Parallel filtering in global gyrokinetic simulations

S. Jolliet [a,*], B.F. McMillan [b], L. Villard [b], T. Vernay [b], P. Angelino [b,1], T.M. Tran [b], S. Brunner [b], A. Bottino [c], Y. Idomura [a]

[a] Japan Atomic Energy Agency, Higashi-Ueno 6-9-3, Taitou, Tokyo 110-0015, Japan
[b] Centre de Recherches en Physique des Plasmas, Association Euratom-Confédération Suisse, Ecole Polytechnique Fédérale de Lausanne, PPB, 1015 Lausanne, Switzerland
[c] Max Planck Institut fuer Plasmaphysik, IPP-EURATOM Association, Garching, Germany

ARTICLE INFO

ABSTRACT

In this work, a Fourier solver [B.F. McMillan, S. Jolliet, A. Bottino, P. Angelino, T.M. Tran, L. Villard, Comp. Phys. Commun. 181 (2010) 715] is implemented in the global Eulerian gyrokinetic code GT5D [Y. Idomura, H. Urano, N. Aiba, S. Tokuda, Nucl. Fusion 49 (2009) 065029] and in the global Particle-In-Cell code ORB5 [S. Jolliet, A. Bottino, P. Angelino, R. Hatzky, T.M. Tran, B.F. McMillan, O. Sauter, K. Appert, Y. Idomura, L. Villard, Comp. Phys. Commun. 177 (2007) 409] in order to reduce the memory of the matrix associated with the field equation. This scheme is verified with linear and nonlinear simulations of turbulence. It is demonstrated that the straight-field-line angle is the coordinate that optimizes the Fourier solver, that both linear and nonlinear turbulent states are unaffected by the parallel filtering, and that the $k_\parallel$ spectrum is independent of plasma size at fixed normalized poloidal wave number.

1. Introduction

It is now commonly accepted that plasma turbulence is responsible for the anomalous transport observed in Tokamaks [1]. The best numerical tools to study this issue are gyrokinetic codes, which consistently solve the Vlasov–Maxwell system [2]. The gyrokinetic theory assumes that the typical frequency of micro-instabilities is much smaller than the cyclotron frequency, thus reducing the number of dimensions from 6 to 5. However, when deriving gyrokinetic equations (see for e.g. [3]), other small parameters are introduced: this is called the gyrokinetic ordering. In particular, it is assumed that the parallel wavenumber $k_\parallel$ is small ($k_\parallel \rho_s \sim \mathcal{O}(\rho^*)$, $\rho^* = \rho_s/a \sim 10^{-2} - 10^{-3}$, where ρ_s is the ion sound gyroradius and a is the minor radius of the Tokamak) whereas the perpendicular wavenumber can be large ($k_\perp \rho_s \sim \mathcal{O}(1)$). This assumption is based on the theoretical argument that small parallel wavelengths are Landau damped and has been observed experimentally [4]. This strong anisotropy of plasma turbulence is the starting point of the so-called flux-tube codes [5], which solve the turbulence on a field-aligned domain. Field-aligned coordinates allow a huge reduction of computational requirements, but may unfortunately be inconvenient once used in a global code due to the magnetic shear: non-rational field lines do not close on themselves and special care must be employed to ensure the poloidal periodicity of the perturbations. Several techniques exist to avoid this problem. In [6], the shifting-metric procedure is applied to have a locally orthogonal coordinate system at each poloidal plane. Another useful technique is to use quasi-ballooning coordinates [7], where the parallel coordinate is not exactly aligned to give straightforward boundary conditions in the poloidal and toroidal directions. This technique is used

---

* Corresponding author.
  E-mail address: jolliet.sebastien@jaea.go.jp (S. Jolliet).
[1] Now at Institute of Chemical Sciences and Engineering, Ecole Polytechnique Fédérale de Lausanne, Switzerland.

by several global gyrokinetic codes [8–10]. Quasi-ballooning coordinates result in an improved scaling of CPU time ($\propto (\rho^*)^{-2}$) compared to the $(\rho^*)^{-3}$ scaling with standard unaligned coordinates, due to the low resolution needed in the parallel direction. These coordinates act as a natural filter for unwanted high frequencies and allow larger time steps. However, implementing a field-aligned solver is rather complicated due to non-rectangular grids and the treatment of the magnetic axis is generally avoided. Due to magnetic shear, the grid is distorted which may result in poor resolution for radial derivatives. This can be resolved by using an unstructured grid [34]. Unfortunately, the field equation is in that case solved with an iterative solver whose convergence is not guaranteed and depends on the physical problem. For these reasons, many of the global gyrokinetic codes [23,11–13] still solve the field equation on the poloidal plane with direct solvers. The GT3D code [23] uses a ballooning phase factor to extract analytically the $k_\parallel = 0$ structure at a given magnetic surface: the grid resolution of the field equation can be strongly reduced. However, when solving the field equation on the poloidal plane, the discretized spectrum may contain modes with $k_\parallel \rho_s \gg \rho^*$ that are unphysical. Indeed, Particle-In-Cell (PIC) simulations may be polluted by high parallel components generated by inherent statistical noise [12]. This has been resolved by applying a Fourier filter specifically designed to remove high $k_\parallel \rho_s$ modes on the perturbed density [12]. This scheme has been recently improved by applying the same filtering procedure to the potential [14] and results in massive computational savings in global Particle-In-Cell (PIC) codes: memory is decreased by two orders of magnitude, the number of Fourier modes is decreased by one order of magnitude and the number of particles required for a given accuracy is decreased by the same factor as the number of particle per Fourier modes dictates the noise level [15]. It means that in PIC codes, parallel filtering cannot be dissociated from the number of markers. On the other hand, Eulerian codes are free from such noise, and influences of the filtering can be clearly estimated by comparing filtered and non-filtered simulations at a fixed number of grid points. Therefore, this work presents the implementation of the Fourier filtering technique in the Eulerian code GT5D [11] and in the PIC code ORB5 [12].

The rest of this paper is organized as follows. Section 2 briefly presents both codes, the implementation of the Fourier solver [14], and further focuses on the choice of the poloidal angle. The solver is verified with linear and nonlinear simulations in Section 3. Then, turbulent spectra are studied in Section 4, and conclusions are given in Section 5.

2. Implementation of the Fourier solver

2.1. The GT5D code

The detailed implementation of the GT5D code can be found in Refs. [11,16]. It is briefly summarized for completeness. GT5D is a five-dimensional full-f Vlasov code that solves a gyrokinetic equation [17] in Tokamaks:

$$\frac{\partial \mathcal{J}f}{\partial t} + \nabla \cdot (\mathcal{J}\dot{\mathbf{R}}f) + \frac{\partial}{\partial v_\parallel}(\mathcal{J}\dot{v}_\parallel f) = \mathcal{J}[C(f) + S_{\mathrm{src}}(f) + S_{\mathrm{snk}}(f)] \tag{1}$$

where $f(\mathbf{R}, v_\parallel, \mu, t)$ is the ion guiding-center distribution function, $\mathbf{R}$ is the guiding-center position, $v_\parallel$ is the velocity parallel to the magnetic field, μ is the magnetic moment and $\mathcal{J}$ is the phase-space Jacobian. The nonlinear equations of motion ($\dot{\mathbf{R}}$, $\dot{v}_\parallel$) are obtained from a Hamiltonian approach:

$$H = \frac{1}{2}m_i v_\parallel^2 + \mu B + q_i \langle \phi \rangle_\alpha \tag{2}$$

$$\dot{\mathbf{R}} = v_\parallel \mathbf{b} + \frac{c}{q_i B_\parallel^*} \times \left(q_i \nabla \langle \phi \rangle_\alpha + m_i v_\parallel^2 \mathbf{b} \cdot \nabla \mathbf{b} + \mu \nabla B \right) \tag{3}$$

$$\dot{v}_\parallel = -\frac{\mathbf{B}^*}{m_i B_\parallel^*} \cdot (q_i \nabla \langle \phi \rangle_\alpha + \mu \nabla B) \tag{4}$$

where $\mathbf{B} = B\mathbf{b}$ is the magnetic field, $\mathbf{B}^* = \mathbf{B} + B v_\parallel / \Omega_i \nabla \times \mathbf{b}, B_\parallel^* = \mathbf{b} \cdot \mathbf{B}^*, \Omega_i = q_i B/(m_i c)$ is the cyclotron frequency and $\langle \cdot \rangle_\alpha = 1/(2\pi)\oint \cdot \mathrm{d}\alpha$ is the gyro-averaging operator where α is the gyro-phase angle. The equations of motion are obtained through $\dot{\mathbf{R}} \equiv \{\mathbf{R}, H\}$ and $\dot{v}_\parallel \equiv \{v_\parallel, H\}$, where $\{F, G\}$ is the Poisson bracket operator defined by:

$$\{F, G\} \equiv \frac{\Omega_i}{B}\left(\frac{\partial F}{\partial \alpha}\frac{\partial G}{\partial \mu} - \frac{\partial G}{\partial \alpha}\frac{\partial F}{\partial \mu}\right) + \frac{\mathbf{B}^*}{m_i B_\parallel^*} \cdot \left(\nabla F \frac{\partial G}{\partial v_\parallel} - \nabla G \frac{\partial F}{\partial v_\parallel}\right) - \frac{c}{q_i B_\parallel^*}\mathbf{b} \cdot (\nabla F \times \nabla G) \tag{5}$$

The collision operator $C(f)$ is a linearized, drift-kinetic Fokker–Planck operator [18] $C(f) \equiv C_T(\delta f) + C_F(f)$, where $C_T(\delta f)$ is the test-particle operator and $C_F(f)$ is the field-particle operator. In particular, the field-particle operator is constructed numerically in order to conserve density, parallel momentum and energy up to machine precision [19]. Finite Larmor Radius (FLR) effects are neglected.

