov equation. *Comp. Phys. Comm*, 180(10):1730 – 1745, 2009.

[CRS12] Nicolas Crouseilles, Ahmed Ratnani, and Eric Sonnendrücker. An isogeometric analysis approach for the study of the gyrokinetic quasi-neutrality equation. *J. Comput. Phys.*, 231(2):373–393, January 2012.

[CW03a] J. Candy and R.E. Waltz. Anomalous transport scaling in the DIII-D Tokamak matched by supercomputer simulation. *Phys. Rev. Lett.*, 91:045001, 2003.

[CW03b] J. Candy and R.E. Waltz. An Eulerian gyrokinetic-Maxwell solver. *J. Comput. Phys.*, 186(2):545–581, 2003.

[CWD04] J. Candy, R. E. Waltz, and W. Dorland. The local limit of global gyrokinetic simulations. *Physics of Plasmas*, 11(5):L25–L28, 2004.

[DBB+00] A. M. Dimits, G. Bateman, M. A. Beer, B. I. Cohen, W. Dorland, G. W. Hammett, C. Kim, J. E. Kinsey, M. Kotschenreuther, A. H. Kritz, et al. Comparisons and physics basis of tokamak transport models and turbulence simulations. *Phys. Plasmas*, 7(3):969–983, 2000.

[DeB01] C. DeBoor. *A practical guide to splines.* Springer-Verlag, New York, applied mathematical sciences 27 edition, 2001.

[DGBG00] G. Depret, X. Garbet, P. Bertrand, and A. Ghizzo. Trapped-ion driven turbulence in tokamak plasmas. *Plasma Phys. Control. Fusion*, 23(42):949–971, 2000.

[DGS+08] G. Darmet, Ph. Ghendrih, Y. Sarazin, X. Garbet, and V. Grandgirard. Intermittency in flux driven kinetic simulations of trapped ion turbulence. *Communications in Nonlinear Science and Numerical Simulation*, 13(1):53 – 58, 2008. Vlasovia 2006: The Second International Workshop on the Theory and Applications of the Vlasov Equation.

[DHCS91] W. D. D'haeseleer, W. N. G. Hitchon, J. D. Callen, and J. L. Shohet. *Flux Coordinates and Magnetic Structure, A Guide to a Fundamental Tool of Plasma Theory.* Springer, Berlin, 1991.

[DJKR00] W. Dorland, F. Jenko, M. Kotschenreuther, and B.N. Rogers. Electron temperature gradient turbulence. *Phys. Rev. Lett.*, (85):5579–5582, 2000.

[DK95] Richard E. Denton and M. Kotschenreuther. [delta]f algorithm. *Journal of Computational Physics*, 119(2):283–294, 1995.

[DKG+13] P Diamond, Y Kosuga, Ö Gürcan, C.J. McDevitt, T.S. Hahm, N. Fedorczak, J.E. Rice, W.X. Wang, S. Ku, J.M. Kwon, G. Dif-Pradalier, J. Abiteboul, L. Wang, W.H. Ko, Y.J. Shi, K. Ida, W. Solomon, H. Jhang, S.S. Kim, S.Yi, S.H. Ko, Y. Sarazin, R. Singh, and C.S. Chang. An overview of intrinsic torque and momentum transport bifurcations in toroidal plasmas. *Nuclear Fusion*, 53:104019, 2013.

[DPDG+11] G. Dif-Pradalier, P. H. Diamond, V. Grandgirard, Y. Sarazin, J. Abiteboul, X. Garbet, Ph. Ghendrih, G. Latu, A. Strugarek, S. Ku, and C. S. Chang. Neoclassical physics in full distribution function gyrokinetics. *Phys. Plasmas*, 18(6), 2011.

[DPGS+08a] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin, X. Garbet, and Ph. Ghendrih. Defining an equilibrium state in global full-f gyrokinetic models. *Communications in Nonlinear Science and Numerical Simulation*, 13(1):65 – 71, 2008. Vlasovia 2006: The Second International Workshop on the Theory and Applications of the Vlasov Equation.

[DPGS+08b] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin, X. Garbet, Ph. Ghendrih, and P. Angelino. On the influence of initial state on gyrokinetic simulations. *Phys. Plasmas*, 15(4), 2008.

[DPGS+09] G. Dif-Pradalier, V. Grandgirard, Y. Sarazin, X. Garbet, and Ph. Ghendrih. Interplay between gyrokinetic turbulence, flows, and collisions: Perspectives on transport and poloidal rotation. *Phys. Rev. Lett.*, 103:065002, Aug 2009.

[DPHG+15] G. Dif-Pradalier, G. Hornung, Ph. Ghendrih, Y. Sarazin, F. Clairet, L. Vermare, P. H. Diamond, J. Abiteboul, T. Cartier-Michaud, C. Ehrlacher, D. Esteve, X. Garbet, V. Grandgirard, Ö.D. Gürcan, P. Hennecquin, Y. Kosuga, G. Latu, P. Maget, P. Morel, C. Norscini, R. Sabot, and A. Storelli. Finding the elusive ExB staircase in magnetised plasmas. *Phys. Rev. Lett.*, 114:085004, 2015.

[Dur98] D. R. Durran. *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer-Verlag, New York, 1998.

[DWBC96] A.M. Dimits, T.J. Williams, J.A. Byers, and B.I. Cohen. Scalings of Ion-Temperature-Gradient-Driven anomalous transport in tokamaks. *Phys. Rev. Lett.*, 77(1):71–74, Jul 1996.

[DZS+13] R. J. Dumont, D. Zarzoso, Y. Sarazin, X. Garbet, A. Strugarek, J. Abiteboul, T. Cartier-Michaud, G. Dif-Pradalier, Ph. Ghendrih, J-B. Girardo, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Interplay between fast ions and turbulence in magnetic fusion plasmas. *Plasma Physics and Controlled Fusion*, 55(12):124012, 2013.

[EGS+15] D. Estève, X. Garbet, Y. Sarazin, V. Grandgirard, T. Cartier-Michaud, G. Dif-Pradalier, P. Ghendrih, G.Latu, and C. Norscini. A multi-species collisional operator for full-f gyrokinetics. *Phys. Plasmas*, (22):122506, 2015.

[FB05] M. R. Feix and P. Bertrand. A universal model: The Vlasov Equation. *Transport Theory and Statistical Physics*, 34(1):7–62, 2005.

[FSA+08] G. L. Falchetto, B. D. Scott, P. Angelino, A. Bottino, T. Dannert, V. Grandgirard, S. Janhunen, F. Jenko, S. Jolliet, A. Kendl, B. F. McMillan, V. Naulin, A. H. Nielsen, M. Ottaviani, A. G. Peeters, M. J. Pueschel, D. Reiser, T. T. Ribeiro, and M. Romanelli. The European turbulence code benchmarking effort: turbulence driven by thermal gradients in magnetically confined plasmas. *Plasma Physics and Controlled Fusion*, 50(12):124015, 2008.

[FSB01] F. Filbet, E. Sonnendrücker, and P. Bertrand. Conservative numerical schemes for the Vlasov equation. *J. Comput. Phys.*, (172):166–187, 2001.

[Fu08] G. Y. Fu. Energetic-Particle-Induced Geodesic Acoustic Mode. *Phys. Rev. Lett.*, 101:185002, Oct 2008.

[GBB+06] V. Grandgirard, M. Brunetti, P. Bertrand, N. Besse, X. Garbet, P. Ghendrih, G. Manfredi, Y. Sarazin, O. Sauter, E. Sonnendrücker, J. Vaclavik, and L. Villard. A drift-kinetic semi-lagrangian 4D code for ion turbulence simulation. *Journal of Computational Physics*, 217(2):395 – 423, 2006.

[GDPN+09] X. Garbet, G. Dif-Pradalier, C. Nguyen, Y. Sarazin, V. Grandgirard, and Ph. Ghendrih. Neoclassical equilibrium in gyrokinetic simulations. *Phys. Plasmas*, 16(6), 2009.

[GES+13] X. Garbet, D. Esteve, Y. Sarazin, J. Abiteboul, C. Bourdelle, G. Dif-Pradalier, P. Ghendrih, V. Grandgirard, G. Latu, and A. Smolyakov. Turbulent acceleration and heating in toroidal magnetized plasmas. *Phys. Plasmas*, 20(7), 2013.

[GIVW10] X. Garbet, Y. Idomura, L. Villard, and T.H. Watanabe. Gyrokinetic simulations of turbulent transport. *Nuclear Fusion*, 50(4):043002, 2010.

[GLB+11] Tobias Goerler, Xavier Lapillonne, Stephan Brunner, Tilman Dannert, Frank Jenko, Sohrab Khosh Aghdam, Patrick Marcus, Ben F. McMillan, Florian Merz, Olivier Sauter, Daniel Told, and Laurent Villard. Flux- and gradient-driven global gyrokinetic simulation of tokamak turbulence. *Physics of Plasmas*, 18(5):056103, 2011.

[GPS02] H. Goldstein, C. Poole, and J. Safko. *Classical Mechanics*. Addison-Wesley, San Francisco, 3rd ed. edition, 2002.

[GS13] V. Grandgirard and Y. Sarazin. Gyrokinetic simulations of magnetic fusion plasmas. *Panoramas et synthèses*, (39-40):91–176, 2013.

[GSA+07] V. Grandgirard, Y. Sarazin, P. Angelino, A. Bottino, N. Crouseilles, G. Darmet, G. Dif-Pradalier, X. Garbet, Ph. Ghendrih, S. Jolliet, G. Latu, E. Sonnendrücker, and L. Villard. Global full-f gyrokinetic simulations of plasma turbulence. *Plasma Physics and Controlled Fusion*, 49(12B):B173, 2007.

[GSG+08] V. Grandgirard, Y. Sarazin, X. Garbet, G. Dif-Pradalier, Ph. Ghendrih, N. Crouseilles, G. Latu, E. Sonnendrücker, N. Besse, and P. Bertrand. Computing ITG turbulence with a full-f semi-lagrangian code. *Communications in Nonlinear Science and Numerical Simulation*, 13(1):81 – 87, 2008. Vlasovia 2006: The Second International Workshop on the Theory and Applications of the Vlasov Equation.

[GTH+16] T. Goerler, N. Tronko, W.A Hornsby, R. Kleiber, C. Norscini, V. Grandgirard, F. Jenko, and E. Sonnendrücker. Intercode comparison of gyrokinetic global electromagnetic modes. *Phys. Plasmas (submitted)*, 2016.

[GZD+14] J.-B. Girardo, D. Zarzoso, R. Dumont, X. Garbet, Y. Sarazin, and S. Sharapov. Relation between energetic and standard geodesic acoustic modes. *Physics of Plasmas*, 21(092507), 2014.

[HKKS+01] J. A. Heikkinen, T. P. Kiviniemi, T. Kurki-Suonio, A. G. Peeters, and S. K. Sipilä. Particle simulation of the neoclassical plasmas. *J. Comput. Phys.*, 173(2):527–548, 2001.

[HM03] R.D. Hazeltine and J.D. Meiss. *Plasma confinement*. Dover publication, 2003.

[HMB+15] W A Hornsby, P Migliano, R Buchholz, D Zarzoso, F J Casson, E Poli, and A G Peeters. On seed island generation and the non-linear self-consistent interaction of the tearing mode with electromagnetic gyro-kinetic turbulence. *Plasma Phys. Control. Fusion*, 57(5):054018, 2015.

[HO13] F. Hariri and M. Ottaviani. A flux-coordinate independent field-aligned approach to plasma turbulence simulations. *Computer Physics Communications*, 184(11):2419 – 2429, 2013.

[HP90] G. W. Hammett and F. W. Perkins. Fluid moment models for Landau damping with application to the ion-temperature-gradient instability. *Phys. Rev. Lett.*, 64(25):3019–3022, Jun 1990.

[HR99] F.L. Hinton and M.N. Rosenbluth. Dynamics of axisymmetric ExB and poloidal flows in tokamaks. *Plasma Phys. Control Fusion*, 41, 1999.

[HS05] P. Helander and D.J. Sigmar. *Collisional Transport in Magnetized Plasmas*. Cambridge Monographs on Plasma Physics. Cambridge University Press, 2005.

[HTK+02] R. Hatzky, T.M. Tran, A. Konies, R. Kleiber, and S. J. Allfrey. Energy conservation in a nonlinear gyrokinetic particle-in-cell code for ion-temperature-gradient-driven modes in theta-pinch geometry. *Phys. Plasmas*, 9(3):898–912, 2002.

[HW06] F. L. Hinton and R. E. Waltz. Gyrokinetic turbulent heating. *Phys. Plasmas*, 13(10):102301, 2006.

[Ich92] S. Ichimaru. *Statistical Plasma Physics, Volume I: Basic Principles (Frontiers in Physics)*. Basic Books, 1992.

[Ido14] Yasuhiro Idomura. Full-f gyrokinetic simulation over a confinement time. *Phys. Plasmas*, 21(2), 2014.

[IIK+08] Yasuhiro Idomura, Masato Ida, Takuma Kano, Nobuyuki Aiba, and Shinji Tokuda. Conservative global gyrokinetic toroidal full-f five-dimensional Vlasov simulation. *Comp. Phys. Comm*, 179(6):391–403, 2008.

[IITV07] Y. Idomura, M. Ida, S. Tokuda, and L. Villard. New conservative gyrokinetic full-f Vlasov code and its comparison to gyrokinetic [delta]f particle-in-cell code. *J. Comput. Phys.*, 226(1):244 – 262, 2007.

[INJ14] Y. Idomura, M. Nakata, and S. Jolliet. Progress of full- f gyrokinetic simulation toward reactor relevant numerical experiments. *Plasma and Fusion Research*, 9:3503028, 2014.

[ITE99] ITER Physics Expert Group on Confinement. Chapter 2: Plasma confinement and transport. *Nuclear Fusion*, 39(12):2175, 1999.

[ITKW01] Y. Idomura, S. Tokuda, Y. Kishimoto, and M. Wakatani. Gyrokinetic theory of drift waves in negative shear tokamaks. *Nuclear Fusion*, 41(4):437, 2001.

[IWS06] Yasuhiro Idomura, Tomo-Hiko Watanabe, and Hideo Sugama. Kinetic simulations of turbulent fusion plasmas. *Comptes Rendus Physique*, 7(6):650–669, 2006.

[JBA+07] S. Jolliet, A. Bottino, P. Angelino, R. Hatzky, T. M. Tran, B.F. Mcmillan, O. Sauter, K. Appert, Y. Idomura, and L. Villard. A global collisionless PIC code in magnetic coordinates. *Comp. Phys. Comm*, 177(5):409 – 425, 2007.

[JDKR00] F. Jenko, W. Dorland, M. Kotschenreuther, and B.N. Rogers. Massively parallel Vlasov simulation of electromagnetic drift-wave turbulence. *Comp. Phys. Comm*, (125):196–209, 2000.

[Kro12] John A. Krommes. The gyrokinetic description of microturbulence in magnetized plasmas. *Annual Review of Fluid Mechanics*, 44:175–201, 2012.

[Lan36] L.D. Landau. The transport equation in the case of Coulomb interactions. *Phys. Z. Sowj. Union*, 10, 1936.

[LB58] A. Lenard and I.B. Bernstein. Plasma oscillations with diffusion in velocity space. *Physical Review*, 112(5), 1958.

[LBD+08] X. Lapillonne, S. Brunner, T. Dannert, S. Jolliet, A. Marinoni, L. Villard, T. Gorler, F. Jenko, and F. Merz. Effects of geometry on linear and non-linear gyrokinetic simulations, and development of a global version of the GENE code. *AIP conference proceedings, Joint Varenna-Lausanne International Workshop on Theory of Fusion Plasmas*, 1069:289–294, 2008.

[LCG11] G. Latu, N. Crouseilles, and V. Grandgirard. Parallel bottleneck in the Quasineutrality solver embedded in GYSELA. INRIA Research Report RR-7595, April 2011.

[LCGS07] G. Latu, N. Crouseilles, V. Grandgirard, and E. Sonnendrücker. Gyrokinetic semi-lagrangian parallel simulation using a hybrid OpenMP/MPI programming. In *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, volume 4757 of *Lecture Notes in Computer Science*, pages 356–364, Paris, France, 2007. Springer.

[Lee83] W. W. Lee. Gyrokinetic approach in particle simulation. *Physics of Fluids*, 26(2):556–562, 1983.

[LGA+12] G. Latu, V. Grandgirard, J. Abiteboul, M. Bergot, N. Crouseilles, X. Garbet, Ph. Ghendrih, M. Mehrenberger, Y. Sarazin, H. Sellama, E. Sonnendrücker, and D. Zarzoso. Accuracy of unperturbed motion of particles in a gyrokinetic semi-lagrangian code. INRIA Research Report RR-8054, Sep 2012.

[LGA+14] G. Latu, V. Grandgirard, J. Abiteboul, N. Crouseilles, G. Dif-Pradalier, X. Garbet, P. Ghendrih, M. Mehrenberger, E. Sonnendrücker, and Y. Sarazin. Improving conservation properties of a 5D gyrokinetic semi-lagrangian code. *The European Physical Journal D*, 68(11), 2014.

[LGC+11] G. Latu, V. Grandgirard, N. Crouseilles, R. Belaouar, and E. Sonnendrücker. Some parallel algorithms for the Quasineutrality solver of GYSELA. INRIA Research Report RR-7591, April 2011.

[LGCDP12] G. Latu, V. Grandgirard, N. Crouseilles, and G. Dif-Pradalier. Scalable Quasineutral solver for gyrokinetic simulation. Research Report RR-7611, INRIA, May 2012.

[LHL+98] Z. Lin, T. S. Hahm, W. W. Lee, W. M. Tang, and R. B. White. Turbulent transport reduction by Zonal Flows: Massively parallel simulations. *Science*, 281(5384):1835–1837, 1998.

[LMG+10] X. Lapillonne, B. F. McMillan, T. Görler, S. Brunner, T. Dannert, F. Jenko, F. Merz, and L. Villard. Nonlinear quasisteady state benchmark of global gyrokinetic codes. *Physics of Plasmas*, 17(11), 2010.

[MJT+08] B. F. McMillan, S. Jolliet, T. M. Tran, L. Villard, A. Bottino, and P. Angelino. Long global gyrokinetic simulations: Source terms and particle noise control. *Physics of Plasmas*, 15(5):052308, 2008.

[MJT+09] B. F. McMillan, S. Jolliet, T. M. Tran, L. Villard, A. Bottino, and P. Angelino. Avalanchelike bursts in global gyrokinetic simulations. *Phys. Plasmas*, 16(2):022310, 2009.

[MLB+10] B. F. McMillan, X. Lapillonne, S. Brunner, L. Villard, S. Jolliet, A. Bottino, T. Görler, and F. Jenko. System size effects on gyrokinetic turbulence. *Phys. Rev. Lett.*, 105:155001, Oct 2010.

[MOT77] W. M. Manheimer, E. Ott, and W. M. Tang. Anomalous electron-ion energy exchange from the trapped electron mode. *Phys. Fluids*, 20(5):806, 1977.

[MV07] M. Mehrenberger and E. Violard. A Hermite-type adaptive semi-Lagrangian scheme. *International Journal of Applied Mathematics and Computer Science*, 17(3):329–334, 2007.

[NFA+08] R. Nazikian, G. Y. Fu, M. E. Austin, H. L. Berk, R. V. Budny, N. N. Gorelenkov, W. W. Heidbrink, C. T. Holcomb, G. J. Kramer, G. R. McKee, M. A. Makowski, W. M. Solomon, M. Shafer, E. J. Strait, and M. A. Van Zeeland. Intense Geodesic Acousticlike Modes driven by suprathermal ions in a Tokamak plasma. *Phys. Rev. Lett.*, 101:185001, Oct 2008.

[NHD+05] W. M. Nevins, G. W. Hammett, A. M. Dimits, W. Dorland, and D. E. Shumaker. Discrete particle noise in particle-in-cell simulations of plasma microturbulence. *Phys. Plasmas*, 12(12):122305, 2005.

[NY99] T. Nakamura and T. Yabe. Cubic Interpolated Propagation scheme for solving the hyper-dimensional Vlasov-Poisson equation in phase space. *Comp. Phys. Comm*, (120):122–154, 1999.

[Ott11] M. Ottaviani. An alternative approach to field-aligned coordinates for plasma turbulence simulations. *Physics Letters A*, 375(15):1677 – 1685, 2011.

[PC08] F. I. Parra and P. J. Catto. Limitations of gyrokinetics on transport time scales. *Plasma Phys. Control. Fusion*, 50(6):065014, 2008.

[PC10] F. I. Parra and P. J. Catto. Transport of momentum in full f gyrokinetics. *Phys. Plasmas*, 17(5):056106, 2010.

[PDM+07] E.J. Doyle (Chair Transport Physics), W.A. Houlberg (Chair Confinement Database, Modelling), Y. Kamada (Chair Pedestal, Edge), V. Mukhovatov (co Chair Transport Physics), T.H. Osborne (co Chair Pedestal, Edge), A. Polevoi (co Chair Confinement Database, Modelling), G. Bateman, J.W. Connor, J.G. Cordey (retired), T. Fujita, X. Garbet, T.S. Hahm, L.D. Horton, A.E. Hubbard, F. Imbeaux, F. Jenko, J.E. Kinsey, Y. Kishimoto, J. Li, T.C. Luce, Y. Martin, M. Ossipenko, V. Parail, A. Peeters, T.L. Rhodes, J.E. Rice, C.M. Roach, V. Rozhansky, F. Ryter, G. Saibene, R. Sartori, A.C.C. Sips, J.A. Snipes, M. Sugihara, E.J. Synakowski, H. Takenaga, T. Takizuka, K. Thomsen, M.R. Wade, H.R. Wilson, ITPA Transport Physics Topical Group, ITPA Confinement Database, Modelling Topical Group, ITPA Pedestal, and Edge Topical Group. Chapter 2: Plasma confinement and transport. *Nuclear Fusion*, 47(6):S18, 2007.

[PGM02] D.Ya Petrina, V.I Gerasimenko, and P.V. Malyshev. *Mathematical foundations of classical statistical mechanics: continuous system.* Taylor and Francis inc., advanced studies in contemporary mathematics edition, 2002.

[PL93] S.E Parker and W.W. Lee. A fully nonlinear characteristic method for gyrokinetic simulation. *Phys. Fluids B*, 5(1):77–86, 1993.

[PS04] A. G. Peeters and D. Strintzi. The effect of a uniform radial electric field on the toroidal ion temperature gradient mode. *Phys. Plasmas*, 11(8):3748–3751, 2004.

[RH98] M.N Rosenbluth and F.L Hinton. Poloidal flow driven by Ion-Temperature-Gradient turbulence in tokamaks. *Phys. Rev. Lett.*, 80(4), 1998.

[RLRG15] F. Rozar, G. Latu, J. Roman, and V. Grandgirard. Toward memory scalability of GYSELA code for extreme scale computers. *Concurrency and computation: Practice and Experience*, 27(4):994–1009, 2015.

[RSM+15] F. Rozar, G. Steiner, Ch. Latu, M. Mehrenberger, V. Grandgirard, J. Bigot, T. Cartier-Michaud, and J. Roman. Optimization of the gyroaverage operator based on Hermite interpolation. In *CEMRACS 2014*, volume submitted of *ESAIM: Proc.*, Luminy, France, 2015.

[SC91] A. Staniforth and J. Côté. Semi-Lagrangian integration schemes for atmospheric models - A review. *Monthly Weather Review*, (119):2206–2223, 1991.

[SCM+07] M. Shimada, D.J. Campbell, V. Mukhovatov, M. Fujiwara, N. Kirneva, K. Lackner, M. Nagami, V.D. Pustovitov, N. Uckan, J. Wesley, N. Asakura, A.E. Costley, A.J.H. Donné, E.J. Doyle, A. Fasoli, C. Gormezano, Y. Gribov, O. Gruber, T.C. Hender, W. Houlberg, S. Ide, Y. Kamada, A. Leonard, B. Lipschultz, A. Loarte, K. Miyamoto, V. Mukhovatov, T.H. Osborne, A. Polevoi, and A.C.C. Sips. Chapter 1: Overview and summary. *Nuclear Fusion*, 47(6):S1, 2007.

[SDD96] R.D. Sydora, V.K. Decyk, and J.M. Dawson. Fluctuation-induced heat transport results from a large global 3D toroidal particle simulation model. *Plasma Phys. Control. Fusion*, 38(12A):A281, 1996.

[SDPZ+09] Y. Sarazin, G. Dif-Pradalier, D. Zarzoso, X. Garbet, Ph. Ghendrih, and V. Grandgirard. Entropy production and collisionless fluid closure. *Plasma Physics and Controlled Fusion*, 51(11):115003, 2009.

[SG10] B. Schroeder and G.A. Gibson. A large-scale study of failures in High-Performance Computing systems. *IEEE Transactions on Dependable and Secure Computing*, 7(4), 2010.

[SGA+10] Y. Sarazin, V. Grandgirard, J. Abiteboul, S. Allfrey, X. Garbet, Ph. Ghendrih, G. Latu, A. Strugarek, and G. Dif-Pradalier. Large scale dynamics in flux driven gyrokinetic turbulence. *Nuclear Fusion*, 50(5):054004, 2010.

[SGA+11] Y. Sarazin, V. Grandgirard, J. Abiteboul, S. Allfrey, X. Garbet, Ph. Ghendrih, G. Latu, A. Strugarek, G. Dif-Pradalier, P.H. Diamond, S. Ku, C.S. Chang, B.F. McMillan, T.M. Tran, L. Villard, S. Jolliet, A. Bottino, and P. Angelino. Predictions on heat transport and plasma rotation from global gyrokinetic simulations. *Nuclear Fusion*, 51(10):103023, 2011.

[SGDP+06a] Y Sarazin, V Grandgirard, G Dif-Pradalier, E Fleurence, X Garbet, Ph Ghendrih, P Bertrand, N Besse, N Crouseilles, E Sonnendrücker, G Latu, and E Violard. Impact of large scale flows on turbulent transport. *Plasma Physics and Controlled Fusion*, 48(12B):B179, 2006.

[SGDP+06b] Y. Sarazin, V. Grandgirard, G. Dif-Pradalier, X. Garbet, and Ph. Ghendrih. Interplay between transport barriers and density gradient. *Phys. Plasmas*, 13(9), 2006.

[SGF+05] Y. Sarazin, V. Grandgirard, E. Fleurence, X. Garbet, Ph. Ghendrih, P. Bertrand, and G. Depret. Kinetic features of interchange turbulence. *Plasma Phys. Control. Fusion*, 47(10):1817–1840, 2005.

[SMC+15] Ch. Steiner, M. Mehrenberger, N. Crouseilles, V. Grandgirard, G. Latu, and F Rozar. Gyroaverage operator for a polar mesh. *The European Physical Journal D*, 69(1):18, 2015.

[SRBG99] E. Sonnendrücker, J. Roche, P. Bertrand, and A. Ghizzo. The semi-Lagrangian method for the numerical resolution of Vlasov equation. *J. Comput. Phys.*, 149(2):201–220, 1999.

[SS10] B. D. Scott and J. Smirnov. Energetic consistency and momentum conservation in the gyrokinetic description of tokamak plasmas. *Phys. Plasmas*, 17(11):112302, 2010.

[SSZ+13a] A. Strugarek, Y. Sarazin, D. Zarzoso, J. Abiteboul, A. S. Brun, T. Cartier-Michaud, G. Dif-Pradalier, X. Garbet, Ph. Ghendrih, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Ion transport barriers triggered by plasma polarization in gyrokinetic simulations. *Plasma Physics and Controlled Fusion*, 55(7):074013, 2013.

[SSZ+13b] A. Strugarek, Y. Sarazin, D. Zarzoso, J. Abiteboul, A. S. Brun, T. Cartier-Michaud, G. Dif-Pradalier, X. Garbet, Ph. Ghendrih, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Unraveling quasiperiodic relaxations of transport barriers with gyrokinetic simulations of tokamak plasmas. *Phys. Rev. Lett.*, 111:145001, Oct 2013.

[Str68] G. Strang. On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.*, 5:506–517, 1968.

[SW06] H. Sugama and T.-H. Watanabe. Collisionless damping of Zonal Flows in helical systems. *Phys. Plasmas*, 13(012501), 2006.

[TBG+13] O. Thomine, J. Bigot, V. Grandgirard, G. Latu, Ch. Passeron, and F. Rozar. An asynchronous writing method for restart files in the GYSELA code in prevision of exascale systems. In *CEMRACS 2012*, volume 43 of *ESAIM: Proc.*, pages 108–116, Luminy, France, 2013.

[TGL+08] P. W. Terry, M. Greenwald, J.-N. Leboeuf, G. R. McKee, D. R. Mikkelsen, W. M. Nevins, D. E. Newman, D. P. Stotler, Task Group on Verification Validation, U.S. Burning Plasma Organization, , and U.S. Transport Task Force. Validation in fusion research: Towards guidelines and best practices. *Phys. Plasmas*, 15(6):062503, 2008.

[VB92] John Villasenor and Oscar Buneman. Rigorous charge conservation for local electromagnetic field solvers. *Computer Physics Communications*, 69(2-3):306 – 316, 1992.

[VBB+10] L. Villard, A. Bottino, S. Brunner, A. Casati, J. Chowdhury, T. Dannert, R. Ganesh, X. Garbet, T. Görler, V. Grandgirard, R. Hatzky, Y. Idomura, F. Jenko, S. Jolliet, S. Khosh Aghdam, X. Lapillonne, G. Latu, B. F. McMillan, F. Merz, Y. Sarazin, T. M. Tran, and T. Vernay. Gyrokinetic simulations of turbulent transport: size scaling and chaotic behaviour. *Plasma Physics and Controlled Fusion*, 52(12):124038, 2010.

[Viled] C. Villani. *Landau damping.* SMF, panoramas et synthèses edition, to be published.

[Wes97] J. Wesson. *Tokamaks.* Okford Science Publications, 1997.

[WHL+07] W. X. Wang, T. S. Hahm, W. W. Lee, G. Rewoldt, J. Manickam, and W. M. Tang. Nonlocal properties of gyrokinetic turbulence and the role of ExB flow shear. *Phys. Plasmas*, 14(7):072306, 2007.

[WS08] R. E. Waltz and G. M. Staebler. Gyrokinetic theory and simulation of turbulent energy exchange. *Phys. Plasmas*, 15(1):014505, 2008.

[WSD+97] R. E. Waltz, G. M. Staebler, W. Dorland, G. W. Hammett, M. Kotschenreuther, and J. A. Konings. A gyro-Landau-fluid transport model. *Phys. Plasmas*, 4(7):2482–2496, 1997.

[XC06] Yong Xiao and Peter J. Catto. Short wavelength effects on the collisionless neoclassical polarization and residual zonal flow leve. *Phys. Plasmas*, 13(10), 2006.

[XR91] X.Q. Xu and M.N. Rosenbluth. Numerical simulation of Ion-Temperature-Gradient-driven modes. *Physics of Fluids B: Plasma Physics*, 3, 1991.

[ZCS96] F. Zonca, L. Chen, and R.A Santoro. Kinetic theory of low-frequency alfvén modes in tokamaks. *Plasma Physics and Controlled Fusion*, 38(11):2011, 1996.

[ZGS+12] D. Zarzoso, X. Garbet, Y. Sarazin, R. Dumont, and V. Grandgirard. Fully kinetic description of the linear excitation and nonlinear saturation of fast-ion-driven geodesic acoustic mode instability. *Phys. Plasmas*, 19(2), 2012.

[ZSG+13] D. Zarzoso, Y. Sarazin, X. Garbet, R. Dumont, A. Strugarek, J. Abiteboul, T. Cartier-Michaud, G. Dif-Pradalier, Ph. Ghendrih, V. Grandgirard, G. Latu, Ch. Passeron, and O. Thomine. Impact of Energetic-Particle-Driven Geodesic Acoustic Modes on turbulence. *Phys. Rev. Lett.*, 110:125002, Mar 2013.

[ZWS02] M. Zerroukat, N. Wood, and A. Staniforth. SLICE: A semi-Lagrangian inherently conserving and efficient scheme for transport problems. *Quarterly Journal of the Royal Meteorological Society*, 128(586):2801–2820, 2002.

[ZWS07] M. Zerroukat, N. Wood, and A. Staniforth. Application of the parabolic spline method (PSM) to a multi-dimensional conservative semi-lagrangian transport scheme (SLICE). *J. Comput. Phys.*, 225(1):935–948, 2007.

GYROKINETIC SIMULATIONS OF MAGNETIC FUSION PLASMAS

by

Virginie Grandgirard, Yanick Sarazin

Abstract. — This lecture presents the gyrokinetic framework and details the various numerical schemes used in nonlinear simulations to compute turbulent transport in magnetic fusion plasmas. The basic features of tokamak magnetic configuration and of fusion plasmas are recalled. Fundamental elements of the gyrokinetic theory are carefully introduced, including the derivation of velocity drifts and of the quasi-neutrality. From the numerical point of view, the main focus is put on the three existing classes of numerical methods, namely Particle-In-Cell, Eulerian and semi-Lagrangian. Their properties are discussed, and their strengths and weaknesses are exhaustively reviewed.

1. Introduction

In magnetic fusion devices, the power gain strongly increases with the energy confinement time. As a matter of fact, the quality of the plasma energy confinement largely determines the size and therefore the cost of a fusion reactor. This confinement time turns out to be mainly governed by the plasma turbulence which develops in such devices – of relative magnitude of a few percents in the hot core – and the associated transport. Understanding its origin and properties in view of its possible control is one of the critical issues in fusion science [**37**]. Such a quest belongs to the more general framework of general studies on turbulence, which is considered as one of the most difficult problems in physics, because of its inherent nonlinear character with many degrees of freedom. Plasma turbulence is rich of at least three additional properties with respect to neutral fluids: (*i*) Multiple fluids (an electron fluid and other ion fluids) with considerably different mobility are coupled through electromagnetic fields and weak collisional interactions. (*ii*) Strong confinement magnetic fields provide highly anisotropic turbulent structures. (*iii*) The inhomogeneities in density, temperature, and magnetic fields place the plasma naturally out of

thermodynamical equilibrium, and tend to excite several micro-instabilities over a wide spectral range. In particular, the toroidal magnetic configuration strongly affects the linear properties of micro-instabilities as well as nonlinearly evolved turbulent structures. Finally, because of the weak collisionality in the hot core of fusion plasmas, which is often said to be collisionless (for thermal particles, the mean free path is expected to be of the order of several kilometers in ITER), kinetic resonances as well as significant deviations from the Maxwellian of the distribution function of the various species and particle orbit effects cannot be ignored. Because of this almost collisionless character, conventional fluid models becomes insufficient, and kinetic descriptions of the plasma in phase space are required. Kinetic models are drastically more demanding in terms of numerical resources than fluid models. In such first-principle descriptions of plasmas, the six dimensional evolution equation for the distribution function – Vlasov or Fokker-Planck equations – is solved for each specie, coupled to the self-consistent equations for the electromagnetic fields, namely Maxwell's equations. These coupled equations are non-linear. No general analytic solution of these equations exists, such that numerical simulations are necessary. Fortunately, as far as turbulent fluctuations are concerned, they develop at much lower typical frequencies than the high frequency cyclotron motion. Therefore, this 6D problem can be restricted to a 5D one by incorporating part of this small scale temporal behavior into the larger scales temporal dynamics of both the distribution function and the fields. The useful part of the distribution function then evolves in a five dimensional phase space generated by four slow variables and an adiabatic invariant. This model is known as the *gyrokinetic* model. In a first approximation, the gyrokinetic distribution function can be thought of as deduced from its six-dimensional kinetic analogue by an average procedure over the fast-varying gyrophase angle. This model consensually provides today's deepest insight on plasma behavior. But even with this dimensional reduction, the task is not easy at all. Solving 5D gyrokinetic equations for each specie reveals extremely challenging. First-principle gyrokinetic codes, which have been developed for this stage, make an intensive use of massively parallel supercomputers and require state-of-the-art high performance computing (HPC). They have greatly benefited from the drastic increase of both hardware capabilities (at present, Petaflop calculators are able to process 10^{15} floating point operations per second) and of new computational techniques based on fast solvers, massive parallelization protocols, etc. These numerical tools have already played an important role in clarifying a number of pending issues in turbulent transport. Enlightening exhaustive reviews on the subject can be found in references [**142, 132, 79, 58**], listed in chronological order.

The development of such gyrokinetic codes would not be possible without a strong collaboration between physicists, mathematicians and now computer

scientists. The numerical schemes, developed for Vlasov equations in 2 to 6 dimensional phase spaces, have evolved all along the last twenty-five years, in direct link with the evolution of HPC resources. Historically, particle in cell methods (PIC) [**16**] have been most popular, and represent widely adopted approaches to numerical simulations of kinetic plasmas. They used to be considered as the most efficient tool to describe plasma dynamics, essentially because they are capable of describing many physical phenomena in the full dimensional case, at relatively small computational costs. However, it is well known that the numerical noise inherent to PIC methods constitutes a strong limiting factor to accurately describe the distribution function in phase space on long time scales. Moreover, the numerical noise only slowly decreases, like $1/\sqrt{N}$, when the number N of particles is increased. To remedy this problem, alternative methods have also been developed. Eulerian methods, in which the Vlasov equation is discretized on a mesh of phase space, are one of those. Among them, the Fourier transform uses fast Fourier transform of the distribution function in phase space. Other methods like the Fourier-Hermite or Hermite transform methods exploit the fact that the Hermite basis can offer some advantage when modeling distribution functions with Gaussian-shaped profiles. However, due to wave-particle resonances which play an important role in the collisionless or weakly collisional regime, fine-scale structures inherently develop in velocity space. This requires the spectral expansion of a large number of Hermite modes to achieve high accuracy. Some of these algorithms use artificial damping to smooth out these fine-scale structures [**124**]. The drawback is that the collisionless nature of the system is lost, leading to distortions of the distribution function [**123**]. Another drawback of such methods is that they are only valid for periodic boundary conditions [**87, 88**]. Consequently, for non-periodic boundary conditions, Gibbs oscillations form at the boundary of the grid and lead to spurious oscillations which can propagate to the entire distribution function. A finite element method has also been proposed [**159, 158**]. Although it is well suited for complex boundary conditions which may arise in many practical applications, the numerical resolution can become cumbersome when dealing with the Vlasov equation in high dimension. Another scheme for the Vlasov equation is the flux corrected transport (FCT) [**18, 17**], or more recently the flux balance method (FBM) [**53**]: the basic idea is to compute the average of the Vlasov equation solution in each cell of the phase space grid by a conservative method. One of the common flaws of these algorithms is the non-preservation of the positivity, which reveals problematic for long simulation runs since numerical oscillations develop. Attempts to overcome this problem of the positive and flux conservative (PFC) methods have been tried [**55**]. Finally, the semi-Lagrangian

method aims at taking advantages of both Lagrangian and Eulerian numerical schemes. It consists of computing the distribution function at each grid point by following the particle trajectories backward in time. To compute the origin of the characteristics, high-order interpolation methods are needed. E. Sonnendrücker et al. proposed the cubic spline reconstruction which gives very good results [**126**], at the expense of the loss of the local character of the reconstruction. Nakamura and Yabe also presented the cubic interpolated propagation (CIP) method based on the approximation of the gradients of the distribution function in order to use a Hermite interpolation [**107**]. This method is very stable but very expensive in terms of computational memory since it requires the storage not only of the distribution function f, but also of its gradients $\nabla_x f$ and $\nabla_v f$. Such a constraint is also present for the Morinishi's scheme [**104**]. As will be detailed in these notes, all the various schemes have both advantages and drawbacks. The diversity of numerical approaches existing now in gyrokinetic codes clearly constitute a strength with regard to the complexity of the physical problem. As a result, benchmarks become crucial, as exemplified by the growing number of papers devoted to this difficult task in the community of gyrokinetic code development.

The reminder of the paper is organized as follows. After a general introduction in section 1, we briefly describe the basic features of magnetic fusion research and of fusion plasmas in section 2. In section 3, we describe a theoretical hierarchy of plasma physics, with a focus on plasma kinetic theory in section 4. The gyrokinetic approach specific to magnetic fusion plasmas is described in section 5. Section 6 is devoted to the description of the numerical methods used in gyrokinetic codes, divided in three categories PIC, Eulerian and Semi-Lagrangian approach. In section 7, some differences among the various methods regarding the treatment of some of the main physical plasma properties are highlighted. Finally, a summary is given in section 8.

2. Brief introduction to plasma fusion

The increase in energy needs and the fact that fossil fuels are running out make indispensable the development of new sources of energies. To produce energy out of matter, it is necessary to carry out a transformation in which, between the initial and final state, a small proportion of the body mass involved disappears. This mass defect may then be found in the form of energy through the well-known formula $E = mc^2$, where E is the produced energy, m the mass that has disappeared and c the speed of light. Chemical reactions involve atoms or molecules and their electrons. As such, the corresponding energies lie in the tens of electron-Volt range (the ionisation energy of hydrogen is 13.6eV). As far as nuclear reactions are concerned, there exist two main

types, both in the range of Mega electron-Volt[(1)]. The fission reaction consists in splitting the nucleus of a sufficiently heavy atom (such as the uranium or plutonium atoms) to make lighter atoms. Conversely, very light atomic nuclei are joined together in the fusion reaction to build heavier atoms. Fission is at work in present nuclear reactors while fusion is still at the stage of research. These researches on controlled fusion are conducted on the most accessible fusion reaction which is the Deuterium-Tritium reaction. In this reaction two nuclei of Deuterium and Tritium (the heavy isotopes of hydrogen, respectively with 1 and 2 neutrons) combine into an alpha particle (Helium nucleus) and a neutron.

$$D + T \rightarrow \quad {}^4He\,(3.52MeV) + n\,(14.06MeV) \tag{1}$$

The total kinetic energy retrieved from the reaction is about $17.6MeV$, one fifth carried out by the Helium nucleus, the remainder by the neutron (this ratio comes from the conservation of momentum during the reaction: $m_{He}v_{He} = m_n v_n$, which implies that $E_{He} = \frac{1}{2}m_{He}v_{He}^2 = \frac{m_{He}}{m_n}\frac{v_{He}^2}{v_n^2}E_n = \frac{1}{4}E_n$). Deuterium is widely available from water, since it constitutes 0.015% of all hydrogen atoms. Tritium is radioactive with a relatively short half-life of 12.3 years, so that its natural occurrence is negligible. It has to be produced. Fortunately, the neutron which is created by the reaction (1) can be used to breed new tritium out of lithium directly in the wall of the reaction vessel. In order to overcome the electrostatic repulsion between them, the nuclei must have a temperature greater than a hundred million degrees[(2)] or about 10keV ($1eV \approx 11\,600\ ^oC$). At such temperatures, electrons are completely detached from the nuclei, such that the hot "gas" is no longer composed of neutral atoms, but of positively (ions) and negatively (electrons) charged particles. This fourth state of matter is called plasma. Due to the presence of these charge carriers, the plasma is electrically conductive so that it strongly responds to electromagnetic fields. Magnetic confinement fusion attempts to

(1) The ratio between the energy retrieved from chemical and nuclear reactions directly relates to the characteristic interaction lengths of the underlying forces: the Coulomb interaction links electrons to the nucleus on distances $r^{Coulomb}$ of the order of a few tens of Angström ($1\AA = 10^{-10}m$), while the nuclear force (also called residual strong force) binds neutrons and protons in nuclei at distances $r^{nuclear}$ of the order of one Fermi ($1Fermi = 10^{-15}m$). Since the potential energy associated to each of these central forces (there also exists a weak noncentral component of the nuclear force) decays like $1/r$, with r the distance between the two interacting particles, it comes: $E_{pot}^{Coulomb}/E_{pot}^{nuclear} \sim r^{nuclear}/r^{Coulomb} \sim 10^{-6}$. The precise ratio involves the interaction constants for each of these forces.

(2) As a matter of fact, the minimal distance between two colliding thermal particles at such a temperature is about $10^{-13}m$, still much too large for the nuclear force to overcome the repulsive Coulomb force. Fusion reactions take place thanks to the tunnel effect of quantum mechanics.

create the conditions needed for fusion energy production by confining the plasma with strong magnetic fields. Any static and homogeneous magnetic field restricts the perpendicular (to the magnetic field lines) motion of charged particles to gyro-orbits. Parallel to the field lines, particles move freely (up to magnetic, and possibly electric, mirror effects). In order to keep the volume of the container finite, the field lines are usually bent to a torus. It turns out that charged particles immersed in a curved magnetic field are subject to drifts. Should the magnetic field be purely toroidal (i.e. with circular field lines), these drifts would be strong enough to prevent any confinement on long time durations in a volume of acceptable size [**149**]. This problem is solved by twisting the magnetic field lines, thanks to the creation of an additional poloidal component of the magnetic field. The average pitch of the field line, defined by the ratio of toroidal revolutions per poloidal revolution of a field line, corresponds to the so-called safety factor q. If q is not a rational number, the field line covers a so-called flux surface. The field lines at different radial positions inside the toroidal plasma vessel define nested flux surfaces. The two most important concepts for magnetic confinement fusion essentially differ in the way the twisting of the field lines is achieved. In stellarators, as Wendelstein 7-X in construction at Greifswald in Germany, the twisted magnetic field that is needed for confinement is entirely generated by the external field coils. In tokamaks (Fig. 1), the set of external field coils produces a purely toroidal magnetic field. The poloidal component of the magnetic field is created by the strong electric current induced in the plasma along the toroidal direction. Most of fusion experiments in the world, including the International Thermonuclear Experimental Reactor (ITER), now under construction at Cadarache, France, follow this concept.

In view of producing energy, the ratio between the power from fusion reactions P_{fusion} and the external additional power P_{add} supplied to the plasma by the heating systems must be greater than 1. This ratio is called the quality factor $Q = P_{fusion}/P_{add}$. The present record of fusion power in D-T plasmas has been achieved in the European tokamak JET, with 16 Mega-Watt produced corresponding to $Q = 0.64$. In practice, reaching ignition (condition under which the plasma can be maintained by fusion reactions without external energy input) is not mandatory for viable power plant. Finite values close to $Q = 20$ are often thought to be enough for the economic viability of a reactor. In ITER, whose objective is to demonstrate the practical feasibility of a fusion-based production of energy on an industrial level, the target is $Q = 5$ in steady conditions, and $Q = 10$ during transients. In this latter case, a significant fraction of the heating power of the plasma, namely 2/3, comes from the plasma itself via fusion reactions.

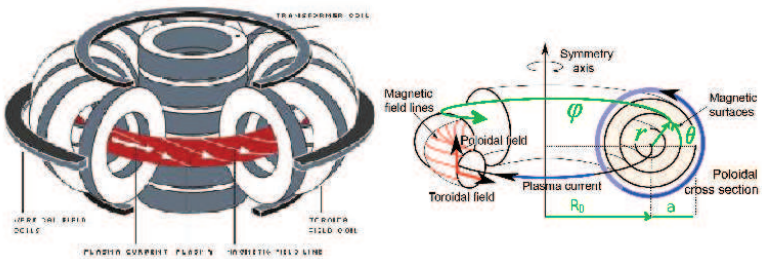


FIGURE 1. Left: Schematic view of the coil system and magnetic field of a tokamak. Right: Corresponding idealized toroidal magnetic geometry and its adopted notations.

The Lawson criterion is the constraint for reaching self-plasma heating (the *ignition* condition), assuming that the power of Helium ashes is entirely deposited on fuel ions, namely Deuterium and Tritium ions. It is reached when this fraction P_α of the fusion power counterbalances all power losses W/τ_E, with W the plasma internal energy and τ_E the energy confinement time. In this case, the energy confinement time is equal to the Lawson time $\tau_{Lawson} = W/P_\alpha$. Lawson criterion states that the triple product $nT\tau_E$ (n is the density, T the temperature) must be larger than a critical value of the order of $3\,10^{21} m^{-3} keV\, s^{-1}$. In other words, to be able to produce energy from fusion reactions, a sufficiently hot (T) and dense (n) plasma must be confined efficiently (τ_E). The difficulty resides in obtaining the three parameters simultaneously. Indeed, for example, when density is increased by means of gas injection, or when temperature is increased additional heating, the energy confinement time degrades in tokamak plasmas. In magnetic fusion devices, the maximal achievable plasma density n (number of particles per volume unit) is theoretically upper bounded by the strongest achievable magnetic field. Indeed, confinement is effective provided the thermal energy density nT remains lower than the magnetic energy density $B^2/2\mu_0$, with μ_0 the permeability of free space. The ratio of the former on the latter defines the dimensionless beta parameter β. As a matter of fact, tokamak plasmas are subject to large scale instabilities well below the critical value $\beta = 1$ (β is typically of a few percents in tokamaks), which appear redhibitory for the confinement. In such devices, densities are of the order of a few $10^{20} m^{-3}$. Such densities are much lower, typically by 5 orders of magnitudes, than the density of air at normal pressure and temperature. Given the very large

temperatures of such plasmas, the pressure is however of a few atmospheres. For these reasons, much effort is devoted to scenarios aiming at improving the confinement time τ_E. As seen on figure 2 a small increase of τ_E can have a big impact on the fusion performances. This confinement time, which is basically

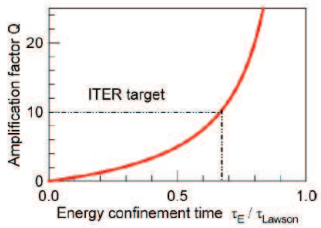


FIGURE 2. Quality factor Q increases with energy confinement time τ_E. Q is proportional to $\tau_E = \tau_E/(\tau_{\rm Lawson} - \tau_E)$.

a thermal relaxation time, is mainly governed by conductive losses. It turns out that these losses are essentially of turbulent nature. *Understanding turbulent transport in magnetized plasma is one of the key open issues in magnetic confinement fusion research.*

3. Theoretical Hierarchy of Plasma Physics

The self consistent treatment of plasma evolution requires to solving both the dynamics of the electromagnetic fields, governed by Maxwell's equations, and the plasma response to these fields. Plasma response can be described by a hierarchy of models. The basic and most precise model of this hierarchy is the microscopic description. At this level, the plasma is an ensemble of several species of charged particles (for neutral plasmas, these are electrons with charge $q_e = -e$ and mass m_e and at least one species of ions with charge $q_i = Z_i e$ and mass m_i) characterized by their positions $\mathbf{x}$ and velocities $\mathbf{v}$. Their individual motion is governed by Newton's equation, the only relevant force being the Lorentz one (Coulomb plus Laplace). For very fast particles, the relativistic formulation should be retained. As shown *e.g.* by Poincaré, the minimal phase space where all the possible trajectories of any dynamical system are represented is six-dimensional: 3D in configuration space (referred as *real space*) and 3D in *velocity space* (or momentum space in the general case). In the following, the vector $\mathbf{x} = (x_1, x_2, x_3)$ refers to position in real

space and $\mathbf{v} = (v_1, v_2, v_3)$ in velocity space. The subscript s refers to the various species, with charge q_s and mass m_s. SI units are adopted. Newton's law then reads as follows:

$$m_s \frac{\mathrm{d}\mathbf{v}_s^i}{\mathrm{d}t} = q_s(\mathbf{E} + \mathbf{v}_s^i \times \mathbf{B}) \tag{2}$$

with i ($1 \leq i \leq N_s$) any particle of species s. Here, $\mathbf{E}$ is the electric field and $\mathbf{B}$ the magnetic field. We introduce the scalar ϕ and vector $\mathbf{A}$ potential such that $\mathbf{E} = -\nabla\phi - \partial_t \mathbf{A}$. The dynamics of these fields obey Maxwell's equations:

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \qquad \text{Gauss} \tag{3}$$

$$-\frac{1}{c^2}\frac{\partial \mathbf{E}}{\partial t} + \nabla \times \mathbf{B} = \mu_0 \mathbf{j} \qquad \text{Ampère} \tag{4}$$

$$\nabla \cdot \mathbf{B} = 0 \qquad \text{flux conservation} \tag{5}$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0 \qquad \text{Faraday} \tag{6}$$

The fields depend on the charge density $\rho(\mathbf{x}, t)$ and current density $\mathbf{j}(\mathbf{x}, t)$ of the plasma. Let the plasma consists of N_s particles of s species at positions $(\mathbf{x}_s^i, \mathbf{v}_s^i)$, with $i = 1, \cdots, N_s$. Considering point-like particles, the number density of species s is

$$n_s(\mathbf{x}, t) = \sum_{i=1}^{N_s} \delta(\mathbf{x} - \mathbf{x}_s^i(t))$$

The total charge density is then

$$\rho(\mathbf{x}, t) = \sum_s q_s n_s(\mathbf{x}, t) \tag{7}$$

The current density $\mathbf{j}$ is obtained from the mean velocities

$$n_s \mathbf{v}_s(\mathbf{x}, t) = \sum_{i=1}^{N_s} \mathbf{v}_s^i(t)\, \delta(\mathbf{x} - \mathbf{x}_s^i(t))$$

as

$$\mathbf{j}(\mathbf{x}, t) = \sum_s q_s \mathbf{v}_s(\mathbf{x}, t)\, n_s(\mathbf{x}, t) \tag{8}$$

The complete description of plasma dynamics is given by Newton's law (Eq. (2)) and Maxwell's equations (Eqs. (3)-(6)). Such a microscopic approach requires to solve N coupled equations in 6D phase space for the $N = \sum_s N_s$ particles of the system. Considering fusion plasmas of about $10^{20} m^{-3}$ ions and electrons, this *many-body model* still remains out of reach for nowadays supercomputers.

Next, the *kinetic models* proceed from the statistical description of the plasma: the particle distribution function $f_s(\mathbf{x}, \mathbf{v}, t)$ is introduced, counting the number of particles of species s in the infinitesimal volume $\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}$ of the 6D phase space centered on the position $(\mathbf{x}, \mathbf{v})$. Although the precise locations of individual particles are lost, the detailed knowledge of particle motion is required to evaluate the dynamics of f_s. In this sense kinetic theory is still microscopic, even though statistical averages are employed.

The last step of the hierarchy consists in further reducing the degrees of freedom of the kinetic theory by integrating over the 3D velocity space. Then, a hierarchy of so-called fluid moments can be constructed by weighting the distribution function f_s by tensors of the velocity of rank k before integration. The first moments are the density, the flow velocity (vector), and the pressure (tensor of rank 2). These are macroscopic quantities depending on space and time only. In such *macroscopic or fluid models*, the knowledge of the individual particle motions is no longer required. The major difficulty of the fluid approach resides in the fact that each moment of order k depends on the moment $(k+1)$, such that an additional hypothesis is required to close the system of otherwise infinite set of fluid equations. For weakly collisional media such as fusion plasmas, no satisfactory *closure* exists, conversely to neutral fluids for which the fluid description remains fully appropriate. Still, solving 3D fluid equations is certainly the most convenient and fastest way to solve the problem given the set of well established numerical techniques and the wealth of results obtained in the field of fluid turbulence. However, it is known that the stability threshold given by fluid equations is lower than the actual (kinetic) value [**49**]. It is also well established that fluid description usually overestimates turbulent fluxes [**49**]. This discrepancy comes partly from the resonant interactions between waves and particles (Landau resonances), which cannot be properly accounted for by fluid models. Also, large scale axi-symmetric flows of fusion plasmas, known as zonal flows, which play an important role in regulating turbulence, tend to be over-damped in fluid models.

Several attempts have been made to propose optimized closures, in view of accounting as much as possible for some kinetic effects, such as Landau resonances (see e.g. [**68**]-[**145**]-[**119**]). Comparing fluid and kinetic simulations provides a stringent test of these closure schemes. This exercise has revealed much more difficult than expected, pointing out irreconcilable discrepancies [**49**]-[Y. Sarazin, V. Grandgirard, G. Dif-Pradalier et al., *Plasma Phys. Control. Fusion* **48** (2006) B179]. Fortunately, the dramatic increase of numerical resources and performances now enables the direct treatment of kinetic models, or at least of the gyro-kinetic ones (see section 5).

4. Plasma kinetic theory

4.1. Liouville theorem. — Let us consider a canonical Hamiltonian system: $\mathbf{q} = \{\mathbf{q}_i\}_{1\leq i\leq N}$ denote the generalized coordinates, $\mathbf{p} = \{\mathbf{p}_i\}_{1\leq i\leq N}$ their conjugate momenta and $H(\{\mathbf{q}_i, \mathbf{p}_i\})$ the Hamiltonian. Let also the phase-space distribution $\mathcal{D}_N(\mathbf{q}, \mathbf{p})$ determine the probability $\mathcal{D}_N(\mathbf{q}, \mathbf{p})\, \mathrm{d}^N q\, \mathrm{d}^N p$ that the system is in the infinitesimal volume of phase-space $\mathrm{d}^N q\ \mathrm{d}^N p$ around the position $(\mathbf{q}, \mathbf{p})$. The equilibrium statistical mechanics of such a canonical Hamiltonian system is governed by Liouville's theorem, which states that the *phase-space distribution function is constant along the trajectories of the system* – that is, the density $\mathcal{D}_N$ of N system points in the vicinity of a given system point travelling through phase-space is constant in time. The proof directly follows from the continuity equation fulfilled by $\mathcal{D}_N$, namely:

$$\frac{\partial \mathcal{D}_N}{\partial t} + \sum_{i=1}^{N} \left[\frac{\partial}{\partial \mathbf{q}_i} \left(\mathcal{D}_N \frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} \right) + \frac{\partial}{\partial \mathbf{p}_i} \left(\mathcal{D}_N \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) \right] = 0 \tag{9}$$

and the fact that the generalized velocity field $(\dot{\mathbf{p}}, \dot{\mathbf{q}})$ in phase space is divergenceless, as a direct consequence of the Hamilton equations of motion

$$\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} = \frac{\partial H}{\partial \mathbf{p}_i}$$
$$\frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} = -\frac{\partial H}{\partial \mathbf{q}_i}$$

Developing eq. (9) leads to

$$\frac{\partial \mathcal{D}_N}{\partial t} + \sum_{i=1}^{N} \left(\frac{\partial \mathcal{D}_N}{\partial \mathbf{q}_i} \frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} + \frac{\partial \mathcal{D}_N}{\partial \mathbf{p}_i} \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) + \mathcal{D}_N \sum_{i=1}^{N} \left[\frac{\partial}{\partial \mathbf{q}_i} \left(\frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} \right) + \frac{\partial}{\partial \mathbf{p}_i} \left(\frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) \right] = 0$$

The last sum vanishes in virtue of Hamilton's equations. The resulting equation states that the convective derivative of the density $\mathrm{d}\mathcal{D}_N / \mathrm{d}t$ is equal to 0:

$$\frac{\mathrm{d}\mathcal{D}_N}{\mathrm{d}t} = \frac{\partial \mathcal{D}_N}{\partial t} + \sum_{i=1}^{N} \left(\frac{\partial \mathcal{D}_N}{\partial \mathbf{q}_i} \frac{\mathrm{d}\mathbf{q}_i}{\mathrm{d}t} + \frac{\partial \mathcal{D}_N}{\partial \mathbf{p}_i} \frac{\mathrm{d}\mathbf{p}_i}{\mathrm{d}t} \right) = 0 \tag{10}$$

Integrating by parts Liouville's equation (10) over the variables leads to a chain of $(N-1)$ equations where the j-th equation connects the j-th and $(j+1)$-th particle density probability functions, with $\mathcal{D}_j = \mathcal{D}_j(\mathbf{q}_1 \cdots \mathbf{q}_j, \mathbf{p}_1 \cdots \mathbf{q}_j)$. The truncation of this BBGKY hierarchy of equations is a common starting point for many applications of kinetic theory. In particular, truncation at the first equation or the first two equations can be used to derive classical Boltzmann equations and their first order corrections. This derivation is out of the scope of this paper, and can be found in *e.g.* [**113**]. In the following, we focus on

the kinetic description of the plasma turbulence and more precisely on the numerical solving of *Boltzmann equation* and of its collisionless form, namely *Vlasov equation*.

4.2. Vlasov-Maxwell and Vlasov-Poisson system. — In a high temperature fusion plasmas, the kinetic energy is much larger than the average potential (Coulombian) energy between particles, such that particles are weakly coupled. In weakly coupled plasmas, multiple particle correlations involving more than two particles are neglected. Let us denote $f_s \equiv f_s(\mathbf{x}, \mathbf{v}, t)$ the 6D distribution function, which represents the density of particles species s in the phase space $(\mathbf{x}, \mathbf{v})$ at time t. Its evolution is governed by Boltzmann equation

$$\frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \frac{\partial f_s}{\partial \mathbf{x}} + q_s \left(\mathbf{E} + \mathbf{v} \times \mathbf{B} \right) \frac{\partial f_s}{\partial \mathbf{v}} = C(f_{s'}, f_s) \tag{11}$$

In the collisionless limit, the collision operator $C(f_{s'}, f_s)$ is neglected, and equation (11) reduces to the Vlasov equation (or collisionless Boltzmann equation)

$$\frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \frac{\partial f_s}{\partial \mathbf{x}} + q_s \left(\mathbf{E} + \mathbf{v} \times \mathbf{B} \right) \frac{\partial f_s}{\partial \mathbf{v}} = 0 \tag{12}$$

The electromagnetic fields $\mathbf{E}$ and $\mathbf{B}$ are governed by Maxwell's equations

$$\nabla \cdot \mathbf{E} = \frac{1}{\epsilon_0} \sum_s q_s n_s \tag{13}$$

$$-\frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t} + \nabla \times \mathbf{B} = \mu_0 \sum_s \mathbf{j}_s \tag{14}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{15}$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0 \tag{16}$$

where the source terms, the density $n_s(\mathbf{x}, t)$ and the current density $\mathbf{j}_s(\mathbf{x}, t)$, correspond to the first two velocity moments of f_s

$$n_s(\mathbf{x}, t) = q_s \int f_s(\mathbf{x}, \mathbf{v}, t)\, \mathrm{d}\mathbf{v}$$
$$\mathbf{j}_s(\mathbf{x}, t) = q_s \int f_s(\mathbf{x}, \mathbf{v}, t)\, \mathbf{v}\, \mathrm{d}\mathbf{v}$$

The Vlasov-Maxwell system, equations (12)-(16), provides the kinetic framework for describing high temperature collisionless plasmas. The Vlasov-Poisson model is obtained by neglecting the time fluctuations of the magnetic field $\mathbf{B}$.

4.3. Advective and conservative forms of Vlasov equation. — In the following, the subscript s of the considered species is dropped for clarity. Let us denote $\mathbf{Z}$ the six-dimensional phase space vector $\mathbf{Z} = \{\mathbf{x}, \mathbf{v}\}$ and $\nabla_{(\mathbf{x},\mathbf{v})}$ the six-dimensional phase-space derivative:

$$\nabla_{(\mathbf{x},\mathbf{v})} = \{\nabla_{\mathbf{x}}, \nabla_{\mathbf{v}}\} = \left\{\frac{\partial}{\partial \mathbf{x}}, \frac{\partial}{\partial \mathbf{v}}\right\} = \frac{\partial}{\partial \mathbf{Z}} \tag{17}$$

Vlasov equation (12) translates into an *advection equation* in phase-space of the function $f : \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}$ (with $d = 6$)

$$\frac{\partial}{\partial t} f(\mathbf{Z}, t) + \mathbf{U}(\mathbf{Z}, t) \cdot \nabla_{(\mathbf{x},\mathbf{v})} f(\mathbf{Z}, t) = 0 \tag{18}$$

Here, we have introduced the six-dimensional phase-space flow $\mathbf{U} : \mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}$, defined as the total time derivative of $\mathbf{Z}$:

$$\mathbf{U}(\mathbf{Z}, t) = \{\mathbf{U}_{\mathbf{x}}, \mathbf{U}_{\mathbf{v}}\} = \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} = \left\{\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}, \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\right\} = \{\mathbf{v}, \mathbf{E} + \mathbf{v} \times \mathbf{B}\} \tag{19}$$

Let us now consider the differential system

$$\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} = \mathbf{U}(\mathbf{Z}(t), t) \tag{20}$$

$$\mathbf{Z}(s) = \mathbf{z} \tag{21}$$

Definition 1. — *The solutions of equation* (20) *are called the characteristics of the advection equation* (18). *Let us denote* $\mathbf{Z}(t; \mathbf{z}, s)$ *the solution of* (20)-(21)

The existence, uniqueness and regularity of the solutions of the previous differential equations (20)-(21) derive from the classical theorem of differential equation theory, which reads as follows (its proof can be found in *e.g.* [**3**]):

Theorem 1. — *Let us assume* $\mathbf{U} \in C^{k-1}(\mathbb{R}^d \times [0, T])$, $\nabla \mathbf{U} \in C^{k-1}(\mathbb{R}^d \times [0, T])$ *with* $k \in \mathbb{N}, k > 1$ *and that, for* $\kappa \geq 1$*:*

$$|\mathbf{U}(\mathbf{z}, t)| \leq \kappa(1 + |\mathbf{z}|) \qquad \forall t \in [0, T] \quad \forall \mathbf{z} \in \mathbb{R}^d$$

then $\forall s \in [0, T]$ *and* $\mathbf{z} \in \mathbb{R}^d$*, there exists a unique solution* $\mathbf{Z} \in C^k([0, T] \times \mathbb{R}^d \times [0, T])$ *of equations* (20)-(21).

Also, a few properties of the characteristics are useful to some of the various numerical approaches described in the following sections. These properties are summarized in the following proposition 1 (the proof can be found in appendix C):

Proposition 1. — *Under the same assumptions as for theorem 1, the following properties hold:*

a) $\forall t_1, t_2, t_3 \in [0, T]$ *and* $\mathbf{z} \in \mathbb{R}^d$

$$\mathbf{Z}(t_3; \mathbf{Z}(t_2; \mathbf{z}, t_1), t_2) = \mathbf{Z}(t_3; \mathbf{z}, t_1)$$

b) $\forall (t, s) \in [0, T]^2$*, the application* $\mathbf{z} \mapsto \mathbf{Z}(t; \mathbf{z}, s)$ *is a* C^1*-diffeomorphism of* $\mathbb{R}^d$ *with inverse* $\mathbf{y} \mapsto \mathbf{Z}(s; \mathbf{y}, t)$.

c) The jacobian $J(t; 1, s) = \det(\nabla_{\mathbf{z}} \mathbf{Z}(t; \mathbf{z}, s))$ *satisfies* $J > 0$ *and*

$$\frac{\partial J}{\partial t} = (\nabla \cdot \mathbf{U})(\mathbf{Z}(t; \mathbf{z}, s))\, J$$

In particular, if $\nabla \cdot \mathbf{U} = 0$*, then* $J(t; 1, s) = J(s; 1, s) = \det \mathbb{I}_d = 1$ *where* $\mathbb{I}_d$ *is the identity matrix of order* d.

Also notice that equation (18) can be formulated as follows:

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t} + \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} \cdot \nabla_{\mathbf{z}} f = 0 \tag{22}$$

The total time derivative of f is equal to 0. Said differently, the fundamental property of the Vlasov equation is that *the distribution function* f *is constant along its characteristics.* This property is one of the foundations of many numerical schemes, including both Lagrangian and semi-Lagrangian numerical approaches, as will be detailed in the following.

Since the phase space element is incompressible in the Vlasov equation $\nabla_{(x,v)} \mathbf{U} = 0$ – Liouville's theorem applies – then the advective form of the Vlasov equation (18) also writes

$$\frac{\partial}{\partial t} f(\mathbf{Z}, t) + \nabla_{(x,v)} \cdot (\mathbf{U}(\mathbf{Z}, t)\, f(\mathbf{Z}, t)) = 0 \tag{23}$$

Equation (23) corresponds to the *conservative form* of the Vlasov equation. Liouville's theorem guaranties that the advective and the conservative forms of the Vlasov equation are equivalent. Both formulations are used in the numerical treatments, depending on the adopted numerical scheme. For instance, the following proposition 2 is the backbone of the Eulerian approach (see section 6.3)

Proposition 2. — *Let* $\mathbf{Z}(t; \mathbf{z}, s)$ *be the solution of Eqs.* (20)-(21), *and* $J(t; 1, s) = \det \nabla_{\mathbf{z}} \mathbf{Z}(t; \mathbf{z}, s)$ *the jacobian. Using the positivity of the jacobian (cf proposition 1), it can be proved (see* [**21**] *for instance) that the solution of the transport equation* (23) *can be expressed as*

$$f(\mathbf{Z}, t) = f(\mathbf{Z}(t; \mathbf{z}, s), s) \nabla_{\mathbf{z}} \mathbf{Z}(t; \mathbf{z}, s) \tag{24}$$

Such a result directly follows from the particle conservation along the characteristics

$$\forall K \subset \mathbb{R}^d, \qquad \int_K f(\mathbf{Z},t)d\mathbf{Z} = \int_{\mathbf{Z}(t;K,s)} f(\mathbf{Z},s)d\mathbf{Z} \tag{25}$$

with

$$\mathbf{Z}(t;K,s) = \{\mathbf{y} \in \mathbb{R}^d : \quad \mathbf{y} = \mathbf{Z}(t;\mathbf{z},s); \quad \mathbf{z} \in K\}$$

4.4. Conservation properties of the Vlasov-Maxwell system. — Let us call the Vlasov equation (12) coupled to Maxwell's equations (13)- (16) the *non-linear Vlasov equation*. This non-linear equation satisfies a certain number of conservation properties. Such exact properties reveal particularly powerful when developing numerical schemes: inherently conserving numerical methods can be envisaged, or, alternatively, they become stringent verification tests for numerical schemes and simulation results.

Proposition 3. — *The Vlasov equation insures the conservation of the number N of particles in a closed phase space volume V, apart from exchanges with the exterior. In this case, in agreement with Liouville's theorem:*

$$N = \int_V f(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = \int_V f(\mathbf{Z},t)\,\mathrm{d}V = \text{const} \tag{26}$$

As shown below, this simply results from the fact that the total derivative of f is equal to 0:

$$\frac{\mathrm{d}N}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}\int_V f(\mathbf{Z},t)\,\mathrm{d}V = \int_V \frac{\partial f}{\partial t}\,\mathrm{d}V = -\int_V \mathbf{U}\cdot\nabla_{(x,v)} f\,\mathrm{d}V$$

Using Gauss's theorem, it can be recast as follows

$$\int_V \mathbf{U}\cdot\nabla_{(x,v)} f\,\mathrm{d}V = \oint_{S(V)} f\,\mathbf{n}\cdot\mathbf{U}\,\mathrm{d}S \tag{27}$$

where $S(V)$ stands for the surface of volume V and $\mathbf{n}$ is its normal unit vector. Then

$$\frac{\partial N}{\partial t} = -\oint_{S(V)} f(\mathbf{n}\cdot\mathbf{U})\,\mathrm{d}S \tag{28}$$

This relation states that the rate of change of the number of particle inside the volume V is equal to the integral of the flux of f across the surface $S(V)$ of this volume. Hence, N remains constant for vanishing fluxes at the boundaries.

Proposition 4. — *Let the distribution function $f(\mathbf{x},\mathbf{v},t)$ in the phase space $(\mathbf{x},\mathbf{v}) \in \mathbb{R}^d \times \mathbb{R}^d, d = 1,\cdots 3$ be governed by the Vlasov-Maxwell equations. Then, denoting $f_0(\mathbf{x},\mathbf{v})$ the distribution function at initial time $t = 0$, f satisfies the following properties:*

– *Maximum principle*

$$0 \le f(\mathbf{x},\mathbf{v},t) \le max_{(x,v)}(f_0(\mathbf{x},\mathbf{v})) \quad \forall t \in \mathbb{R}^+ \tag{29}$$

– *Conservation of the volume. For all volume V of phase-space*

$$\int_V f(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = \int_{\mathbf{U}^{-1}(V)} f_0(\mathbf{y},\mathbf{u})\,\mathrm{d}\mathbf{y}\,\mathrm{d}\mathbf{u} \quad \forall t \in \mathbb{R}^+ \tag{30}$$

– *Conservation of the L^p norms, $p \in \mathbb{N}$, $1 \le p \le \infty$*

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d\times\mathbb{R}^d} (f(\mathbf{x},\mathbf{v},t))^p\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = 0 \quad \forall t \in \mathbb{R}^+ \tag{31}$$

– *Conservation of the kinetic entropy*

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(-\int_{\mathbb{R}^d\times\mathbb{R}^d} f(\mathbf{x},\mathbf{v},t)\ln f(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}\right) = 0 \quad \forall t \in \mathbb{R}^+ \tag{32}$$

– *Conservation of the energy,*

- *In the case of the Vlasov-Maxwell system:*

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}&\int_{\mathbb{R}^d\times\mathbb{R}^d} m|\mathbf{v}|^2 f(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} \\ &+ \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d}\left(\varepsilon_0|E(\mathbf{x},t)|^2 + \frac{1}{\mu_0}|B(\mathbf{x},t)|^2\right)\mathrm{d}\mathbf{x} = 0 \quad \forall t \in \mathbb{R}^+\end{aligned} \tag{33}$$

- *In the case of the Vlasov-Poisson system:*

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d\times\mathbb{R}^d} m|\mathbf{v}|^2 f(\mathbf{x},\mathbf{v},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} + \frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d}\varepsilon_0|E(\mathbf{x},t)|^2\,\mathrm{d}\mathbf{x} = 0 \quad \forall t \in \mathbb{R}^+ \tag{34}$$

The maximum principle especially insures that $f(\mathbf{x},\mathbf{v},t)$ remains positive for all time t if the initial data $f_0(\mathbf{x},\mathbf{v})$ is positive everywhere. The proofs of all these assertions are given in appendix D.

The entropy conservation is the consequence of the fact that the Vlasov-Maxwell system is *time reversible*. Indeed, let us consider the following time reversal transformation: $t \to -t$; $\mathbf{x} \to \mathbf{x}$; $\mathbf{v} \to -\mathbf{v}$; $\mathbf{E} \to \mathbf{E}$ and $\mathbf{B} \to -\mathbf{B}$. This transformation leaves Vlasov equation invariant. This can be understood as follows: starting from some initial state and evolving f in time according to the Vlasov-Maxwell system, and deciding at a certain time to reverse the velocity of all particles (leading to the reversal of the magnetic field $\mathbf{B}$), then the system would trace back to its initial state (see *e.g.* Villani's lecture notes [**141**])

Finally, it is worth noticing that Maxwell's equations imply that charge density ρ and current density $\mathbf{j}$ satisfy the continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{j} = 0 \tag{35}$$

In other words, should this relation not hold, one can easily verify that there is no solution to Maxwell's equations. When the sources of Maxwell's equations are computed numerically by solving the Vlasov equation, they do not necessarily verify a discrete equivalent of the continuity equation, which would be compatible with the discrete form of Maxwell's equations. To address this problem, numerical methods for deriving sources that satisfy the discrete continuity equation have been developed (cf *e.g.* [**144**]). A rigorous mathematical theory for these generalized formulations of Maxwell's equations is derived in [**7**].