The source operator is $S_{\mathrm{src}} = A_{\mathrm{src}}(\mathbf{R})\tau_{\mathrm{src}}^{-1}(f_{M1} - f_{M2})$, where A_{src} is a deposition profile, f_{M1} and f_{M2} are (shifted) Maxwellian distributions and τ_{src} is a time constant. τ_{src} is set by imposing zero particle and momentum input, but a fixed power input P_{in}:

$$0 = \int S_{\mathrm{src}}\mathrm{d}^6 Z = \int m_i v_\parallel S_{\mathrm{src}}\mathrm{d}^6 Z \tag{6}$$

$$P_{\text{in}} = \int \left(\frac{1}{2} m_i v_\parallel^2 + \mu B \right) S_{\text{src}} \mathrm{d}^6 Z \tag{7}$$

The sink operator is $S_{\text{snk}} = A_{\text{snk}}(\mathbf{R}) \tau_{\text{snk}}^{-1}(f_0 - f)$, where A_{snk} is a deposition profile, f_0 is the initial distribution and τ_{snk} is a time constant.

Self-consistency is imposed by the quasi-neutrality equation:

$$-\nabla_\perp \cdot \frac{\rho_{ti}^2}{\lambda_{Di}^2} \nabla_\perp \phi + \frac{1}{\lambda_{De}^2} \left(\phi - \langle \phi \rangle_f \right) = 4\pi e \left[\int f \delta(\mathbf{R} + \rho - \mathbf{x}) \mathrm{d}^6 Z - n_{0e} \right] \tag{8}$$

where $\mathbf{R} + \rho$ is the particle position, $\mathrm{d}^6 Z = \mathcal{J} \mathrm{d}\mathbf{R} \mathrm{d}v_\parallel \mathrm{d}\mu \mathrm{d}\alpha = m_i^2 B_\parallel^* \mathrm{d}\mathbf{R} \mathrm{d}v_\parallel \mathrm{d}\mu \mathrm{d}\alpha$ is the phase space volume, ρ_{ti} is the Larmor Radius evaluated with the thermal velocity v_{ti}, λ_{Di}, λ_{De} are the ion and electron Debye lengths, $\langle \cdot \rangle_f$ is a flux-surface-average operator and n_{0e} is the equilibrium electron density. Electrons are adiabatic. Dirichlet boundary conditions are imposed at the plasma edge while no boundary condition is imposed at the magnetic axis.

The gyrokinetic Poisson bracket operator is discretized using a 4th order non-dissipative conservative finite difference scheme [20], and the collision operator is discretized with a 6th order centered finite difference scheme, thus enabling long and accurate computation of the turbulence. The time integration is performed using the 2nd order semi-implicit Runge–Kutta method [21] and a stiff linear term involving the parallel streaming is treated implicitly based on a generalized conjugate residual method [22]. The separation of linear and nonlinear motions is written $H(R, Z, \varphi, v_\parallel, \mu) = H_0(R, Z, v_\parallel, \mu) + H_1(R, Z, \varphi, \mu)$, where R, Z, φ are cylindrical coordinates. Since μ appears as a parameter in these equations, the computational cost of the Vlasov solver scales as $(\rho^*)^{-3}$.

GT5D is heavily parallelized with MPI and openMP. A simulation uses $N = N_{\text{MPI}} N_{\text{OMP}}$ processors, where N_{OMP} is the number of processors of the openMP parallelization and N_{MPI} is the number of processors for MPI parallelization. The distribution function is parallelized over a 3D decomposition domain $N_{\text{MPI}} = N_\mu N_{pt}$, where N_μ is the number of points in the μ direction and $N_{pt} = N_{pR} N_{pZ}$ is the number of processors in the $(R - Z)$ direction. The field-solver is parallelized in the φ direction over N_{pt} processors. Within a time iteration, parallel communications appear in three different stages. Firstly, communications in the R and Z directions must be performed after the advection of the distribution function (the derivative operator may require the value of f at a grid point located in another processor). Secondly, the perturbed density in the r.h.s. of the quasi-neutrality equation includes an integral over μ, which translates into a sum over the processors in the μ direction. Also, since the distribution function is parallelized in the (R,Z) directions but the perturbed density must be parallelized in the φ direction, a parallel data transpose must be performed. Thirdly, the collision operator involves derivatives in the μ direction which would require parallel communications. By applying a parallel data transpose before computing the collision operator, the distribution function, originally parallelized with N_{pt} processors in the (R,Z) direction and with N_μ processors in the μ direction, becomes parallelized with $N_{pt} N_\mu$ processors in the (R,Z) direction. In this way, every processors contains all grid points in velocity space.

GT5D has been benchmarked [16,19] against the global PIC code GT3D [23] for collisionless ITG turbulence, and against the global PIC code FORTEC-3D [24] for neoclassical physics.

2.2. The ORB5 code

The ORB5 code [12] is a global PIC code whose gyrokinetic model is very similar to the one used in GT5D: the quasi-neutrality equation is identical and the trajectory of numerical markers is evolved according to Eqs. (2) and (3). Collisions are neglected. The distribution function f is decomposed into a Maxwellian $f_0(\hat{\psi}, \epsilon, \mu)$ and a perturbed part $\delta f(\mathbf{z}, t)$ where $\hat{\psi}$ is a constant of motion derived from the canonical angular momentum to avoid the generation of spurious zonal flows [25], ϵ is the kinetic energy per mass unit, μ is the magnetic moment per mass unit and $\mathbf{z}$ are the phase-space coordinates. The Vlasov equation is:

$$\frac{\mathrm{d}\delta f}{\mathrm{d}t} = \tau(\mathbf{E}) + S_{NC}(\mathbf{z}, t) \tag{9}$$

$$\tau(\mathbf{E}) = -\frac{\mathrm{d}f_0}{\mathrm{d}t}\bigg|_1 \tag{10}$$

$S_{NC}(\mathbf{z}, t)$ is the noise-control operator [26]. It is composed of a Krook term $-\gamma_K \delta f(\mathbf{z}, t)$ and a correction term such that $S_{NC}(\mathbf{z}, t)$ does not alter an arbitrary set of moments. In the present simulation, these moments are the density and the long time structure of zonal flows. The Krook operator introduces some numerical dissipation essential to reach a thermodynamical steady-state [27]. In addition, the noise-control operator introduces energy into the system in order to restore the temperature gradient to its initial value: the ORB5 simulations presented here are nearly-fixed-gradient simulations, in contrast to GT5D simulations which are fixed-flux. Equations of motion are integrated with a 4th-order Runge–Kutta scheme. Before solving the quasi-neutrality equation, the perturbed density is Fourier-filtered in order to remove the unphysical high parallel wavenumber components from the numerical representation of the system [12], which greatly decreases the noise of the simulation. ORB5 is also massively parallelized with MPI. The markers are parallelized in the toroidal direction, and a 2D parallelization scheme based on the domain cloning algorithm is further used [28]. The number of processors is therefore $P = P_\varphi P_C$, where P_φ is the number of processors in the toroidal direction and P_C is the number of clones. The quasi-neutrality

equation is parallelized in the toroidal direction (in Fourier space) over P_φ processors. There are three communication stages: after the particle pushing, the attributes of some markers must be communicated toroidally. Before solving the quasi-neutrality equation, the perturbed density must be summed over the clones, and finally a parallel data transpose must be performed on the perturbed density in order to compute the Fourier transforms in the toroidal direction. The parallel data transpose is usually the most costly communication operation and scales with $(\rho^*)^{-3}$. While it is negligible for standard cases $\rho^* = 1/150$, it will become dominant at ITER sizes $\rho^* \sim 1/1000$.