5. Gyrokinetic theory

The computational effort to numerically solve the 6-dimensional Vlasov-Maxwell system still remains out of reach for present day supercomputers. As shown in figure 3, it involves a wide range of spatio-temporal scales – from light waves to macroscopic plasma instabilities. When dealing with micro-turbulence studies, it is therefore critical to restrict the problem to the relevant space and time scales. The strong magnetic fields – a few Tesla – used in magnetic confinement fusion devices lead to the fast gyration of charged particles around magnetic field lines. Especially, the cyclotron frequency $\Omega_s = (e_s B_0)/m_s$ is large compared to the characteristic frequency of micro-turbulence, which is of the order of the diamagnetic frequency $\omega \sim \omega_s^* \sim (k_\theta \rho_s) v_{th,s}/L_p$ (with k_θ the poloidal wave vector, $\rho_s = v_\perp/\Omega_s$ the Larmor radius, $v_{th,s} = (T_s/m_s)^{1/2}$ the thermal speed and $L_p = |\nabla \ln p_0|^{-1}$ the characteristic gradient length of the mean pressure profile. In this framework, the single particle motion consists of the superposition of the fast periodic gyro-motion and the slow drift of the guiding-centre, as shown in Fig. 4. Besides, experimental observations in core plasmas of magnetic confinement fusion devices suggest that small scale turbulence, responsible for anomalous transport, obeys the following ordering in a small parameter ϵ_g

$$\frac{\omega}{\Omega_s} \sim \frac{e_s \phi}{T_s} \sim \frac{\delta n_s}{n_0} \sim \frac{B_1}{B_0} \sim \frac{k_\parallel}{k_\perp} \sim \frac{\rho_s}{L_n} \sim \mathcal{O}(\epsilon_g) \tag{36}$$

where the subscript s refers to the particle species. T_s is the temperature, δn_s the perturbed density, n_0 the equilibrium density, $\mathbf{B}_0$ and $\mathbf{B}_1$ are respectively the equilibrium and the perturbed magnetic field, $k_\parallel = \mathbf{k} \cdot \mathbf{b}$ and

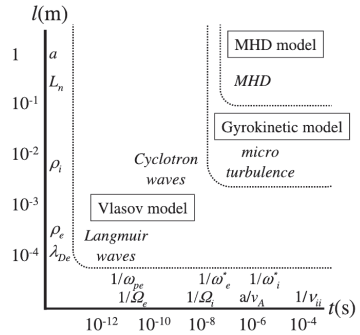


FIGURE 3. Typical space and time range scales in fusion plasmas and applicability of Vlasov, gyrokinetic and MHD models. Here, ω_{ps} is the plasma oscillation frequency, Ω_s is the cyclotron frequency, ω_s^* is the diamagnetic rotation frequency, v_A is the Alfvèn velocity, ν_{ii} is the ion-ion collision frequency, λ_{Ds} is the Debye length, ρ_s is the Larmor radius, L_n is the characteristic gradient length of the equilibrium density profile, a is the plasma size and s denotes the particle species. (figure from [**79**])

$k_\perp = |\mathbf{k} \times \mathbf{b}|$ are parallel and perpendicular components of the wave vector $\mathbf{k}$ with $\mathbf{b} = \mathbf{B}_0/B_0$, $\rho_s = v_\perp/\Omega_s$ is the Larmor radius, $L_n = |\nabla \ln n_0|^{-1}$ is the characteristic length of n_0. Within this *gyro-ordering*, the so-called *gyrokinetic model* can be derived (see *e.g.* [**94**]) by eliminating high-frequency processes characterized by $\omega > \Omega_s$. The phase space is reduced from 6 to 5 dimensions, while retaining crucial kinetic effects such as finite Larmor radius effects. Numerically speaking, the computational cost is dramatically reduced because the limitations on the time step and the grid discretization are relaxed from $\omega_{ps}\,\Delta t < 1$ and $\Delta x < \lambda_{Ds}$ to $\omega_s^*\,\Delta t < 1$ and $\Delta x < \rho_s$ (with ω_{ps} the plasma oscillation frequency and λ_{Ds} the Debye length). It is also important to note that the magnetic moment, $\mu_s = m_s v_\perp^2/(2B_0)$ becomes an adiabatic invariant. In terms of simulation cost, this last point is convenient because μ_s plays the role of a parameter. This means that the problem to treat is not a true 5D problem but rather a 4D problem parametrized by μ_s. Note that μ_s looses its invariance property in the presence of collisions. Such a numerical drawback can be overcome by considering reduced collisions operators acting in the $v_\parallel$ space only, while still recovering the results of the neoclassical theory [**57**].

The resulting *gyrokinetic equation* is today the most advanced framework to

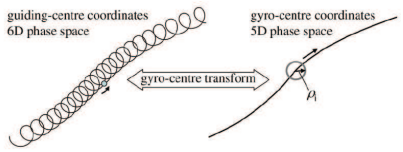


FIGURE 4. The phase-space reduction from 6 to 5 dimensions results from the gyro-centre transform, which aims at eliminating the fast gyro-motion. (figure from [**79**])

describe plasma micro-turbulence. It reads as follows (see also Eqs (17)-(20) in [**58**]):

$$B_\parallel^* \frac{\mathrm{d}\bar{f}_s}{\mathrm{d}t} + \boldsymbol{\nabla}\cdot\left(B_\parallel^* \frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t}\,\bar{f}_s\right) + \frac{\partial}{\partial v_{G\parallel}}\left(B_\parallel^* \frac{\mathrm{d}v_{G\parallel}}{\mathrm{d}t}\,\bar{f}_s\right) = 0 \tag{37}$$

In the electrostatic limit, the equations of motion of the guiding centers are given below:

$$B_\parallel^* \frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t} = v_{G\parallel}\mathbf{B}^* + \frac{\mathbf{b}}{e_s}\times\boldsymbol{\nabla}\Xi \tag{38}$$

$$B_\parallel^* \frac{\mathrm{d}v_{G\parallel}}{\mathrm{d}t} = -\frac{\mathbf{B}^*}{m_s}\cdot\boldsymbol{\nabla}\Xi \tag{39}$$

$$\begin{aligned} \text{with}\quad \boldsymbol{\nabla}\Xi &= \mu_s\boldsymbol{\nabla}B + e_s\boldsymbol{\nabla}\bar{\phi} \\ \mathbf{B}^* &= \mathbf{B} + (m_s/e_s)\,v_{G\parallel}\boldsymbol{\nabla}\times\mathbf{b} \end{aligned}$$

Here, $\bar{\phi}$ denotes the gyro-averaged electric potential (see section 5.1). The scalar $B_\parallel^*$ is simply $B_\parallel^* = \mathbf{B}^*\cdot\mathbf{b}$, with $\mathbf{b} = \mathbf{B}/B$ the unit vector along the magnetic field line at the guiding-center position. As shown in appendix A, these equations of motion naturally arise (up to small $j_\parallel$ terms) when considering the fast time average of Newton's equations in the adiabatic limit, i.e. when the characteristic gradient lengths of the equilibrium magnetic field are small with respect to the Larmor radius.

For an overview and a modern formulation of the gyrokinetic derivation, one encourages the reader to refer to the review paper by A.J. Brizard and T.S. Hahm [**23**]. The advantage of this new approach based on Lagrangian formalism and Lie perturbation theory [**30, 31**] is to preserve the first principles by construction, such as the symmetry and conservation properties of the Vlasov equation (see section 4.4) – particle number, momentum, energy and entropy.

5.1. Average over the cyclotron motion. — The gyro-radius $\boldsymbol{\rho}_s$ is transverse to $\mathbf{b} = \mathbf{B}/B$ and depends on the gyrophase angle φ_c:

$$\boldsymbol{\rho}_s = \frac{\mathbf{b}\times\mathbf{v}}{\Omega_s} = \rho_s\left[\cos\varphi_c\,\mathbf{e}_{\perp 1} + \sin\varphi_c\,\mathbf{e}_{\perp 2}\right] \tag{40}$$

Here $\mathbf{e}_{\perp 1}$ and $\mathbf{e}_{\perp 2}$ are the unit vectors of a cartesian basis in the plane perpendicular to the magnetic field direction $\mathbf{b}$. Let $\mathbf{x}_G$ be the guiding-center radial coordinate and $\mathbf{x}$ the position of the particle in the real space. These two quantities differ by a Larmor radius $\boldsymbol{\rho}_s$:

$$\mathbf{x} = \mathbf{x}_G + \boldsymbol{\rho}_s$$

The gyro-average $\bar{g}$ of any function g depending on the spatial coordinates corresponds to the following operation:

$$\begin{aligned} \bar{g}(\mathbf{x}_G, v_\perp) &= \oint_0^{2\pi} \frac{\mathrm{d}\varphi_c}{2\pi} g(\mathbf{x}) \\ &= \left\{\oint_0^{2\pi} \frac{\mathrm{d}\varphi_c}{2\pi}\exp(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\right\} g(\mathbf{x}_G) \end{aligned} \tag{41}$$

The operator $\mathrm{e}^{\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}$ corresponds to the change of coordinates $(\mathbf{x},\mathbf{v}) \to (\mathbf{x}_G,\mathbf{v}_G)$. The inverse operator governing the transformation $(\mathbf{x}_G,\mathbf{v}_G) \to (\mathbf{x},\mathbf{v})$ simply reads $\mathrm{e}^{-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}$. This gyro-average process consists in computing an average on the Larmor circle. It tends to damp any fluctuation which develops at sub-Larmor scales. Introducing $\hat{g}(\mathbf{k})$ the Fourier transform of g, with $\mathbf{k}$ the wave vector, then the operation of gyro-average reads:

$$\begin{aligned} \bar{g}(\mathbf{x}_G, v_\perp) &= \int_0^{2\pi}\frac{\mathrm{d}\varphi_c}{2\pi}\int_{-\infty}^{+\infty}\frac{\mathrm{d}^3\mathbf{k}}{(2\pi)^3}\,\hat{g}(\mathbf{k})\exp\{i\mathbf{k}\cdot(\mathbf{x}_G+\boldsymbol{\rho}_s)\} \\ &= \int_{-\infty}^{+\infty}\frac{\mathrm{d}^3\mathbf{k}}{(2\pi)^3}\left[\int_0^{2\pi}\frac{\mathrm{d}\varphi_c}{2\pi}\exp(ik_\perp\rho_s\cos\varphi_c)\right]\hat{g}(\mathbf{k})\exp(i\mathbf{k}\cdot\mathbf{x}_G) \\ &= \int_{-\infty}^{+\infty}\frac{\mathrm{d}^3\mathbf{k}}{(2\pi)^3}\,J_0(k_\perp\rho_s)\hat{g}(\mathbf{k})\mathrm{e}^{i\mathbf{k}\cdot\mathbf{x}_G} \end{aligned} \tag{42}$$

In the following, we shall denote indistinctly $J.\phi$ or $\bar{\phi}$ or even $\langle\phi\rangle$ the gyroverage of any scalar field ϕ. Here, $k_\perp$ is the norm of the transverse component of the wave vector $\mathbf{k}_\perp = \mathbf{k} - (\mathbf{b}.\mathbf{k})\mathbf{b}$, and J_0 is the Bessel function of first order. In Fourier space, the gyro-average reduces to the multiplication by the Bessel function of argument $k_\perp\rho_s$. This operation is straightforward in simple geometry with periodic boundary conditions, such as in local codes (see section 7.2). Conversely, in the case of global codes (see section 7.2), the use of Fourier transform is not applicable for two main reasons: (i) radial boundary conditions are non periodic, and (ii) the radial dependence of the Larmor radius has to be accounted for. Several approaches have been developed to overcome

this difficulty. One of those consists in approximating the Bessel function with Padé expansion $J_{\text{Padé}}(k_\perp\rho_s) = 1/\left[1+(k_\perp\rho_s)^2/4\right]$ (*e.g.* see [**121**]). Such a Padé representation then requires the inversion of the Laplacian operator $\nabla_\perp^2$ in real space. As seen on figure 5 this approximation gives the correct limit in the large wavelengths limit $k_\perp\rho_c \ll 1$, while keeping $J_{\text{Padé}}$ finite in the opposite limit $k_\perp\rho_s \to \infty$. The drawback is an overdamping of small scales: in the limit of large arguments $x \to \infty$, $J_{\text{Padé}}(x) \to 4/x^2$, whereas $J_0 \to (2/\pi x)^{1/2}\cos(x-\pi/4)$ (see figure 6).

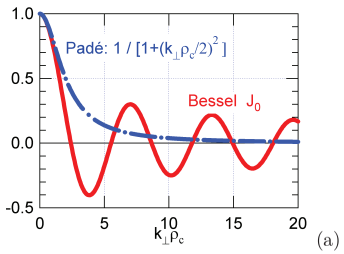


FIGURE 5. The zero-th order Bessel function $J_0(k_\perp\rho_s)$ compare to its Padé approximation $1/\left[1+(k_\perp\rho_s)^2/4\right]$.

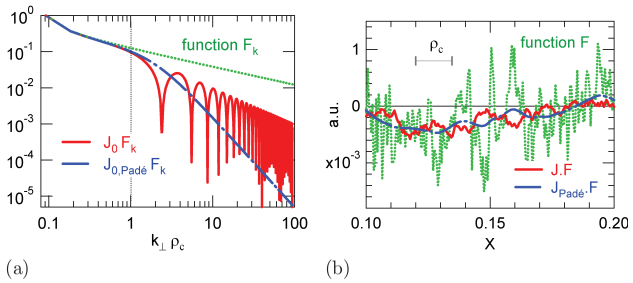


FIGURE 6. Exact and approximated gyro-average operators applied on an arbitrary funcion F_k exhibiting a broad spectrum ranging from low to large wavelengths as compared with the Larmor radius ρ_c: (a) Representation in the Fourier space, (b) Representation in the real space (figures from [**121**]).

Another possibility, for this gyro-averaging process, is to use a quadrature formula. In this context, the integral over the gyro-ring is usually approximated by a sum over four points on the gyro-ring [**94**]. Notice that, for small arguments only, i.e. for large scale fluctuations $k_\perp\rho_s \ll 1$, this is rigorously equivalent to considering the Taylor expansion of the Bessel function at order two in the small argument limit, namely $J_0(k_\perp\rho_s) \simeq 1-(k_\perp\rho_s)^2/4$, and to computing the transverse Laplacian at second order using finite differences [(3)]. Perhaps counter intuitively, note however that this 4 point average is no longer equivalent to $[1-(k_\perp\rho_s)^2/4]$ in the opposite limit $k_\perp\rho_s \gg 1$. Indeed, the difference between *distant* points can no longer be interpreted as the finite difference expression of a differential operator: the latter can be approximated by finite differences in the limit of small distances only. In particular, it is important to realize that, while $[1-(k_\perp\rho_s)^2/4]$ diverges at small scales, the 4 point average still leads to the damping of sub-Larmor scales. This method has been extended to achieve accuracy for large Larmor radius [**72**], i.e the number of points (starting with four) is linearly increased with the gyro-radius to guarantee the same number of points per arclength on the gyro-ring. In this approach –used *e.g.* in [**76**] and [**83**]– the points that are equidistantly distributed over the ring are rotated for each particle (or marker) by a random angle calculated every time step. This is performed on a finite element formalism and enables therefore high order accuracy by keeping the matricial formulation. In [**38**] the influence of the interpolation operator (which is of great importance when the quadrature points do not coincide with the grid points) is studied and shows that cubic splines are good candidates. Besides, new methods based on the direct integration of the gyroaverage operator are also described and compared to the two previous approaches.

5.2. Quasi-neutrality. — Fusion plasma turbulence typically develops at Larmor scales, which are larger than the Debye length λ_D. The Larmor radius of thermal Deuterium nuclei is $\rho_D \approx 4.6\,10^{-3}\,T_{[keV]}^{1/2}/B_{[T]}\,m \approx 4.1\,10^{-3}\,m$, for $T = 20keV$ and $B = 5T$, while

$$\lambda_D \approx 2.35\,10^{-5}(T_{[keV]}/n_{10^{20}m^{-3}})^{1/2}m \approx 10^{-4}m$$

for $n = 10^{20}m^{-3}$. The electron Larmor radius ρ_e may eventually compete with λ_D, although both experimental observations and numerical simulations suggest that electron turbulence saturates at scales larger than ρ_e. In this case, Poisson's equation can be safely replaced by the quasi-neutrality constraint, namely $\sum_s e_s n_s = 0$. In the gyrokinetic framework, the additional complexity

(3) Indeed, let x and y be cartesian transverse coordinates, with grid spacing equal to ρ_s. At second order in both $k_\perp\rho_s$ and in ρ_s, the approximation of J_0 translates in real space as follows: $J_0.\hat{\phi}_k \to (1+\rho_s^2\nabla_\perp^2/4)\phi \simeq \phi_{i+1,j}+\phi_{i,j+1}+\phi_{i,j-1}+\phi_{i-1,j}$.

comes from the fact that each particle density $n_s = \int \mathrm{d}^3\mathbf{v} f_s$ needs being related to the distribution function of the gyro-centers $\bar{f}_s$. Both distribution functions relate to each other in the following way:

$$f_s(\mathbf{x},\mathbf{v},t) = \bar{f}_s(\mathbf{x}_G,\mathbf{v}_G,t) + \frac{e_s}{B}\left\{\phi(\mathbf{x},t) - \bar{\phi}(\mathbf{x}_G,\mathbf{v}_G,t)\right\}\ \partial_\mu \bar{f}_{eq,s}(\mathbf{x}_G,\mathbf{v}_G) \tag{43}$$

Such a relationship results from the infinitesimal contact (or canonical) transformation that relates the canonical variables $(\mathbf{x},\mathbf{v})$ and $(\mathbf{x}_G,\mathbf{v}_G)$, as detailed in appendix B. Equation (43) allows one to relate particle density n_s and guiding-center density n_{Gs}:

$$\begin{aligned} n_s(\mathbf{x},t) &= \int \mathrm{d}^3\mathbf{v}\bar{f}_s(\mathbf{x}_G,\mathbf{v}_G,t) \\ &\quad + \int \mathrm{d}^3\mathbf{v}\frac{e_s}{B}\left\{\phi(\mathbf{x},t) - \bar{\phi}(\mathbf{x}_G,\mathbf{v}_G,t)\right\}\ \partial_\mu \bar{f}_{eq,s}(\mathbf{x}_G,\mathbf{v}_G) \end{aligned}$$

The first integral on the right hand side corresponds to the gyro-center density $n_{G,s}(\mathbf{x},t)$. The last integral on the right hand side is the polarization density $n_{pol,s}$. Further recalling that $\mathbf{x}_G = \mathbf{x} - \boldsymbol{\rho}_s$, it follows:

$$n_{G,s}(\mathbf{x},t) = \int \mathcal{J}_v\, \mathrm{d}\mu\, \mathrm{d}v_{G\|}\ J.\bar{f}_s(\mathbf{x},\mathbf{v},t) \tag{44}$$

$$\begin{aligned} n_{pol,s}(\mathbf{x},t) &= \int \mathcal{J}_v\, \mathrm{d}\mu\, \mathrm{d}v_{G\|}\frac{e_s}{B}\int_0^{2\pi}\frac{\mathrm{d}\varphi_c}{2\pi}\ \Big\{\mathrm{e}^{-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}\partial_\mu f_{eq,s}(\mathbf{x},\mathbf{v}) \\ &\qquad \left[1 - \mathrm{e}^{-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}\left\langle \mathrm{e}^{\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}}\right\rangle\right]\phi(\mathbf{x},\mathbf{v},t)\Big\} \end{aligned} \tag{45}$$

where $\mathcal{J}_v = (2\pi B_\|^*/m_s)$ stands for the Jacobian in the velocity space, and the brackets $\langle ... \rangle$ for the average over the gyro-phase: $\langle ... \rangle \equiv \int_0^{2\pi} ... \frac{\mathrm{d}\varphi_c}{2\pi}$.

It is usual – although not always legitime – to consider the large wavelength limit $k_\perp \rho_s \ll 1$ of eq. (45), for which some particularly compact expression of $n_{pol,s}$ can be derived. In such a limit, one can restrict the Taylor expansion of the operator to the leading terms of the development only. Details of the calculus are given in appendix B. At second order in $k_\perp \rho_s$, it comes:

$$n_{pol,s}(\mathbf{x},t) \simeq \boldsymbol{\nabla}_\perp \cdot \left(\frac{m_s n_{eq,s}}{e_s B^2}\boldsymbol{\nabla}_\perp \phi(\mathbf{x},t)\right) \tag{46}$$

The polarisation density of electrons can usually be neglected because of their small inertia. If the electron response is assumed adiabatic, the quasi-neutrality then further reduces to:

$$\frac{e}{T_e}\left(\phi - \langle\phi\rangle_{FS}\right) - \frac{1}{n_{eq}}\boldsymbol{\nabla}_\perp \cdot \left(\frac{m_s n_{eq}}{e_s B^2}\boldsymbol{\nabla}_\perp \phi\right) = \frac{1}{n_{eq}}\int \mathcal{J}_v\, \mathrm{d}\mu\, \mathrm{d}v_{G\|} J.\bar{f}_i - 1 \tag{47}$$

with $\langle\phi\rangle_{FS}$ the flux surface average of ϕ, and we recall that $J.\bar{f}_i$ is the gyro-average of the ion gyro-center distribution function.

6. Numerical models for Vlasov equations

Solving the set of gyrokinetic equations is very challenging. To summarize, it consists of the evolution equation of the distribution function f in the 4D phase space parametrized by the adiabatic invariant μ, the set of four coupled ordinary differential equations (ODE) for the characteristics, and 3D integro-differential equations for the fields, namely Poisson's equation (or the quasi-neutrality equation) and the Ampère law (if magnetic perturbations are taken into account). This set of equations is nonlinear, the dominant quadratic nonlinearity being due to the $\mathbf{E}\times\mathbf{B}$ advection terms. The quasi-neutrality equation is generally solved by using Fourier projection in all the periodic directions and finite differences or finite elements in the others. Concerning the global algorithm, the challenge consists in finding numerical schemes which preserve the first principles such as the conservation of the Casimir invariants, the phase space volume and the total energy. Various numerical schemes have been explored until now. They can be classified in Lagrangian, Eulerian and semi-Lagrangian. In this section, the main schemes used in gyrokinetic codes are described, emphasizing their own advantages and drawbacks. The comparison between PIC and Eulerian approaches in terms of operation numbers is detailed in *e.g.* [**27, 52**]. The interest of developing concurrently these three approaches is crucial for the gyrokinetic community. Indeed, due to the extremely challenging computing requirements, each gyrokinetic code runs at the limit of its applicability range. Benchmarks between the different approaches are therefore crucial and more and more encouraged (*e.g.* [**134, 51**]).

6.1. PIC approach. — The Lagrangian-PIC (particles in cell) approach looks for solutions of the Vlasov equation in terms of the solution of the ODE of motion for macro-particles. Each macro-particle represents a large number of the plasma particles. The particle orbits are the characteristics of the Vlasov equation. In this case, the phase space density along the trajectories is preserved by construction (Lagrange), while charge and current densities need being computed, by extrapolation, on Eulerian (i.e. fixed in space) grid points. PIC codes have the enormous advantages of being simple, robust and easily scalable. Their disadvantage is the numerical noise, inherent to the limited number of macro-particles, which can cause numerical collisions and artificial dissipation. Many codes are based on this PIC approach. Let us mention Parker's code [**111**], Sydora's code [**129**], PG3EQ [**46**], GTC [**97**], ELMFIRE [**73**], GT3D [**76**], ORB5 [**19, 83**], and GTS [**146**].

6.1.1. Statistical noise reduction. — When applied to plasma studies, the PIC-approach consists in two distinct steps. In the first one, the self-consistent fields generated by the distribution of computational particles in a multidimensional phase space are calculated. In the second step, the particle orbits are followed, given the electromagnetic fields. PIC simulations are intrinsically *noisy*. This noise is predominantly associated with the first step, where the two first moments of the distribution function are calculated to find the source terms in Poisson's and Ampere's equations. The relatively small number of tracers in phase space, which are governed by the computational particle positions, introduces errors at this stage. From the mathematical point of view, Aydemir [**6**] pointed out the fact that the Lagrange-PIC algorithms can be viewed as statistical methods to obtain, via *Monte Carlo integration*, estimates of the moments of the distribution function. The standard deviation of this estimator is known to scale like $1/\sqrt{N}$, with N the number of markers (see Appendix E-section E.1 for details). This statistical error is called "numerical noise". The main problem for non-linear gyrokinetic simulations is that the noise level *a priori* accumulates in time [**110**]. Even small errors in the evaluation of theses moments can cause systematic corruptions of the simulation results on relatively short period of time. The reduction of numerical noise is right from the start a matter of intense research, and lots of improvements have been obtained in this domain since five years. Let us specifically mention the achievements made on the ORB5 gyrokinetic PIC-code [**83**], which are summarized in [**143**].

It has long been appreciated that the statistical noise of the PIC method in the framework of gyrokinetic simulations [**94**] can be considerably reduced by using the δf scheme [**111**]. It consists in decomposing the distribution function as follows: $f = f_0 + \delta f$, with f_0 a time-independent (or slowly evolving) function of phase-space coordinates and δf sampled with markers (cf. section 7.1 for more details on this approximation). By making connection with Monte Carlo schemes, Aydemir [**6**] has shown that the δf method can be seen as an example of the control variate method of variance reduction. The sampling noise error is thus reduced by the factor $|\delta f|/f_0$, which is much smaller than unity for core plasma turbulence. An improved δf scheme, called "adjustable control variate", has been developed and applied to electromagnetic gyrokinetic simulations. This new approach succeeded in reducing the required number of markers for a given precision by orders of magnitude [**71**]. Control variate is not the only variance reduction technique available under a Monte Carlo scheme. Importance sampling also efficiently reduces the error in moment estimates. This "optimized loading" scheme [**72**] consists in finding an appropriate distribution of the marker positions compared to the simple "proportional loading". Hatzky et al. [**72**] has also shown an example of the two methods used in conjunction, resulting in dramatic improvements

in the quality of gyrokinetic simulations. The classical techniques, which are developed for noise reduction in the Monte Carlo evaluation of integrals and are at the basis of all these improvements, are detailed in Appendices E.2 and E.3. Apart from these classical Monte Carlo approaches, knowledge of the underlying physics can inspire other major improvements.

(i) The choice of f_0 is critical in order to keep $|\delta f|/f_0$ as small as possible. In this regard, choosing $f_0 = f_{\rm eq}$, with $f_{\rm eq}$ the initial equilibrium state, looks highly desirable. However, since f can evolve away from $f_{\rm eq}$, an intuitive idea would be to evolve f_0 as well, in such a way to "follow" f. One such technique has been implemented in collisional Monte Carlo simulations [**26**]. In this case an appropriate choice for $f_0(t)$ was a shifted Maxwellian distribution, evolved using fluid equations. A more general technique is proposed in [**2**]. In this approach, δf is directly obtained from the constancy of f along orbits $\mathbf{Z}(t)$, since $\delta f(\mathbf{Z}_j(t)) = f(\mathbf{Z}_j(t_0)) - f_0(\mathbf{Z}_j(t))$.

(ii) In any case, performing long simulation runs – and reaching turbulent steady-state – while keeping the numerical noise at weak levels is very challenging. A noise control algorithm using modified Krook operator looks promising [**101**]. Interestingly, such a Krook operator also adds finite dissipation, which has be proven to be mandatory to reach a statistically converged state [**91**]. In grid-based codes (Eulerian or semi-Lagrangian), numerical dissipation is introduced either by using dissipative discretization or interpolation schemes, or by adding hyper-diffusion terms [**114**].

(iii) Another physical property of turbulence in tokamaks can be fruitfully exploited, namely its strong anisotropy. Indeed, micro-turbulence modes are characterized by small parallel wave vectors as compared to transverse ones: $|k_\parallel|\rho_s \leq \rho^* \ll 1$ (more precisely, $k_\parallel \sim 1/qR$, with q the safety factor and R the major radius), consistently with the gyro-ordering given in Eq. (36). Jolliet et al. [**83**] have developed a filter which takes advantage of this strong anisotropy of the perturbations. They have shown that filtering the modes, which may be present in the simulation but do not satisfy this ordering, is a very efficient way to significantly reduce numerical noise. The technique consists in applying a *field-aligned Fourier filter*. The technique essentially consists in setting to zero all (m, n) modes (m and n being the poloidal and toroidal wave numbers, respectively) of the perturbed density appearing in the quasi-neutrality equation which fulfill the following relation: $|m| \geq q(r)|n| + \Delta m$, where $\Delta m \in N$ is the prescribed width of the retained modes. Δm typically ranges from 5 to 10 (see Jolliet's thesis [**82**] for more details). It corresponds to a diagonal filter in Fourier space, instead of a square filter generally used. The main interest is that, in this case, the signal to noise ratio scales like the number of markers per Fourier mode retained by the filter, and no longer

like the number of markers per numerical degree of freedom of the field representation.

(iv) Finally, solutions have been found to face the problem known as the *entropy paradox* [**148**] inherent to the Monte Carlo approach used in the PIC codes. Indeed, the fluctuation entropy, proportional to the sum of the squared weights $\langle w^2 \rangle$, increases with the turbulence and one observes that low order moments saturate while the fluctuation entropy continually increases in time (see *e.g.* [**84**] for more explanations). Therefore, as a paradox, on top of the physical nonzero value of the fluctuation entropy there is a numerical growth of this quantity. One way to resolve this entropy paradox is to introduce a physical dissipation by introducing a collision operator. Unfortunately, this is a very difficult task in a 5D PIC code. An alternative, originally proposed by Krommes [**90**], consists of introducing a Krook-like dissipation term. The aim of this term is to control the variance of weights and therefore to control the noise. This algorithm (with the difference that the variance of the weights is not frozen to a given value) has been successfully implemented in the ORB5 code [**100, 84**].

6.1.2. Global PIC-algorithm. — The algorithms of PIC codes (see Fig. 7) can be decomposed in three main steps: *(i)* sampling initial positions in phase space (loading of markers); *(ii)* following marker orbits in 5D (pushing) and *(iii)* obtaining the source terms for the field equations at each time step (charge and current assignment). These three steps are detailed in appendix F.

6.2. Time-splitting scheme. — Let us consider the Vlasov equation, written in the form (18), where $\mathbf{U}$ is the divergence-free vector field having up to six components in the full 6D case. For example, in the case of the 3D electrostatic Vlasov equation, these fields are $\mathbf{Z} = (x, y, z, v_x, v_y, v_z)$ and $\mathbf{U} = (v_x, v_y, v_z, \mathbf{E}_x, \mathbf{E}_y, \mathbf{E}_z)$, all the components of the electric field depending on x, y, z and t. The Vlasov equation can also be written in its conservative form (we recall that $\mathbf{U}$ is divergence free):

$$\frac{\partial}{\partial t} f(\mathbf{Z}, t) + \mathrm{div}_{(x,v)} \left(\mathbf{U}(\mathbf{Z}, t)\, f(\mathbf{Z}, t) \right) = 0$$

Splitting the components of $\mathbf{Z}$ into two sets $\mathbf{x}$ and $\mathbf{v}$, the previous equation can then be written as

$$\frac{\partial}{\partial t} f(\mathbf{x}, \mathbf{v}, t) + \mathrm{div}_x \left(\mathbf{U}_x(\mathbf{x}, \mathbf{v}, t)\, f(\mathbf{x}, \mathbf{v}, t) \right) + \mathrm{div}_v \left(\mathbf{U}_v(\mathbf{x}, \mathbf{v}, t)\, f(\mathbf{x}, \mathbf{v}, t) \right) = 0$$

where the components of the advection field $\mathbf{U}_x$ and $\mathbf{U}_v$ are defined by Eq. (19). At this stage, it is convenient to split both parts and to treat them separately (this operation is known as the *splitting method* or the operator

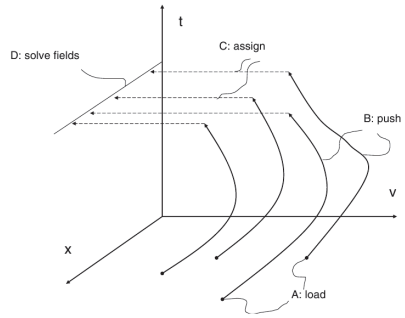


FIGURE 7. In Lagrangian-PIC methods, marker initial positions are loaded pseudo- (or quasi-) randomly in phase space (A). Markers are evolved along their orbits (B). Charge and current perturbations are assigned (projected) to real space (C). Field equations are solved (D), e.g. on a fixed grid in real space. (figure from [**79**])

decomposition):

$$\frac{\partial}{\partial t} f(\mathbf{x}, \mathbf{v}, t) + \mathrm{div}_x \left(\mathbf{U}_x(\mathbf{x}, \mathbf{v}, t)\, f(\mathbf{x}, \mathbf{v}, t) \right) = 0 \quad \text{at fixed } \mathbf{v} \tag{48}$$

and

$$\frac{\partial}{\partial t} f(\mathbf{x}, \mathbf{v}, t) + \mathrm{div}_v \left(\mathbf{U}_v(\mathbf{x}, \mathbf{v}, t)\, f(\mathbf{x}, \mathbf{v}, t) \right) = 0 \quad \text{at fixed } \mathbf{x} \tag{49}$$

Notice that the semi-Lagrangian scheme (see section 6.4) does not solve Vlasov equation in the conservative form, but rather in the advective form in order to make full use of the backward characteristic method. In this case, eqs. (48) and (49) can be put in the advective form

$$\frac{\partial f}{\partial t} + \mathbf{U}_x \cdot \nabla_x f = 0 \tag{50}$$

$$\frac{\partial f}{\partial t} + \mathbf{U}_v \cdot \nabla_v f = 0 \tag{51}$$

if and only if both velocity field are divergence free independently, namely

$$\mathrm{div}_x(\mathbf{U}_x(\mathbf{x}, \mathbf{v}, t)) = 0 \tag{52}$$

$$\mathrm{div}_v(\mathbf{U}_v(\mathbf{x}, \mathbf{v}, t)) = 0 \tag{53}$$

Conversely, should equations (52) and (53) not be satisfied, then splitting Eq. (18) would be equivalent to solving advective equations with source terms

$$\frac{\partial f}{\partial t} + \mathbf{U}_x \cdot \nabla_x f = -f \mathrm{div}_x(\mathbf{U}_x) \tag{54}$$

$$\frac{\partial f}{\partial t} + \mathbf{U}_v \cdot \nabla_v f = -f \mathrm{div}_v(\mathbf{U}_v) \tag{55}$$

The divergence-free property of the full advection field $\mathbf{U}$ should *a priori* guaranty that

$$\mathrm{div}_x(\mathbf{U}_x) = -\mathrm{div}_v(\mathbf{U}_v)$$

However, the source terms in Eqs. (54) and (55) do not cancel each other since they are not computed at the same time. As a consequence, this introduces systematic cumulative errors at each time step, resulting in poor density conservation (see *e.g.* [**74, 108**]). This constraint of divergence free independence is one drawback for semi-Lagrangian approach in high dimensionality.

For the following, let suppose (52) and (53) satisfied, then the error associated to the splitting can be estimated by writing the Vlasov equation formally as follows

$$\frac{\partial f}{\partial t} = (A + B) f \tag{56}$$

where A and B stand for differential operators in phase space (for instance, $A = -\mathbf{v} \cdot \nabla_{\mathbf{x}}$ and $B = -(q_s/m_s)\mathbf{E} \cdot \nabla_{\mathbf{v}}$, such that $\mathbf{U}_x = \mathbf{v}$ and $\mathbf{U}_v = \mathbf{E}$), assumed constant in the time interval $[t_n, t_{n+1}]$. The formal solution of this equation (56) on Δt time step reads

$$f(t + \Delta t) = \exp\left[\Delta t (A + B)\right] f(t)$$

Let us split equation (56) as follows

$$\frac{\partial f}{\partial t} = A f(t) \tag{57}$$

$$\frac{\partial f}{\partial t} = B f(t) \tag{58}$$

The solutions are, respectively

$$f(t + \Delta t) = \exp(\Delta t A) f(t) \quad \text{and} \quad f(t + \Delta t) = \exp(\Delta t B) f(t)$$

The *standard splitting method* consists in successively solving Eq. (57) on time step Δt followed by Eq. (58), also on time step Δt. On one time step, one then obtains

$$\tilde{f}(t + \Delta t) = \exp(\Delta t B) \exp(\Delta t A) f(t)$$

If the operators A and B commute, then $\exp(\Delta t B) \exp(\Delta t A) = \exp\left[\Delta t (A + B)\right]$, which corresponds to the solution of the whole equation, eq. (56). In this case, the splitting is exact. This is especially true when the advections consist of constant coefficients. However, in most cases, A and B do not commute.

In this case, Strang has shown [**128**] that the splitting error can be drastically reduced by solving symmetrically Eq. (57) on half a time step $\Delta t/2$, then Eq. (58) on a time step Δt and finally again Eq. (57) on half a time step $\Delta t/2$. This method is called *Strang splitting*. It leads to the formal solution

$$\tilde{f}(t + \Delta t) = \exp(\frac{\Delta t}{2} A) \exp(\Delta t B) \exp(\frac{\Delta t}{2} A) f(t)$$

Proposition 5. — *The splitting error on one time step, when the operators do not commute, is of order* 1 *in time for the standard splitting and of order* 2 *in time for the Strang splitting.*

Remark 1. — *The Strang splitting can be generalized to more than two operators. If $A = A_1 + \cdots + A_n$, the following decomposition preserves the second order accuracy in time*

$$\exp(\frac{\Delta t}{2} A_1) \cdots \exp(\frac{\Delta t}{2} A_{n-1}) \exp(\Delta t A_n) \exp(\frac{\Delta t}{2} A_{n-1}) \cdots \exp(\frac{\Delta t}{2} A_1)$$

The proof for three operators can be found in appendix G. The advantage of such a method is to transform six-dimensional systems into successive advection or conservative equations immersed in lower dimensional spaces. This technique has been used more than thirty years ago to solve Vlasov equation in 2D [**124**]. It is now currently used in Eulerian and semi-Lagrangian approaches as detailed in the following.

6.3. Eulerian approach. — The Eulerian approach consists in discretizing the phase space on a fixed grid, and in applying finite differences, finite volumes and/or Fourier transforms to model the differential and integral operators (see *e.g.* [**27**] for a review and figure 8 for a schematic representation). Eulerian schemes are not subject to the issue of marker sampling noise which is critical in Lagrangian-PIC methods. Conversely, when explicit time integration is performed, they are limited by the Courant-Friedrichs-Lewy (CFL) stability condition, which constraints the maximum time step as a function of grid space resolution. Several gyrokinetic codes are based on Eulerian schemes. They are called *Eulerian* (or sometimes *Vlasov*) *codes*. The following list (not exhaustive) names the mostly cited Eulerian codes, as they appeared chronologically in the literature: GS2 [**89, 47**], GYRO [**29**], GENE [**81, 102**], GKV [**147**], GKW [**112**] and GT5D [**78, 77**].

Most of the Eulerian schemes are based on standard time-splitting schemes (see section 6.2), such that the Vlasov solver transforms into successive spatial and velocity space updates [**34**]. Subsequently, without any loss of generality, we can restrict the analysis to the one-dimensional problem

$$\partial_t f + u(x,t) \partial_x f = 0, \qquad \forall (x,t) \in [x_{\min}, x_{\max}] \times \mathbb{R}+ \tag{59}$$

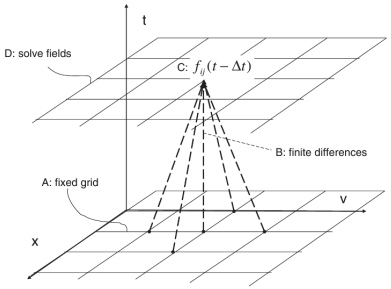


FIGURE 8. In Eulerian methods, a fixed grid is defined in phase space (A). Finite difference expressions are used (B) in order to obtain the value of f at grid points at the next time step (C). Field equations are then solved (D) after integration over velocity space. (figure from [**58**])

The transporting velocity field $u(x,t)$ is assumed to be smooth enough; for instance u is Lipschitz continuous. Besides, taking into account the divergence-free property of the flow $\partial_x u(x,t) = 0$, we will consider the 1D conservative form

$$\partial_t f + \partial_x(u(x,t)f) = 0, \qquad \forall (x,t) \in [x_{\min}, x_{\max}] \times \mathbb{R}+ \tag{60}$$

Then, by analogy with Eqs. (20) and (21), let us introduce the characteristic trajectories which are solutions of the differential system associated to the transport equation

$$\begin{cases} \frac{\mathrm{d}X}{\mathrm{d}t}(t) = u(X(t),t) \\ X(s) = x \end{cases} \tag{61}$$

Notation 1. — *Let us denote $X(t;x,s)$ the solution of Eq.* (61)*, i.e. the position X at time t of the point that was at x at time s.*

Notation 2. — *Let us also introduce the computational 1D domain $\Omega = [x_{\min}, x_{\max}]$ divided by a finite set of mesh points $\{x_{i+1/2}\}_{i\in I}$*

$$x_{\min} = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N-\frac{1}{2}} < x_{N+\frac{1}{2}} = x_{\max}$$

into N Eulerian Control Volumes (ECV) such that

$$ECV_i \equiv \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right] \quad \textit{with centre} \quad x_i \equiv \frac{1}{2}\left(x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}}\right)$$

and unequal spacing

$$\Delta x_i \equiv x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \quad \textit{for} \quad i = 1, 2, \cdots, N$$

Let us integrate (60) with respect to x between two arbitrary moving boundaries $x_1 = x_1(x,t)$ and $x_2 = x_2(x,t)$ and using Leibniz's rule,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\int_{x_1(x,t)}^{x_2(x,t)} f(x,t)\mathrm{d}x - \left(f(x_2,t)\frac{\mathrm{d}x_2}{\mathrm{d}t} - f(x_1,t)\frac{\mathrm{d}x_1}{\mathrm{d}t}\right) \\ + \left(f(x_2,t)u(x_2,t) - f(x_1,t)u(x_1,t)\right) = 0 \end{aligned} \tag{62}$$

If the boundaries x_1 and x_2 are moving with the fluid, i.e if

$$\frac{\mathrm{d}x_1}{\mathrm{d}t} = u(x_1,t), \quad \frac{\mathrm{d}x_2}{\mathrm{d}t} = u(x_2,t)$$

then (62) simplifies to the classical integral conservation form

$$\frac{\mathrm{d}}{\mathrm{d}t}M(x_1,x_2,t) = \frac{\mathrm{d}}{\mathrm{d}t}\left(\int_{x_1(t)}^{x_2(t)} f(x,t)\mathrm{d}x\right) = 0 \tag{63}$$

which simply states that the quantity $M(x_1,x_2,t)$ contained between any two boundaries $x_1(t)$ and $x_2(t)$, moving with the fluid, is invariant in time.

At first time, let us assume that u is independent of both f and x. Then, the solution of the advection equation (59) at time Δt is simply the uniform translation of the distribution function by $u\Delta t$.

Proposition 6. — *There are two particular properties of great importance in an advection solver:*

(a) The method should not introduce false extrema. This is equivalent to saying that the monotonous character of any function should be preserved: for $\lambda = u\Delta t/\Delta x_i$ (λ being the CFL number), if $0 < \lambda < 1$ and $f_{i-1}^n < f_i^n < f_{i+1}^n$ then $f_{i-1}^{n+1} < f_i^{n+1} < f_{i+1}^{n+1}$ (the same should hold for the monotonous decreasing case),

(b) The method should not accentuate already existing extrema: for $0 < \lambda < 1$, if $f_{i-1}^n < f_i^n > f_{i+1}^n$ then $f_i^n \geq \max\{f_i^{n+1}, f_{i+1}^{n+1}\}$.

Properties (a) and (b) together imply that the method is positivity-preserving and total variation diminishing (TVD).

Let us take the example of an uniform mesh of discretization step Δx for simplicity. The difficulty is that f is only known on the discrete grid points $x_j = j\Delta x$. One solution would be to interpolate between the grid points by some function $\tilde{f}(y)$ and write $f(y_i, t+\Delta t) = \tilde{f}(y_i - \lambda, t)$, where $y = x/\Delta x$ (i.e. $y_i = i$) and λ is the distance in y by which f must be shifted. However, by

a corollary of Godunov's theorem[(4)], any interpolation scheme that is higher than first order violates properties (a) and (b). As discussed in the following, satisfying these properties has been one of the leitmotive for the development of new schemes since more than 10 years.

Let us come back to the general case of non constant advection field $u(x,t)$. The goal is to solve equation (60) on a grid, assuming the function f is smooth in each elementary cell I_i. The main idea is to replace $f(x,t)$ by its smoothed approximation (average quantity)

$$\bar{f}_i(t) = \frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} f(x,t)dx, \quad i = 1,2,\cdots,N \tag{64}$$

The values of the approximated distribution function $\bar{f}_i$ are assumed to be known at time $t^n = n\Delta t$ for all $i = 1,\cdots,N$ The new values of $\bar{f}_i$ at time t^{n+1} can be retrieved by using the conservation property (63) of $\int_{x_1(t)}^{x_2(t)} f(x,t)\mathrm{d}x$ where $x_1(t)$ move from $x_{i-1/2}$ (resp. $x_{i+1/2}$) at time t^n to $X(t^{n+1};x_{i-1/2},t^n)$ (resp. $X(t^{n+1};x_{i+1/2},t^n)$) at time t^{n+1}.

$$\bar{f}_i^{n+1} = \bar{f}_i(t^{n+1}) = \frac{1}{\Delta x_i}\int_{X(t^{n+1};x_{i-1/2},t^n)}^{X(t^{n+1};x_{i+1/2},t^n)} f(x,t^n)dx \tag{65}$$

In the case u is a constant, one gets:

$$\bar{f}_i^{n+1} = \frac{1}{\Delta x_i}\int_{x_{i-1/2}-u\Delta t}^{x_{i+1/2}-u\Delta t} f(x,t^n)dx$$

A geometric representation of this process is shown in Figure 9 for u positive.

As far as the general case is concerned, let us further define,

$$\Phi_{i+1/2}(t^n) = \int_{X(t^{n+1};x_{i+1/2},t^n)}^{x_{i+1/2}} f(x,t^n)dx = \int_{x_{i+1/2}-\alpha_i}^{x_{i+1/2}} f(x,t^n)dx \tag{66}$$

where

$$\alpha_i = x_{i+1/2} - X(t^{n+1};x_{i+1/2},t^n)$$

One then finally obtains the conservative scheme

$$\bar{f}_i^{n+1} = \bar{f}_i^n - \frac{1}{\Delta x_i}\left[\Phi_{i+1/2}(t^n) - \Phi_{i-1/2}(t^n)\right] \tag{67}$$

Here, $\Phi_{i\pm1/2}$ represent the exchange of cell i with cells $(i\pm1)$. These fluxes can be either positive or negative. However, mass conservation states that the source terms (resp. losses) for cell i are exactly compensated by the losses (resp. sources) for the neighboring cells.

(4) Godunov's theorem (1954) states that linear numerical schemes for solving partial differential equations (PDE's), having the property of not generating new extrema (monotonous scheme), can be at most first-order accurate.

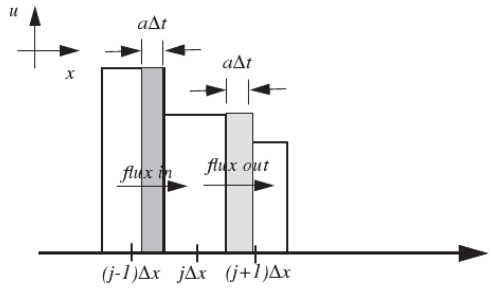


FIGURE 9. Geometric representation of fluxes moving through cell j (Figure taken in [**35**] - For the link with the present text $j \to i$ and $a \to u$).

Although the time-splitting scheme often allows one to replace characteristics curves by straight lines, there are cases where the characteristic curves cannot be computed explicitly. The computation of α_i then requires time discretization of Eqs. (61). The solution proposed in [**55**] for instance, is to use a second-order leap-frog scheme to obtain a fixed point problem where $x^n = X(t^{n+1};x_{i+1/2},t^n)$ is solution of

$$\begin{cases} x_{i+1/2} - x^n = \Delta t u(x^{n+1/2},t^{n+1/2}), \\ t^{n+1/2} = t^n + \frac{\Delta t}{2}, x^{n+1/2} = \frac{1}{2}(x_{i+1/2} + x^n) \end{cases}$$

Such a system can be solved iteratively.

The critical step is now to choose an efficient method to reconstruct the distribution function from the values on each ECV_i.

6.3.1. Flux Balance Method (FBM). — In the Flux Balance Method (FBM), this reconstruction makes use of linear interpolation [**53**]:

$$D_i = \frac{f_{i+1} - f_{i-1}}{2\Delta x_i}$$
$$f_h(x) = f_i + D_i(x - x_i), \qquad \forall x \in [x_{i-1/2}, x_{i+1/2}]$$

then

$$\begin{aligned}\Phi_{i+1/2}(t^n) &= \int_{x_{i+1/2}-\alpha_i}^{x_{i+1/2}} f_h(x,t)dx = \int_{x_{i+1/2}-\alpha_i}^{x_{i+1/2}} \left(f(x_i) + D_i(x-x_i)\right) dx \\ &= \alpha_i f(x_i) + D_i \left[(x-x_i)^2\right]_{x_{i+1/2}-\alpha_i}^{x_{i+1/2}} \\ &= \alpha_i f(x_i) + \frac{D_i}{2}\left[\left(\frac{\Delta x_i}{2}\right)^2 - \left(\frac{\Delta x_i}{2} - \alpha_i\right)^2\right]\end{aligned}$$

And finally,

$$\Phi_{i+1/2}(t^n) = \alpha_i f(x_i) + \frac{D_i}{2}\alpha_i(\Delta x_i - \alpha_i)$$

This FBM scheme is only second-order accurate in space. Besides, there is no guarantee that it preserves monotonicity, and that it does not introduce false extrema. However this method still reveals successful for a variety of Vlasov problems [**53**] when coupled to smoothing and averaging techniques to dissipate fine-scale structures. It is also the starting point of most of Eulerian methods. Several improvements of the FBM method have been proposed, e.g. by Arber and Vann [**5**] in two directions: *(i)* increasing the order of the gradient D_i ; *(ii)* applying Van-Leer gradient limiter (for instance) to the gradient D_i before calculating the cell boundary fluxes. In this paper [**5**], the Piecewise Parabolic Method (PPM) (a new third-order scheme) is compared to second-order spatially accurate schemes (FBM and Van Leer-Limited scheme), and also to fourth order schemes (Flux-Corrected Transport (FCT)) and the sixth-order Compact Finite Difference (Compact). Arber and Vann conclude that maintaining positivity is less important than correctly dissipating fine-scale structures which naturally arise when solving many Vlasov problems.

6.3.2. The Essentially Non-Oscillatory method (ENO). — The essentially non-oscillatory (ENO) schemes, first introduced by Harten and Osher [**70**] and Harten et al. [**69**], can achieve uniformly high-order accuracy with sharp, essentially non-oscillatory shock transitions. In Vlasov equation solution, there is no shock development but ENO scheme can be useful to treat the problem of stiff gradients which appear in phase space. This scheme is based on *Newton Divided Difference* interpolation. The method is detailed in Appendix I. This approach has been implemented by Filbet et al. [**55**] up to the fourth order. Its advantage is to control spurious oscillations. However it does not ensure positivity and Filbet concludes that it is too dissipative to accurately describe the distribution function because the entropy and the L^2 norm are strongly decreasing. Umeda also concludes that WENO (Weighted ENO) schemes are well-designed for shocks and discontinuities but not necessarily appropriate

for the advection of sinusoidal waves. Still, this method has been successfully applied by Qiu and Christlieb [**116**] coupled with the semi-Lagrangian method (described in section 6.4). The third order scheme was shown not to be sufficient, while fifth or seventh-order ones can give satisfactory results.

6.3.3. Positive Flux Conservative scheme (PFC). — This method was introduced in 2001 by Filbet [**55**]. It also uses antiderivative functions for the reconstruction (cf. section 6.3.2). The antiderivative is approximated by Lagrange polynomials. In [**55**] the test is performed for the polynomial of degree two or three on the interval $[x_{i-1/2}, x_{i+1/2}]$. For the sake of simplicity, let us consider that the propagation velocity $u(x,t)$ is positive and second order. Then the approximation is given by:

$$f_h(x) = f_i + \epsilon_i(x-x_i)\frac{f_{i+1}-f_i}{\Delta x_i}, \quad \forall x \in [x_{i-1/2}, x_{i+1/2}] \tag{68}$$

where the slope corrector ϵ_i is defined as

$$\epsilon_i = \begin{cases} \min\left(2(f_i - f_{\min}); (f_{i+1}-f_i)\right) & \text{if } f_{i+1}-f_i \geq 0 \\ \min\left(-2(f_{\max} - f_i); (f_{i+1}-f_i)\right) & \text{if } f_{i+1}-f_i < 0 \end{cases} \tag{69}$$

with $f_{\min} = 0$ and $f_{\max} = f_\infty$. This approximation satisfies the conservation of the average

$$\text{for all } i \in I, \quad \int_{x_{i-1/2}}^{x_{i+1/2}} f_h(\xi)\,\mathrm{d}\xi = \Delta x_i f_i$$

Also, it preserves the maximum principle

$$\text{for all } x \in [x_{\min}, x_{\max}], \quad 0 \leq f_h(x) \leq f_\infty$$

The proof can be found in reference [**55**]. This reconstruction (68) and (69) is then used to compute the fluxes given by Eq. (66). The drawback of such a method, as exemplified in 4D drift-kinetic simulations of ion temperature gradient driven turbulence [**63**], is that it is dissipative, such that the total energy is not conserved.

Several modified versions of the PFC method have been proposed. Schmitz and Grauer [**122**] have modified the limiter, using $f_{\max} = \infty$, to avoid the decrease of the maximum value of the profile and to allow the profile to rise uncontrollably. Unfortunately, in this method, the profile, which can exhibit local maxima and/or minima, is not necessarily non-oscillatory. Umeda [**138**], has defined a non-oscillatory scheme (Positive Interpolation for Conservation), in the sense that the already-existing extrema are kept, by changing the extrema of Filbet's slope corrector by

$$f_{\max} = \max[f_{\max 1}; f_{\max 2}]$$
$$f_{\min} = \max[f_{\min 1}; f_{\min 2}]$$

where

$$\begin{aligned} f_{\max 1} &= \max[\max[f_{i-1}, f_i]; \min[2f_{i-1} - f_{i-2}; 2f_i - f_{i+1}]], \\ f_{\max 2} &= \max[\max[f_{i+1}, f_i]; \min[2f_{i+1} - f_{i+2}; 2f_i - f_{i-1}]], \\ f_{\min 1} &= \min[\min[f_{i-1}, f_i]; \max[2f_{i-1} - f_{i-2}; 2f_i - f_{i+1}]], \\ f_{\min 2} &= \min[\min[f_{i+1}, f_i]; \max[2f_{i+1} - f_{i+2}; 2f_i - f_{i-1}]] \end{aligned}$$

This new PFC scheme has been applied successfully to nonlinear beam-plasma interactions [**118**].

Anyway the choice between one or another method is not simple and depends on the problem being solved. The Flux Balance method (FBM) [**53**], the Positive and Flux Conservative method (PFC) [**55**], the Finite Difference Method (FDM) [**4**], a spectral method [**87, 88**], the Constrained Interpolation Profile scheme (CIP) (see section 6.3.4) [**107**] have been compared in [**54**] in the context of the non-linear 1D1D problem of Landau damping or for the two stream instability problem. Filbet et al. conclude that there is no clear winner, each method having pros and cons. However, there are some indications in the sense that: *(i)* for problems which needs a good precision of the distribution function in regions where its values are small (*e.g.* beam halo problems) the PFC algorithm appears to be more efficient since it preserves positivity without introducing any numerical oscillation ; *(ii)* for problem where the distribution function remains smooth (*e.g.* Landau damping) a high-order method like CIP scheme can be preferred.

6.3.4. The CIP method. — The Constrained Interpolation Profile/Cubic-Interpolated Pseudo-particle algorithm (CIP) has been under development since the middle of the 1980s. The CIP scheme is a low diffusion and stable algorithm which can solve hyperbolic equations up to the 3rd order accuracy in space and time [**139**]. This scheme has been successfully applied to various complex fluid flow problems, covering both compressible and incompressible flows (*e.g.* [**130, 153**]), and to various Vlasov-Poisson problems [**107, 157**] dedicated for instance to simulations of harmonic Langmuir waves [**137**]. The main concept of the CIP, which makes the scheme quite different from the other advection solvers, is to treat the spatial derivatives of the interpolation function as dependent variables. These derivatives serve as free parameters in the interpolating procedure and are calculated consistently with the master equation. In others words, in CIP method, the f value is interpolated by a cubic polynomial as follows,

$$f_h(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + f_i'(x - x_i) + f_i$$

The values of f are known $f_i (i = 1, \cdots, N)$ at all grid points. There are three parameters a_i, b_i and $f_i' \equiv \frac{\partial f}{\partial_x}|_{x_i}$ which need being determined. It has

been shown (*e.g.* [**130**]) that the spatial derivative $\partial_x f$ must be determined by using an expression obtained as a differential operation to the advection equation with respect to spatial variable, i.e.

$$\frac{\partial}{\partial t}\left(\frac{\partial f}{\partial x}\right) + u\frac{\partial}{\partial x}\left(\frac{\partial f}{\partial x}\right) = -\frac{\partial u}{\partial x}\frac{\partial f}{\partial x} \tag{70}$$

Therefore, if f and $\partial_x f$ at all grid points are given by (59) and (70), only two conditions are required to determine the coefficients a and b. Imposing the continuity of both f and $\partial_x f$ at all grid points $i+1$ provides these two additional conditions. One finally obtains:

$$\begin{aligned} a_i &= \frac{(f_i' + f_{i+1}')}{\Delta x_i^2} + \frac{2(f_i - f_{i+1})}{\Delta x_i^3} \\ b_i &= \frac{3(f_{i+1} - f_i)}{\Delta x_i^2} - \frac{(2f_i' + f_{i+1}')}{\Delta x_i} \end{aligned}$$

As a remark, equation (70) is usually solved by time splitting the advection $\partial_t(\partial_x f) + u\partial_x(\partial_x f) = 0$ and non-advection parts $\partial_t(\partial_x f) = -\partial_x u \partial_x f$, the second one being solved with finite differences. This CIP algorithm has been extended to nonlinear [**154**] and multidimensional cases [**155**]. A first variant, based on rational function interpolation (R-CIP), has been proposed by Xiao et al. [**151, 152**]. The advantage of the R-CIP algorithm is that it is convex-concave preserving as well as monotone preserving. It is well established that these CIP-type methods show good conservation of mass, even though they are written in a non-conservative form. However, in some special cases, problems can be encountered when exact conservation of mass is required. For instance, Lesur et al. [**95**] have shown that it can be problematic for energetic beam-driven instability study, both for the CIP and for the R-CIP variant. Indeed, in the Berk-Breizman model (cf. *e.g.* [**8, 9, 140**]), the CIP method leads to numerical oscillations in the velocity direction in the region of large gradient of the distribution –which appears between a cold bulk and a beam. In the case of the Vlasov equation, Nakamura and Yabe [**107, 156**] have proved that it is possible, to improve the CIP method so as to exactly conserve the mass. The CIP-CSL scheme is derived from the standard CIP scheme by constraining the reconstruction of the interpolation. A fourth-order polynomial is chosen as the interpolation function instead of a third order. This permits to add a supplementary constraint on the integrated value of f as

$$\int_{x_i}^{x_{i+1}} f^n dx = \rho_{i+1/2}^n$$

It is shown in [**156**] that this additional constraint improves the conservation property even though f is advanced in a non-conservative form. This permits

the CIP-CSL scheme to ensure a quick convergence without unfavorable numerical oscillations [**95**]. Several variants have been proposed in this sense for multidimensional problems using directional time-splitting [**108**] or not (cf. *e.g.* [**131, 135**]). There are many problems for which time-splitting technique produces unacceptable numerical results. An example given by Levy et al. [**96**] is a pure shear wave oblique to the grid, where the time-splitting procedure, will smear out the oblique wave. In that respect, developing CIP algorithm without time-splitting corresponds to a highly desirable improvement. To conclude CIP type schemes are robust algorithms. It is however important to keep in mind that the additional storage of partial derivatives or integrals can be a significant drawback when applied to five– or even six–dimensional phase-space Vlasov codes.

6.3.5. Morinishi's scheme. — To finish with the description of the Eulerian methods, we will briefly describe the new finite difference operator proposed by Morinishi et al [**104, 106**]. The main particularity of this operator is that it has been constructed to conserve both momentum and kinetic energy. Another point is that it can be easily extended to higher order accuracy. As in the CIP approach, the Morinishi's scheme idea comes from the fact that numerical schemes can be improved by advancing more fields than the distribution function alone. Conversely to the CIP scheme it is f^2 which is involved and not the derivatives of f. Indeed, Arakawa et al. [**4**] have pointed out that numerical instabilities which appear when numerical oscillations due to aliasing errors are produced –linked for instance to inevitable filamentation in velocity space–, may be avoided by conserving the square quantity, which bounds the amplitude of the numerical oscillations. Similar remarks have also been done in more recent gyrokinetic Vlasov simulations [**136, 29**]. The Morinishi's scheme has been successfully applied to incompressible neutral fluid simulations [**104**] and has also been recently adapted to compressible flow equations [**105**]. As far as Vlasov simulations are concerned, this scheme constitutes the backbone of the global full-f gyrokinetic 5D code GT5D [**75**] and has already proved very accurate when running long time simulations.
Let us again consider the simple 1D conservative form of the Vlasov equation $\partial_t f + \partial_x(u(x)f) = 0$. If we introduce the same symbolic notations as in [**104**], this equation can be written in the generic form

$$\frac{\partial f}{\partial t} + (\text{Conv.}) = 0$$

where the quantity (Conv.) may be written in the following three types of conservative forms, the divergence form (Div.), the advection form (Adv.)

and the skew-symmetric form (Skew.):

$$(\text{Div.}) \equiv \frac{\partial(uf)}{\partial x} \tag{71}$$

$$(\text{Adv.}) \equiv u\frac{\partial f}{\partial x} = (\text{Div.}) - f(\text{Cont.}) \tag{72}$$

$$(\text{Skew.}) \equiv \frac{1}{2}\frac{\partial(uf)}{\partial x} + \frac{1}{2}u\frac{\partial f}{\partial x} = (\text{Div.}) - \frac{1}{2}f(\text{Cont.}) \tag{73}$$

where

$$(\text{Cont.}) \equiv \frac{\partial u}{\partial x}$$

(Div.) trivially conserves the $L1$ norm, while it is the case for (Adv.) and (Skew.) if (Cont.) is equal to 0. Then, multiplying the Vlasov equation by f yields the continuity equation for $f^2/2$,

$$\frac{\partial(f^2/2)}{\partial t} + f(\text{Conv.}) = 0$$

where $f(\text{Conv.})$ can also be written using the above three operators,

$$f(\text{Div.}) = \frac{\partial(uf^2/2)}{\partial x} + \frac{1}{2}f^2(\text{Cont.}) \tag{74}$$

$$f(\text{Adv.}) = \frac{\partial(uf^2/2)}{\partial x} - \frac{1}{2}f^2(\text{Cont.}) \tag{75}$$

$$f(\text{Skew.}) = \frac{\partial(uf^2/2)}{\partial x} \tag{76}$$

Therefore, the $L2$ norm is trivially conserved with (Skew.), while (Div.) and (Adv.) are conservative for the $L2$ norm provided that (Cont.) $= 0$. These properties of $L1$ and $L2$ norm conservation are preserved in the discrete form if finite differences are carefully applied. Let us describe the case of a uniform regular grid where both f and u are known at the same grid points (for example with staggered grid system, i.e the velocity u is distributed around the grid point where f is defined, see [**104**]). We use, as in [**104**], the following notations for all quantities A and B,

$$\begin{aligned}
\frac{\delta_n A}{\delta_n x} &\equiv \frac{A(x+nh/2) - A(x-nh/2)}{nh} \\
\bar{A}^n &\equiv \frac{1}{2}A(x+nh/2) + \frac{1}{2}A(x-nh/2) \\
\widetilde{(AB)}^n &\equiv \frac{1}{2}A(x+nh/2)B(x-nh/2) + \frac{1}{2}A(x-nh/2)B(x+nh/2)
\end{aligned}$$

where h is the grid spacing and n an integer. By applying a second-order centered finite difference to Eqs. (71)-(73), the $L1$ norm is automatically conserved by [Div.] while it is true for [Adv.] and [Skew.] only if [Cont.] $= 0$

(the bracket notation [] corresponding to the discrete operators) where the discrete operators are defined as

$$[\text{Div.}] = \frac{\delta_2(uf)}{\delta_2 x}$$

$$[\text{Adv.}] = u\frac{\delta_2 f}{\delta_2 x} = \frac{\delta_1\widetilde{(uf)}^1}{\delta_1 x} - f[\text{Cont.}]$$

$$[\text{Skew.}] = \frac{1}{2}[\text{Div.}] + \frac{1}{2}[\text{Adv.}] = \frac{\delta_1 \bar{u}^1 \bar{f}^1}{\delta_1 x} - \frac{1}{2}f[\text{Cont.}]$$

where

$$[\text{Cont.}] = \frac{\delta_2 u}{\delta_2 x}$$

Similarly the $L2$ norm conservation has to be analysed from the discrete forms of Eqs. (74)-(76),

$$f[\text{Div.}] = \frac{\delta_1 f\widetilde{(uf)}^1}{\delta_1 x} - uf\frac{\delta_2 f}{\delta_2 x}$$

$$f[\text{Adv.}] = 2f[\text{Skew.}] - f[\text{Div.}]$$

$$f[\text{Skew.}] = \frac{\delta_1 \bar{u}^1 \widetilde{(f^2/2)}^1}{\delta_1 x}$$

Therefore, if $[\text{Cont.}] = 0$, the skew-symmetric operator conserves both the $L1$ and $L2$ norms. However, in the case of a non uniform u, the conservation of the $L2$ norm then depends on the time integration scheme. If one uses non-dissipative methods such as the implicit midpoint rule, the $L2$ norm may be exactly conserved. However, such an implicit scheme can be really costly in high dimensionality problems such as 5D gyrokinetic simulations. In this case, fourth-order Runge-Kutta method is preferred even if it leads to small error in the $L2$ norm due to weak numerical dissipation (see [**75**]). Let us notice that this 1D approach can be straightforwardly generalized to higher dimensionality problems and to higher order convective operator (obtained as a combination of the second order one) [**104**].

6.4. Semi-Lagrangian approach. — Semi-Lagrangian schemes have been first used for the advection of vorticity in simplified models of large scale flows. It has gained maturity when the discretization approach was introduced in the relevant context of atmospheric flows. A comprehensive review of semi-Lagrangian methods in this meteorological context until 1990 is due to Staniforth [**127**]. It is also applied to geophysical fluid dynamics (cf. [**48**]). In the following, we focus on the semi-Lagrangian method (SL) applied to plasma turbulence studies. For instance, the SL method has already been applied to

calculate a turbulence driven by passing ions in 2D (1D in space, 1D in velocity) [**126**] and trapped ions in 3D (2D in space, 1D in velocity) [**45, 121**]. More recently, this method was extended to the 4D model (3D in space and $v_\parallel$ (with $\mu = 0$)) of Ion Temperature Gradient (ITG) driven turbulence in cylindrical geometry with the development of the Gysela code (for GYrokinetic SEmi-LAgrangian code) [**63**]. This 4D drift-kinetic slab-ITG version of the code has shown good properties of energy conservation in non-linear regime as well as accurate description of fine spatial scales [**50**]. The gyrokinetic 5D version of the Gysela code is also based on this semi-Lagrangian approach [**64**]. The semi-Lagrangian method is also at the basis of the 4D drift-kinetic code CYGNE [**24**] where the Taylor expansion (see below) is replaced by the Bürlisch-Stoer scheme (for the 2D advection) to increase the spatial accuracy and the logarithmic interpolation technique is used to ensure the positivity of the distribution function. Brunetti et al. have also shown that non-equidistant meshes in radial and parallel velocity directions are a key tool for obtaining accurate results.

The purpose of the semi-Lagrangian method is to take advantage of both the Lagrangian and Eulerian approaches, to accurately describe the phase space, in particular regions where the density is low, as well as enhanced numerical stability. It is based on the fact that the most accurate way to solve convection (or advection) hyperbolic PDE is to use their characteristics along which the distribution function remains constant. This method was primarily developed by Cheng and Knorr [**34**]. It has been cast in more general framework of SL by Sonnendrücker in 1998 [**126**] and Nakamura in 1999 [**107**]. In this approach, the phase-space mesh grid is kept fixed in time (Eulerian method) and the Vlasov equation is integrated along the trajectories (Lagrangian method) using the invariance of the distribution function along the trajectories. To obtain the solution on a phase-space Euler-grid, extrapolation of the distribution function on the neighboring Euler-grid points is then required. Cubic spline interpolation is generally adopted, offering a good compromise between accuracy (small diffusivity) and simplicity (numerical cost) [**55, 13**]. In low-dimensional systems, the semi-Lagrangian method is very efficient. When applied to higher dimensional problems, it face the problem of multidimensional interpolation, which is extremely expensive for high dimensional problems. However, this problem has been partially cured by using the time-splitting idea of Cheng and Knorr [**34**] (see section 6.2). The convergence and stability of such schemes will not be tackled in this paper, for more details see for instance [**11, 13**].

6.4.1. Backward semi-Lagrangian schemes. — The standard semi-Lagrangian approach, described in this section, [**126**] has been recently renamed *Backward semi-Lagrangian approach (BSL)* by its author to make the distinction with a new scheme proposed in 2009 called *Forward semi-Lagrangian approach (FSL)*

[**41**]. The main difference is that the advection equations are solved backward in the first case and forward in the second one. For more details on FSL approach see section 6.4.2. They are both based on the solving of the advective form of Vlasov equation (59) and used the fundamental Vlasov property (22) that the distribution function is constant along its characteristics.

Notation 3. — *Let us consider the computational 1D domain* $[x_{\min}, x_{\max}]$ *and the associated grid (fixed in time) defined by the following finite set of mesh points* $x_i = i\Delta x$ *for all* $i = 0, \cdots, N$ *with* $\Delta x = |x_{\max} - x_{\min}|/N$.

Let us also assume f_n, the distribution function at time t_n, known at each point x_i of this grid. The purpose is to compute the distribution function at time $t_n + \Delta t$ on the same grid.

BSL method consists first in finding for each point x_i the position it had at previous time t_n, i.e $X(t_n; x_i, t_{n+1})$. This is performed by solving backward in time the advection equation (61). Then using (22), the values of f at time $t_{n+1} = t_n + \Delta t$ can be expressed as:

$$f(x_i, t_{n+1}) = f(X(t_n; x_i, t_{n+1}), t_n) \tag{77}$$

where $f(X(t_n; x_i, t_{n+1}), t_n)$ is approximated by interpolation because $f(x, t_n)$ is known on all the mesh points but $X(t_n; x_i, t_{n+1})$ does not necessarily coincide with one of them. This scheme is summarized on Fig. 10 As said before

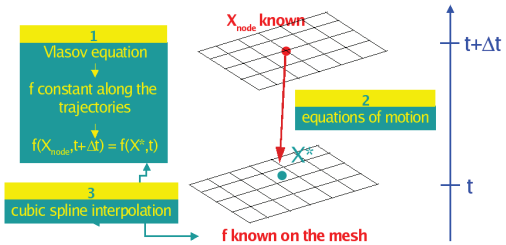


FIGURE 10. Semi-Lagrangian basic concept: a fixed grid is defined in phase space. The orbits are integrated back in time from each grid point (2). The value of f at grid points is obtained by interpolation at the foot of the orbit (3) and using the property $f = \text{const}$ along orbits. (figure from [**63**])

the interpolation currently used is the cubic spline interpolation ([**1, 43**]). Several previous works have shown that cubic interpolation is a good compromise between accuracy and computational cost ([**115, 13**]). The drawback

of cubic spline interpolation is that it does not provide the locality of the reconstruction. All the values of the distribution function are required to interpolate the function at one point. This constraint of global dependency is not adapted to a competitive parallelisation. To overcome this problem, a new approach based on local splines, applied to phase space subdomains, has been developed [**40**] (see Appendix H for more details). The strategy is based on adapted boundary conditions which allow a $\mathcal{C}^1$ reconstructed solution on the global phase space domain. The choice of appropriate Hermite boundary conditions is crucial to obtain a good agreement between a global solution using classical global cubic splines and the new local approach [**40**]. This new approach has been successfully implemented in the GYSELA code (see [**93**] for the parallelism details).

Another difficulty of the BSL approach is to solve backward in time the characteristics defined by $\mathrm{d}X/\mathrm{d}t = u(X(t), t)$ because the field u is only known at time t_n. The field has to be computed iteratively, with Newton fixed point methods or prediction correction algorithms. Indeed, for keeping a scheme of second order, a centered quadrature on two time steps is used (*e.g.* see [**127**])

$$X^{n+1} - X^{n-1} = 2\Delta t\, u^n(X^n) \quad , \quad X^{n+1} + X^{n-1} = 2X^n + \mathcal{O}(\Delta t^2)$$

and X^{n-1} is implicitly computed by using a fixed point procedure such that

$$X^{n+1} - X^{n-1} = 2\Delta t\, u^n\left(\frac{X^{n+1} + X^{n-1}}{2}\right) \tag{78}$$

Theoretically, McDonald ([**98, 99**]) has shown that one should use an interpolation of order one less than for the interpolation used to calculate (77) ; *e.g.* quadratic interpolation of u when using interpolation of f. However in practice, in the context of both passive advection and coupled systems of equations in several spatial dimensions, it has been found that it is sufficient to use linear interpolation for the computation of the displacements, when using cubic interpolation for f (see *e.g.* [**133**]). This technique has also given good results for 4D drift-kinetic turbulence study [**63**].

Finally, the global algorithm in time is a leap-frog algorithm, i.e f^{n+1} is computed from f^{n-1} (see [**63**] for a global scheme description for a 4D problem). The drawback of such an algorithm is that two distribution functions need to be stored at successive times. Besides, an artificial coupling needs to be introduced to avoid even and odd order time approximations to become decoupled after some time. Another solution is to use a predictor-corrector algorithm. Anyway this strategy makes high-order methods quite difficult and expensive. Making the problem explicit enables to get rid of this inconvenient and to use high-order Runge-Kutta methods more easily. This one of the main advantages of the Forward semi-Lagrangian scheme recently proposed by Crouseilles et *al.* [**41**] and described in the following section.

6.4.2. Forward semi-Lagrangian scheme. — As for the BSL approach, the cubic spline are used to approximate distribution functions.

$$f(x, t+\Delta t) = \sum_k c_k(t)\Lambda_k(X(t+\Delta t; x, t)) \tag{79}$$

where Λ_k correspond to the piecewise cubic polynomials defined by (143). In this expression the cubic spline coefficient $c_k(t)$ is associated to the particle located at grid point x_k at time t. It plays the role of the particle weight by analogy to the PIC approach (see section 6.1). All theses cubic spline coefficients are determined by the following interpolation conditions

$$f(x_i, t) = \sum_k c_k(t)\Lambda_k(x_i)$$

Then, taking into account boundary conditions, they can be computed by solving a linear system (see Appendix H).

The full FSL algorithm can be expressed as:

- step 0 : Initialize $f(x_i, t^0)$
- step 1 : Compute the cubic spline coefficients $c_k(t^0)$ such that

$$f(x_i, t^0) = \sum_k c_k(t^0)\Lambda_k(x_i)$$

- step 2 : Integrate the characteristic equation (61) from t^n to t^{n+1} assuming $u(t^n)$ known. This to obtain for each mesh point $X(t^n) = x_i$, the end of the characteristic at time t^{n+1}, i.e $X(t^{n+1}; x_i, t^n)$
- step 3 : Project on the phase space grid using Eq. (79) to get $f(x_i, t^{n+1})$
- step 4 : Compute the cubic spline coefficients $c_k(t^{n+1})$ such that

$$f(x_i, t^{n+1}) = \sum_k c_k(t^{n+1})\Lambda_k(x_i)$$

- Go to step 2 for the next time step

In the case of linear transport equation FSL and BSL methods are identical (see Remark 3.1 in [**41**]). Both methods have been compared for non-linear cases as Vlasov-Poisson model and guiding-center model in the same reference [**41**]. In terms of numerical cost the deposition step (in FSL) and the interpolation step (in BSL) are of same order. The difference come from the computation of the characteristic curves. This step is more expensive in the case of FSL approach due to the fact that a Runge-Kutta of at least 4th order is required to capture non-linear phenomena. This part could be improved by using ODE solvers based on multiple time steps or Taylor expansion. Anyway, the results are in some cases a bit less accurate with respect to the conservation invariants than the one obtained with classical BSL approach, but enable the use of very large time steps without being unstable.