Finally, the filtering procedure is the starting point to define a signal-to-noise ratio [15]. It is generally observed that a signal-to-noise above 10 is required to have converged simulations.

2.3. The Fourier solver

The electrostatic potential is discretized in toroidal Fourier space with B-splines finite elements [29] on a (r, χ) grid with (N_r, N_χ) grid points, where r is a radial coordinate and χ is a poloidal angle-like coordinate.

$$\phi(r, \chi, \varphi) = \sum_{\mu,n} \phi_\mu^{(n)} \Lambda_\mu(r, \chi) e^{in\varphi} \tag{11}$$

where $\Lambda_\mu(r, \chi) = \Lambda_j(r)\Lambda_k(\chi)$ is a product of 1D B-splines, r is the radial coordinate and χ is a poloidal coordinate which will be discussed later. Applying the Galerkin method gives a linear system for each n:

$$\sum_\mu M_{\mu\nu}^{(n)} \phi_\mu^{(n)} = g_\nu^{(n)} \tag{12}$$

with:

$$M_{\mu\nu}^{(n)} = \int d^3x \left[\frac{\rho_{ti}^2}{\lambda_{Di}^2} \nabla_{pol}\Lambda_\mu \cdot \nabla_{pol}\Lambda_\nu + \frac{1}{\lambda_{De}^2}\left(\Lambda_\mu\Lambda_\nu - \delta_{n,0}\langle\Lambda_\mu\rangle_f \Lambda_\nu \right) \right] \tag{13}$$

$$g_\nu^{(n)} = \int d^6Z G_\nu(r, \chi, \mu)\delta f^{(n)} \tag{14}$$

$$G_\nu(r, \chi, \mu) = \frac{1}{2\pi} \int d\alpha \Lambda_\nu[\mathbf{R}(r, \chi) + \rho(\mu, \alpha)] \tag{15}$$

where $\delta f^{(n)}$ is the Discrete Fourier Transform (DFT) of $\delta f = f - f_0$ in the toroidal direction, $\nabla_{pol} = \frac{\partial}{\partial r}\nabla r + \frac{\partial}{\partial \chi}\nabla \chi$ is the gradient in the poloidal plane, $d^3x = J_{r\chi\varphi}drd\theta d\varphi$, $J_{r\chi\varphi}$ is the configuration space Jacobian, and $\langle\cdot\rangle_f$ is the flux-surface average operator:

$$\langle A\rangle_f = \frac{\int d\chi d\varphi J_{r\chi\varphi}(r, \chi)A(r, \chi, \varphi)}{\int d\chi d\varphi J_{r\chi\varphi}(r, \chi)} \tag{16}$$

The matrix is a symmetric band matrix and is solved with LAPACK [30]. Note that any radial or poloidal coordinate can be used: changes are reflected in the spatial Jacobian appearing in the integrals of Eqs. (13) and (14), as well as in ∇_{pol}. The distribution function has N_φ points in the toroidal direction. In order to avoid aliasing effects, toroidal modes with $|n| > N_\varphi/4$ are filtered out. A $n = 0$ matrix and a $n \neq 0$ matrix must be stored for a total number of $2(p + 1)(N_r + p)N_\chi^2 \sim (\rho^*)^{-3}$ complex elements where p is the splines order. For a standard Cyclone [31] simulation of size $\rho^* = 1/150$, $N_r = 128$, $N_\chi = 256$ with quadratic splines, 800 Mb are needed to store the two matrices. ITER plasmas will have typically $\rho^* \sim 1/1000$ which would correspond to a total size of 240 Gb and would be impossible to store on a single node. It is noted that in principle, such a huge operator can be solved via parallel iterative approaches. However, the convergence of such a solver may not be always assured. Therefore, a direct solver with smaller operator sizes is desirable for turbulent simulations, where the robustness of the field-solver is extremely important. Furthermore, solving the field equation on the poloidal plane does not restrict the ratio $k_\parallel/k_\perp$. At mid-radius, the latter can be as high as 0.25, which is inconsistent with the gyrokinetic ordering [32]. From a theoretical point of view, these modes may be safely eliminated. Consequently, the Fourier solver described in Ref. [14] and first applied to the global PIC code ORB5 has been implemented in GT5D as well. The implementation is therefore identical. It is summarized for the sake of completeness. First, the linear system is written in poloidal Fourier space:

$$\sum_\mu \underbrace{FM_{\mu\nu}^{(n)}F^{-1}}_{\equiv \widetilde{M}^{(n)}} \underbrace{F\phi_\mu^{(n)}}_{\equiv \tilde{\phi}_\mu^{(n)}} = \underbrace{Fg_\nu^{(n)}}_{\equiv \tilde{g}_\nu^{(n)}} \tag{17}$$

where F is the combination of the DFT and the filtering operators: by invoking the gyrokinetic ordering $k_\parallel/k_\perp \sim \rho^*$, the quasi-neutrality equation is solved in the subspace of field-aligned modes, characterized by $|m - nq(r)| < \Delta m$, where $q(r)$ is the safety factor profile. Note that this filtering procedure is surface-dependent. Each toroidal mode has a different matrix, but the matrix storage is parallelized in the toroidal direction with P_φ processors, giving a size of $3(N_r + 2)(2\Delta m + 1)^2 N_\varphi/(4P_\varphi) \sim (\rho^*)^{-1}$ as Δm is independent of plasma size if one chooses the straight-field-line angle for poloidal coordinate (see Sections 2.4 and 4), and $N_\varphi/(4P_\varphi) \sim \mathcal{O}(1)$. For $\Delta m = 20$ and $N_\varphi/(4P_\varphi) = 1$, the size becomes 10 Mb for a Cyclone simulation and 80 Mb for an ITER simulation.

There are only a few minor differences between the two implementations. B-splines are polynomials of order p. GT5D uses quadratic splines ($p = 2$) and r as a radial coordinate, while ORB5 uses cubic splines ($p = 3$) and $s = \sqrt{\psi/\psi_{\text{edge}}}$, where ψ is the poloidal flux. Moreover, ORB5 stores N_φ/P_φ matrices per processor. These differences have no impact on the results.

2.4. Choice of Δm

There are a few constraints on the value of Δm. First, $\Delta m \gtrsim 2\hat{s}$, where $\hat{s}$ is the magnetic shear, must hold to avoid discontinuities across magnetic surfaces [12]. The strongest constraint on Δm is in fact linked with the choice of the poloidal coordinate χ. In this work, two different poloidal coordinates will be considered: $\chi = \theta$, where θ is the usual poloidal angle, and $\chi = \theta_*$, where θ_* is the straight-field-line coordinate, defined by:

$$\theta_* = \frac{1}{q(\psi)} \int_0^\theta d\theta' \frac{\mathbf{B} \cdot \nabla \varphi}{\mathbf{B} \cdot \nabla \theta'} = 2 \tan^{-1} \left(\sqrt{\frac{1-\epsilon}{1+\epsilon}} \tan(\theta/2) \right) \tag{18}$$

where ψ is the poloidal flux, $\epsilon = r/R_0$ is the local aspect ratio and the last equality has been obtained by assuming a circular equilibrium. The relation $k_\parallel = (m - nq)/(qR)$ holds only if the straight-field-line coordinate θ_* is used. θ_* is such that $\mathbf{B} \cdot \nabla \varphi = q(\psi) \mathbf{B} \cdot \nabla \theta_*$. In order to understand the link between Δm and the choice of the poloidal angle, consider a single toroidal perturbation propagating along a field line at a given surface r_0:

$$\phi(r_0, \theta_*, \varphi) = \hat{\phi}(r_0) e^{in_0(\varphi - q\theta_*)} \tag{19}$$