6.4.3. Conservative semi-Lagrangian scheme. — The Parabolic Spline Method (PSM) has been first introduced by Zerroukat et *al.* [**160, 161, 162**] in view of solving conservative transport problems for atmospheric flows. This conservative method has been recently studied in the context of the Vlasov equation by Crouseilles et *al.* [**39**]. The PSM scheme is derived from the Piecewise Parabolic Method [**36**] which has been shown to be more accurate for problems with slow spectral decay such as those occurring in atmospheric flow [**161**]. Conversely to the BSL method, the PSM method deals with the conservative form of the Vlasov equation (i.e (60) in 1D case). The PSM is described in 1D in the following, and it can be generalized to higher dimensions. The PSM scheme uses the conservation property (63) of $\int_{x_1(t)}^{x_2(t)} f(x,t)\mathrm{d}x$ as follows:

$$\int_{x_{i-1/2}(t^{n+1})}^{x_{i+1/2}(t^{n+1})} f(x, t^{n+1})\mathrm{d}x = \int_{x_{i-1/2}(t^n)}^{x_{i+1/2}(t^n)} f(x, t^n)\mathrm{d}x \tag{80}$$

Here, $x_{i\pm1/2}(t^n)$ and $x_{i\pm1/2}(t^{n+1})$ derive from each other by moving along the flow trajectory, namely $x_{i\pm1/2}(t^{n+1}) = x_{i\pm1/2}(t^n) + u_{i\pm1/2}(t^n)(t^{n+1} - t^n)$ at leading order. Two schemes can be a priory derived, depending on whether the points $x_{i\pm1/2}$ correspond to grid points at t^n or at t^{n+1}. The former uses the forward lagrangian scheme, while the latter echoes the backward (in time) semi-lagrangian scheme. Let us focus on the latter one. In this case, $x_{i\pm1/2}(t^n) = X(t^n; x_{i\pm1/2}, t^{n+1})$. Let us again use the definition (64) of the averaged value $\bar{f}_i$ on the ECVi. Then, according to (80),

$$\bar{f}^{n+1}(x_i) \equiv \bar{f}_i^{n+1} = \frac{1}{\Delta x}\int_{X(t^n; x_{i-1/2}, t^{n+1})}^{X(t^n; x_{i+1/2}, t^{n+1})} f(x, t^n)\mathrm{d}x \tag{81}$$

Let us introduce the antiderivative function

$$F^n(z) = \int_{x_{1/2}}^{z} f(x, t^n)\mathrm{d}x \tag{82}$$

with $x_{1/2}$ an arbitrary reference point of the domain. For instance, one can choose the first node of the grid $\{x_{i-1/2}\}_{i=1,N+1}$. Then,

$$\bar{f}_i^{n+1} = \frac{1}{\Delta x}\left[F^n(X(t^n; x_{i+1/2}, t^{n+1})) - F^n(X(t^n; x_{i-1/2}, t^{n+1}))\right] \tag{83}$$

is computed in three steps:

a) Compute $F^n(x_{i+1/2})$ for all $i = 1, \cdots, N$ by using the exact cumulative formula

$$F^n(x_{i+1/2}) = F^n(x_{1/2}) + \sum_{k=1}^{i} \bar{f}_k^n \Delta x$$

b) For each ECV_i, solve the nonlinear system already introduced for BSL scheme (cf. (78)) to compute (backward in time) the associated Lagrangian control volume $\left[X\left(t^n; x_{i-1/2}, t^{n+1}\right), X\left(t^n; x_{i+1/2}, t^{n+1}\right)\right]$:

$$X(t^n; x_{i\pm 1/2}, t^{n+1}) - x_{i\pm 1/2} = u\left(\frac{X(t^n; x_{i\pm 1/2}, t^{n+1}) + x_{i\pm 1/2}}{2}\right)\Delta t$$

c) Use a (cubic spline) interpolation to compute $F^n(X(t^n; x_{i\pm 1/2}, t^{n+1}))$. Notice that the nodal values of F^n are all known. The averaged value of f at time t^{n+1} is then computed on each node of the eulerian grid using eq. (83).

Notice that the mean values $\bar{f}$ are computed at the cell center x_i while antiderivative function values are computed at cell boundaries. This scheme is formally fourth order in space and strictly equivalent to the BSL scheme in the case of directional splitting with constant advection, see [**39**]. It is second order in time assuming when Leap-frog, Predictor-corrector or Runge-Kutta time integration is used. One advantage compared to the BSL scheme is that the mass is conserved because

$$\int_{x_{1/2}}^{x_{N+1/2}} f(x, t^{n+1})\mathrm{d}x = \sum_{k=1}^{N} \bar{f}_k^{n+1}\Delta x$$

Another advantage of such a scheme is that its conservative property allows for directional splitting (a d dimension space can be split in d 1D equations). This property has been used for 4D drift-kinetic simulations of plasma turbulence [**22**] where the splitting in four 1D advective equations is not licite for the BL scheme. Promising results have been obtained but Brauenig et al [**22**] insist on the fact that a special attention must be paid to the maximum principle respect. Besides, like all high-order numerical schemes, it can develop spurious oscillations when stiff gradients occur in the distribution function. In this case, the usual technique for finite volume numerical schemes consists in further applying a flux limiter so as to damp these oscillations. Flux limiters add a diffusion operator to the high order scheme at the locations of steep gradients. A new limiter called Slope Limited Splines (SLS) has been tested in [**66**] and compared to other limiters found in the literature such as, the entropic flux limiter [**22**], the Umeda's limiter [**138**] or the Oscillation Limiter [**39**]. As seen in [**66**], the choice of the best limiter is not trivial. This issue can render the implementation of a stable PSM scheme rather complicate.

6.5. Water-Bag approach. — Introduced initially at the end of the 1960's (*e.g.* [**10**]), the Water-Bag model was shown to create a bridge between kinetic and fluid descriptions. It offers an interesting alternative to the usual kinetic description, using the conservation property of the distribution function in the

phase space. Accordingly, a discrete distribution function is assumed along the velocity direction, taking the form of a multi-step-like function. The distribution function is not an approximation but rather a special class of initial conditions. This approach allows one to reduce the full gyrokinetic 5D equation to a set of hydrodynamic equations while keeping its kinetic character. In a recent review [**52**], Feix and Bertrand detail how the Vlasov equation can be viewed as multi-fluid hydrodynamics via water bag and multi water bag models, at least for one-dimensional problems. These authors show how mathematical similarities between 2D incompressible and the 2D phase space fluid can be used in vortex stability problems. This approach has gained renewed interest when applied to linear drift-kinetic 4D modeling [**103, 60**] and gyrokinetic modeling [**86**]. First non-linear results have been recently obtained by Besse and Bertrand [**14**]. This method will not be further detailed in this paper. More details can be found in the previous citations or also in the references [**12, 65**].

6.6. Problem of filamentation and its numerical treatment. — Vlasov equation is non-dissipative such that entropy is conserved. The free-streaming evolution of the distribution function generates finer and finer filamentation of the distribution function in the phase space, down to sub-grid sizes, leading to strong gradients. The evolution of fine scales can be estimated from the solution of the free-streaming Vlasov equation. When Fourier transformed in real space ($f(x, v, t) \to f(k, v, t)$)

$$f(x, v, t) = \int \frac{\mathrm{d}k}{2\pi}\hat{f}(k, v, t)\exp(i(kt)v)$$

each Fourier component develops fine scales in velocity space as time increases. When $\eta = kt$ reaches the inverse of Δv (Δv being the discretization step in velocity space) one can no longer follow the further filamentation of f and the information is lost (see *e.g.* Büchner [**27**]). The continuous development of fine scale structures represents the transfer of entropy from macro to micro velocity scales (see *e.g.* [**147**]). The physically relevant filamentation can lead to non physical oscillations and numerical instabilities in discrete mathematics. The correct treatment of these gradients is a major issue in numerical schemes. In PIC codes, the filamentation is smeared out by the noise resulting from the finite number of macro-particles. In Eulerian approaches, the use of high-order schemes may be counterproductive in order to prevent this filamentation. Indeed, high-order schemes tend to produce Gibbs overshoot unless some additional averaging is applied. Artificial dissipation –smoothing or filtering– is therefore often introduced (*e.g.* see [**87, 88**] for filtration techniques). A related problem is that there is no guarantee, except for first order schemes, that the numerical solution remains positive everywhere. Both of

these problems are identical to the problems encountered in the treatment of shocks in computational fluid mechanics. Ways of avoiding the restrictions imposed on the order of the scheme by Godunov's theorem in fluid dynamics have a direct analogy in the solution of Vlasov problems. This explains why lots of effort have been made to develop positivity-preserving (see *e.g.* [**54**]) and monotonicity-preserving schemes. Arakawa [**4**] has also shown that numerical instabilities due to this inevitable filamentation may be avoided by conserving the square quantity, which bounds the amplitude of numerical oscillations. Brunetti has shown the interest of PFC method in the case of 4D drift-kinetic simulations [**25**]. Notice that this problem of filamentation becomes less severe when collisions are taken into account, although the numerical treatment of collisions is out of the scope of this lecture.

7. Main physic differences in gyrokinetic codes

7.1. δf codes / full-f codes. — Gyrokinetic models can be split in two distinct families with respect to the adopted representation of the distribution function: either full-f or δf models. In the δf model, only perturbations with respect to some prescribed background equilibrium (usually Maxwellian in velocity) are computed. This method [**44**] has been widely used from the beginning of gyrokinetic code development. Its main advantage is to reduce statistical noise, which can become redhibitory in PIC codes, as explained in section E.3. Technically, the distribution function f is decomposed into the time-independent background f_0 and the time-dependent perturbation δf:

$$f = f_0 + \delta f$$

The background f_0 is the prescribed distribution function at time $t = t_0$. It is one of the stationary solutions of the gyrokinetic equation at vanishing gyro-averaged potential $J_0 \cdot \phi = 0$. Perturbations δf are then governed by the following equation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\delta f = -\frac{\mathrm{d}}{\mathrm{d}t}f_0 = -\frac{\partial}{\partial t}f_0 - \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}\cdot\frac{\partial}{\partial \mathbf{Z}}f_0 = -\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}\cdot\frac{\partial}{\partial \mathbf{Z}}f_0 \tag{84}$$

It is clear that, contrary to the full distribution function f, δf is not conserved along the trajectories. Indeed, the right-hand side of eq. (84) is not equal to zero. As far as f_0 is concerned, it is usually assumed Maxwellian $f_0 = f_{\mathrm{eq}}$ (f_{eq} being an equilibrium Maxwellian distribution function). Trajectories are decomposed into equilibrium and perturbed components, i.e.

$$\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} = \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{eq}} + \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{perturb}}$$

The usual form of the δf equation then reads as follows:

$$\frac{\partial}{\partial t}f_{\mathrm{eq}} = \frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{eq}}\cdot\frac{\partial}{\partial \mathbf{Z}}f_{\mathrm{eq}} = 0 \tag{85}$$

$$\text{and}\quad \frac{\mathrm{d}}{\mathrm{d}t}\delta f = -\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t}|_{\mathrm{perturb}}\cdot\frac{\partial}{\partial \mathbf{Z}}f_{\mathrm{eq}} \tag{86}$$

Should all terms be retained in these equations, they would be rigorously equivalent to the originate gyrokinetic equation. However, let us mention a few simplifications likely leading to discrepancies with respect to the exact gyrokinetic equation: (i) f_0 may not be a *true* equilibrium of the system at vanishing electric field, such that ∂f_0 is not exactly zero (eq. (85)); (ii) a few non-linear terms in the left-hand-side of eq. (86) are sometimes neglected, assuming that they are high order with respect to the right-hand-side terms. In particular, if the flux-surface averaged part of δf is frozen in time, one is left with the so-called gradient-driven regime (see section 7.3).

Conversely, in full-f models, the whole distribution function is evolved. Especially, the back reaction of turbulent transport is accounted for in the time evolution of the equilibrium. Full-f models are specifically capable of addressing the relaxation dynamics of equilibrium profiles, either due to fast non-linear transients or governed by the slow drift of the initial profiles towards their relaxed state in the presence of saturated turbulence. Notice that transients are of great importance in confinement devices. The formation of Internal Transport Barrier in the plasma core is one of such examples. In full-f simulations, the turbulent regime is evanescent if no free energy is injected in the system to prevent the inevitable relaxation of equilibrium profiles below the – linear or non-linear – threshold of the underlying instability. A heat source is mandatory in view of exploring the long time – on energy confinement times – behavior of turbulence and transport. Examples of such heat sources are briefly given in section 7.3.

7.2. Local geometry / global geometry. — Accounting for the toroidal magnetic geometry is not trivial and introduces strong anisotropy in the low frequency perturbations. While $k_{\|}\rho_s$ is ordered as a small quantity in the gyrokinetic ordering (36), the perpendicular wavenumber is not assumed small, $k_{\perp}\rho_s \sim \mathcal{O}(1)$. Fluctuations typically have parallel wavelengths of the order of the system size, whereas perpendicular wavelengths are of the order of a few Larmor radii. Moreover, core plasmas are weakly collisional, characterized by mean free paths larger than the system size. The geometry of the magnetic configuration strongly affects micro-instabilities and turbulence. This strong anisotropy can become an advantage in numerical simulations using field-aligned coordinates instead of toric ones. These coordinates can lead to

an order of magnitude improvement of numerical schemes. A further simplification is made in *flux-tube* codes, in which the considered domain remains in the vicinity of a magnetic field line. Scale separation is assumed, fluctuations being at smaller scale than the equilibrium. In this framework, the equilibrium profiles, and more precisely their gradients,are taken constant in time. In such codes (*e.g.* GS2 [**47**], GYRO [**29**]), periodicity is assumed for the fluctuations along the radial direction. Conversely, *global codes* take the geometry of the whole plasma domain into account. In this case, the radial periodicity assumption is irrelevant. As a result, the delicate problem of radial boundary conditions has to be solved.

The flux-tube approach allows for an efficient reduction of CPU time and memory size consumption. Conversely, modeling the whole plasma domain requires huge meshes. This drawback is counterbalanced by the fact that global codes can describe phenomena such as profile shearing, profile relaxation and large scale transport events such as avalanches (see section 7.3), although the latter have recently been also observed in local simulations. The wealth of such physics has encouraged the development of a new generation of global codes since 15 years. Let us mention, for instance, ORB5 [**83**], GT5D [**78**] or GYSELA [**64**] for core plasmas, and XGC1 [**32, 33**] for edge plasmas. Even the flux-tube code GENE [**81**] has recently given birth to a global version [**92**].

7.3. Fixed gradient / flux driven systems. — Let us close this section with differences in the treatment of the boundary conditions. As already mentioned, flux-tube codes use periodic boundary conditions in all directions. As far as global codes are concerned, periodicity is lost in the radial direction. Until recently, gyrokinetic simulations set the system out of equilibrium by imposing two thermal baths as radial boundary conditions. As shown in [**64**], the relaxation of the mean profiles in the center of the radial domain can lead to strong gradients at the radial edges, unless buffer regions are added. This evolution stops when the core gradients reach the instability threshold. It typically takes place on energy confinement time scales. Limited statistics are available to investigate the physics of scaling laws in this case, unless running very small ρ^* simulations. Alternatively, an ad-hoc heat source can be added to force the gradients out of thermodynamical equilibrium. The impact of the characteristics of the source on the turbulent transport properties is certainly an issue. Several types of sources have been developed. A Krook operator is used in [**100, 101**]. In this case, however, the driving flux is not prescribed *a priori*: since the Krook term depends on the actual distribution function, it evolves in time. Conversely, implementing a heat source which is independent of the distribution function provides a way to study forced turbulence at constant-in-time incoming flux. Gradients then self-adjust in response to this flux as a result of turbulent (and possibly collisional) transport. Such a forcing

was successfully exploited in fluid simulations of turbulent transport. Let us remark that, in gyrokinetics, the source is *a priori* 5-dimensions. In particular, its expression in the velocity space has to be considered. Its precise choice depends on the number of fluid moments one wishes to excite. A nice way to decouple the various fluid moments of the source (e.g. source of particles, of momentum and of heat) consists in projecting the velocity-dependent part of the source on the orthogonal basis of Hermite and Laguerre polynomials, of the form

$$S \sim \sum_{\ell=0}^{+\infty}\sum_{h=0}^{+\infty} c_{h\ell} H_h(\bar{v}_\parallel) L_\ell(\bar{\mu}) \mathrm{e}^{-\bar{v}_\parallel^2-\bar{\mu}}$$

where the $c_{h\ell}$ coefficients depend on space coordinates only. Here, $\bar{v}_\parallel$ and $\bar{\mu}$ account for the normalized parallel and transverse velocities: $\bar{v}_\parallel = v_\parallel/\sqrt{2T_s/m}$ and $\bar{\mu} = \mu B/T_s$, with T_s the source temperature. A general feature of flux-driven turbulence simulations is that the turbulent transport exhibits avalanche-like events characterized by large scale intermittent outbursts. This avalanche-like transport had already been documented in fluid simulations [**59, 15**] and has also been reported experimentally in core plasmas of tokamaks [**150**] and of helical devices [**80**]. First calculations of gyrokinetic flux driven systems [**42**] have been performed with a reduced gyrokinetic model for deeply trapped ion turbulence [**45, 121**]. This system is 2D in configuration space, namely the radial coordinate and the angle, and 1D in velocity space, the chosen coordinate being the energy. The same kind of source has recently been implemented in GYSELA code [**120**]. Again, it is shown that these bursts propagate almost ballistically on large radial distances, much larger than the Eulerian correlation length of turbulence. These flux-driven conditions are now addressed in most of global codes, leading to statistical steady-state turbulent regimes [**101, 77**]. First comparisons have been performed this year [**143**].

In this paper, only electrostatic gyrokinetic ion turbulence with adiabatic electron response has been discussed. Such simulations already require large HPC resources. As an example, the full-f global simulation performed with the GYSELA code, at the ITER relevant ρ^* dimensionless parameter $\rho^* = 1/512$, has required 272.10^9 grid points to discretize the 5D phase space $(N_r \times N_\theta \times N_\varphi \times N_{v_\parallel} \times N_\mu = 1024 \times 1024 \times 128 \times 128 \times 16)$ to model ion plasma turbulence on quarter of torus. 1 millisecond of plasma was captured by this simulation which ran during about 31 days on 8192 processors. This is equivalent to about 6.1 million hours on a single processor. Even though the global and full-f gyrokinetic simulation of an ITER-size plasma including both kinetic electrons and electromagnetic fluctuations is still a long way ahead, the

route is already undertaken. Non-adiabatic electron response and electromagnetic effects have given rise to first results [**109, 28, 62**]. The simultaneous treatment of both kinetic ions and kinetic electrons poses new challenges to the numerical simulations. The characteristic time step should be reduced by a factor of the order of the square root of the mass ratio, while the radial and poloidal grid resolution has to be typically increased by up to one order of magnitude as compared to the adiabatic electron case. Up to now, due to computational resource constraints, these simulations are often run with an artificially small mass ratio.

When dealing with electromagnetic effects, the Vlasov-Poisson solver has to be replaced by the Vlasov-Maxwell solver. The modification is far from trivial. First nonlinear electromagnetic global simulations have been recently performed using the ORB5 code [**20**] with gyrokinetic ions and drift-kinetic electrons with $m_i/m_e = 1000$ (for $\rho^* = 1/160$). For such a simulation, the time step was reduced by a factor 20, and the radial resolution was increased by a factor 4 as compared to the adiabatic electron case.

8. Conclusion

Hot magnetized plasmas such as those encountered in controlled magnetic fusion devices require a kinetic description because of their low collisionality. Fortunately, turbulence studies can be performed within a reduced phase space of five dimensions, by averaging out the fast gyro-motion of charged particles around the equilibrium magnetic field lines. Solving 5D gyrokinetic equations for each species still requires state-of-the-art high performance computing techniques, involving massively parallel computers and parallel scalable algorithms. In this paper, details of the main numerical schemes which have been explored until now have been reviewed. Lagrangian, Eulerian or semi-Lagrangian schemes exhibit both advantages and drawbacks which have been highlighted. As a concluding remark, it should be kept in mind that the high-end HPC power requires a continuing adaptation of the numerical algorithms in strong link with the care of parallelization optimization. This clearly poses non trivial challenges.

Appendix A

Guiding-center motion

The dynamics of a non relativistic particle s, of charge e_s and mass m_s, obeys Newton's equation. If an electromagnetic field is present, the Lorentz

force usually largely dominates over gravity:

$$m_s \frac{\mathrm{d}\mathbf{v}_s}{\mathrm{d}t} = e_s\{\mathbf{E}(\mathbf{x},t) + \mathbf{v}_s \times \mathbf{B}(\mathbf{x},t)\} \tag{87}$$

Such an equation is by essence non linear, since the fields have to be taken at the instantaneous position of the particle: $\mathbf{v}_s = \mathrm{d}\mathbf{x}/\,\mathrm{d}t$. In turn, it does not admit integrable solutions in the general case, for arbitrary electric $\mathbf{E}$ and magnetic $\mathbf{B}$ fields. Conversely, strongly magnetized systems allow for a perturbative treatment, which is known as the *adiabatic theory*. Such systems are characterized by slowly varying electromagnetic fields, as compared to the cyclotron frequency $\Omega_s = e_s B/m_s$. The small parameter ε of the development is:

$$\varepsilon = \left|\frac{\partial \log B}{\Omega_s\, \partial t}\right| \sim \frac{|\mathbf{v}_\perp \cdot \nabla_\perp \log B|}{\Omega_s} \sim \frac{\left|v_\parallel \nabla_\parallel \log B\right|}{\Omega_s} \ll 1 \tag{88}$$

with $v_\parallel \equiv \mathbf{v}.\mathbf{b}$, and $\mathbf{b} \equiv \mathbf{B}/B$. Within the framework of the adiabatic theory, let us decompose the particle motion in a fast dynamics of typical frequency Ω_s, and a slow motion on larger time scales. The later quantity can be obtained by averaging over a cyclotron motion: $\langle \mathbf{v} \rangle \equiv \oint \mathbf{v}\, \frac{\mathrm{d}\varphi_c}{2\pi}$, where φ_c stands for the cyclotron phase. Consistently, fast variables – denoted with a tilde in this appendix – have a zero mean: $\mathbf{B} = \langle \mathbf{B} \rangle + \tilde{\mathbf{B}}$, $\mathbf{E} = \langle \mathbf{E} \rangle + \tilde{\mathbf{E}}$ and $\mathbf{v} = \langle \mathbf{v} \rangle + \tilde{\mathbf{v}}$, with $\langle \tilde{\mathbf{B}} \rangle = \langle \tilde{\mathbf{E}} \rangle = \langle \tilde{\mathbf{v}} \rangle = 0$. From now on, $\mathbf{v}_G$ refers to $\langle \mathbf{v} \rangle$: $\mathbf{v}_G \equiv \langle \mathbf{v} \rangle$. As will become clear in the following, the averaged motion refers to that of the guiding-center, hence the subscript "G". Equation (88) implies that $\tilde{B}$ is much weaker than its corresponding averaged component $\langle B \rangle$: $\tilde{B}/\langle B \rangle \sim \varepsilon$. Furthermore, at leading order, $\langle \mathbf{B} \rangle$ can be approximated by its value at the position of the guiding-center $\langle \mathbf{B} \rangle \simeq \mathbf{B}(\mathbf{x}_G) \equiv \mathbf{B}_G$ [(5)]. Conversely, there is no such a hierarchy for the velocities, $\tilde{v}$ and v_G being of the same order of magnitude *a priori*. Accounting for all these definitions, equation (87) is then equivalent to the following system:

$$m_s \frac{\mathrm{d}\mathbf{v}_G}{\mathrm{d}t} = e_s \left\{ \langle \mathbf{E} \rangle + \mathbf{v}_G \times \mathbf{B}_G + \langle \tilde{\mathbf{v}} \times \tilde{\mathbf{B}} \rangle \right\} \tag{89}$$

$$m_s \frac{\mathrm{d}\tilde{\mathbf{v}}}{\mathrm{d}t} = e_s \left\{ \tilde{\mathbf{E}} + \tilde{\mathbf{v}} \times \mathbf{B}_G + \mathbf{v}_G \times \tilde{\mathbf{B}} + \tilde{\mathbf{v}} \times \tilde{\mathbf{B}} - \langle \tilde{\mathbf{v}} \times \tilde{\mathbf{B}} \rangle \right\} \tag{90}$$

Let us first focus on the fast dynamics. Given the large magnitude of the magnetic field in fusion plasmas, the Coulomb force has a much weaker magnitude

[(5)] Let $\mathbf{x}_G$ be the guiding center position, and $\mathbf{x}$ that of the particle, such that $\mathbf{x} = \mathbf{x}_G + \boldsymbol{\rho}_s$. Any vector field $\mathbf{P}(\mathbf{x})$ can then be Taylor expanded as follows: $\mathbf{P}(\mathbf{x}) = \exp(\boldsymbol{\rho}_s \cdot \boldsymbol{\nabla})\, \mathbf{P}(\mathbf{x}_G)$. Noticing that $\boldsymbol{\rho}_s$ only depends on the gyro-phase φ_c, the gyro-average of $\mathbf{P}(\mathbf{x})$ then simply reads: $\langle \mathbf{P}(\mathbf{x}) \rangle = \langle \exp(\boldsymbol{\rho}_s \cdot \boldsymbol{\nabla}) \rangle\, \mathbf{P}(\mathbf{x}_G) = \sum_{n=0}^{\infty} \frac{1}{2n!} \left\langle (\boldsymbol{\rho}_s \cdot \boldsymbol{\nabla})^{2n} \right\rangle\, \mathbf{P}(\mathbf{x}_G)$. Approximating $\langle \mathbf{B} \rangle$ with $\mathbf{B}_G$ is then equivalent to neglecting high order contributions in the summation, namely terms with $n \geq 1$.

than the Laplace force [6]. Consequently, the dominant term in the right hand side of eq. (90) is the second one. At leading order, eq. (90) then reduces to:

$$m_s \frac{\mathrm{d}\tilde{\mathbf{v}}}{\mathrm{d}t} = e_s \tilde{\mathbf{v}} \times \mathbf{B}_G \tag{91}$$

Here, $\mathbf{B}_G$ does no longer depend on the instantaneous position of the particle $\mathbf{x}(t)$. This makes eq. (91) integrable. This is nothing but the cyclotron motion. Integrating once in time, one obtains: $\tilde{\mathbf{v}} = \boldsymbol{\rho}_s \times \boldsymbol{\Omega}_s$, with $\boldsymbol{\rho}_s$ the Larmor radius ($\rho_s = m_s v_\perp / e_s B_G$ and $\mathrm{d}\boldsymbol{\rho}_s / \mathrm{d}t \equiv \tilde{\mathbf{v}}$) and $\boldsymbol{\Omega}_s = e_s \mathbf{B}_G / m_s$ the vector of the cyclotron pulsation.

Let us now consider the averaged motion. The last term on the right hand side of eq. (89) remains to be evaluated. At lowest order in ε, $\tilde{\mathbf{B}}$ can be approximated by the first term of its Taylor expansion:

$$\tilde{\mathbf{B}} = (\boldsymbol{\rho}_s \cdot \boldsymbol{\nabla}) \mathbf{B}_G$$

Using the tensorial expression of the cross-product, straightforward algebra leads to:

$$\begin{aligned} \langle \tilde{\mathbf{v}} \times \tilde{\mathbf{B}} \rangle &= \langle (\boldsymbol{\rho}_s \times \boldsymbol{\Omega}_s) \times (\boldsymbol{\rho}_s \cdot \boldsymbol{\nabla}) \mathbf{B}_G \rangle \\ &= \langle \rho_k \Omega_i \, \rho_p \nabla_p B_k \, \mathbf{e}_i - \rho_i \Omega_k \, \rho_p \nabla_p B_k \, \mathbf{e}_i \rangle \end{aligned}$$

where indices "G" and "s" have been omitted for clarity. Here, the subscripts refer to one of the three coordinates of a cartesian vector basis, with the convention that doubled symbols are summed (e.g. $\boldsymbol{\rho}.\nabla = \rho_p \nabla_p$). The vector $\boldsymbol{\rho}_s$ is characterized by the following coordinates transverse to $\mathbf{B}_G$: $\boldsymbol{\rho}_s = \rho_s (\cos\varphi_c \, \mathbf{e}_i + \sin\varphi_c \, \mathbf{e}_j)$. It follows that $\langle \rho_i \rho_j \rangle = \frac{1}{2} \rho_s^2 \, \delta_{ij}^\perp$, where $\delta_{ij}^\perp$ stands for the modified Kronecker symbol, equal to 1 if both i and j are in the transverse plane and if $i \neq j$, and 0 otherwise. Hence, only those indices $k = p$ and $i = p$ in the first and second terms, respectively, lead to non zero averages:

$$\langle \tilde{\mathbf{v}} \times \tilde{\mathbf{B}} \rangle = \frac{\Omega_s \rho_s^2}{2} \left[\mathbf{b} \left(\boldsymbol{\nabla}_\perp \cdot \mathbf{B}_{G\perp} \right) - \boldsymbol{\nabla}_\perp B_G \right]$$

Then, since $\boldsymbol{\nabla}.\mathbf{B}_G = 0$, it turns out that $\boldsymbol{\nabla}_\perp \cdot \mathbf{B}_{G\perp} = -\nabla_\parallel B_{G\parallel} = -\nabla_\parallel B_G$. One finally gets:

$$\langle \tilde{\mathbf{v}} \times \tilde{\mathbf{B}} \rangle = -\frac{\mu_s}{e_s} \boldsymbol{\nabla} B_G$$

Equation (89) can then be expressed in terms of the averaged, slowly varying quantities, only:

$$m_s \frac{\mathrm{d}\mathbf{v}_G}{\mathrm{d}t} = e_s \left(\langle \mathbf{E} \rangle + \mathbf{v}_G \times \mathbf{B}_G \right) - \mu_s \boldsymbol{\nabla} B_G \tag{92}$$

[6] This characterizes magnetized kinetic plasmas, where the Coulomb potential energy is much smaller than the plasma kinetic energy.

The last term on the right hand side represents the drag force exerted on the guiding center by the small inhomogeneity of the magnetic field at the Larmor radius scale. $\mathbf{v}_G$ represents the guiding-center velocity. It can be decomposed into parallel and transverse components:

$$\mathbf{v}_G \equiv v_{G\parallel} \mathbf{b} + \mathbf{v}_{G\perp}$$

Equation (92) can be used to obtain the expressions of both $v_{G\parallel}$ and $\mathbf{v}_{G\perp}$. In the remainder, the subscript G will be dropped for the magnetic field.

A.1. Transverse drifts. — Projecting eq. (92) onto the transverse plane yields[7]:

$$m_s \left. \frac{\mathrm{d}\mathbf{v}_{G\perp}}{\mathrm{d}t} \right|_\perp + m_s \left. \frac{\mathrm{d}\mathbf{v}_{G\parallel}}{\mathrm{d}t} \right|_\perp = e_s \left(\langle \mathbf{E} \rangle_\perp + \mathbf{v}_G \times \mathbf{B} \right) - \mu_s \boldsymbol{\nabla}_\perp B \tag{93}$$

At leading order, the first term on the left hand side can be neglected, as can be checked *a posteriori* (it turns out to be smaller by a factor ε with respect to the second term on the right hand side). The second term accounts for the curvature of the magnetic field lines, which generates a centrifugal force. It reads:

$$\left. \frac{\mathrm{d}\mathbf{v}_{G\parallel}}{\mathrm{d}t} \right|_\perp = v_{G\parallel} \frac{\mathrm{d}\mathbf{b}}{\mathrm{d}t} = {v_{G\parallel}}^2 \nabla_\parallel \mathbf{b} = {v_{G\parallel}}^2 \frac{\mathbf{N}}{R}$$

where $\mathbf{N}$ is the unit vector normal to magnetic field lines and R their curvature radius. Taking the cross product of eq. (93) with $\mathbf{B}$ leads to the transverse drift velocity in the adiabatic limit, at lowest order:

$$\mathbf{v}_{G\perp} = \frac{\langle \mathbf{E} \rangle \times \mathbf{B}}{B^2} + \frac{\mathbf{B}}{e_s B^2} \times \left[\mu_s B \, \frac{\boldsymbol{\nabla} B}{B} + m_s {v_{G\parallel}}^2 \frac{\mathbf{N}}{R} \right] \tag{94}$$

Further noticing that $\mathbf{N}/R = (\mathbf{b}.\boldsymbol{\nabla})\mathbf{b} = \boldsymbol{\nabla}_\perp \log B + (\mu_0/B) \mathbf{j} \times \mathbf{b}$, eq. (94) also reads:

$$\mathbf{v}_{G\perp} = \frac{\langle \mathbf{E} \rangle \times \mathbf{B}}{B^2} + \frac{m_s {v_{G\parallel}}^2 + \mu_s B}{e_s B} \, \frac{\mathbf{B} \times \boldsymbol{\nabla} B}{B^2} + \frac{m_s {v_{G\parallel}}^2}{e_s B^2} \, \mu_0 \mathbf{j}_\perp \tag{95}$$

The first term is the electric drift velocity $\mathbf{v}_E$. In the electrostatic limit, $\langle \mathbf{E} \rangle = -\boldsymbol{\nabla} \langle \phi \rangle$, and $\mathbf{v}_E$ is equal to $\mathbf{v}_E = \mathbf{B} \times \boldsymbol{\nabla} \langle \phi \rangle / B^2$. The two last terms, denoted $\mathbf{v}_{d,s}$ hereafter, are the magnetic drifts. They are made of the so-called *grad-B* and curvature drifts. They are of the order of $(T_s / e_s B R)$ for thermal particles. In tokamaks, these drifts are essentially along the vertical direction. They lead to vertical charge separation, ions and electrons drifting in opposite directions. By using the condition for magnetic equilibrium in tokamaks, stating that $\mathbf{j} \times \mathbf{B} = \boldsymbol{\nabla} p$ (with $\mathbf{j}$ the plasma current and p its pressure), it can be shown that the last term on the right hand side of eq.

[7] Such a projection corresponds to taking the scalar product with the transverse tensor $\mathbb{I}_\perp = \mathbb{I} - \mathbf{b} \otimes \mathbf{b}$, with $\mathbb{I}$ the unit tensor of rank 2 and $\otimes$ the exterior product.

(95) is smaller than the second one by the factor $\beta \equiv p/(B^2/2\mu_0)$, the ratio of kinetic to magnetic energy (μ_0 the permeability of free space). This beta parameter barely exceeds a few percents in tokamaks.

A.2. Parallel dynamics. — The parallel projection of eq. (92) yields:

$$m_s \frac{\mathrm{d}\mathbf{v}_{G\|}}{\mathrm{d}t} \cdot \mathbf{b} + m_s \frac{\mathrm{d}\mathbf{v}_{G\perp}}{\mathrm{d}t} \cdot \mathbf{b} = -e_s \nabla_\| \langle\phi\rangle - \mu_s \nabla_\| B \tag{96}$$

The first term on the left hand side reduces to $m_s\, \mathrm{d}v_{G\|}/\,\mathrm{d}t$. The second term can be evaluated as follows:

$$\frac{\mathrm{d}\mathbf{v}_{G\perp}}{\mathrm{d}t} \cdot \mathbf{b} = \left\{ \left(\mathbf{v}_{G\perp} \cdot \boldsymbol{\nabla}_\perp + v_{G\|} \nabla_\| \right) \ \mathbf{v}_{G\perp} \right\} .\mathbf{b}$$

Since it is proportional to $v_{G\perp}^2$ (with $v_{G\perp} \equiv \|\mathbf{v}_{G\perp}\|$), the first term on the right hand side of this development, namely $(\mathbf{v}_{G\perp} \cdot \boldsymbol{\nabla}_\perp)\mathbf{v}_{G\perp}$, can be neglected. This simplification is consistent with the one made earlier, when $(\,\mathrm{d}\mathbf{v}_{G\perp}/\,\mathrm{d}t)_\perp$ was neglected in the derivation of $\mathbf{v}_{G\perp}$ at leading order. Using properties of the cross-product and the expression of $\mathbf{v}_{G\perp}$ eq. (95), the later expression can be transformed as follows in the electrostatic limit:

$$\begin{aligned} \frac{\mathrm{d}\mathbf{v}_{G\perp}}{\mathrm{d}t} \cdot \mathbf{b} &\simeq v_{G\|} \left\{ (\mathbf{b} \cdot \boldsymbol{\nabla})\, \mathbf{v}_{G\perp} \right\} \cdot \mathbf{b} \\ &= -v_{G\|} \left\{ \mathbf{v}_{G\perp} \times \left(\frac{\mathbf{B} \times \boldsymbol{\nabla} B}{B^2} + \frac{\mu_0 \mathbf{j}}{B} \right) \right\} \cdot \mathbf{b} \\ &= v_{G\|} \left\{ \frac{\mathbf{B} \times \boldsymbol{\nabla} B}{B^3} \cdot \boldsymbol{\nabla}\langle\phi\rangle + \frac{\mu_0 \mathbf{j}_\perp}{B^2} \cdot \left(\boldsymbol{\nabla}\langle\phi\rangle + \frac{\mu_s}{e_s} \boldsymbol{\nabla} B \right) \right\} \end{aligned}$$

Introducing the quantity $\boldsymbol{\nabla}\Xi = \mu_s \boldsymbol{\nabla} B + e_s \boldsymbol{\nabla}\langle\phi\rangle$, the parallel dynamics of the guiding-center finally reads:

$$\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t} = -\frac{1}{m_s} \left(\mathbf{b} + \frac{m_s v_{G\|}}{e_s B^2} \mu_0 \mathbf{j}_\perp \right) \cdot \boldsymbol{\nabla}\Xi - v_{G\|} \frac{\mathbf{B} \times \boldsymbol{\nabla} B}{B^3} \cdot \boldsymbol{\nabla}\langle\phi\rangle \tag{97}$$

The term depending on the transverse current $\mathbf{j}_\perp$ is smaller than the other ones by the ratio β. All others are *a priori* of the same order of magnitude.