In the large aspect ratio approximation with elliptic surfaces, $\theta_* = \theta - \epsilon\kappa \sin\theta + \mathcal{O}(\epsilon^2)$ where κ is the plasma elongation and so:

$$\phi(r_0, \theta, \varphi) = \tilde{\phi}_0(r_0, \theta, \varphi) \sum_{\delta m = -\infty}^{+\infty} J_{\delta m}(x) e^{i\delta m\theta} \tag{20}$$

$$x = -\epsilon n_0 \kappa q \tag{21}$$

where $\tilde{\phi}_0(r_0, \theta, \varphi) = \hat{\phi}(r_0) e^{in_0(\varphi - q\theta)}$, $J_n(x)$ is the Bessel function of the first kind and the relation $e^{ix\sin\theta} = \sum_{n=-\infty}^{+\infty} J_n(x) e^{i\theta}$ has been used. Therefore, one sees from Eqs. (19) and (20) that while the perturbation has one single harmonic $m_0 = n_0 q$ in the θ_* space, its spectrum is broadened in the θ space. The broadening can be seen on Fig. 1, which plots $J_{\delta m}(x)$ as a function of δm for different values of x. As $\lim_{x \to \infty} J_n(x) = \sqrt{2/(\pi x)} \cos(x - n\pi/2 - \pi/4)$, the broadening monotonically increases with the argument of the Bessel functions. Physically, it means that the broadening will increase when the aspect ratio is lowered, when the safety factor and the elongation are increased or when n_0 is increased. The latter happens when $k_\theta \rho_{ti}$ is increased at fixed plasma size, or when the plasma size is increased at fixed $k_\theta \rho_{ti}$. Fig. 1 may be used to determine a lower bound for Δm when θ is used. The present estimation assumes for simplicity a single harmonic perturbation in θ_* space. In gyrokinetic simulations, the global spectra are typically wider. Fig. 2 shows the poloidal local field energy spectrum (defined in Section A) of a linear ORB5 simulation for a Cyclone-based MHD equilibrium with $\kappa = 1.5$, corresponding to the bottom plot of Fig. 1. The spectrum in θ_* space is very narrow, and a value of $\Delta m = 3$ would seem sufficient to capture all relevant modes. On the contrary, the spectrum in θ space is very wide and a value of $\Delta m = 25$ is needed. The estimate from the bottom graph of Fig. 1
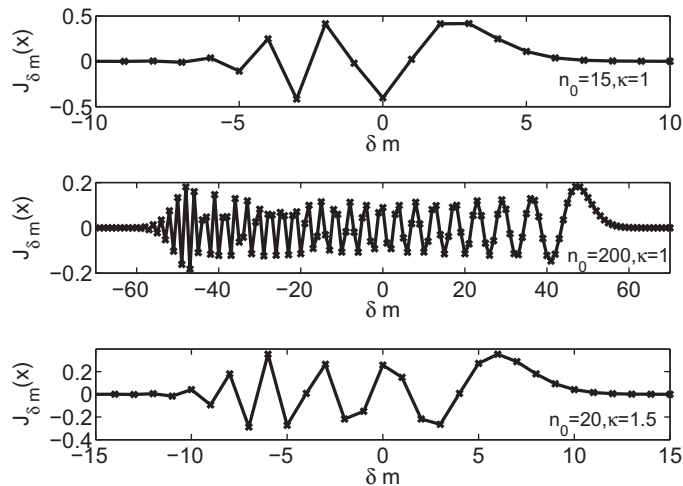

Fig. 1. Broadening amplitude $J_{\delta m}(x = \epsilon\kappa n_0 q)$ as a function of δm for $\epsilon = 0.18$, $q = 1.4$ and different values of n_0 and κ.
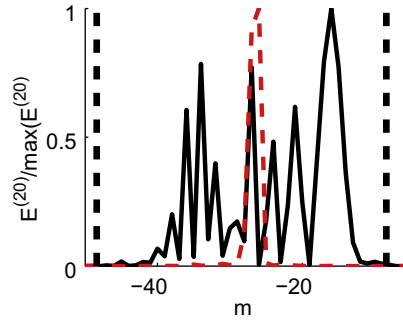
Fig. 2. Poloidal spectrum in θ (black,solid) and in θ_* (red,dashes) of local field energy for a $n_0 = 20$ mode in a MHD equilibrium with $\kappa = 1.5$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

would give a value of $\Delta m \cong 12$. The middle graph of Fig. 1 is an estimate of a $k_\theta \rho_{ti} = 0.5$ ITG mode in an ITER plasma, yielding a minimum value of $\Delta m = 60$. The finite element matrix would at least be one order of magnitude larger if the usual poloidal angle is used. This simple model shows that global codes not using a field-aligned coordinate will need the straight-field-line coordinate instead to simulate large or highly-shaped plasmas. Note also that flux-tube codes often use the "$s - \alpha$" model, in which θ_* is approximated by θ. Neglecting this finite aspect ratio effects strongly affects heat transport [33].

3. Verification

3.1. CPU cost

As mentioned in Section 2.1, the computational cost of the Vlasov part of the GT5D code scales as $(\rho^*)^{-3}$. Using the Fourier solver drastically reduces the memory and also affects the CPU time. Using the Fourier solver in GT5D allows a 3% reduction of the CPU time for standard Cyclone case at $\rho^* = 1/150$. Estimates show that this gain is approximatively similar at $\rho^* = 1/300$. The original field-solver part is dominated by LAPACK operations and communications in the μ direction, which scale as $(\rho^*)^{-3}$. When the Fourier solver is used, these parts theoretically scale as $(\rho^*)^{-1}$ and become subdominant, therefore the $(R - Z) \rightarrow (r - \theta)$ transpose, proportional to $(\rho^*)^{-3}$ will dominate, but tests have shown that this does not happen up to $\rho^* = 1/300$, which was the largest size accessible on the JAEA Altix3700Bx2 system. In practice, the communication scaling can be different depending on the number of processors and the parallel architecture. At $\rho^* = 1/300$, the Fourier solver part including the charge assignment is 3% of the total simulation time.

In the ORB5 code, the number of particle scales as $(\rho^*)^{-2}$ and the solver CPU time, dominated by the parallel data transpose needed to perform the toroidal Fourier transform, scales as $(\rho^*)^{-3}$, but fortunately the latter dominates the former from $\rho^* \cong 1/1000$ only [14]. Note that the original LAPACK solver was consuming 40% of the total CPU time for standard simulations [14]. Consequently, in terms of CPU usage the ORB5 code with the Fourier solver is equivalent to a field-aligned code up to large plasma sizes, with the advantage that the solver is direct and consequently very robust.

Although a direct comparison is beyond the scope of this paper, it is worth commenting other approaches. Quasi-ballooning and unstructured grids resolution N scales as $(\rho^*)^{-2}$. However, simple Jacobi preconditioning leads to a $N^{3/2} \sim (\rho^*)^{-3}$ scaling, which can be improved with multigrid techniques, see Ref. [35]. The parallelized field-aligned solver developed in this work performs well (solve time of the order of a few seconds for a ITER-like plasma).

3.2. Linear and neoclassical validation

The Fourier solver is first verified with linear simulations, solving only one toroidal mode. In ORB5, linear simulations are performed by neglecting the electric field in the equations of motions (2) and (3): in absence of a saturation mechanism, the perturbation grows exponentially and indefinitely. In GT5D, linear simulations must be understood as the linear phase of single n mode nonlinear simulations. Choosing a very small initial perturbation allows a long enough linear phase to extract the growth rate before the nonlinear saturation. Linear results are presented for the GT5D code only, but are also valid for ORB5. The Fourier solver introduces a new parameter Δm with respect to which the simulation must be converged. The unfiltered case (still solved in poloidal Fourier space) will be noted $\Delta m = \infty$. Compared to the simulation in real space, the results are identical up to machine precision. Then, linear Cyclone simulations have been performed (see [16] for the physical and numerical parameters) by using either θ or θ_* as the poloidal coordinate and by varying Δm. Instantaneous growth rates for $\chi = \theta$ and $\chi = \theta_*$ are displayed on Fig. 3. As predicted in the previous section, the converged value of Δm is larger in the θ case compared to the θ_* case. In the latter situation, $\Delta m = 3$ is sufficient to converge the linear growth rate. These results are in agreement with those of [12], where the Fourier filter is applied on the density only. This is indeed not surprising: when applied on the potential, the Fourier filter suppresses the couplings producing and produced by poloidal
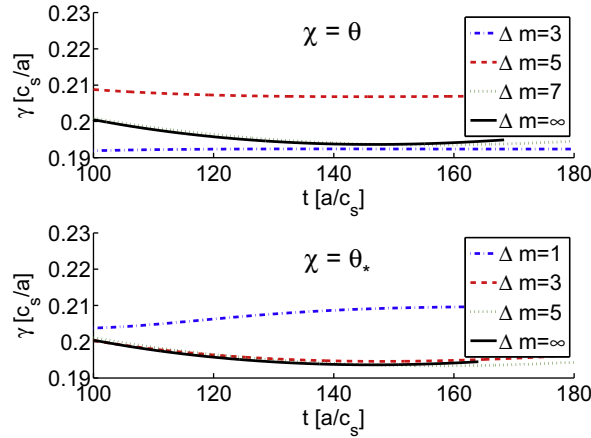
Fig. 3. Instantaneous growth rates as a function of time for $\chi = \theta$ (top) and $\chi = \theta_*$ (bottom), for different values of Δm.

modes $m \gtrless nq(r) \pm \Delta m$. In the linear regime, these couplings are triggered by the poloidal dependence of the magnetic field, which is of the order of the aspect ratio and are expected to be unimportant for modes at the filter boundaries. $\Delta m = 3$ is also sufficient to converge the potential structure on the poloidal plane, as shown by Fig. 4.

The matrix of the $n = 0$ mode differs due to the flux-surface-average operator in Eq. (13), and the $n = 0$ mode is linearly stable. In order to check its correct implementation, neoclassical simulations (axisymmetric limit, no sources) have been performed based on parameters from [11]: the system is initialized with a local Maxwellian distribution which is not a solution of the Poisson bracket operator: a radial electric field is produced to satisfy the ambipolarity condition. Both filtered and unfiltered simulations are identical, showing relative differences of 10^{-5} for all fluxes. The poloidal spectra of the field energy, Fig. 5 are in good agreement. In particular, in the unfiltered case, the $|m| > \Delta m$ components are 10^{-6} times smaller than the dominant $m = 0$ value. Same differences are observed in Rosenbluth-Hinton test [36] with the ORB5 code.

3.3. Nonlinear verification

In the previous section, it has been shown that the Fourier filtering technique does not affect the linear stage of the simulations. The nonlinear picture is more complex. Once ITG turbulence develops, toroidal and poloidal nonlinear couplings become important. Introducing a parallel filter might affect the turbulent transport, since radial fluxes are produced by non-parallel components of the $\mathbf{E} \times \mathbf{B}$ velocity: the convergence with respect to Δm must be checked in the steady-state phase of the simulation. Note that a similar study has been performed for sourceless PIC simulations in [12] when the filter is applied on the density only, and it was found that $\Delta m = 9$ was enough to converge the heat flux. However, these simulations had many drawbacks. Firstly, the plasma size of $\rho^* = 1/40$ was unrealistically small and boundary effects were important. Secondly, the gyrokinetic model did not have any physical or numerical dissipation. Consequently, the final state is quasi-steady and dominated by noise [27]. Finally, there were no sources to maintain the temperature gradient and the final state did not show bursty transport. Now that these problems have been overcome, a nonlinear Δm scan is necessary. It is a tedious task for decaying simulations (when the temperature gradient is allowed to relax): 320 M particles are needed to converge a Cyclone simulation at $\Delta m = 5$ [37], when a noise-control algorithm is used [26]. The situation becomes easier for nearly-fixed-gradient simulations, as the signal-to-noise ratio can be maintained high enough with relatively few markers (80 M for the previous example). It becomes even easier with GT5D, since the Eulerian scheme allows the separation of grid convergence and parallel filtering. It enables a more stringent verification of the latter issue. The aim of this section is to test the convergence of nonlinear simulations with respect to the parallel filter parameter Δm for both PIC and Eulerian
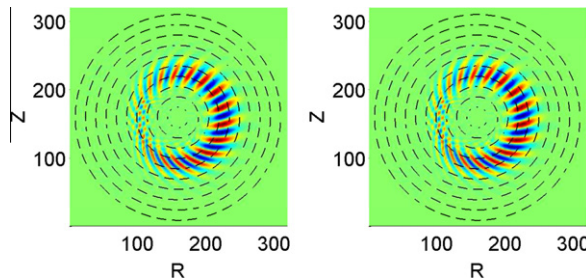

Fig. 4. Poloidal cross section of the potential for the real space simulation (left) and the $\Delta m = 3, \theta_*$ simulation (right). Axis are in ρ_s units.
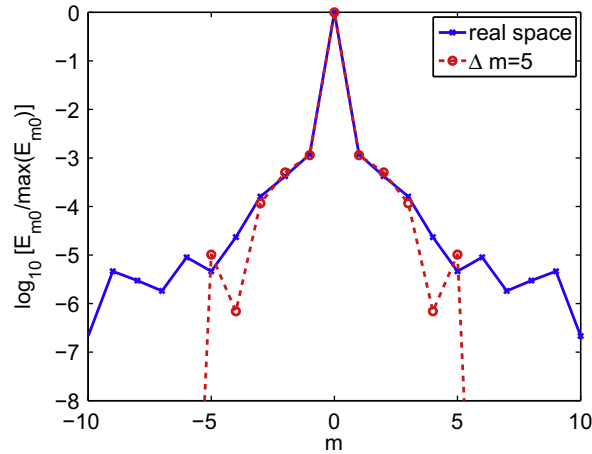
Fig. 5. Poloidal spectrum of the $n = 0$ mode global field energy for an unfiltered and a $\Delta m = 5$, θ_* neoclassical simulation, normalized to its maximum value. The spectra are averaged over the quasi-steady-state phase of the simulations.

methods. It is by no means a benchmarking attempt: fixed-flux and nearly-fixed-gradient simulations will lead to different transport levels.

3.3.1. ORB5 simulations

For numerical reasons it is impossible to perform an unfiltered ORB5 simulation. Consequently, three simulations at $\Delta m = 5, 10$ and 20 have been performed. The number of markers is $80M, 160M$ and $320M$ such that the signal to noise ratio is constant. The initial temperature gradient is $R/L_{Ti} = 7.2$, $\rho^* = 1/155$ and a $1/2$ wedge torus (solving toroidal modes $0, \pm 2, \pm 4, \ldots$) is used. $\Delta t = 51\Omega_i^{-1}$ and the finite element grid's resolution is $(N_s, N_{\theta_*}) = (128, 512)$. The noise-control decay rate is $\gamma_K = 7.8 \cdot 10^{-5}\Omega_i$ which is approximatively 7% of the maximum linear growth rate. The simulations start with a linear phase. Immediately after the saturation, large heat bursts are produced in the so-called transient phase. Finally, a quasi-steady-state establishes, with intermittent outward and inward propagating avalanches. The avalanches dynamics is independent of Δm. In order to assess the influence of this parameter on the system, one must quantify the radial heat transport. This is a difficult task as bursts are chaotic: the turbulence has a so-called intrinsic variability. This phenomenon has been studied in [26,38] for ORB5 simulations: identical simulations differing only by the initialization are performed, and are compared by applying a moving time-average of width Δt_w. A standard deviation over the ensemble of simulations of about 15% is found for $\Delta t_w = 500a/c_s$. In order to measure the influence of Δm on the simulations, the moving time-average procedure is applied, with a starting time $t_0 = 500a/c_s$ and $\Delta t_w = 300a/c_s$. Fig. 6 shows the moving time-average of χ_i/χ_{GB} and R/L_{Ti} for the
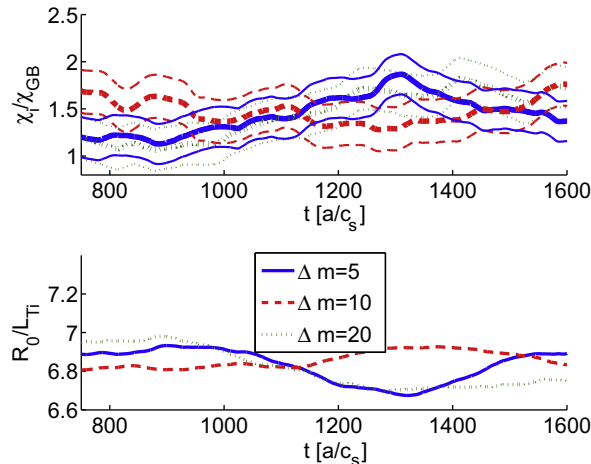

Fig. 6. Temporal evolution of χ_i/χ_{GB} (top) and R/L_{Ti} (bottom) for $\Delta m = 5$ (blue, solid), $\Delta m = 10$ (red, dashed) and $\Delta m = 20$ (green, dotted) ORB5 simulations. For each simulation on the top graph, the thick line is the moving time-average and thin lines represent the intrinsic variability, estimated to be 15% of the space and time-average of χ_i/χ_{GB}. Quantities are averaged between $r/a = 0.35$ and $r/a = 0.65$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

three simulations, where $\chi_i = -Q_i/(n_i\nabla T_i)$, Q_i is the ion heat flux, n_i is the ion density and $\chi_{GB} = \rho_s^2 c_s/a$ is the Gyrobohm normalization. For χ_{i}/χ_{GB}, the intrinsic variability (thin lines), estimated to be 15% of the radial and time-averaged value of χ_{i}/χ_{GB}, is added and can be viewed as the "error bar" for each simulation. The three enclosed regions overlap almost all the time. Then, the values of R/L_{Ti} differ at most by 0.2. This is sufficient to say that the parallel filtering does not influence ORB5 simulations, and that $\Delta m = 5$ can be employed.