A.3. Consistency with respect to standard formulation. — In this section, we show that the previous derivation agrees with the standard formulation of gyrokinetic theory, up to small terms proportional to the parallel plasma current. The standard formulation of the guiding center motion, as

presented in eqs. (38)-(39), can be recast as follows:

$$\begin{aligned} \frac{\mathrm{d}\mathbf{x}_G}{\mathrm{d}t} &= v_{G\|} \mathbf{b} + \frac{\mathbf{B} \times \boldsymbol{\nabla}\langle\phi\rangle}{B B_\|^*} \\ &\quad + \frac{m_s {v_{G\|}}^2 + \mu_s B}{e_s B} \frac{\mathbf{B} \times \boldsymbol{\nabla} B}{B B_\|^*} + \frac{m_s {v_{G\|}}^2}{e_s B B_\|^*} \mu_0 \mathbf{j}_\perp \end{aligned} \tag{98}$$

$$\frac{\mathrm{d}v_{G\|}}{\mathrm{d}t} = -\frac{B}{m_s B_\|^*} \left(\mathbf{b} + \frac{m_s v_{G\|}}{e_s B^2} \mu_0 \mathbf{j} \right) \cdot \boldsymbol{\nabla}\Xi - v_{G\|} \frac{\mathbf{B} \times \boldsymbol{\nabla} B}{B^2 B_\|^*} \cdot \boldsymbol{\nabla}\langle\phi\rangle \tag{99}$$

We recall that $B_\|^* \equiv B + (m_s v_{G\|}/e_s)\,(\boldsymbol{\nabla} \times \mathbf{b}).\mathbf{b} = B\{1 + (m_s v_{G\|}/e_s B^2)\mu_0 j_\|\}$. In the limit $(m_s v_{G\|}/e_s B^2)\mu_0 j_\| \ll 1$, the first term on the right hand side of eq. (99) leads to $\frac{-1}{m_s}[\mathbf{b} + (m_s v_{G\|}/e_s B^2)\mu_0 \mathbf{j}_\perp] \cdot \boldsymbol{\nabla}\Xi$. It then appears that the equations of motion derived from Newton's law, eqs. (95)-(97), and those derived from the phase-space reduction in the Hamiltonian formalism, eqs. (98)-(99), are identical in the limit $(m_s v_{G\|}/e_s B^2)\mu_0 j_\| \to 0$. This would correspond to approximating $B_\|^*$ by B. Such a simplification looks reasonable since this ratio is indeed very small in tokamaks: $(m_s v_{G\|}/e_s B^2)\mu_0 j_\| \sim \rho_*(a/qR)$ for thermal particles, with $\rho_* = (\rho_s/a) \approx 2.10^{-3}$ and $(a/qR) \approx 1/6$ in ITER plasmas. However, keeping such small terms is essential to preserve the Hamiltonian formulation of the problem, thus insuring its conservation properties.

Appendix B
Details of the derivation of particle density from the gyro-center distribution function

B.1. Relationship between $f_s(\mathbf{x}, \mathbf{v}, t)$ and $\bar{f}_s(\mathbf{x}_G, \mathbf{v}_G, t)$. — The old canonical coordinates $(\mathbf{x}, \mathbf{p})$ – that of the particle – and the new ones $(\mathbf{x}', \mathbf{p}')$ – those of the guiding-center, only differ by infinitesimal quantities of order $\epsilon \sim \rho^*$. In such a case, they can be shown to be related, at first order in the small parameter ϵ, by the generating function $S(\mathbf{x}, \mathbf{p})$[8] (which remains to be determined) as follows [**61**] :

$$\begin{aligned} \mathbf{x} - \mathbf{x}' &= \partial_{\mathbf{p}'} S \\ \mathbf{p} - \mathbf{p}' &= -\partial_{\mathbf{x}'} S \end{aligned}$$

Any field f evaluated at the position $(\mathbf{x}, \mathbf{p})$ can then be expressed in terms of its value at $(\mathbf{x}', \mathbf{p}')$, namely:

$$f(\mathbf{x}, \mathbf{p}) = f(\mathbf{x}', \mathbf{p}') + [f, S]_{\mathbf{x}', \mathbf{p}'} + o(\epsilon) \tag{100}$$

[8] Rigorously speaking, the generating function G is $G = \mathbf{x}.\mathbf{p}' + \epsilon S$, where $\mathbf{x}.\mathbf{p}'$ can be shown to be the identity transform.

This is especially the case for the Hamiltonian H:

$$H(\mathbf{x},\mathbf{p}) = H(\mathbf{x}',\mathbf{p}') + [H,S]_{\mathbf{x}',\mathbf{p}'} + o(\epsilon) \tag{101}$$

Since the generating function does not explicitly depend on time, $H(\mathbf{x}',\mathbf{p}')$ also stands for the new Hamilonian with respect to which the new coordinates are canonically conjugated. To avoid any confusion with H, we will denote it $\bar{H}(\mathbf{x}',\mathbf{p}')$ hereafter. As far as the gyrokinetic theory is concerned, the transformation acts on the first pair of conjugate coordinates, namely (φ_c, J_1), with $J_1 = -m_s\mu_s/e_s$. From eq. (101), it comes:

$$\begin{aligned} H(\mathbf{x},\mathbf{p}) &= \bar{H}(\mathbf{x}_G,\mathbf{p}_G) + [\bar{H},S]_{\varphi_c,J_1} \\ &= \bar{H}(\mathbf{x}_G,\mathbf{p}_G) + \Omega_s\,\partial_{\varphi_c}S + \partial_{\varphi_c}\bar{H}\,\partial_\mu S \end{aligned} \tag{102}$$

with $\Omega_s = (e_s/m_s)\partial_\mu\bar{H}$ the cyclotron frequency. The additional imposed constraint is that $\bar{H}$ should not depend on the gyro-angle φ_c. In this case, the last term on the right hand side of eq. (102) vanishes. This allows one to express the generating function S as function of the old and new Hamiltonians:

$$\begin{aligned} S(\mathbf{x},\mathbf{p}) &= \int \frac{\mathrm{d}\varphi_c}{\Omega_s}\,\left\{H(\mathbf{x},\mathbf{p}) - \bar{H}(\mathbf{x}_G,\mathbf{p}_G)\right\} \\ &= \int \frac{m_s\,\mathrm{d}\varphi_c}{B}\,\left\{\phi(\mathbf{x}) - \bar{\phi}(\mathbf{x}_G,\mathbf{p}_G)\right\} \end{aligned} \tag{103}$$

Injecting the expression of S, eq. (103), in equation (100), the distribution function of particles f_s can then be related to the one of the gyro-centers $\bar{f}_s$:

$$f_s(\mathbf{x},\mathbf{v},t) = \bar{f}_s(\mathbf{x}_G,\mathbf{v}_G,t) + \frac{e_s}{B}\left\{\phi(\mathbf{x},t) - \bar{\phi}(\mathbf{x}_G,\mathbf{v}_G,t)\right\}\,\partial_\mu\bar{f}_{eq,s}(\mathbf{x}_G,\mathbf{v}_G)$$

Since the computation is performed at order one in ϵ, the equilibrium distribution function only is retained in the last term.

B.2. Calculus of $n_{pol,s}$. — Notice that the Larmor radius itself depends on spatial location via its B dependence. More precisely, when moving from one position to another $\mathbf{x} \leftrightarrow \mathbf{x}_G$, the following relations hold:

$$\mathbf{x} \to \mathbf{x}_G: \quad \mathbf{x} = \mathbf{x}_G + \boldsymbol{\rho}_s(\mathbf{x}) \tag{104}$$

$$\mathbf{x}_G \to \mathbf{x}: \quad \mathbf{x}_G = \mathbf{x} - \boldsymbol{\rho}_s(\mathbf{x}_G) \simeq \mathbf{x} - \boldsymbol{\rho}_s(\mathbf{x}) + (\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s \tag{105}$$

Then it comes, with the convention that $\boldsymbol{\rho}_s$ stands for $\boldsymbol{\rho}_s(\mathbf{x})$ hereafter:

$$\mathrm{e}^{\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}} \simeq 1 + \boldsymbol{\rho}_s\cdot\boldsymbol{\nabla} + \frac{1}{2}\,(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})^2 \tag{106}$$

$$\mathrm{e}^{-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}} \simeq 1 - \boldsymbol{\rho}_s\cdot\boldsymbol{\nabla} + [(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s]\cdot\boldsymbol{\nabla} + \frac{1}{2}\,(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})^2 \tag{107}$$

With these expressions, and at second order in $k_\perp\rho_s$, eq. (45) reads as follows:

$$\begin{aligned} n_{pol,s}(\mathbf{x},t) \simeq & \int \mathcal{J}_v\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\frac{e_s}{B}\int_0^{2\pi}\frac{\mathrm{d}\varphi_c}{2\pi}\,(1-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\,\partial_\mu f_{eq,s}(\mathbf{x},\mathbf{v}) \\ & \left\{1 - \left[1 - \boldsymbol{\rho}_s\cdot\boldsymbol{\nabla} + ((\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s)\cdot\boldsymbol{\nabla} + \frac{1}{2}(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})^2\right]\right. \\ & \left.\left\langle 1 + \boldsymbol{\rho}_s\cdot\boldsymbol{\nabla} + \frac{1}{2}(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})^2\right\rangle\right\}\phi(\mathbf{x},t) \end{aligned}$$

The gyro-radius vector $\boldsymbol{\rho}_s$ can be written $\boldsymbol{\rho}_s = \rho_s(\cos\varphi_c\mathbf{e}_{\perp 1} + \sin\varphi_c\mathbf{e}_{\perp 2})$ where $(\mathbf{e}_{\perp 1},\mathbf{e}_{\perp 2})$ are the two vectors of the cartesian basis. It follows:

$$\begin{aligned} &\langle\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}\rangle = 0 \\ &\langle(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})^2\rangle = \frac{1}{2}\rho_s\boldsymbol{\nabla}_\perp\cdot(\rho_s\boldsymbol{\nabla}_\perp) \end{aligned} \tag{108}$$

with $\rho_s = |\boldsymbol{\rho}_s|$ the norm of the gyro-radius. Keeping terms up to the second order in $k_\perp\rho_s$ only, one gets:

$$\begin{aligned} n_{pol,s}(\mathbf{x},t) \simeq & \int \mathcal{J}_v\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\frac{e_s}{B}\int_0^{2\pi}\frac{\mathrm{d}\varphi_c}{2\pi}\,(1-\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\,\partial_\mu f_{eq,s}(\mathbf{x},\mathbf{v}) \\ & \left\{1 - \left[1 - \boldsymbol{\rho}_s\cdot\boldsymbol{\nabla} + ((\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s)\cdot\boldsymbol{\nabla} + \frac{1}{2}\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\right.\right. \\ & \left.\left. + \frac{1}{4}\rho_s\boldsymbol{\nabla}_\perp.(\rho_s\boldsymbol{\nabla}_\perp)\right]\right\}\phi(\mathbf{x},t) \\ = & -\int \mathcal{J}_v\,\mathrm{d}\mu\,\mathrm{d}v_{G\|}\frac{e_s}{B}\left\langle\left[\frac{1}{2}\,(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}))\right.\right. \\ & \left. + ((\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s)\cdot\boldsymbol{\nabla} + \frac{1}{4}\rho_s\boldsymbol{\nabla}_\perp.(\rho_s\boldsymbol{\nabla}_\perp)\right]\phi\,\frac{\partial f_{eq,s}}{\partial\mu} \\ & \left. + (\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}\phi)\left(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}\frac{\partial f_{eq,s}}{\partial\mu}\right)\right\rangle \end{aligned}$$

(109)

Three brackets remain being calculated. The first one has already been computed (cf. eq. (108)). The two others read as follows:

$$\langle((\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla})\boldsymbol{\rho}_s)\cdot\boldsymbol{\nabla}\rangle = \frac{1}{4}\,(\boldsymbol{\nabla}_\perp\rho_s^2)\cdot\boldsymbol{\nabla}_\perp$$

$$\langle(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}\phi)\,(\boldsymbol{\rho}_s\cdot\boldsymbol{\nabla}\partial_\mu f_{eq,s})\rangle = \frac{1}{2}\,\rho_s^2\,\boldsymbol{\nabla}_\perp\phi\cdot\boldsymbol{\nabla}_\perp\partial_\mu f_{eq,s}$$

It then comes, further noticing that $\rho_s \boldsymbol{\nabla}_\perp \cdot (\rho_s \boldsymbol{\nabla}_\perp \phi) = \frac{1}{2}\left(\boldsymbol{\nabla}_\perp \rho_s^2\right) \cdot \boldsymbol{\nabla}_\perp \phi + \rho_s^2 \boldsymbol{\nabla}_\perp^2 \phi$:

$$\begin{aligned} n_{pol,s}(\mathbf{x},t) &\simeq -\int \mathcal{J}_v \, \mathrm{d}\mu \, \mathrm{d}v_{G\|} \frac{e_s}{B} \left[\frac{1}{2}\left(\boldsymbol{\nabla}_\perp \rho_s^2\right) \cdot \boldsymbol{\nabla}_\perp \phi + \frac{1}{2} \rho_s^2 \boldsymbol{\nabla}_\perp^2 \phi \right. \\ &\quad \left. + \frac{1}{2} \rho_s^2 \boldsymbol{\nabla}_\perp \phi \cdot \boldsymbol{\nabla}_\perp \right] \partial_\mu f_{eq,s} \\ &= \int \mathcal{J}_v \, \mathrm{d}\mu \, \mathrm{d}v_{G\|} \frac{e_s}{2B} \partial_\mu \left[\left(\boldsymbol{\nabla}_\perp \rho_s^2\right) \cdot \boldsymbol{\nabla}_\perp \phi + \rho_s^2 \boldsymbol{\nabla}_\perp^2 \phi \right. \\ &\quad \left. + \rho_s^2 \boldsymbol{\nabla}_\perp \phi \cdot \boldsymbol{\nabla}_\perp \right] f_{eq,s} \end{aligned}$$

At this point, notice that $B^{-1}\partial_\mu = (m_s v_\perp)^{-1} \partial_{v_\perp}$ commutes with the operator in configuration space $\boldsymbol{\nabla}_\perp$. For the same reason, $\int \mathcal{J}_v \, \mathrm{d}\mu$ also commutes with the operator $\boldsymbol{\nabla}_\perp$. Since $B^{-1}\partial_\mu \rho_s^2 = 2m_s/(e_s B)^2$, one finally obtains the following expression for the polarization density:

$$\begin{aligned} n_{pol,s}(\mathbf{x},t) &\simeq \int \mathcal{J}_v \, \mathrm{d}\mu \, \mathrm{d}v_{G\|} \, \boldsymbol{\nabla}_\perp \cdot \left(\frac{m_s}{e_s B^2} \, f_{eq,s} \boldsymbol{\nabla}_\perp \phi \right) \\ &= \boldsymbol{\nabla}_\perp \cdot \left(\int \mathcal{J}_v \, \mathrm{d}\mu \, \mathrm{d}v_{G\|} \frac{m_s}{e_s B^2} \, f_{eq,s} \boldsymbol{\nabla}_\perp \phi \right) \\ &= \boldsymbol{\nabla}_\perp \cdot \left(\frac{m_s n_{eq,s}}{e_s B^2} \boldsymbol{\nabla}_\perp \phi(\mathbf{x},t) \right) \end{aligned} \tag{110}$$

Appendix C
Proof of characteristic properties

a) The points $\mathbf{Z}(t_1;\mathbf{z},t_1)$, $\mathbf{Z}(t_2;\mathbf{z},t_1)$ and $\mathbf{Z}(t_3;\mathbf{z},t_1)$ are on the same characteristics which is located at $\mathbf{z}$ at time t_1: $\mathbf{Z}(t_1) = \mathbf{z}$. Solving the differential system (20)- (21) with each one of these points as initial condition yields the same solution. In particular, it follows that:

$$\mathbf{Z}(t_3;\mathbf{Z}(t_2;\mathbf{z},t_1),t_2) = \mathbf{Z}(t_3;\mathbf{z},t_1) \tag{111}$$

b) By taking $t_1 = t_3$ in Eq. (111), it comes:

$$\mathbf{Z}(t_3;\mathbf{Z}(t_2;\mathbf{z},t_3),t_2) = \mathbf{Z}(t_3;\mathbf{z},t_3) = \mathbf{z}$$

It appears that $\mathbf{Z}(t_3;\cdot,t_2)$ is the inverse of $\mathbf{Z}(t_2;\cdot,t_3)$ in the sense that these applications allow one to explore a given characteristics forward and backward in time. Both applications are C^1 in virtue of theorem 1.

c) Let the jacobian J be defined as

$$J(t;1,s) = \det(\nabla \mathbf{Z}(t;\mathbf{z},s)) = \det\left(\left(\frac{\partial \mathbf{Z}_i(t;\mathbf{z},s)}{\partial z_j} \right) \right)_{1 \le i,j \le d}$$

Let $\mathbf{Z}$ satisfy $\frac{\mathrm{d}\mathbf{Z}}{\mathrm{d}t} = \mathbf{U}(\mathbf{Z}(t),t)$ then $\frac{\mathrm{d}\mathbf{Z}_i}{\mathrm{d}t} = \mathbf{U}_i(\mathbf{Z}(t),t)$ for all $1 \le i \le d$. Consequently, $\nabla \mathbf{Z}$ fulfills the following equation:

$$\frac{\mathrm{d}}{\mathrm{d}t} \nabla \mathbf{Z}_i = \sum_{k=1}^{d} \nabla \mathbf{Z}_k \cdot \frac{\partial U_i}{\partial z_k}$$

Definition 2. — *Let f be an application $f : E^d \mapsto \mathbb{R}$. f is a d- linear alternating map if:*

- *f is linear with respect to each variable, i.e for $x_1, \cdots, x_d, x_i'$ vectors and two scalars a and b*

$$f(x_1, \cdots, x_{i-i}, ax_i + bx_i', x_{i+1}, \cdots, x_n) = af(x_1, \cdots, x_n) + bf(x_1, \cdots, x_i', \cdots, x_n)$$

- *f is an alternating map, i.e:*

$$[\exists i \neq j, x_i = x_j] \Rightarrow f(x_1, \cdots, x_n) = 0$$

Remark 2. — *Let M be a $(d \times d)$ matrix. Then the determinant of M is a d-linear alternating map with respect to the columns of M. Let us denote $(\cdot, \cdots, \cdot)$ this d-linear alternating map. The determinant of M can be written as $\det M = (M_1, \cdots, M_d)$ where M_j represents the j-th column of M.*

Using the notation of the previous definition and remark, then

$$\begin{aligned} \frac{\partial J}{\partial t}(t;1,s) &= \frac{\partial}{\partial t} \det(\nabla \mathbf{Z}(t;\mathbf{z},s)) \\ &= \left(\frac{\partial \nabla Z_1}{\partial t}, \nabla Z_2, \cdots, \nabla Z_d\right) + \cdots + \left(\nabla Z_1, \nabla Z_2, \cdots, \frac{\partial \nabla Z_d}{\partial t}\right) \\ &= \left(\sum_{k=1}^{d} \nabla Z_k \cdot \frac{\partial U_1}{\partial z_k}, \nabla Z_2, \cdots, \nabla Z_d\right) \\ &\quad + \cdots + \left(\nabla Z_1, \nabla Z_2, \cdots, \sum_{k=1}^{d} \nabla Z_k \cdot \frac{\partial U_d}{\partial z_k}\right) \\ &= \frac{\partial U_1}{\partial z_1} J + \cdots + \frac{\partial U_d}{\partial z_d} J \end{aligned}$$

In compact form, it reads $\frac{\partial J}{\partial t}(t;1,s) = (\nabla \cdot \mathbf{U}) J$. Besides, $\nabla \mathbf{Z}(s;\mathbf{z},s) = \nabla \mathbf{z} = \mathbb{I}_d$ such that $J(s;\mathbf{z},s) = \det \mathbb{I}_d = 1$. The jacobian J turns out to satisfy the differential equation

$$\frac{\mathrm{d}J}{\mathrm{d}t} = (\nabla \cdot \mathbf{U}) J \quad \text{and} \quad J(s) = 1$$

which has for unique solution $J(t) = \exp(\int_s^t \nabla \cdot \mathbf{U} dt) > 0$. In particular, if $\nabla \cdot \mathbf{U} = 0$, then $J(t; 1, s) = 1$ for all t.

Appendix D
Proof of the conservation properties of Vlasov-Maxwell equations

The system defining the characteristics associated to the Vlasov-Maxwell equations reads:

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{v}(t), \tag{112}$$

$$\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = \frac{q}{m}\left(\mathbf{E}(\mathbf{x}(t), t) + \mathbf{v}(t) \times \mathbf{B}(\mathbf{x}(t), t)\right) \tag{113}$$

Let us note $\mathbf{Z}(t; \mathbf{z}_0, s) = \{\mathbf{x}(t; \mathbf{x}_0, \mathbf{v}_0, s), \mathbf{v}(t; \mathbf{x}_0, \mathbf{v}_0, s)\}$ the unique solution at time t of the system (112)-(113) which takes the value $\{\mathbf{x}_0, \mathbf{v}_0\}$ at time s, or simply $\mathbf{z}(t) = \{\mathbf{x}(t), \mathbf{v}(t)\}$.

D.1. Proof of the maximum principle. — Using (112)-(113), the Vlasov equation can be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(f(\mathbf{x}(t), \mathbf{v}(t)\right) = 0$$

then

$$f(\mathbf{x}, \mathbf{v}, t) = f_0(\mathbf{x}(0; \mathbf{x}, \mathbf{v}, t), \mathbf{v}(0; \mathbf{x}, \mathbf{v}, t))$$

Since f_0 is positive, f satisfies the maximum principle

$$0 \leq f(\mathbf{x}, \mathbf{v}, t) \leq max_{(x,v)}(f_0(\mathbf{x}, \mathbf{v}))$$

D.2. Proof of the volume conservation. — By integrating on an arbitrary volume V of the phase space and by using the fact that f is conserved along its characteristics, then

$$\begin{aligned}\int_V f(\mathbf{x}, \mathbf{v}, t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} &= \int_V f(\mathbf{x}(t; \mathbf{x}, \mathbf{v}, t), \mathbf{v}(t; \mathbf{x}, \mathbf{v}, t))\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} \\ &= \int_V f(\mathbf{x}(0; \mathbf{x}, \mathbf{v}, t), \mathbf{v}(0; \mathbf{x}, \mathbf{v}, t))\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} \\ &= \int_V f_0(\mathbf{x}(0; \mathbf{x}, \mathbf{v}, t), \mathbf{v}(0; \mathbf{x}, \mathbf{v}, t))\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}\end{aligned}$$

Then by performing the change of variable

$$\mathbf{y} = \mathbf{x}(0; \mathbf{x}, \mathbf{v}, t) \quad , \quad \mathbf{u} = \mathbf{v}(0; \mathbf{x}, \mathbf{v}, t)$$

and using the fact that the jacobian of this transformation is equal to 1 (due to proposition 1 and because $\nabla_{(x,v)} \cdot \mathbf{U} = 0$) then

$$\int_V f(\mathbf{x}, \mathbf{v}, t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = \int_{\mathbf{U}^{-1}(V)} f_0(\mathbf{y}, \mathbf{u})\,\mathrm{d}\mathbf{y}\,\mathrm{d}\mathbf{u}$$

D.3. Proof of the L^p norm conservation. — Let the function Υ be defined as $\mathcal{C}^1$ such that $\Upsilon \in \mathcal{C}^1(\mathbb{R}^+, \mathbb{R}^+)$. We are going to prove that:

$$I = \frac{\mathrm{d}}{\mathrm{d}t}\left(\int_{\mathbb{R}^d \times \mathbb{R}^d} \Upsilon(f(\mathbf{x}, \mathbf{v}, t))\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}\right) = 0$$

In particular, by taking $\Upsilon(y) = y^p$, this is equivalent to saying that all L^p norms, for $1 \leq p \leq +\infty$ are preserved (proof of Eq. (31)). The infinite norm L^∞ is also conserved due to the maximum principle. Moreover, taking $\Upsilon(y) = -y\ln(y)$, we obtain the conservation of the kinetic entropy (proof of Eq. (32)).
Using the definition of the total time derivative ($\frac{\mathrm{d}\cdot}{\mathrm{d}t} = \partial_t + \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}\nabla_{\mathbf{x}} \cdot + \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t}\nabla_{\mathbf{v}} \cdot$), I is equivalent to

$$I = \int \frac{\mathrm{d}}{\mathrm{d}t}[\Upsilon(f(\mathbf{x}, \mathbf{v}, t))]\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = \int \frac{\mathrm{d}\Upsilon}{\mathrm{d}f}\frac{\mathrm{d}f}{\mathrm{d}t}\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = 0$$

D.4. Proof of the energy conservation. — To obtain the energy conservation property of the Vlasov-Maxwell system, the first step is to multiply the Vlasov equation (12) by $\frac{1}{2}m\mathbf{v} \cdot \mathbf{v} = \frac{1}{2}mv^2$ and to integrate on the phase space,

$$\begin{aligned}\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d \times \mathbb{R}^d} mv^2 f\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} + \frac{1}{2}\int_{\mathbb{R}^d \times \mathbb{R}^d} m\nabla_x \cdot (v^2\mathbf{v}f)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} \\ + \frac{1}{2}\int_{\mathbb{R}^d \times \mathbb{R}^d} mv^2\nabla_v \cdot ((\mathbf{E} + \mathbf{v} \times \mathbf{B})f)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = 0\end{aligned}$$

Since f vanishes at infinity faster than any power of v:

$$\int_{\mathbb{R}^d \times \mathbb{R}^d} \nabla_x \cdot (v^2\mathbf{v}f)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} = 0 \tag{114}$$

Integrating by parts, the third term leads to:

$$\begin{aligned}&\frac{m}{2}\int_{\mathbb{R}^d \times \mathbb{R}^d} v^2\nabla_v \cdot ((\mathbf{E} + \mathbf{v} \times \mathbf{B})f)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} \\ &= -\int_{\mathbb{R}^d \times \mathbb{R}^d} \mathbf{v} \cdot ((\mathbf{E} + \mathbf{v} \times \mathbf{B})f)\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} + \text{surface term} \\ &= -\int_{\mathbb{R}^d} \mathbf{E} \cdot \mathbf{j}\,\mathrm{d}\mathbf{x}\end{aligned}$$

where the surface term vanishes for the same reason than before (i.e f vanishes at infinity faster than any power of v) and where j stands for the current

$\mathbf{j} = \int \mathbf{v} f \, \mathrm{d}\mathbf{v}$. Therefore, the time derivative of the kinetic energy reads as follows:

$$\frac{m}{2}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathbb{R}^d\times\mathbb{R}^d} v^2 f \, \mathrm{d}\mathbf{x}\, \mathrm{d}\mathbf{v} = \int_{\mathbb{R}^d} \mathbf{E}\cdot\mathbf{j} \, \mathrm{d}\mathbf{x} \tag{115}$$

The second step consists in evaluating the time derivative of the electromagnetic energy $\mathcal{E}_{\mathrm{EM}} = \frac{1}{2}\int \left(\epsilon_0 E^2 + \frac{B^2}{\mu_0}\right) \mathrm{d}\mathbf{x}$. It reads

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{E}_{\mathrm{EM}} = \int \left(\epsilon_0 \mathbf{E}\cdot\partial_t\mathbf{E} + \frac{\mathbf{B}}{\mu_0}\cdot\partial_t\mathbf{B}\right) \mathrm{d}\mathbf{x}$$

Using Maxwell-Faraday (16) and Maxwell-Ampère equation (14) to compute $\partial_t\mathbf{E}$ and $\partial_t\mathbf{B}$, respectively, one obtains:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{E}_{\mathrm{EM}} &= \int \left(\frac{\mathbf{E}\cdot\nabla\times\mathbf{B}}{\mu_0} - \mathbf{E}\cdot\mathbf{j} - \frac{\mathbf{B}\cdot\nabla\times\mathbf{E}}{\mu_0}\right) \mathrm{d}\mathbf{x} \\ &= -\int \mathbf{E}\cdot\mathbf{j}\,\mathrm{d}\mathbf{x} - \int \nabla\cdot\left(\frac{\mathbf{E}\times\mathbf{B}}{\mu_0}\right) \mathrm{d}\mathbf{x}\end{aligned}$$

The latter term is a surface term: it accounts for the exchange of the electromagnetic energy with the outside world via the Poyinting vector $\frac{\mathbf{E}\times\mathbf{B}}{\mu_0}$. It vanishes for an isolate system. In this case, it readily appears that the total energy, namely the kinetic plus the electromagnetic energy, is conserved

$$\frac{\mathrm{d}}{\mathrm{d}t}\left\{\int \frac{1}{2}mv^2 f \, \mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v} + \int \frac{1}{2}\left(\epsilon_0 E^2 + \frac{B^2}{\mu_0}\right)\right\} = 0$$

Remark 3. — *In a finite domain, as used in a numerical solving, the condition* (114) *could not be exactly* 0 *and would depend on the choice of the boundary conditions.*

Appendix E
Monte-Carlo interpretation for PIC simulations

E.1. Monte-Carlo evaluation of integrals. — In the following, the term "Monte-Carlo" refers to the estimation of multidimensional integrals using statistical sampling techniques. In general, particular form of the integrand and how it is sampled in the volume of interest determine the accuracy of the estimates. Since the 1950s, the Monte Carlo community has developed a number of techniques aiming at minimizing the error in the estimates and which increase the efficiency of the calculations. Readers unfamiliar with the techniques and the terminology of the Monte-Carlo method can refer to *e.g.*

Hammersley and Handscomb [**67**], Fishman [**56**] or more recently [**85**]. For the discussion let us consider the general integral of the form:

$$I(\Upsilon) = \int_V \Upsilon(\mathbf{Z}) f(\mathbf{Z}) \, \mathrm{d}\tau \tag{116}$$

where $\mathrm{d}\tau = \mathcal{J} d\mathbf{x}\, d\mathbf{v}$ is the volume element, $\Upsilon(\mathbf{Z})$ is any function of the phase-space coordinates $\mathbf{Z} = (\mathbf{x}, \mathbf{v})$ and $f(\mathbf{Z}, t)$ is the distribution function of some population of N_s particles, i.e

$$\int_V f(\mathbf{Z}) \, \mathrm{d}\tau = N_s$$

For instance, $I(\Upsilon)$ would be the number density in configuration space if $\Upsilon = 1$, and the integral is over the velocity space. Let us treat $\mathbf{Z}$ as a continuous random variable with probability density function (PDF) $p(\mathbf{Z})$ in the phase-space volume V, the sampling distribution satisfying,

$$\int_V p(\mathbf{Z}) \, \mathrm{d}\tau = 1 \tag{117}$$

The basic principle of Monte-Carlo methods consists in reformulating equation (116) as follows:

$$I(\Upsilon) = \mathbb{E}_p(g(\mathbf{Z})) = \int_V g(\mathbf{Z}) p(\mathbf{Z}) \, \mathrm{d}\tau \tag{118}$$

where $\mathbb{E}_p(g)$ is the *expected value* of the random variable

$$g \equiv (\Upsilon(\mathbf{Z}) f(\mathbf{Z}))/p(\mathbf{Z}) \tag{119}$$

under the probability density $p(\mathbf{Z})$. Let us also define the *variance* of g by

$$\mathcal{V}(g) = \sigma_g^2 = \int_V (g - \mathbb{E}_p(g))^2 p(\mathbf{Z}) \, \mathrm{d}\tau \tag{120}$$

The idea is to generate an independent random sample $(\mathbf{Z}_1, \mathbf{Z}_2, \cdots, \mathbf{Z}_N)$ for the random variable $\mathbf{Z}$ of probability $p(\mathbf{Z})$ and to calculate a new estimate (called *Monte-Carlo estimate*) as a function of this sampling. The law of large numbers (first fundamental theorem of probability) suggests to generate this estimate with the empiric mean

$$\tilde{g}_N = \frac{1}{N}\sum_{j=1}^{N} g(\mathbf{Z}_j) \tag{121}$$

Notice that the N number of markers upper bounded due to the computational limits, such that $N \ll N_s$, where N_s is the number of physical particles of the

s specie. Then, let S_N be the random variable characterized by $\mathbb{E}_p(S_N) = 0$ and $\sigma(S_N) = 1$:

$$S_N \equiv \frac{\tilde{g}_N - \mathbb{E}_p(\tilde{g}_N)}{\sigma_g/\sqrt{N}}$$

where the unbiased estimate $\tilde{g}_N$ is defined by Eq. (121), the expected value $\mathbb{E}_p$ by Eq. (118) and the square root of the variance σ_g by Eq. (120)

Theorem 2. — **The central limit theorem** *(second fundamental theorem of probability) states that S_N converges in distribution to the standard normal distribution $\mathbb{N}(0;1)$ as N approaches infinity. Convergence in distribution means that if $\Phi(z)$ is the cumulative distribution function of $\mathbb{N}(0;1)$, i.e*

$$\Phi(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right) \mathrm{d}t = \textit{error function}$$

then for every real number z, we have

$$\lim_{n\to\infty} P(S_N \leq z) = \Phi(z)$$

Therefore it is possible to define a *confidence interval* that quantifies the degree of agreement of the estimate $\tilde{g}_N$ with respect to the moment integral $I(\Upsilon)$. Let e_n be this error, then for a *confidence level* $(1-\alpha)$ $(\alpha \in \mathbb{R})$,

$$|e_n| \leq z_{1-\alpha/2} \frac{\sigma_g}{\sqrt{N}} \tag{122}$$

where the real $z_{1-\alpha/2}$ is the $(1-\alpha)$-th percentile of the distribution. For instance let us take $1-\alpha = 0.95$, then

$$P(-z_{1-\alpha/2} \leq S_N \leq z_{1-\alpha/2}) = 1 - \alpha = 0.95$$

where $z_{1-\alpha/2}$ derives from the cumulative distribution function:

$$\Phi(z_{1-\alpha/2}) = P(S_N \leq z_{1-\alpha/2}) = 1 - \frac{\alpha}{2} = 0.975$$
$$z_{1-\alpha/2} = \Phi^{-1}(\Phi(z_{1-\alpha/2})) = \Phi^{-1}(0.975) = 1.96$$

This can be interpreted as follows: given the number N of discrete markers to sample the continuous distribution function in the phase space, the integral (116) can be approached down to an error of $|e_n| \leq \Phi^{-1}(1-\alpha/2)\frac{\sigma_g}{\sqrt{N}}$ (i.e $|e_n| \leq 1.96\frac{\sigma_g}{\sqrt{N}}$ in the example) within the confidence level of $100(1-\alpha)$ percent, where α is any real comprised between 0 and 1 (i.e 95% here).

To summarize, $\tilde{g}_N$ is the *unbiased* (i.e $\mathbb{E}_p(\tilde{g}_N) = \mathbb{E}_p(g(\mathbf{Z}))$) and consistent (i.e $\mathcal{V}(\tilde{g}_N) = \sigma_g^2/N \to 0$ as $N \to \infty$) estimate of the expected value of g, with the standard error $\epsilon \simeq \sigma_g/\sqrt{N}$, where σ_g is the standard deviation defined by

Eq. (120). Finally, a valid Monte-Carlo estimate of the moment integral $I(\Upsilon)$ (Eq. 116) is given by

$$I(\Upsilon) = \frac{1}{N}\sum_{j=1}^{N} \frac{\Upsilon(\mathbf{Z}_j) f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)} + \epsilon \quad \text{with} \quad \epsilon \simeq \frac{\sigma_g}{\sqrt{N}} \tag{123}$$

Besides, let $f_K(\mathbf{Z})$ be the *Klimontovitch density*

$$f_K(\mathbf{Z}) = \frac{1}{\mathcal{J}}\sum_{j=1}^{N} w_j \delta(\mathbf{Z} - \mathbf{Z}_j) \quad \text{with the weight} \quad w_j = \frac{1}{N}\frac{f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)} \tag{124}$$

It readily appears that, for any volume element Ω in V the moments $I(\Upsilon)$ (Eq. (123)) of f can be expressed as

$$\int_\Omega \Upsilon(\mathbf{Z}) f(\mathbf{Z})\, \mathrm{d}\tau = \int_\Omega \Upsilon(\mathbf{Z}) f_K(\mathbf{Z})\, \mathrm{d}\tau + \epsilon, \quad \epsilon \simeq \frac{\sigma_g}{\sqrt{N}} \tag{125}$$

Remark that, in practice, σ_g is unknown and must be estimated. One possibility is to use the discrete variance

$$\sigma_g^2 \simeq \frac{1}{N}\sum_{j=1}^{N} (g(\mathbf{Z}_j) - \tilde{g}_N)^2$$

Several methods, called *variance reduction techniques*, efficiently improve the accuracy–or reduce the computation time– by replacing $g(\mathbf{Z})$ by another random variable. Two of them are particularly widespread in plasma particle simulations. These are the importance sampling and the control variates. The backbone of these methods are detailed in the following sections.

E.2. Importance sampling. — The main idea of importance sampling is to use non-uniform marker probability, proportional to the distribution function, conversely to the simplest Monte Carlo method where it is uniform (with $p(\mathbf{Z}) = \frac{1}{V}$, $w_j = \frac{V}{N} f(\mathbf{Z}_j)$ and $g = \Upsilon(\mathbf{Z}) f(\mathbf{Z}) V$). This leads to a more frequent sampling of the most "important" region of the phase-space.

$$p(\mathbf{Z}) = \frac{1}{N_s} f(\mathbf{Z}), \quad w_j = \frac{N_s}{N} \quad \text{and} \quad g = N_s \Upsilon(\mathbf{Z}) \tag{126}$$

The first advantage is that there is no information storage required for the weights, because they are the same for each marker $w_j = N_s/N$.

Remark 4. — *In this case, Lagrangian markers are called "macro-particles", each representing N_s/N physical particles.*

The second and most important point is that this choice reduces the variation in g because it only comes from the function $\Upsilon(\mathbf{Z})$ since $f/p = \text{const}$. This importance sampling method is adopted in all present PIC simulations.

E.3. Control variates - The δf method. — The control variates method is another intuitively obvious approach that aims at reducing the variance in $I(\Upsilon)$ by replacing, as often as possible, the Monte Carlo estimate by analytic or numerical calculations which are more accurate. Assume that there exists a function f_0, formally called the *control variate*, such that

(i) moments of f_0 can be found easily (preferentially analytically),
(ii) at all times, the physical distribution function $f(\mathbf{Z})$ remains close to $f_0(\mathbf{Z})$ in the sense $\|f - f_0\|/\|f\| \ll 1$ where $\|\cdot\|$ is some arbitrary norm.