3.3.2. GT5D simulations

One unfiltered ($\Delta m = \infty$) and two filtered ($\Delta m = 20$ and $\Delta m = 5$) Cyclone GT5D simulations have been performed. $\rho^* = 1/150$ and 1/3 wedge torus (solving toroidal modes $0, \pm3, \pm6, \ldots$) is simulated. The initial temperature gradient is $R/L_{Ti} = 10$. The numerical parameters are $\Delta t = 5\Omega_i^{-1}$, $(N_R, N_Z, N_\varphi, N_{v_\parallel}, N_\mu) = (160, 160, 64, 80, 20)$ and $(N_r, N_\chi) = (128, 256)$ for the finite element grid. Only 16 toroidal modes are kept in the system and the others are filtered out. This ensures a minimum resolution of 4 points per wavelength to avoid aliasing effects. The input heating power is $P_{in} = 2MW$. The heat deposition profile extends between $r/a = 0$ and $r/a = 0.5$, while the sink deposition profile extends from $r/a = 0.8$ to $r/a = 1.0$ with $\tau_{snk}^{-1} = 0.1\, v_{ti}/a$ (see [11]). The numerical scheme is a 4th order centered difference scheme and is therefore non dissipative. The L2-norm conservation inherent to this scheme prevents any growth of spurious modes that would be caused by numerical dispersion. Physical collisions prevent filamentation in velocity space.

The three simulations show similar particle number and energy conservation properties. The spatio-temporal evolution of χ_{i}/χ_{GB} is shown on the top graph of Fig. 7. Here, simulations with different initial conditions could not be performed due to the huge cost of a simulation (250'000 CPU hours on the JAEA Altix3700Bx2 system), but it can be anticipated that the
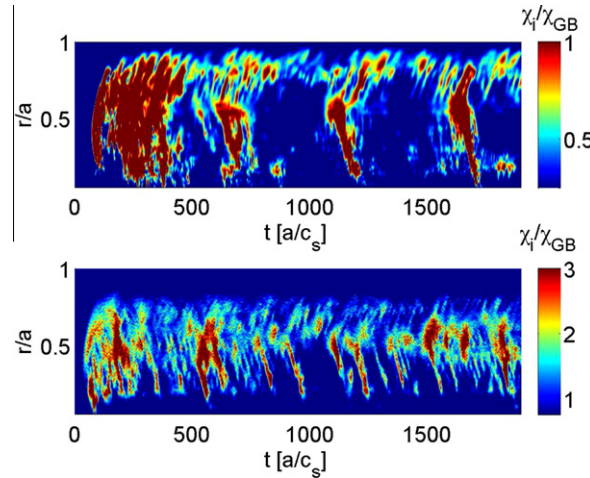

Fig. 7. Spatio-temporal evolution of χ_{i}/χ_{GB} for the $\Delta m = \infty$ GT5D (top) and $\Delta m = 5$ ORB5 (bottom) simulations. The minimum (resp. maximum) of the color bar is chosen as one half (resp. two times) the average on the radial regions of interest.
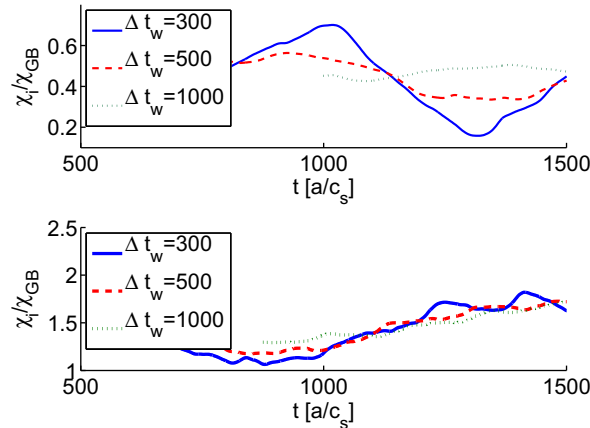

Fig. 8. Moving time-average of χ_{i}/χ_{GB} for the $\Delta m = 20$ GT5D (top) and $\Delta m = 5$ ORB5 (bottom) simulations for $\Delta t_w = 300a/c_s$ (solid line, blue), $\Delta t_w = 500a/c_s$ (dashed line, red) and $\Delta t_w = 1000a/c_s$ (dotted, green). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
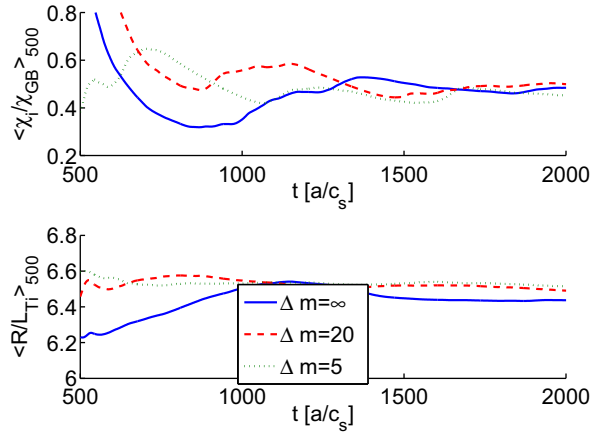
Fig. 9. Temporal evolution of $\langle\chi_i/\chi_{GB}\rangle_{500}(t)$ (top) and $\langle R/L_{Ti}\rangle_{500}(t)$ (bottom) for $\Delta m = \infty$ (blue, solid), $\Delta m = 20$ (red, dashed) and $\Delta m = 5$ (green, dotted) GT5D simulations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
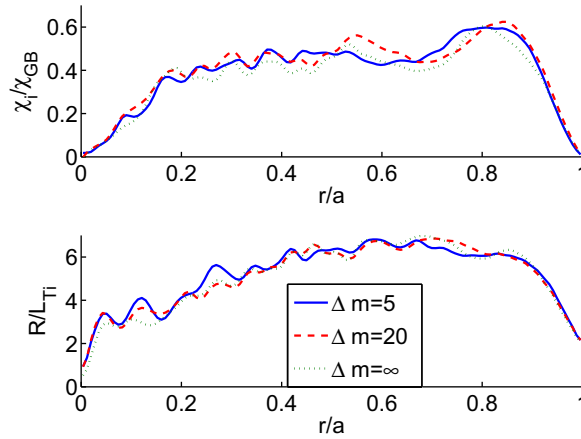

Fig. 10. Radial profile of χ_i/χ_{GB} (top) and R/L_{Ti} (bottom) for $\Delta m = \infty$ (blue, solid), $\Delta m = 20$ (red, dashed) and $\Delta m = 5$ (green, dotted) GT5D simulations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

intrinsic variability of GT5D should be equal or higher than the one of ORB5 since GT5D operates much closer to the critical gradient ($R/L_{Ti,GT5D} \cong 6.5, R/L_{Ti,ORB5} \cong 7.2$ and $R/L_{Ti,crit} = 6.0$). For that reason, the bursts in GT5D are less frequent but more important in relative value (see Fig. 7). Consequently, the moving time-average procedure must be adapted in that situation. This is further confirmed on Fig. 8, where moving time-averages with different Δt_w are shown. For ORB5, $\Delta t_w = 300a/c_s$ seems enough to reproduce the quantitative transport, but for GT5D $\Delta t > 500a/c_s$ is required. However, in fixed-flux simulations the turbulent transport and the temperature profile, constrained by a power balance relation, should reach a steady-state (in the average sense). The turbulent diffusivity is affected by the radial electric field, which, due to collisions, is constrained by the force balance relation (this is not necessarily the case in ORB5). A measure of the steadiness of the simulation can be obtained by defining the operator $\langle A\rangle_{t0}(t)$ as being the average of a physical quantity A between the times t_0 and t. Fig. 9 shows the result for $A = \chi_i/\chi_{GB}$ and $A = R/L_{Ti}$ with $t_0 = 500a/c_s$. A radial average is applied in the source free region $0.5 < r/a < 0.8$. At the end of the time window, $t = 2000$, the relative difference between the values of χ_i/χ_{GB} is 10%, and R/L_{Ti} values differ by 0.1. Then, Fig. 10 shows the radial profiles of χ_i/χ_{GB} and R/L_{Ti} averaged between 500 and $2000a/c_s$, where one sees that changing Δm does not modify the global shape and the local variations of these profiles. These results demonstrate that parallel filtering does not influence the steady-state of GT5D simulations.