Then the error in the estimate $I(\Upsilon)$, can be reduced by rewriting the integral in the form

$$I(\Upsilon) = \int_V \Upsilon(\mathbf{Z}) f_0(\mathbf{Z})\,\mathrm{d}\tau + \int_V \Upsilon(\mathbf{Z})\delta f\,\mathrm{d}\tau \quad \text{where} \quad \delta f = f(\mathbf{Z}) - f_0(\mathbf{Z})$$

and by applying a Monte Carlo technique to the second integral only. Using the same technique as before the Monte Carlo estimate for $I(\Upsilon)$ is then given by

$$I(\Upsilon) = I_0(\Upsilon) + \frac{1}{N}\sum_{j=1}^{N} \frac{\Upsilon(\mathbf{Z}_j) f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)} + \epsilon_{\delta g} \quad \text{with} \quad \epsilon_{\delta g} \simeq \frac{\sigma_{\delta g}}{\sqrt{N}} \tag{127}$$

where

$$I_0(\Upsilon) = \int_V \Upsilon(\mathbf{Z}) f_0(\mathbf{Z})\,\mathrm{d}\tau$$

can be computed analytically. The function δg is defined as $\delta g = \Upsilon(\mathbf{Z})\delta f(\mathbf{Z})/p(\mathbf{Z})$ while $\sigma_{\delta g}$ the deviation of δg is given by

$$\mathcal{V}(\delta g) = \sigma_{\delta g}^2 = \int_V (\delta g - \mathbb{E}_p(\delta g))^2 p(\mathbf{Z})\,\mathrm{d}\tau \tag{128}$$

Comparing the error in the Monte Carlo estimates (123) and (127) (respectively proportional to σ_g given by Eq. (120) and $\sigma_{\delta g}$ given by Eq. (128)), the advantage of this control variate technique becomes evident. Indeed, the noise is reduced by a factor $\delta f/f$ for the same number of sample points. Notice however that, if one chooses the Klimontovitch density for $\delta f_K(\mathbf{Z})$, then

$$\delta f_K(\mathbf{Z}) = \frac{1}{\mathcal{J}}\sum_{j=1}^{N} w_j \delta(\mathbf{Z} - \mathbf{Z}_j) \quad \text{with the weight} \quad w_j = \frac{1}{N}\frac{\delta f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)} \tag{129}$$

It appears that the weights w_j are no longer constant in time, in this case:

$$\frac{\mathrm{d}w_j}{\mathrm{d}t}(t) = \frac{1}{N}\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\delta f(\mathbf{Z}_j)}{p(\mathbf{Z}_j)}\right) = -\frac{1}{N}\left[\frac{1}{p(\mathbf{Z})}\frac{\mathrm{d}f_0}{\mathrm{d}t}\right]_{\mathbf{Z}=\mathbf{Z}_j}$$

For the same reasons as discussed in the previous section, the choice $p(\mathbf{Z}) = f(\mathbf{Z})/N_s$ is better than the choice of an uniform density.

$$\frac{\mathrm{d}w_j}{\mathrm{d}t}(t) = -\frac{N_s}{N}\left(1 - \frac{N}{N_s}w_j\right)\left[\frac{1}{f_0}\frac{\mathrm{d}f_0}{\mathrm{d}t}\right]_{\mathbf{Z}=\mathbf{Z}_j} \tag{130}$$

Note however, that Eq. (130) does not need being solved for the weights w_j to be calculated, since $\delta f(\mathbf{Z}_j)/f(\mathbf{Z}_j)$ can be calculated directly.

To conclude, the control variate-δf method reduces noise by reducing the size of the Monte Carlo contribution to $I(\Upsilon)$. It is important to point out that this method also concentrates all the relevant physics on the small integral of δf and its time evolution; thus, the accuracy of the method crucially depends on accurate evaluations of the moments of δf. For this reason, there are two complementary requirements:

(i) Low noise is only accomplished by ensuring that $\|f - f_0\|/\|f\| \ll 1$,
(ii) Accuracy is only possible if the relative error in $\delta I(\Upsilon)$ is small, i.e $\|\epsilon_{\delta g}/\delta I(\Upsilon)\| \ll 1$, where $\delta I(\Upsilon) = I(\Upsilon) - I_0(\Upsilon)$ and $\epsilon_{\delta g}$ is defined in Eq. (127) and (128).

The first requirement can be achieved with a well-chosen control variate f_0 and already for a modest number N of macro-particles. Conversely, the second requires a large number of markers, since $\epsilon_{\delta g}/\delta I(\Upsilon) \sim 1/\sqrt{N}$.

Appendix F

Particle In Cell global scheme

As discussed in appendix E, PIC methods substitute to the distribution function f the sum of Dirac distributions localized at the phase space positions $\mathbf{Z}_j(t) = (\mathbf{x}_j(t), \mathbf{v}_j(t))_{1\le j\le N}$, where N stands for the number of macro-particles of weight w_j. The approximated distribution function, denoted f_N, exhibits some analogy with the one proposed by Klimontovitch, except that weights w_j are introduced:

$$f_N(\mathbf{x}, \mathbf{v}, t) = \sum_{j=1}^{N} w_j \delta(\mathbf{x} - \mathbf{x}_j(t))\delta(\mathbf{v} - \mathbf{v}_j(t))$$

Positions $\mathbf{x}_j^0$, velocities $\mathbf{v}_j^0$ and weights w_j are initialized such that $f_{N0} \equiv f_N(\mathbf{x}, \mathbf{v}, 0)$ approaches the initial distribution function $f_0(\mathbf{x}, \mathbf{v})$. f_N is evolved in time by following the macro-particle trajectories. The following system of

differential equations is solved:

$$\begin{aligned}\frac{\mathrm{d}\mathbf{x}_j}{\mathrm{d}t} &= \mathbf{v}_j\\ \frac{\mathrm{d}\mathbf{v}_j}{\mathrm{d}t} &= q_s\left(\mathbf{E}(\mathbf{x}_j,t)+\mathbf{v}_j\times\mathbf{B}(\mathbf{x}_j,t)\right)\\ \mathbf{x}_j(0) &= \mathbf{x}_j^0, \quad \mathbf{v}_j(0)=\mathbf{v}_j^0\end{aligned}$$

The electromagnetic field is then self-consistently computed from Maxwell's equations by using classical numerical schemes such as finite elements, finite differences or spectral methods. PIC algorithms still exhibit two additional degrees of freedom, depending (i) on the kind of strategy which is retained to construct the initial distribution function f_{N0}, and (ii) on the numerical method which is used to solve the differential equation system governing the characteristics.

F.1. Choice of initial condition. — Two strategies can be followed to construct the initial condition f_{N0}. The first one is a deterministic. First construct the phase space mesh, which can be uniform or non-uniform. Then consider that the barycenters of the cells correspond to the initial phase space positions of the macro-particles $(\mathbf{x}_j^0,\mathbf{v}_j^0)$. In this case, the weight w_j is simply the integral of f_0 on the corresponding elementary volume element $w_j=\int_{V_j} f_0(\mathbf{x},\mathbf{v})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}$. It readily appears that $\sum_{j=1}^N w_j=\int f_0(\mathbf{x},\mathbf{v})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{v}$. The second strategy consists in choosing the initial positions of the macro-particle at random (or pseudo-randomly, cf. appendix E). The associated weights are then the PDF of the f_0 distribution.

F.2. Coupling between particles and mesh. — From the discrete representation f_N of the continuous distribution function f, the value of the distribution function cannot be *easily* extrapolated to any position that lies outside the finite set of macro-particle positions $(\mathbf{x}_i,\mathbf{v}_i)$. Conversely, the field solver acts on some prescribed mesh. Therefore, regularizing f_N represents the mandatory preliminary step before the fields can be calculated. This regularization is performed by using convolution kernels. Numerically, B-splines of fixed order reveal advantageous as convolution kernels (for more details on B-splines, please refer to [**43**]).

The charge and current densities ρ and $\mathbf{j}$ entering Maxwell's equations are computed from the discrete distribution function f_N:

$$\rho_N = q_s\sum_j w_j\delta(\mathbf{x}-\mathbf{x}_j), \quad \mathbf{j}_N = q_s\sum_j w_j\delta(\mathbf{x}-\mathbf{x}_j)\mathbf{v}_j$$

with q_s the particle charge of the s species. The convolution kernel S is then employed to calculate ρ and $\mathbf{j}$ at all mesh points:

$$\rho_h(\mathbf{x},t) = \int S(\mathbf{x}-\mathbf{x}')\rho_N(\mathbf{x}')\mathrm{d}\mathbf{x}' = q_s\sum_j w_j S(\mathbf{x}-\mathbf{x}_j) \tag{131}$$

$$\mathbf{j}_h(\mathbf{x},t) = \int S(\mathbf{x}-\mathbf{x}')\mathbf{j}_N(\mathbf{x}')\mathrm{d}\mathbf{x}' = q_s\sum_j w_j S(\mathbf{x}-\mathbf{x}_j)\mathbf{v}_j \tag{132}$$

The same kernel S is used to regularize the fields and to extrapolate the value of the fields at the particle positions. Then,

$$\mathbf{E}_h(\mathbf{x},t) = \sum_j \mathbf{E}_j(t)S(\mathbf{x}-\mathbf{x}_j), \quad \mathbf{B}_h(\mathbf{x},t) = \sum_j \mathbf{B}_j(t)S(\mathbf{x}-\mathbf{x}_j) \tag{133}$$

The simplest choice for the kernel – and probably the one used predominantly in the literature – is to take $S=S^1$, which corresponds to linear interpolation. Cubic splines S^3 are also frequently used. In this case, the regularized field is no more equal to $\mathbf{E}_j$ at mesh points but to $\frac{1}{6}\mathbf{E}_{j-1}+\frac{2}{3}\mathbf{E}_j+\frac{1}{6}\mathbf{E}_{j+1}$.

F.3. Time scheme for the particles. — For the sake of simplicity, let us focus on the Vlasov-Poisson case in the following, by neglecting the time evolution of the magnetic field. In this case, the trajectories of the macro-particles are governed by:

$$\frac{\mathrm{d}\mathbf{x}_j}{\mathrm{d}t} = \mathbf{v}_j, \quad \frac{\mathrm{d}\mathbf{v}_j}{\mathrm{d}t} = q_s\mathbf{E}(\mathbf{x}_j,t) \tag{134}$$

This system is Hamiltonian. It is usually solved by using a symplectic numerical schemes. The most frequently used is the *Verlet scheme*, which we detail hereafter. Assuming that $\mathbf{x}_j^n$, $\mathbf{v}_j^n$ and $\mathbf{E}_j^n$ are known quantities, then

$$\mathbf{v}_j^{n+\frac{1}{2}} = \mathbf{v}_j^n + q_s\frac{\Delta t}{2}\mathbf{E}_j^n(\mathbf{x}_j^n) \tag{135}$$

$$\mathbf{x}_j^{n+1} = \mathbf{x}_j^n + \Delta t\mathbf{v}_j^{n+\frac{1}{2}} \tag{136}$$

$$\mathbf{v}_j^{n+1} = \mathbf{v}_j^{n+\frac{1}{2}} + q_s\frac{\Delta t}{2}\mathbf{E}_j^{n+1}(\mathbf{x}_j^{n+1}) \tag{137}$$

Step (137) requires the knowledge of the electric field at time t_{n+1}. It can be calculated after step (136) by solving the Poisson equation. Indeed, this latter equation yields $\mathbf{E}_j^{n+1}$ from ρ_h^{n+1}, where ρ_h^{n+1} is entirely governed by the space positions $\mathbf{x}_j^{n+1}$ only, Eq. (131).

F.4. Global PIC-algorithm. — To summarize, here are the four main iterative steps performed in PIC codes in order to advance in time from t_n to t_{n+1}:

a) Compute the charge density ρ_h and the current density $\mathbf{j}_h$ on mesh points, using Eq. (131) and Eq. (132),
b) Compute the electric field by solving Poisson equation (with classical mesh solver as finite difference, spectral methods or finite elements, $\cdots$),
c) Compute the electric field at the particle positions, following Eq. (133),
d) Advance the particles by solving equation (134) with, for instance, the symplectic Verlet scheme (135)-(136).

Appendix G

Strang time-splitting: a scheme of second order in time

Formal proof. — Notice that the Vlasov equation can formally be written as follows:

$$\frac{\partial f}{\partial t} + (A+B+C)f = 0 \tag{138}$$

where A, B and C are differential operators in phase space (for instance, in the case of the 4D drift-kinetic ITG model [**63**], these are $A = v_{ExB}\vec{\nabla}\cdot$, $B = v_\parallel \frac{\partial \cdot}{\partial z}$ and $C = \dot{v_\parallel}\frac{\partial \cdot}{\partial v_\parallel}$). On the one hand, the formal solution of Eq. (138) on one time step Δt is given by:

$$f(t+\Delta t) = f(t)\exp\left[-\int_t^{t+\Delta t}(A+B+C)dt'\right]$$

For the sake of simplicity, let us neglect the time evolution of the operators between t_n and t_{n+1}. In this case, the previous equation simply reads:

$$f(t+\Delta t) = f(t)\exp\left[-(A+B+C)\Delta t\right] \tag{139}$$

On the other hand, the formal solution resulting from the time-splitting scheme is given by:

$$\begin{aligned}\tilde{f}(t+\Delta t) = f(t)\exp\left(-\frac{A\Delta t}{2}\right)\exp\left(-\frac{B\Delta t}{2}\right)\exp\left(-C\Delta t\right)\\ \times\exp\left(-\frac{B\Delta t}{2}\right)\exp\left(-\frac{A\Delta t}{2}\right)\end{aligned} \tag{140}$$

Let us prove that both expressions Eq. (139) and Eq. (140) are equivalent at second order in Δt.

Computation of $I = \exp\left[-(A+B+C)\Delta t\right]$. — The Taylor expansion at second order of $\exp\left[-(A+B+C)\Delta t\right]$ yields:

$$\begin{aligned} I &= \exp\left[(-A+B+C)\Delta t\right]\\ I &= 1-(A+B+C)\Delta t + \frac{\Delta t^2}{2}(A^2+B^2+C^2\\ &\quad +AB+BA+AC+CA+BC+CB) + \mathcal{O}(\Delta t^3)\end{aligned} \tag{141}$$

Computation of $J = \exp(-\frac{A\Delta t}{2})\exp(-\frac{B\Delta t}{2})\exp(-C\Delta t)\exp(-\frac{B\Delta t}{2})\exp(-\frac{A\Delta t}{2})$. — Similarly,

$$\begin{aligned} J_0 &\equiv \exp\left(-\frac{B\Delta t}{2}\right)\exp\left(-\frac{A\Delta t}{2}\right)\\ &= \left(1-\frac{B\Delta t}{2}+\frac{B^2\Delta t^2}{8}+\mathcal{O}(\Delta t^3)\right)\left(1-\frac{A\Delta t}{2}+\frac{A^2\Delta t^2}{8}+\mathcal{O}(\Delta t^3)\right)\\ &= \left[1-\left(\frac{B}{2}+\frac{A}{2}\right)\Delta t + \Delta t^2\left(\frac{B^2}{8}+\frac{A^2}{8}+\frac{BA}{4}\right)\right]+\mathcal{O}(\Delta t^3)\end{aligned}$$

and

$$\begin{aligned} J_1 &\equiv \exp(-C\Delta t)\exp\left(-\frac{B\Delta t}{2}\right)\exp\left(-\frac{A\Delta t}{2}\right) = \exp(-C\Delta t)\times J_0\\ &= \left(1-C\Delta t+\frac{C^2\Delta t^2}{2}\right)\times J_0 + \mathcal{O}(\Delta t^3)\\ &= \left[1-\left(\frac{A}{2}+B+C\right)\Delta t + \right.\\ &\quad \left.\Delta t^2\left(\frac{C^2}{2}+\frac{B^2}{8}+\frac{A^2}{8}+\frac{BA}{4}+\frac{CA}{2}+\frac{CB}{2}\right)\right]+\mathcal{O}(\Delta t^3).\end{aligned}$$

Hence

$$\begin{aligned} J_2 &= \exp\left(-\frac{B\Delta t}{2}\right)\times J_1\\ &= \left(1-\frac{B\Delta t}{2}+\frac{B^2\Delta t^2}{8}\right)\times J_1\\ &= \left[1-\left(\frac{A}{2}+B+C\right)t + \right.\\ &\quad \left.\Delta t^2\left(\frac{C^2}{2}+\frac{B^2}{2}+\frac{A^2}{8}+\frac{BA}{2}+\frac{CA}{2}+\frac{CB}{2}+\frac{BC}{2}\right)\right]+\mathcal{O}(\Delta t^3)\end{aligned}$$

In the end, one obtains

$$J = \exp\left(-\frac{A\Delta t}{2}\right)\times J_2 = \left(1-\frac{A\Delta t}{2}+\frac{A^2\Delta t^2}{8}\right)\times J_2 + \mathcal{O}(\Delta t^3)$$

which leads to the final expression

$$J = 1 - (A+B+C)\Delta t + \Delta t^2 \left(\frac{C^2}{2} + \frac{B^2}{2} + \frac{A^2}{2} + \frac{AB+BA}{2} + \frac{AC+CA}{2} + \frac{BC+CB}{2} \right) + \mathcal{O}(\Delta t^3) \tag{142}$$

It appears that Eq. (142) is equivalent to Eq. (141) up the the second order term in Δt, such that

$$f(t+\Delta t) = \tilde{f}(t+\Delta t) + \mathcal{O}(\Delta t^3)$$

It proves that Eq. (139) and Eq. (140) are equivalent at second order. ■

Appendix H

A new interpolation approach for Semi-Lagrangian scheme

Interpolation represents one of the steps of semi-Lagrangian numerical schemes which require much CPU time. It is most often performed using cubic splines. The drawback of splines is that the knowledge of the function on all the mesh nodes is required to interpolate this function on any single point. This global dependency conflicts with any tentative of competitive parallelization. The original approach presented in this appendix has been explicitly developed to overcome this problem (cf [**40**]). The basic idea of this so-called *local spline* approach is to split the global domain in separated sub-domains, where standard cubic splines are applied ([**1, 43**]). The difficulty then relies on the choice of appropriated boundary conditions for these sub-domains.

H.1. Domain decomposition in 1D. — Let $g(x)$ be any function defined on the global domain $[x_0, x_{N_x}] \subset \mathbb{R}$. Let us split this domain into several sub-domains, each being associated to one processor. The sub-domain associated to processor p is generically called $[x_{p0}, x_{pK}]$ with the notation $x_{pi} = x_{p0} + ih$, where h is the cell size $h = (x_{pK} - x_{p0})/K$ and $K \in \mathbb{N}$ the number of cells in the sub-domain. Let us now restrict the study of g to $g : x \mapsto g(x)$ on the interval $[x_{p0}, x_{pK}]$. Boundary conditions are non-periodic in this sub-domain. The projection s_{loc} of g on the basis of cubic splines reads:

$$g(x) \simeq s_{\text{loc}}(x) = \sum_{\alpha=-1}^{K+1} c_\alpha \Lambda_\alpha(x)$$

with Λ_α the piecewise cubic polynomials defined as follows, for any integer α in the interval $-1 \le \alpha \le K+1$:

$$\Lambda_\alpha(x) = \frac{1}{6h^3} \begin{cases} (x - x_{\alpha-2})^3 & \text{if } x_{\alpha-2} \le x \le x_{\alpha-1} \\ h^3 + 3h^2(x - x_{\alpha-1}) + 3h(x - x_{\alpha-1})^2 & \\ \quad -3(x - x_{\alpha-1})^3 & \text{if } x_{\alpha-1} \le x \le x_\alpha \\ h^3 + 3h^2(x_{\alpha+1} - x) + 3h(x_{\alpha+1} - x)^2 & \\ \quad -3(x_{\alpha+1} - x)^3 & \text{if } x_\alpha \le x \le x_{\alpha+1} \\ (x_{\alpha+2} - x)^3 & \text{if } x_{\alpha+1} \le x \le x_{\alpha+2} \\ 0 & \text{otherwise} \end{cases} \tag{143}$$

Using Hermite boundary conditions, the interpolated quantity s_{loc} is uniquely determined by the $(K+3)$ following equations:

$$g(x_{pi}) = s_{\text{loc}}(x_{pi}) = \frac{1}{6}c_{i-1} + \frac{2}{3}c_i + \frac{1}{6}c_{i+1} \qquad \forall i = 0, \cdots, K \tag{144}$$

$$g'(x_{pi}) = s'_{\text{loc}}(x_{pi}) = -\frac{1}{2h}c_{i-1} + \frac{1}{2h}c_{i+1} \qquad \text{for } i = 0 \text{ and } i = K \tag{145}$$

Finally, the spline coefficient vector $\mathbf{c} = [c_{-1}, \cdots, c_{K+1}]^T$ fulfills the $(K+3) \times (K+3)$ linear system $A\mathbf{c} = \mathbf{g}$, where $\mathbf{g}$ is the vector $\mathbf{g} = [g'(x_{p0}), g(x_{p0}), \cdots, g(x_{pK}), g'(x_{pK})]$ and A the matrix:

$$A = \frac{1}{6} \begin{pmatrix} -3/h & 0 & 3/h & 0 & \cdots & 0 \\ 1 & 4 & 1 & & & \\ & \ddots & \ddots & \ddots & & \\ & & & 1 & 4 & 1 \\ 0 & \cdots & 0 & -3/h & 0 & 3/h \end{pmatrix}$$

Notice that the A matrix can easily be factorized in LU form, where L is a lower triangular matrix with all diagonal terms equal to 1, and U is an upper triangular matrix. The standard algorithm leading to LU factorization

is based on Gauss elimination [9]:

$$L = \begin{pmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ -h/3 & 1 & 0 & \ddots & & \vdots \\ 0 & l_1 & 1 & \ddots & & \\ 0 & 0 & \ddots & \ddots & 0 & \vdots \\ \vdots & \ddots & \ddots & l_K & 1 & 0 \\ 0 & \cdots & 0 & -(3l_K)/h & (3l_{K+1})/h & 1 \end{pmatrix} \quad \text{and} \quad U = \frac{1}{6}\begin{pmatrix} -3/h & 0 & 3/h & 0 & \cdots & 0 \\ 0 & d_1 & 2 & 0 & & \vdots \\ 0 & 0 & d_2 & 1 & \ddots & \vdots \\ 0 & 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & d_{K+1} & 1 \\ 0 & \cdots & 0 & 0 & 0 & (3d_{K+2})/h \end{pmatrix}$$

where l_i and d_i are obtained as follows:

$d_1 = 4,\ l_1 = 1/4,\ d_2 = 4 - 2l_1 = 7/2,$
for $i = 2, K$
. $l_i = 1/d_i,$
. $d_{i+1} = 4 - l_i,$
end for
$l_{K+1} = 1/d_{K+1},\ d_{K+2} = 1 - l_{K+1}$

The LU decomposition of A needs only to be computed ones, at the beginning of the numerical code. At each time step, the matrix system $A\mathbf{c} = \mathbf{g}$ is then solved in two steps. The lower triangular matrix system $L\mathbf{x} = \mathbf{g}$ is first solved. The second step then consists in solving the upper triangular matrix system $U\mathbf{c} = \mathbf{x}$.

[9] Let L and U be defined by

$$L = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ l_{2,1} & 1 & 0 & \cdots & 0 \\ l_{3,1} & l_{3,2} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ l_{n,1} & l_{n,2} & \cdots & l_{n,n} & 1 \end{pmatrix} \quad \text{and} \quad U = \begin{pmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n} \\ 0 & u_{2,2} & \cdots & u_{2,n} \\ 0 & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & u_{n,n} \end{pmatrix}$$

The algorithm of LU factorization reads as follows:

$u_{1,1} = a_{1,1}$
for $j = 2, \cdots, n$
. $u_{1,j} = a_{1,j}$
. $l_{j,1} = a_{j,1}/a_{1,1}$
end for
for $i = 2, \cdots, n-1$
. $u_{i,i} = a_{i,i} - \sum_{k=1}^{i-1} l_{i,k}\, u_{k,i}$
. for $j = i+1, \cdots, n$
. $u_{i,j} = a_{i,j} - \sum_{k=1}^{i-1} l_{i,k}\, u_{k,j}$
. $l_{i,j} = \frac{1}{u_{i,i}}\left[a_{j,i} - \sum_{k=1}^{i-1} l_{j,k}\, u_{k,i}\right]$
. end for
end for
$u_{n,n} = a_{n,n} - \sum_{k=1}^{n-1} l_{n,k}\, u_{k,n}$

H.2. Approximation of the interface derivatives. — One of the most delicate points in the local cubic splines approach is to provide "good" approximations of the interface derivatives $g'(x_{p0})$ and $g'(x_{pK})$. Good approximations correspond to minimal error between the local and the global splines. Several classical solutions, like finite differences of various orders, or ad-hoc cubic spline approximations, have been explored in reference [**40**]. These authors show that the most robust approximation which remains valid with a relatively small number of grid points employs special combinations of cubic spline coefficients. Let us detail this method. According to Eq. (145), $c_{i-1} = \frac{3}{2}g_{i-1} - \frac{1}{4}c_{i-2} - \frac{1}{4}c_i$ and $c_{i+1} = \frac{3}{2}g_{i+1} - \frac{1}{4}c_i - \frac{1}{4}c_{i+2}$ where g_{i+1} (resp. g_{i-1}) represents the value of g at point $x_{p(i+1)}$ (resp. $x_{p(i-1)}$). Replacing c_{i-1} and c_{i+1} by these expressions in Eq. (144) leads to the iterative equality:

$$s'_i = \frac{3}{4h}(g_{i+1} - g_{i-1}) - \frac{1}{4}(s'_{i+1} + s'_{i-1}) \tag{146}$$

with the notation $s'_i = s'_{\text{loc}}(x_{pi})$. s'_{i+1} and s'_{i-1} can be computed using the same procedure. These derivatives involve s'_{i+2} and s'_{i-2}. Injecting these expressions in Eq. (146) then yields:

$$s'_i = \frac{6}{7h}(g_{i+1} - g_{i-1}) - \frac{3}{14h}(g_{i+2} - g_{i-2}) + \frac{1}{14}(s'_{i+2} + s'_{i-2}) \tag{147}$$

Again, s'_{i+2} and s'_{i-2} can be evaluated from Eq. (147), provided s'_{i+4} and s'_{i-4} are introduced. Incorporating these expressions in Eq. (147) then leads to the following approximation of s'_i:

$$\begin{aligned} s'_i = {} & \frac{39}{49\alpha h}(g_{i+1} - g_{i-1}) - \frac{3}{14\alpha h}(g_{i+2} - g_{i-2}) + \frac{3}{49\alpha h}(g_{i+3} - g_{i-3}) \\ & - \frac{3\gamma}{h}(g_{i+4} - g_{i-4}) + \gamma(s'_{i+4} + s'_{i-4}) \end{aligned}$$

where $\alpha = 1 - 2/14^2$ and $\gamma = 1/(14^2\alpha)$. The procedure is iterated once more, leading to the final retained approximation for s'_i, namely:

$$\begin{aligned} \beta h\, s'_i = {} & \frac{1}{49\alpha}(39 - 3\gamma)(g_{i+1} - g_{i-1}) - \frac{3}{14\alpha}(1 - \gamma)(g_{i+2} - g_{i-2}) \\ & + \frac{1}{49\alpha}(3 - 39\gamma)(g_{i+3} - g_{i-3}) - 3\gamma(g_{i+4} - g_{i-4}) \\ & + \frac{39\gamma}{49\alpha}(g_{i+5} - g_{i-5}) - \frac{3\gamma}{14\alpha}(g_{i+6} - g_{i-6}) + \frac{3\gamma}{49\alpha}(g_{i+7} - g_{i-7}) \\ & + 3\gamma^2(g_{i+8} - g_{i-8}) - \gamma^2 h(s'_{i+8} + s'_{i-8}) \end{aligned}$$

with $\beta = 1 - 2\gamma^2$. So as to close the system, s'_{i+8} and s'_{i-8} are evaluated using 4th order finite differences:

$$\begin{cases} s'_{i+8} = (g_{i+6} - 8g_{i+7} + 8g_{i+9} - g_{i+10})/12h \\ s'_{i-8} = (g_{i-10} - 8g_{i-9} + 8g_{i-7} - g_{i-6})/12h \end{cases}$$

For instance, when using 20 values of g, the approximation of $s'_{\text{loc}}(x_{pi})$ is given by (notice that $w_0 = 0$):

$$s'_{\text{loc}}(x_{pi}) = \sum_{j=1}^{10} w_j \frac{(g_{i+j} - g_{i-j})}{h}$$

where the w_j coefficients are given in table (1)

w_1	w_2	w_3	w_4
15126/18817	−4053/18817	1086/18817	−291/18817
0.8038475846	−0.2153903385	0.05771376946	−0.01546473933
w_5	w_6	w_7	w_8
78/18817	−503/451608	17/56451	−3/37634
$4.145187862E-3$	$-1.113797807E-3$	$3.011461267E-4$	$-7.971515119E-5$
w_9	w_{10}		
1/56451	−1/451608		
$1.771447804E-5$	$-2.214309755E-6$		

TABLE 1. Coefficients for the approximation of the interface derivatives

This approach can be viewed as the generalization of the standard cubic splines. Indeed, should the number of sub-domains be equal to one, corresponding actually to the full domain, this technique turns out to be equivalent to cubic splines.

Appendix I
The ENO reconstruction

I.1. Newton Divided Differences. — Let us consider the $m+1$ pairs $(x_i, f(x_i))$, and notice $\Pi_m f(x_i)$ the *interpolating polynomial* such that

$$\Pi_m f(x_i) = a_m x_i^m + \cdots + a_1 x_i + a_0 = f(x_i), \quad i = 0, \cdots, m \tag{148}$$

Such a polynomial is indeed unique (as proven *e.g.* in reference [**117**]). It takes the *Lagrange* form

$$\Pi_m f(x) = \sum_{i=0}^{m} f(x_i) l_i(x) \tag{149}$$

with

$$l_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{m} \frac{x - x_j}{x_i - x_j}, \quad i = 0, \cdots, m$$

However this Lagrange formulation does not reveal optimal from the numerical point of view. An alternative form with lower computational cost makes use of the *Newton Divided Differences formula*. The goal is to replace Π_m by the sum of Π_{m-1} (with $\Pi_{m-1} f(x_i) = f(x_i)$ for $i = 0, \cdots, m-1$ only) and the polynomial q_m of degree m, which depends on the x_i and on a single unknown coefficient, i.e.

$$\Pi_m f(x) = \Pi_{m-1} f(x) + q_m(x) \tag{150}$$

where $q_m \in \mathbb{P}_m$. From Eq. (150), it appears that q_m must satisfy

$$q_m(x) = a_m (x - x_0) \cdots (x - x_{m-1}) = a_m w_m(x)$$

where the *nodal polynomial* w_m of degree m has been introduced

$$w_m(x) = \prod_{i=0}^{m-1} (x - x_i)$$

The expression of the unknown coefficient a_m directly derives from the definition Eq. (150), and from the equality $\Pi_m f(x_m) = f(x_m)$:

$$a_m = \frac{f(x_m) - \Pi_{m-1} f(x_m)}{w_m(x_m)}$$

a_m is called the *m-th Newton Divided Difference*, also generally denoted by

$$a_m = f[x_0, x_1, \cdots, x_m], \quad \text{for } m \geq 1$$

Eq. (150) can then be reformulated as follows:

$$\Pi_m f(x) = \Pi_{m-1} f(x) + w_m(x) f[x_0, x_1, \cdots, x_m]$$

Eq. (150) can be recursively applied to $\Pi_{m-1} f(x)$, $\Pi_{m-2} f(x)$, ..., down to $\Pi_0 f(x)$. Taking $f(x_0) = f[x_0]$ and $w_0 = 1$ then provides what is commonly called the *Newton Divided Difference formula* for the interpolating polynomial

$$\Pi_m f(x) = \sum_{i=0}^{m} w_i(x) f[x_0, x_1, \cdots, x_i] \tag{151}$$

In view of obtaining an explicit expression for $a_m = f[x_0, x_1, \cdots, x_m]$, first remark that $\Pi_m f$ can also be expressed as

$$\Pi_m f(x) = \sum_{i=0}^{m} \frac{w_{m+1}(x)}{(x - x_i) w'_{m+1}(x_i)} f(x_i) \tag{152}$$

Equating both expressions of $\Pi_m f$, namely Eq. (151) and Eq. (152), leads to the following explicit expression of a_m:

$$f[x_0, x_1, \cdots, x_m] = \sum_{i=0}^{m} \frac{f(x_i)}{w'_{m+1}(x_i)} \tag{153}$$

Finally, an algebraic manipulation of (153) yields the *recursive formula* which reveals particularly efficient for computing divided differences

$$f[x_0] = f(x_0) \tag{154}$$
$$f[x_0, x_1, \cdots, x_m] = \frac{f[x_1, x_2, \cdots, x_m] - f[x_0, x_1, \cdots, x_{m-1}]}{x_m - x_0}, \quad m \geq 1$$

I.2. One Dimensional reconstruction. — Let us use the grid definition introduced in section 6.3. The cell average of any function $g(x)$ is defined by

$$\bar{g}_i = \frac{1}{\Delta x_i} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} g(\xi)\, \mathrm{d}\xi \tag{155}$$

The objective is to find the polynomial $p_i(x)$, of $k-1$ degree at most, defined on each interval I_i, which is the k-th order accurate approximation of the function $g(x)$, whatever $x \in I_i$:

$$p_i(x) = g(x) + \mathcal{O}(\Delta x_i^k), \quad x \in I_i,\ i = 1, \cdots, N$$

Given the localized interval I_i, and the order of accuracy k, let us define the $S(i)$ *stencil* based on r cells to the left, s cells to the right and I_i itself (see figure 11), with $r + s + 1 \equiv k$:

$$S(i) \equiv \{I_{i-r}, \cdots, I_{i+s}\}$$

There exists an unique polynomial $p_i(x)$ of degree at most $k - 1 = r + s$, for

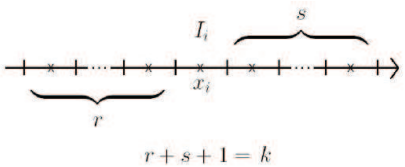


FIGURE 11. k-th order stencil for cell I_i

which the average on each cell of $S(i)$ is equal to that of $g(x)$

$$\bar{g}_j = \frac{1}{\Delta x_j} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} p_i(\xi)\, \mathrm{d}\xi, \quad j = i - r, \cdots, i + s \tag{156}$$

The quantity $\bar{g}_j \Delta x_j$ represents a *constraint* on the choice of $p_i(x)$. Let us denote $G(x)$ the antiderivative function of $g(x)$

$$G(x) \equiv \int_{-\infty}^{x} g(\xi)\, \mathrm{d}\xi \tag{157}$$

Here, the lower limit $-\infty$ can be replaced by any fixed number. Eq. (155) allows one to express $G(x_{i+\frac{1}{2}})$ as function of the averages $\bar{g}_j$:

$$G(x_{i+\frac{1}{2}}) = \sum_{j=-\infty}^{i} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} g(\xi)\, \mathrm{d}\xi = \sum_{j=-\infty}^{i} \bar{g}_j \Delta x_j \tag{158}$$

It follows that the antiderivative function $G(x)$ can be exactly computed at the cell boundaries as long as the averages $\{\bar{g}_j\}$ are known. Let us denote $P_i(x)$ the unique polynomial of degree smaller or equal to k, which interpolates $G(x_{j+\frac{1}{2}})$ at the following $k+1$ points

$$x_{i-r-\frac{1}{2}}, \cdots, x_{i+s+\frac{1}{2}} \tag{159}$$

Let then $p_i(x)$ be its derivative: $p_i(x) \equiv P_i'(x)$. Equality (156) is then easily proven, and

$$P_i'(x) = G'(x) + \mathcal{O}(\Delta x_i^k), \quad x \in I_i$$

I.3. ENO algorithm. — The following subsection proposes a quick introduction on the ENO method. For a deeper pedagogical insight, the interested reader can for instance refer to the lecture notes of Shu [**125**]. The basic idea of the ENO method consists in using an adaptive stencil which excludes the discontinuous cell in the stencil, if possible. The exclusion criteria is given by using a Newton Divided Differences which plays the role of an indicator of smoothness of the function g. Indeed, let us consider the Newton formulation of the interpolation polynomial (see section I.1). Similarly to Eq. (154), let us define the 0-th degree divided differences of function $G(x)$ as

$$G[x_{i-\frac{1}{2}}] \equiv G(x_{i-\frac{1}{2}}) \tag{160}$$

and the j-th degree divided differences, for $j \geq 1$ as

$$G[x_{i-\frac{1}{2}}, \cdots, x_{i+j-\frac{1}{2}}] \equiv \frac{G[x_{i+\frac{1}{2}}, \cdots, x_{i+j-\frac{1}{2}}] - G[x_{i-\frac{1}{2}}, \cdots, x_{i+j-\frac{3}{2}}]}{x_{i+j-\frac{1}{2}} - x_{i-\frac{1}{2}}} \tag{161}$$

We note that, in virtue of Eq. (158), the 0-th degree divided differences of $\bar{g}$ are the first degree divided differences of $G(x)$:

$$G[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] = \frac{G(x_{i+\frac{1}{2}}) - G(x_{i-\frac{1}{2}})}{x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}} = \bar{g}_i$$

It follows that the Newton form of the k-th degree interpolation polynomial $P(x)$, which interpolates $G(x)$ at the $k+1$ points (159), can be expressed using the divided differences (160) and (161):

$$P(x) = \sum_{j=0}^{k} G[x_{i-r-\frac{1}{2}}, \cdots, x_{i-r+j-\frac{1}{2}}] \prod_{m=0}^{j-1} (x - x_{i-r+m-\frac{1}{2}}) \tag{162}$$

and then deduce for $p(x)$

$$p(x) = \sum_{j=1}^{k} G[x_{i-r-\frac{1}{2}}, \cdots, x_{i-r+j-\frac{1}{2}}] \sum_{m=0}^{j-1} \prod_{\substack{l=0 \\ l\neq m}}^{j-1} (x - x_{i-r+l-\frac{1}{2}}) \tag{163}$$

The following properties (proposition 7) of the Newton Divided Difference are at the basis of the ENO method.