4. Convergence of nonlinear spectra

In pure ITG turbulence, the adiabaticity of electrons implies that the entire linear spectrum lies in the $0 < k_\theta\rho_{ti} < 1$ range. It peaks at around $k_\theta\rho_{ti} \sim 0.35$ and then decreases due to FLR effects. Physically converged nonlinear simulations require that the relevant nonlinear spectrum stays within this range. This is fortunately the case as nonlinear couplings trigger an energy

cascade towards longer wavelengths [39]. In the previous section, it was demonstrated that the filtering procedure did not affect the fluxes. It must be checked that the same holds for the turbulent spectra.

4.1. GT5D spectra

Fig. 11 shows, for the $\Delta m = 20$ simulation, the time evolution of the local ($r_0/a = 0.7$) poloidal spectrum of the $n = 12$ mode, which is the most dominant mode in the quasi-steady-state phase. At the end of the linear phase ($t \cong 80$), large couplings are seen. However, at that time the contribution of the $n = 12$ mode on the global field energy of the system is negligible because the system is initialized with a single toroidal mode $n \neq 12$ which dominates the other toroidal modes by several orders of magnitudes. On this surface, $q(r_0) = 2$ and the mode is peaked around $m = 24$. The relative mode amplitude hardly exceeds 10% outside $|\Delta m| < 5$. Consequently, $\Delta m = 20$ seems large enough to capture the avalanche dynamics on the $n = 12$ mode. However, Δm might be a function of n, or equivalently of $k_\theta \rho_{ti}$. In fact, from a slab toroidal ITG dispersion relation, one can derive the maximum $k_\parallel$ (or Δm) to have an instability [40]:

$$\Delta m < \frac{R_0}{L_{Ti}} \frac{1/2 k_\theta \rho_{ti} \Lambda_0(\xi_i)}{\sqrt{1 + \tau^{-1}} \sqrt{\tau^{-1} + 1 - \Lambda_0(\xi_i)}} \times \sqrt{1 - \frac{2L_{Ti}}{L_{ni}} - 2\xi_i \frac{\Lambda_1(\xi_i) - \Lambda_0(\xi_i)}{\Lambda_0(\xi_i)}} \tag{22}$$

where $\xi_i = (k_\theta \rho_{ti})^2$, $\Lambda_n(x) = e^{-x} I_n(x)$, $I_n(x)$ is the nth order modified Bessel function of the first kind, $\tau = T_e/T_i$ and L_{ni} is the characteristic length of density. The r.h.s. of this equation is an increasing function with ξ_i: although it is local and contains many assumptions, it predicts an increase of Δm with the toroidal mode number. This has been verified by plotting the local poloidal field energy spectrum (averaged in the quasi-steady-state phase) for $n = 12$, $n = 24$ and $n = 36$, corresponding to $k_\theta \rho_{ti} = 0.235$, $k_\theta \rho_{ti} = 0.47$ and $k_\theta \rho_{ti} = 0.94$, respectively (see top graph of Fig. 12). The $k_\theta \rho_{ti} = 0.47$ mode has a slightly broader spectrum, confirming the qualitative picture given by the linear dispersion relation. The $k_\theta \rho_{ti} = 0.94$ spectrum is significantly broader. The broadening may be enhanced by finite size of splines for high m numbers. High $k_\theta \rho_{ti}$ modes are 1–2 orders of magnitude smaller than the dominant low $k_\theta \rho_{ti}$ modes. This is confirmed by Fig. 13, where the time-averaged global field energy is plotted against the toroidal wave number. The $k_\theta \rho_{ti}$ dependency of Δm has been further checked for a larger plasma size $\rho^* = 1/230$ filtered simulation: N_R, N_Z, N_φ, N_r and N_χ have been multiplied by 1.5. The width of each poloidal spectra for same values of $k_\theta \rho_{ti}$ is identical to the $\rho^* = 1/150$ value: the parallel filtering is independent of plasma size.

Fig. 13 shows that the field energy of toroidal modes decays similarly for all Δm values. For the latter case, the small value of Δm alters the broad spectra of high $k_\theta \rho_{ti}$ modes (see Fig. 13) and thus slightly modifies the decay exponent. Spectra averages over the heating or source-free regions decay with the same exponent, suggesting that micro-scale turbulent cascades are not affected by the differences in macro-scale conditions such as heating and geometry (local q and ϵ). On the other hand, in the outer region where a sink is located, toroidal modes have high $k_\theta \rho_{ti}$ and the broadening observed on Fig. 12 accentuates, which leads to a stronger decay in the $\Delta m = 5$ case. However, only a small fraction of the field energy is being filtered when a narrow filter is used. Therefore, although the $\Delta m = 5$ simulation has a slightly altered energy spectrum, final heat flux and temperature gradient are converged in the statistical sense.

4.2. ORB5 spectra

The time evolution of the local ($r_0/a = 0.5$) poloidal spectrum of the $n = 12$ mode is displayed on the right plot of Fig. 11 for the $\Delta m = 20$ simulation. Like in the GT5D case, the quasi-totality of the spectrum is contained in $|\Delta m| < 5$ at all times. The right plot of Fig. 13 shows the global toroidal energy spectrum as a function of n, time-averaged in the nonlinear phase of the
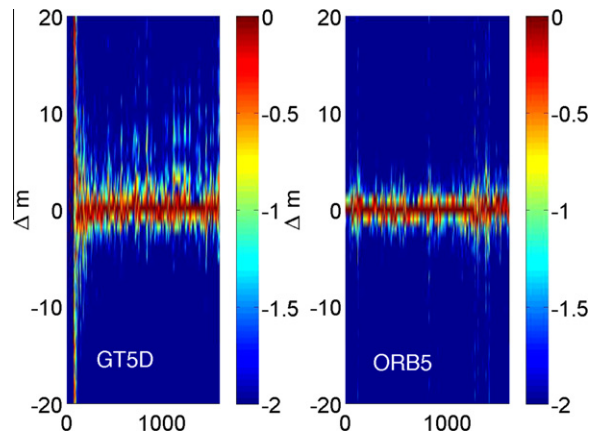

Fig. 11. Local poloidal spectrum of $n = 12$ versus time, in logarithmic scale, for $\Delta m = 20$ GT5D (left) and ORB5 (right) simulations. At each time, the spectrum is normalized to its maximum value.
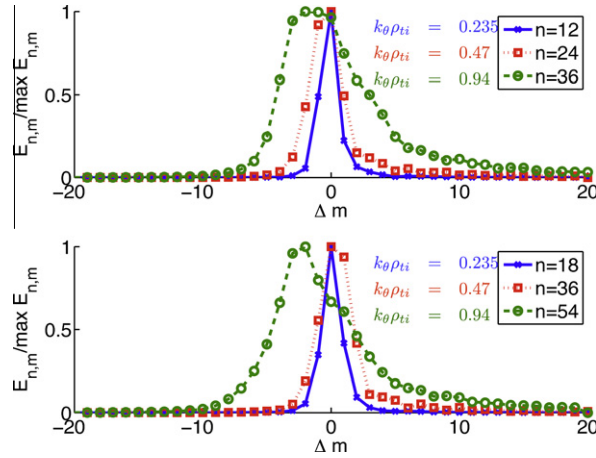
Fig. 12. Local (at $r/a = 0.7$) poloidal field energy spectrum for $k_\theta\rho_{ti} = 0.235$, $k_\theta\rho_{ti} = 0.47$ and $k_\theta\rho_{ti} = 0.94$ modes for $\Delta m = 20$ filtered simulations at $\rho_* = 1/150$ (top) and $\rho_* = 1/230$ (bottom). All spectra are time-averaged in the quasi-steady-state phase of the simulations, and then normalized to their maximum value.
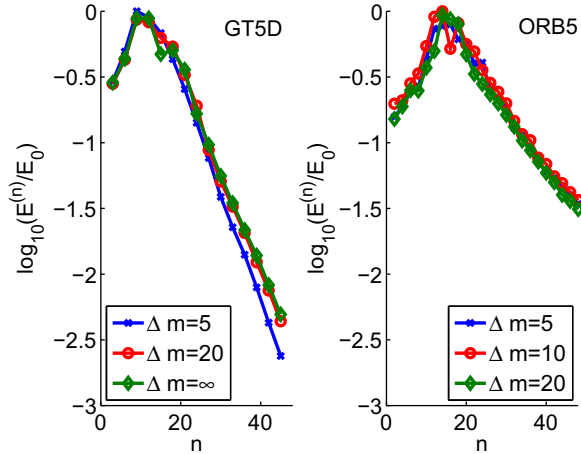

Fig. 13. Global field energy versus toroidal modes for GT5D (left) and ORB5 (right) simulations. For each code, a common normalization E_0 is chosen, and all the data are averaged between 500 and $1900a/c_s$.

simulation. The decay is again similar for each value of Δm, showing that $\Delta m = 5$ is large enough to converge the turbulent spectra. However, the $\Delta m = 5$ does not show a stronger decay towards high $k_\theta\rho_{Li}$ as was observed in GT5D simulations. This may be due to the different treatment of the toroidal angle: it is treated with Fourier transforms in GT5D and with splines in ORB5. Therefore, for high n numbers, splines may introduce an additional smoothing thus modifying the turbulent spectrum.