Proposition 7. — *The divided difference is a measurement of the smoothness of the function inside the stencil in the sense that:*

– *If $G(x) \in \mathcal{C}^j([x_{i-\frac{1}{2}}, x_{i+j-\frac{1}{2}}]$, then there exists*

$$\xi \in [x_{i-\frac{1}{2}}, x_{i+j-\frac{1}{2}}], \quad G[x_{i-\frac{1}{2}}, \cdots, x_{i+j-\frac{1}{2}}] = \frac{1}{j!}\frac{\mathrm{d}^j G}{\mathrm{d}x^j}(\xi) \tag{164}$$

– *If the k-th derivative of $G(x)$ is discontinuous with $0 \leq k \leq j$ on the interval $[x_{i-\frac{1}{2}}, x_{i+j-\frac{1}{2}}]$, then*

$$G[x_{i-\frac{1}{2}}, \cdots, x_{i+j-\frac{1}{2}}] = \mathcal{O}\left(\frac{1}{\Delta x^{k-j}}\right)[\omega^{(k)}] \tag{165}$$

where $\omega^{(k)}$ represents the jump of the k-th derivative.

Indeed, the ENO reconstruction consists in choosing the stencil for which the approximation is the smoothest, i.e. for which the divided differences is the smallest in module. The *ENO procedure* can be summarized as follows:

a) First define the polynomial of degree one interpolating the function $G(x)$ on the two-point stencil $\widetilde{S}(i) = \{x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\}$ and set $d_1(i) = i$,

b) Assuming that we gave the polynomial of degree k interpolating the function $G(x)$ at the points

$$x_{d_k(i)-\frac{1}{2}}, \cdots, x_{d_k(i)+k-\frac{1}{2}}$$

c) To find the polynomial $p_{k+1}(x)$, we consider $k+2$ points obtained by adding to the previous ones the first point on the left or on the right, and choose the point for which the divided difference is the smallest

$$d_{k+1}(i) = \begin{cases} d_k(i) - 1 & \text{if condition (166)} \\ d_k(i) & \text{else} \end{cases}$$

with the condition

$$\left|G[x_{d_k(i)-\frac{3}{2}}, \cdots, x_{d_k(i)+k-\frac{1}{2}}]\right| \leq \left|G[x_{d_k(i)-\frac{1}{2}}, \cdots, x_{d_k(i)+k+\frac{1}{2}}]\right| \tag{166}$$

d) Repeat step c) unless the order required is obtained,

e) Finally, used the reconstruction method to compute the approximation of $g(x)$.

References

[1] J. Ahlberg, E. Nilson & J. Walsh – *The theory of splines and their applications (1st edition)*, new york ed., Academic Press, 1967.

[2] S. Allfrey & R. Hatzky – A revised delta f algorithm for nonlinear pic simulations, *Comp. Phys. Comm* **2** (2003), no. 154, p. 98–104.

[3] H. Amman – *Ordinary differential equations*, De Gruyter, 1990.

[4] A. Arakawa – Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. part i, *J. Comput. Phys.* **135** (1997), no. 2, p. 103–114.

[5] T. D. Arber & R. G. L. Vann – A critical comparison of eulerian-grid-based vlasov solvers, *J. Comput. Phys.* (2002), no. 180, p. 339–357.

[6] A. Y. Aydemir – A unified monte carlo interpretation of particle simulations and applications to non-neutral plasmas, *Phys. Plasmas* **1** (1994), no. 4, p. 822–831.

[7] R. Barthelme, P. Ciarlet & E. Sonnendrücker – Generalized formulations of maxwell's equations for numerical vlasov-maxwell simulations, *Mathematical Models and Methods in Applied Science* **17** (2007), no. 5, p. 657–680.

[8] H. L. Berk, B. N. Breizman & M. Pekker – Numerical simulation of bump-on-tail instability with source and sink, *Phys. Plasmas* **2** (1995), no. 8, p. 3007–3016.

[9] H. L. Berk, B. N. Breizman & M. Pekker – Nonlinear dynamics of a driven mode near marginal stability, *Phys. Rev. Lett.* **76** (1996), no. 8, p. 1256–1259.

[10] P. Bertrand & M. R. Feix – Non linear electron plasma oscillation: the "water bag model", *Physics Letters A* **28** (1968), no. 1, p. 68 – 69.

[11] N. Besse – Convergence of a semi-lagrangian scheme for the one-dimensional vlasov-poisson system, *SIAM Journal on Numerical Analysis* **42** (2004), no. 1, p. 350–382.

[12] N. Besse & P. Bertrand – Quasilinear analysis of the gyro-water-bag model, *EPL (Europhysics Letters)* **83** (2008), no. 2, p. 25003.

[13] N. Besse & M. Mehrenberger – Convergence of classes of high-order semi-lagrangian schemes for the vlasov-poisson system, *Mathematics of Computation* **77** (2008), no. 61, p. 93–123.

[14] N. Besse & P. Bertrand – Gyro-water-bag approach in nonlinear gyrokinetic turbulence, *J. Comput. Phys.* **228** (2009), no. 11, p. 3973 – 3995.

[15] P. Beyer, S. Benkadda, X. Garbet & P. H. Diamond – Nondiffusive transport in tokamaks: Three-dimensional structure of bursts and the role of zonal flows, *Phys. Rev. Lett.* **85** (2000), no. 23, p. 4892–4895.

[16] C. Birdsall & A. Langdon – *Plasma physics via computer simulation*, McGraw-Hill, 1985.

[17] J. P. Boris & D. L. Book – Flux-corrected transport. iii. minimal-error fct algorithms, *Journal of Computational Physics* **20** (1976), no. 4, p. 397 – 431.

[18] J. P. Boris & D. L. Book – Flux-corrected transport, *J. Comput. Phys.* **135** (1997), no. 2, p. 172 – 186.

[19] A. Bottino, A. G. Peeters, R. Hatzky, S. Jolliet, B. F. McMillan, T. M. Tran & L. Villard – Nonlinear low noise particle-in-cell simulations of

electron temperature gradient driven turbulence, *Phys. Plasmas* **14** (2007), no. 1, p. 010701.

[20] A. Bottino, B. Scott, S. Brunner, B. McMillan, T. Tran, T. Vernay, L. Villard, S. Jolliet, R. Hatzky & A. Peeters – Global nonlinear electromagnetic simulations of tokamak turbulence, *Plasma Science, IEEE Transactions on* **38** (2010), no. 9, p. 2129 –2135.

[21] F. Bouchut, F. Golse & M. Pulvirenti – *Kinetic equations and aymptotic theory*, Gauthier-Villars, Paris, 2000.

[22] J. P. Braeunig, N. Crouseilles, V. Grandgirard, G. Latu, M. Mehrenberger & E. Sonnendrücker – Some numerical aspects of the conservative psm scheme in a 4d drift-kinetic code, *INRIA report* (2009), no. 7109, p. 1–37.

[23] A. Brizard & T. Hahm – Foundations of nonlinear gyrokinetic theory, *Rev. Mod. Phys.* **79** (2007), no. 2, p. 421–468.

[24] M. Brunetti, V. Grandgirard, O. Sauter, J. Vaclavik & L. Villard – A semi-lagrangian code for nonlinear global simulations of electrostatic drift-kinetic itg modes, *Comp. Phys. Comm* **163** (2004), no. 1, p. 1 – 21.

[25] ———, Fine-scale structure and negative-density regions: Comparison of numerical methods for solving the advection equation, *Transport Theory and Statistical Physics* **34** (2005), no. 3-5, p. 261–274.

[26] S. Brunner, E. Valeo & J. A. Krommes – Collisional delta-f scheme with evolving background for transport time scale simulations, *Phys. Plasmas* **6** (1999), no. 12, p. 4504–4521.

[27] J. Büchner – Vlasov code simulation, *Advanced Methods for Space Simulations* (2007), p. 23–46.

[28] J. Candy, R. E. Waltz, M. R. Fahey & C. Holland – The effect of ion-scale dynamics on electron-temperature-gradient turbulence, *Plasma Phys. Control. Fusion* **49** (2007), no. 8, p. 1209.

[29] J. Candy & R. Waltz – An eulerian gyrokinetic-maxwell solver, *J. Comput. Phys.* **186** (2003), no. 2, p. 545–581.

[30] J. R. Cary – Lie transform perturbation theory for hamiltonian systems, *Physics Reports* **79** (1981), no. 2, p. 129 – 159.

[31] J. R. Cary & R. G. Littlejohn – Noncanonical hamiltonian mechanics and its application to magnetic field line flow, *Annals of Physics* **151** (1983), no. 1, p. 1 – 34.

[32] C. S. Chang & S. Ku – Spontaneous rotation sources in a quiescent tokamak edge plasma, *Phys. Plasmas* **15** (2008), no. 6, p. 062510.

[33] C. S. Chang, S. Ku, P. H. Diamond, Z. Lin, S. Parker, T. S. Hahm & N. Samatova – Compressed ion temperature gradient turbulence in diverted tokamak edge, *Phys. Plasmas* **056108** (2009), no. 5, p. 16.

[34] C. Cheng & G. Knorr – The integration of the vlasov equation in configuration spaces, *J. Comput. Phys.* (1976), no. 22, p. 330–351.

[35] P. Colella & E. G. Puckett – *Modern numerical methods for fluid flow*, Class Notes, university of California, 1997.

[36] P. Colella & P. R. Woodward – The piecewise parabolic method (ppm) for gas-dynamical simulations, *J. Comput. Phys.* **54** (1984), no. 1, p. 174 – 201.

[37] iter physics expert group on confinement – Chapter 2: Plasma confinement and transport, *Nuclear Fusion* **39** (1999), no. 12, p. 2175.

[38] N. Crouseilles, M. Mehrenberger & H. Sellama – Numerical solution of the gyroaverage operator for the finite gyroradius guiding-center model, *CICP* (2010), p. 1.

[39] N. Crouseilles, M. Mehrenberger & E. Sonnendrücker – Conservative semi-lagrangian schemes for vlasov equations, *J. Comput. Phys.* **229** (2010), no. 6, p. 1927 – 1953.

[40] N. Crouseilles, G. Latu & E. Sonnendrücker – A parallel vlasov solver based on local cubic spline interpolation on patches, *J. Comput. Phys.* **228** (2009), no. 5, p. 1429 – 1446.

[41] N. Crouseilles, T. Respaud & E. Sonnendrüker – A forward semi-lagrangian method for the numerical solution of the vlasov equation, *Comp. Phys. Comm* **180** (2009), no. 10, p. 1730 – 1745.

[42] G. Darmet, P. Ghendrih, Y. Sarazin, X. Garbet & V. Grandgirard – Intermittency in flux driven kinetic simulations of trapped ion turbulence, *Communications in Nonlinear Science and Numerical Simulation* **13** (2008), no. 1, p. 53–58, Vlasovia 2006: The Second International Workshop on the Theory and Applications of the Vlasov Equation.

[43] C. DeBoor – *A practical guide to splines*, applied mathematical sciences 27 ed., Springer-Verlag, New York, 2001.

[44] R. E. Denton & M. Kotschenreuther – [delta]f algorithm, *Journal of Computational Physics* **119** (1995), no. 2, p. 283–294.

[45] G. Depret, X. Garbet, P. Bertrand & A. Ghizzo – Trapped-ion driven turbulence in tokamak plasmas, *Plasma Phys. Control. Fusion* **23** (2000), no. 42, p. 949–971.

[46] A. Dimits, T. Williams, J. Byers & B. Cohen – Scalings of ion-temperature-gradient-driven anomalous transport in tokamaks, *Phys. Rev. Lett.* **77** (1996), no. 1, p. 71–74.

[47] W. Dorland, F. Jenko, M. Kotschenreuther & B. Rogers – Electron temperature gradient turbulence, *Phys. Rev. Lett.* (2000), no. 85, p. 5579–5582.

[48] D. R. Durran – *Numerical methods for wave equations in geophysical fluid dynamics*, Springer-Verlag, New York, 1998.

[49] A. D. *et al* – Comparisons and physics basis of tokamak transport models and turbulence simulations, *Phys. Plasmas* **7** (2000), no. 3, p. 969–983.

[50] S. Y. *et al* – Interplay between transport barriers and density gradient, *Phys. Plasmas* **13** (2006), no. 9, p. 092307.

[51] G. L. Falchetto, B. D. Scott, P. Angelino, A. Bottino, T. Dannert, V. Grandgirard, S. Janhunen, F. Jenko, S. Jolliet, A. Kendl, B. F. McMillan, V. Naulin, A. H. Nielsen, M. Ottaviani, A. G. Peeters, M. J. Pueschel, D. Reiser, T. T. Ribeiro & M. Romanelli – The european turbulence code benchmarking effort: turbulence driven by thermal gradients in magnetically confined plasmas, *Plasma Phys. Control. Fusion* **50** (2008), no. 12, p. 124015.

[52] M. R. Feix & P. Bertrand – A universal model: The vlasov equation, *Transport Theory and Statistical Physics* **34** (2005), no. 1, p. 7–62.

[53] E. Fijalkow – A numerical solution of the vlasov equation, *Comp. Phys. Comm* (1999), no. 116, p. 319–328.

[54] F. Filbet – Comparison of eulerian vlasov solvers, *Computer Physics Communications* **150** (2003), no. 3, p. 247 – 266.

[55] F. Filbet, E. Sonnendrücker & P. Bertrand – Conservative numerical schemes for the vlasov equation, *J. Comput. Phys.* (2001), no. 172, p. 166–187.

[56] G. Fishman – Monte carlo, control variates and stochastic ordering, *SIAM J. Sci. Stat. Comput.* **10** (1989), no. 1, p. 187–204.

[57] X. Garbet, G. Dif-Pradalier, C. Nguyen, Y. Sarazin, V. Grandgirard & P. Ghendrih – Neoclassical equilibrium in gyrokinetic simulations, *Phys. Plasmas* **16** (2009), no. 6, p. 062503.

[58] X. Garbet, Y. Idomura, L. Villard & T. Watanabe – Gyrokinetic simulations of turbulent transport, *Nuclear Fusion* **50** (2010), no. 4, p. 043002.

[59] X. Garbet, Y. Sarazin, P. Beyer, P. Ghendrih, R. Waltz, M. Ottaviani & S. Benkadda – Flux driven turbulence in tokamaks, *Nuclear Fusion* **39** (1999), no. 11Y, p. 2063.

[60] X. Garbet, Y. Sarazin, V. Grandgirard, G. Dif-Pradalier, G. Darmet, P. Ghendrih, P. Angelino, P. Bertrand, N. Besse, E. Gravier, P. Morel, E. Sonnendrücker, N. Crouseilles, J.-M. Dischler, G. Latu, E. Violard, M. Brunetti, S. Brunner, X. Lapillonne, T.-M. Tran, L. Villard & M. Boulet – Beyond scale separation in gyrokinetic turbulence, *Nuclear Fusion* **47** (2007), no. 9, p. 1206.

[61] H. Goldstein, C. Poole & J. Safko – *Classical mechanics*, 3rd ed. ed., Addison-Wesley, San Francisco, 2002.

[62] T. Görler & F. Jenko – Scale separation between electron and ion thermal transport, *Phys. Rev. Lett.* **100** (2008), no. 18, p. 185002.

[63] V. Grandgirard, M. Brunetti, P. Bertrand, N. Besse, X. Garbet, P. Ghendrih, G. Manfredi, Y. Sarazin, O. Sauter, E. Sonnendrücker, J. Vaclavik & L. Villard – A drift-kinetic semi-lagrangian 4d code for ion turbulence simulation, *J. Comput. Phys.* **217** (2006), no. 2, p. 395 – 423.

[64] V. Grandgirard, Y. Sarazin, P. Angelino, A. Bottino, N. Crouseilles, G. Darmet, G. Dif-Pradalier, X. Garbet, P. Ghendrih, S. Jolliet, G. Latu, E. Sonnendrücker & L. Villard – Global full-f gyrokinetic simulations of plasma turbulence, *Plasma Physics and Controlled Fusion* **49** (2007), no. 12B, p. B173.

[65] E. Gravier, R. Klein, P. Morel, N. Besse & P. Bertrand – Gyrokinetic-water-bag modeling of low-frequency instabilities in a laboratory magnetized plasma column, *Phys. Plasmas* **15** (2008), no. 12, p. 122103.

[66] J. Guterl, J. P. Braeunig, N. Crouseilles, V. Grandgirard, G. Latu, M. Mehrenberger & E. Sonnendrücker – Test of some numerical limiters for the conservative psm scheme for 4d drift-kinetic simulations, *INRIA report* **RR-7467** (2010), p. 1–63.

[67] J. Hammerslay & D. Handscomb – *Monte carlo methods*, Methuen and Co, London, 1964.

[68] G. W. Hammett & F. W. Perkins – Fluid moment models for landau damping with application to the ion-temperature-gradient instability, *Phys. Rev. Lett.* **64** (1990), no. 25, p. 3019–3022.

[69] A. Harten, B. Engquist, S. Osher & S. R. Chakravarthy – Uniformly high-order accurate essentially non-oscillatory schemes, iii, *J. Comput. Phys.* (1987), no. 71, p. 231–303.

[70] A. Harten & S. Osher – Uniformly high-order accurate non-oscillatory schemes, i, *SIAM Journal on Numerical Analysis* (1987), no. 24, p. 279–309.

[71] R. Hatzky, A. Könies & A. Mishchenko – Electromagnetic gyrokinetic pic simulation with an adjustable control variates method, *J. Comput. Phys.* **225** (2007), no. 1, p. 568–590.

[72] R. Hatzky, T. Tran, A. Konies, R. Kleiber & S. J. Allfrey – Energy conservation in a nonlinear gyrokinetic particle-in-cell code for ion-temperature-gradient-driven modes in theta-pinch geometry, *Phys. Plasmas* **9** (2002), no. 3, p. 898–912.

[73] J. A. Heikkinen, T. P. Kiviniemi, T. Kurki-Suonio, A. G. Peeters & S. K. Sipilä – Particle simulation of the neoclassical plasmas, *J. Comput. Phys.* **173** (2001), no. 2, p. 527–548.

[74] F. Huot, A. Ghizzo, P. Bertrand, E. Sonnendrücker & O. Coulaud – Instability of the time splitting scheme for the one-dimensional and relativistic vlasov-maxwell system, *J. Comput. Phys.* **185** (2003), no. 2, p. 512–531.

[75] Y. Idomura, M. Ida, S. Tokuda & L. Villard – New conservative gyrokinetic full-f vlasov code and its comparison to gyrokinetic [delta]f particle-in-cell code, *J. Comput. Phys.* **226** (2007), no. 1, p. 244 – 262.

[76] Y. Idomura, S. Tokuda, Y. Kishimoto & M. Wakatani – Gyrokinetic theory of drift waves in negative shear tokamaks, *Nuclear Fusion* **41** (2001), no. 4, p. 437.

[77] Y. Idomura, H. Urano, N. Aiba & S. Tokuda – Study of ion turbulent transport and profile formations using global gyrokinetic full- f vlasov simulation, *Nuclear Fusion* **49** (2009), no. 6, p. 065029.

[78] Y. Idomura, M. Ida, T. Kano, N. Aiba & S. Tokuda – Conservative global gyrokinetic toroidal full-f five-dimensional vlasov simulation, *Comp. Phys. Comm* **179** (2008), no. 6, p. 391–403.

[79] Y. Idomura, T.-H. Watanabe & H. Sugama – Kinetic simulations of turbulent fusion plasmas, *Comptes Rendus Physique* **7** (2006), no. 6, p. 650–669.

[80] S. Inagaki, N. Tamura, T. Tokuzawa, K. Ida, T. Shimozuma, S. Kubo, H. Tsuchiya, Y. Nagayama, K. Kawahata, S. Sudo, K. Itoh & S.-I. Itoh – Radial structure of fluctuation in electron itb plasmas of lhd, *proceedings of the 23rd IAEA Fusion Energy Conference, (Daejon, Korea)* (2010), no. EXC/7-4Ra, p. 1–7.

[81] F. Jenko, W. Dorland, M. Kotschenreuther & B. Rogers – Massively parallel vlasov simulation of electromagnetic drift-wave turbulence, *Comp. Phys. Comm* (2000), no. 125, p. 196–209.

[82] S. Jolliet – *Gyrokinetic particle-in-cell global simulations of ion-temperature-gradient and collisionless-trapped-electron-mode turbulence in tokamaks*, switzerland ed., Ecole Polytechnique Fédérale de Lausanne, 2008.

[83] S. Jolliet, A. Bottino, P. Angelino, R. Hatzky, T. M. Tran, B. Mcmillan, O. Sauter, K. Appert, Y. Idomura & L. Villard – A global collisionless pic code in magnetic coordinates, *Comp. Phys. Comm* **177** (2007), no. 5, p. 409 – 425.

[84] S. Jolliet, B. F. McMillan, T. Vernay, L. Villard, A. Bottino & P. Angelino – Quasisteady and steady states in global gyrokinetic particle-in-cell simulations, *Phys. Plasmas* **16** (2009), no. 5, p. 052307.

[85] M. H. Kalos & P. A. Whitlock – *Monte carlo methods (second, revised and enlarged edition)*, Wiley-VCH, 2008.

[86] R. Klein, E. Gravier, P. Morel, N. Besse & P. Bertrand – Gyrokinetic water-bag modeling of a plasma column: Magnetic moment distribution and finite larmor radius effects, *Phys. Plasmas* **16** (2009), no. 8, p. 082106.

[87] A. J. Klimas – A method for overcoming the velocity space filamentation problem in collisionless plasma model solutions, *J. Comput. Phys.* **68** (1987), no. 1, p. 202 – 226.

[88] A. J. Klimas & W. M. Farrell – A splitting algorithm for vlasov simulation with filamentation filtration, *J. Comput. Phys.* **110** (1994), no. 1, p. 150 – 163.

[89] M. Kotschenreuther, G. Rewoldt & W. M. Tang – Comparison of initial value and eigenvalue codes for kinetic toroidal plasma instabilities, *Comp. Phys. Comm* **88** (1995), no. 2-3, p. 128–140.

[90] J. A. Krommes – Thermostatted delta f, *Phys. Plasmas* **6** (1999), no. 5, p. 1477–1494.

[91] J. A. Krommes & G. Hu – The role of dissipation in the theory and simulations of homogeneous plasma turbulence, and resolution of the entropy paradox, *Phys. Plasmas* **1** (1994), no. 10, p. 3211–3238.

[92] X. Lapillonne, S. Brunner, T. Dannert, S. Jolliet, A. Marinoni, L. Villard, T. Gorler, F. Jenko & F. Merz – Effects of geometry on linear and non-linear gyrokinetic simulations, and development of a global version of the gene code, *AIP conference proceedings, Joint Varenna-Lausanne International Workshop on Theory of Fusion Plasmas* **1069** (2008), p. 289–294.

[93] G. Latu, N. Crouseilles, V. Grandgirard & E. Sonnendrücker – Gyrokinetic simulations in general geometry and applications to collisional damping of zonal flows, *Recent Advances in Parallel Virtual Machine and MPI* **4757** (2007), p. 356–364.

[94] W. W. Lee – Gyrokinetic approach in particle simulation, *Physics of Fluids* **26** (1983), no. 2, p. 556–562.

[95] M. Lesur, Y. Idomura & X. Garbet – Fully nonlinear features of the energetic beam-driven instability, *Phys. Plasmas* **16** (2009), no. 9, p. 092305–092317.

[96] D. W. Levy, K. G. Powell & B. van Leer – Use of a rotated riemann solver for the two-dimensional euler equations, *J. Comput. Phys.* **106** (1993), no. 2, p. 201 – 214.

[97] Z. Lin, T. S. Hahm, W. W. Lee, W. M. Tang & R. B. White – Turbulent transport reduction by zonal flows: Massively parallel simulations, *Science* **281** (1998), no. 5384, p. 1835–1837.

[98] A. McDonald – Accuracy of multiply-upstream, semi-lagrangian advective schemes, *Monthly Weather Review* **112** (1984), no. 6, p. 1267–1275.

[99] ———, Accuracy of multiply-upstream, semi-lagrangian advective schemes 2, *Monthly Weather Review* **114** (1987), p. 1446–1450.

[100] B. F. McMillan, S. Jolliet, T. M. Tran, L. Villard, A. Bottino & P. Angelino – Long global gyrokinetic simulations: Source terms and particle noise control, *Physics of Plasmas* **15** (2008), no. 5, p. 052308.

[101] ———, Avalanchelike bursts in global gyrokinetic simulations, *Phys. Plasmas* **16** (2009), no. 2, p. 022310.

[102] F. Merz – *Gyrokinetic simulation of multimode plasma turbulence*, Thesis, Universität Münster, 2009.

[103] P. Morel, E. Gravier, N. Besse, R. Klein, A. Ghizzo, P. Bertrand, X. Garbet, P. Ghendrih, V. Grandgirard & Y. Sarazin – Gyrokinetic modeling: A multi-water-bag approach, *Phys. Plasmas* **14** (2007), no. 11, p. 112109.

[104] Y. Morinishi, T. S. Lund, O. V. Vasilyev & P. Moin – Fully conservative higher order finite difference schemes for incompressible flow, *J. Comput. Phys.* **143** (1998), no. 1, p. 90–124.

[105] Y. Morinishi – Skew-symmetric form of convective terms and fully conservative finite difference schemes for variable density low-mach number flows, *J. Comput. Phys.* **229** (2010), no. 2, p. 276 – 300.

[106] Y. Morinishi, O. V. Vasilyev & T. Ogi – Fully conservative finite difference scheme in cylindrical coordinates for incompressible flow simulations, *J. Comput. Phys.* **197** (2004), no. 2, p. 686 – 710.

[107] T. Nakamura & T. Yabe – Cubic interpolated propagation scheme for solving the hyper-dimensional vlasov-poisson equation in phase space, *Comp. Phys. Comm* (1999), no. 120, p. 122–154.

[108] T. Nakamura, R. Tanaka, T. Yabe & K. Takizawa – Exactly conservative semi-lagrangian scheme for multi-dimensional hyperbolic equations with directional splitting technique, *J. Comput. Phys.* **174** (2001), no. 1, p. 171 – 207.

[109] W. M. Nevins, J. Candy, S. Cowley, T. Dannert, A. Dimits, W. Dorland, C. Estrada-Mila, G. W. Hammett, F. Jenko, M. J. Pueschel & D. E. Shumaker – Characterizing electron temperature gradient turbulence via numerical simulation, *Phys. Plasmas* **13** (2006), no. 12, p. 122306.

[110] W. M. Nevins, G. W. Hammett, A. M. Dimits, W. Dorland & D. E. Shumaker – Discrete particle noise in particle-in-cell simulations of plasma microturbulence, *Phys. Plasmas* **12** (2005), no. 12, p. 122305.

[111] S. Parker & W. Lee – A fully nonlinear characteristic method for gyrokinetic simulation, *Phys. Fluids B* **5** (1993), no. 1, p. 77–86.

[112] A. G. Peeters & D. Strintzi – The effect of a uniform radial electric field on the toroidal ion temperature gradient mode, *Phys. Plasmas* **11** (2004), no. 8, p. 3748–3751.

[113] D. Petrina, V. Gerasimenko & P. Malyshev – *Mathematical foundations of classical statistical mechanics: continuous system*, advanced studies in contemporary mathematics ed., Taylor and Francis inc., 2002.

[114] M. J. Pueschel, T. Dannert & F. Jenko – On the role of numerical dissipation in gyrokinetic vlasov simulations of plasma microturbulence, *Comp. Phys. Comm* **181** (2010), no. 8, p. 1428–1437.

[115] D. K. Purnell – Solution of the advective equation by upstream interpolation with a cubic spline, *Monthly Weather Review* **104** (1976), no. 1, p. 42–48.

[116] J.-M. Qiu & A. Christlieb – A conservative high order semi-lagrangian weno method for the vlasov equation, *J. Comput. Phys.* **229** (2010), no. 4, p. 1130 – 1149.

[117] A. Quarteroni, R. Sacco & F. Saleri – *Numerical mathematics*, second edition ed., Springer, Berlin, 2007.

[118] C.-M. Ryu, T. Rhee, T. Umeda, P. H. Yoon & Y. Omura – Turbulent acceleration of superthermal electrons, *Phys. Plasmas* **14** (2007), no. 10, p. 100701.

[119] Y. Sarazin, G. Dif-Pradalier, D. Zarzoso, X. Garbet, P. Ghendrih & V. Grandgirard – Entropy production and collisionless fluid closure, *Plasma Phys. Control. Fusion* **51** (2009), no. 11, p. 115003.

[120] Y. Sarazin, V. Grandgirard, J. Abiteboul, S. Allfrey, X. Garbet, P. Ghendrih, G. Latu, A. Strugarek & G. Dif-Pradalier – Large scale dynamics in flux driven gyrokinetic turbulence, *Nuclear Fusion* **50** (2010), no. 5, p. 054004.

[121] Y. Sarazin, V. Grandgirard, E. Fleurence, X. Garbet, P. Ghendrih, P. Bertrand & G. Depret – Kinetic features of interchange turbulence, *Plasma Phys. Control. Fusion* **47** (2005), no. 10, p. 1817–1840.

[122] H. Schmitz & R. Grauer – Darwin-vlasov simulations of magnetised plasmas, *Journal of Computational Physics* **214** (2006), no. 2, p. 738–756.

[123] J. W. Schumer & J. P. Holloway – Vlasov simulations using velocity-scaled hermite representations, *J. Comput. Phys.* **144** (1998), no. 2, p. 626 – 661.

[124] M. M. Shoucri & R. R. J. Gagné – Splitting schemes for the numerical solution of a two-dimensional vlasov equation, *J. Comput. Phys.* **27** (1978), no. 3, p. 315–322.

[125] C.-W. Shu – Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, *NASA Report* (1997), no. CR-97-206253, p. 1–78.

[126] E. Sonnendrücker, J. Roche, P. Bertrand & A. Ghizzo – The semi-lagrangian method for the numerical resolution of vlasov equation, *J. Comput. Phys.* **149** (1999), no. 2, p. 201–220.

[127] A. Staniforth & J. Côté – Semi-lagrangian integration schemes for atmospheric models - a review, *Monthly Weather Review* (1991), no. 119, p. 2206–2223.

[128] G. Strang – On the construction and comparison of difference schemes, *SIAM J. Numer. Anal.* **5** (1968), p. 506–517.

[129] R. Sydora, V. Decyk & J. Dawson – Fluctuation-induced heat transport results from a large global 3d toroidal particle simulation model, *Plasma Phys. Control. Fusion* **38** (1996), no. 12A, p. A281.

[130] H. Takewaki, A. Nishiguchi & T. Yabe – Cubic interpolated pseudo-particle method (cip) for solving hyperbolic-type equations, *J. Comput. Phys.* **61** (1985), no. 2, p. 261 – 268.

[131] K. Takizawa, T. Yabe & T. Nakamura – Multi-dimensional semi-lagrangian scheme that guarantees exact conservation, *Comp. Phys. Comm* **148** (2002), no. 2, p. 137 – 159.

[132] W. Tang & V. Chan – Advances and challenges in computational plasma science, *Plasma Physics and Controlled Fusion* **47** (2005), no. 2, p. R1.

[133] C. Temperton & A. Staniforth – An efficient two-time-level semi-lagrangian semi-implicit integration scheme, *Quarterly Journal of the Royal Meteorological Society* **113** (1987), no. 477, p. 1025–1039.

[134] P. W. Terry, M. Greenwald, J.-N. Leboeuf, G. R. McKee, D. R. Mikkelsen, W. M. Nevins, D. E. Newman, D. P. Stotler, T. G. on Verification Validation, U. B. P. Organization, & U. T. T. Force – Validation in fusion research: Towards guidelines and best practices, *Phys. Plasmas* **15** (2008), no. 6, p. 062503.

[135] K. Toda, Y. Ogata & T. Yabe – Multi-dimensional conservative semi-lagrangian method of characteristics cip for the shallow water equations, *J. Comput. Phys.* **228** (2009), no. 13, p. 4917 – 4944.

[136] H. S. Tomo-Hiko Watanabe & T. Sato – A nondissipative simulation method for the drift kinetic equation, *J. Phys. Soc. Japan* **70** (2001), p. 3565–3576.

[137] T. Umeda, M. Ashour-abdalla & D. Schriver – Comparison of numerical interpolation schemes for one-dimensional electrostatic vlasov code, *Journal of Plasma Physics* **72** (2006), no. 06, p. 1057–1060.

[138] T. Umeda – A conservative and non-oscillatory scheme for vlasov code simulations, *Eart Planets Space* (2008), no. 60, p. 773–779.

[139] T. Utsumi, T. Kunugi & J. Koga – A numerical method for solving the one-dimensional vlasov–poisson equation in phase space, *Comp. Phys. Comm* **108** (1998), no. 2-3, p. 159 – 179.

[140] R. Vann – *Characterization of fully nonlinear berk-breizman phenomenology*, thesis, university of Warwick, 2003.

[141] C. Villani – *Landau damping*, panoramas et synthèses ed., SMF, to be published.

[142] L. Villard, S. Allfrey, A. Bottino, M. Brunetti, G. Falchetto, V. Grandgirard, R. Hatzky, J.Nührenberg, A. Peeters, O. Sauter, S. Sorge & J. Vaclavik – Full radius linear and nonlinear gyrokinetic simulations for tokamaks and stellarators: zonal flows, applied exb flows, trapped electrons and finite beta, *Nuclear Fusion* **44** (2004), no. 1, p. 172.

[143] L. Villard, A. Bottino, S. Brunner, A. Casati, J. Chowdhury, T. Dannert, R. Ganesh, X. Garbet, T. Görler, V. Grandgirard, R. Hatzky, Y. Idomura, F. Jenko, S. Jolliet, S. K. Aghdam, X. Lapillonne, G. Latu, B. F. McMillan, F. Merz, Y. Sarazin, T. M. Tran & T. Vernay – Gyrokinetic simulations of turbulent transport: size scaling and chaotic behaviour, *Plasma Phys. Control. Fusion* **52** (2010), no. 12, p. 124038.

[144] J. Villasenor & O. Buneman – Rigorous charge conservation for local electromagnetic field solvers, *Computer Physics Communications* **69** (1992), no. 2-3, p. 306 – 316.

[145] R. E. Waltz, G. M. Staebler, W. Dorland, G. W. Hammett, M. Kotschenreuther & J. A. Konings – A gyro-landau-fluid transport model, *Phys. Plasmas* **4** (1997), no. 7, p. 2482–2496.

[146] W. X. Wang, T. S. Hahm, W. W. Lee, G. Rewoldt, J. Manickam & W. M. Tang – Nonlocal properties of gyrokinetic turbulence and the role of [bold e] x [bold b] flow shear, *Phys. Plasmas* **14** (2007), no. 7, p. 072306.

[147] T. H. Watanabe & H. Sugama – Velocity-space structures of distribution function in toroidal ion temperature gradient turbulence, *Nuclear Fusion* **46** (2006), p. 24–32.

[148] T. Watanabe & H. Sugama – Kinetic simulation of a quasisteady state in collisionless ion temperature gradient driven turbulence, *Phys. Plasmas* **9** (2002), no. 9, p. 3659–3662.

[149] J. Wesson – *Tokamaks*, Okford Science Publications, 1997.

[150] K. L. Wong, M. S. Chu, T. C. Luce, C. C. Petty, P. A. Politzer, R. Prater, L. Chen, R. W. Harvey, M. E. Austin, L. C. Johnson, R. J. La Haye & R. T. Snider – Internal kink instability during off-axis electron cyclotron current drive in the diii-d tokamak, *Phys. Rev. Lett.* **85** (2000), no. 5, p. 996–999.

[151] F. Xiao, T. Yabe & T. Ito – Constructing oscillation preventing scheme for advection equation by rational function, *Comp. Phys. Comm* **93** (1996), no. 1, p. 1–12.

[152] F. Xiao, T. Yabe, G. Nizam & T. Ito – Constructing a multi-dimensional oscillation preventing scheme for the advection equation by a rational function, *Comp. Phys. Comm* **94** (1996), no. 2-3, p. 103 – 118.

[153] T. Yabe – A universal cubic interpolation solver for compressible and incompressible fluids, *Shock Waves* **1** (1991), no. 3, p. 187–195.

[154] T. Yabe & T. Aoki – A universal solver for hyperbolic equations by cubic-polynomial interpolation 1. one-dimensional solver, *Comp. Phys. Comm* **66** (1991), no. 2-3, p. 219 – 232.

[155] T. Yabe, T. Ishikawa, P. Y. Wang, T. Aoki, Y. Kadota & F. Ikeda – A universal solver for hyperbolic equations by cubic-polynomial interpolation 2. two- and three-dimensional solvers, *Comp. Phys. Comm* **66** (1991), no. 2-3, p. 233 – 242.

[156] T. Yabe, R. Tanaka, T. Nakamura & F. Xiao – An exactly conservative semi-lagrangian scheme (cipcsl) in one dimension, *Monthly Weather Review* **129** (2001), no. 2, p. 332–344.

[157] T. Yabe, F. Xiao & T. Utsumi – The constrained interpolation profile method for multiphase analysis, *J. Comput. Phys.* **169** (2001), no. 2, p. 556 – 593.

[158] S. Zaki, T. Boyd & L. Gardner – A finite element code for the simulation of one-dimensional vlasov plasmas. ii. applications, *J. Comput. Phys.* **79** (1988), no. 1, p. 200 – 208.

[159] S. Zaki, L. Gardner & T. Boyd – A finite element code for the simulation of one-dimensional vlasov plasmas. i. theory, *J. Comput. Phys.* **79** (1988), no. 1, p. 184 – 199.

[160] M. Zerroukat, N. Wood & A. Staniforth – A monotonic and positive-definite filter for a semi-lagrangian inherently conserving and efficient (slice) scheme, *Quarterly Journal of the Royal Meteorological Society* **131** (2005), no. 611, p. 2923–2936.

[161] ———, The parabolic spline method (psm) for conservative transport problems, *International Journal for Numerical Methods in Fluids* **51** (2006), no. 11, p. 1297–1318.

[162] M. Zerroukat, N. Wood & A. Staniforth – Application of the parabolic spline method (psm) to a multi-dimensional conservative semi-lagrangian transport scheme (slice), *J. Comput. Phys.* **225** (2007), no. 1, p. 935–948.

Virginie Grandgirard, Yanick Sarazin, CEA/DSM/IRFM, Association Euratom-CEA, Cadarache, 13108 St Paul-lez-Durance, France.
E-mail : virginie.grandgirard@cea.fr