5. Conclusions

In this paper, the implementation of a Fourier solver described in [14] has been successfully applied to the global gyrokinetic codes GT5D and ORB5. This scheme drastically reduces the memory needed to solve the matrix of the field equation, turning the scaling from $(\rho^*)^{-3}$ to $(\rho^*)^{-1}$. It is shown that the use of the straight-field-line angle optimizes the memory reduction, and will be mandatory for ITER-size plasmas. Linearly, $\Delta m = 3$ is sufficient to get a converged simulation. Then, for the first time, the convergence of nonlinear global heated simulations with respect to the parallel wave number is demonstrated, for both the Eulerian and the PIC numerical scheme. The required width Δm slightly increases with the normalized poloidal wave number $k_\theta\rho_{ti}$ but is independent of plasma size. In ITG turbulence where only $k_\theta\rho_{ti} \ll 1$ are relevant, keeping $2\Delta m + 1 = 11$ poloidal modes per toroidal mode and per magnetic surface is sufficient to converge the ion heat diffusivity and the turbulent spectra. This number is especially important in PIC simulations because it is proportional to the needed number of markers. In addition to the different numerical schemes, GT5D and ORB5 use different source models. Results show that avalanche-like ITG transport (produced by non-zero parallel components of the electrical field) is not affected

by parallel filtering for both fixed-power and nearly-fixed-gradient simulations. The Fourier solver is therefore a robust alternative to true field-aligned solvers.

Acknowledgments

Appendix A. Definition of field energy

The plasma perturbed field energy is defined by:

$$E = \frac{q_i}{2} \int d^3x \phi(\mathbf{x}) \delta n(\mathbf{x}) \tag{A.1}$$

where $\delta n(\mathbf{x}) = \int f \delta(\mathbf{R} + \rho - \mathbf{x}) d^6 Z - n_{0e}$ is the perturbed density. Introducing the potential discretization (11) gives:

$$E = \frac{q_i N_\varphi}{2} \sum_{v,n} \phi_v^{(n)} g_v^{(-n)} = \sum_{\mu,v,n} \phi_v^{(n)} M_{\mu v}^{(n)} \left(\phi_\mu^{(n)} \right)^\dagger \tag{A.2}$$

where Eq. (12) has been used, $v = (jk)$ describes radial index j and poloidal index k, and $\phi_v^{(n)} = \left(\phi_v^{(-n)} \right)^\dagger$ due to the realness of ϕ ($\mathbf{x}$). The field energy is positive because $M_{\mu v}^{(n)}$ is positive definite. The sum over n goes from $-N_\varphi/2 + 1$ to $N_\varphi/2$ but the realness of $\phi(\mathbf{x})$ can again be exploited to give:

$$E = \frac{N_\varphi q_i}{2} \sum_j \sum_{n=0}^{N_\varphi/2} E_j^{(n)} \tag{A.3}$$

$$E_j^{(n)} = \sum_k c_n \{ z_{jk}^{(n)} + [z_{jk}^{(n)}]^\dagger \} \tag{A.4}$$

$$z_{jk}^{(n)} = \phi_{jk}^{(n)} g_{jk}^{(-n)} \tag{A.5}$$

$$c_n = \left(1 - \frac{\delta_{n0}}{2} - \frac{\delta_{n,N_\varphi/2}}{2} \right) \tag{A.6}$$

$E_j^{(n)}$ is the *local* field energy for magnetic surface j. The *global* field energy is $\sum_j E_j^{(n)}$. Using a DFT in the poloidal direction gives:

$$E = \frac{q_i N_\varphi N_\chi}{2} \sum_j \sum_{n=0}^{N_\varphi/2} \sum_{m=-N_\chi/2}^{N_\chi/2} E_j^{(n,m)} \tag{A.7}$$

$$E_j^{(m,n)} = c_n \left\{ z_j^{(m,n)} + [z_j^{(m,n)}]^\dagger \right\} \tag{A.8}$$

$$z_j^{(m,n)} = \phi_j^{(m,n)} g_j^{(-m,-n)} \tag{A.9}$$

References

[1] X. Garbet, Plasma Phys. Control. Fusion 43 (2001) A251.
[2] X. Garbet, Y. Idomura, L. Villard, T.-H. Watanabe, Nucl. Fusion 50 (2010) 043002.
[3] T.S. Hahm, Phys. Fluids 31 (1988) 2670.
[4] S.J. Zweben, S.S. Medley, Phys. Fluids B 1 (1989) 2058.
[5] M.A. Beer, S.C. Cowley, G.W. Hammett, Phys. Plasmas 2 (1995) 2687.
[6] B. Scott, Phys. Plasmas 8 (2001) 447.
[7] A.M. Dimits, Phys. Rev. E 48 (1993) 4070.
[8] J.A. Heikkinen, S.J. Janhunen, T.P. Kiviniemi, F. Ogando, J. Comput. Phys. 227 (2008) 5582.
[9] W.X. Wang, Z. Lin, W.M. Tang, W.W. Lee, S. Ethier, J.L.V. Lewandowski, G. Rewoldt, T.S. Hahm, J. Manickam, Phys. Plasmas 13 (2006) 092505.
[10] Z. Lin, T.S. Hahm, W.W. Lee, W.M. Tang, R.B. White, Science 281 (1998) 1835.
[11] Y. Idomura, H. Urano, N. Aiba, S. Tokuda, Nucl. Fusion 49 (2009) 065029.
[12] S. Jolliet, A. Bottino, P. Angelino, R. Hatzky, T.M. Tran, B.F. McMillan, O. Sauter, K. Appert, Y. Idomura, L. Villard, Comput. Phys. Commun. 177 (2007) 409.
[13] V. Grandgirard, Y. Sarazin, P. Angelino, A. Bottino, N. Crouseilles, G. Darmet, G. Dif-Pradalier, X. Garbet, P. Ghendrih, S. Jolliet, G. Latu, E. Sonnendrucker, L. Villard, Plasma Phys. Control. Fusion 49 (2007) B173.
[14] B.F. McMillan, S. Jolliet, A. Bottino, P. Angelino, T.M. Tran, L. Villard, Comput. Phys. Commun. 181 (2010) 715.
[15] A. Bottino, A.G. Peeters, R. Hatzky, S. Jolliet, B.F. McMillan, T.M. Tran, L. Villard, Phys. Plasmas 14 (2007) 010701.
[16] Y. Idomura, M. Ida, T. Kano, N. Aiba, S. Tokuda, Comput. Phys. Commun. 179 (2008) 391.
[17] A.J. Brizard, T.S. Hahm, Rev. Modern Phys. 79 (2007) 421.
[18] X.Q. Xu, M.N. Rosenbluth, Phys. Fluids B 3 (1991) 627.
[19] S. Satake, Y. Idomura, H. Sugama, T.-H. Watanabe, Comput. Phys. Commun. 181 (2010) 1069.
[20] Y. Idomura, M. Ida, S. Tokuda, L. Villard, J. Comput. Phys. 226 (2007) 244.
[21] X. Zhong, J. Comput. Phys. 128 (1996) 19.
[22] S.C. Eisenstat, H.C. Elman, M.H. Schultz, SIAM J. Numer. Anal. 20 (1983) 345.

[23] Y. Idomura, S. Tokuda, Y. Kishimoto, Nucl. Fusion 43 (2003) 234.
[24] S. Satake, M. Okamoto, N. Nakajima, H. Sugama, M. Yokoyama, C.D. Beidler, Nucl. Fusion 45 (2005) 1362.
[25] P. Angelino, A. Bottino, R. Hatzky, S. Jolliet, O. Sauter, T.M. Tran, L. Villard, Phys. Plasmas 13 (2006) 052304.
[26] B.F. McMillan, S. Jolliet, T.M. Tran, L. Villard, A. Bottino, P. Angelino, Phys. Plasmas 15 (2008) 052308.
[27] S. Jolliet, B.F. McMillan, T. Vernay, L. Villard, A. Bottino, P. Angelino, Phys. Plasmas 16 (2009) 052307.
[28] R. Hatzky, Parallel Comput. 32 (2006) 325.
[29] K. Höllig, Finite Element Methods with B-Splines, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 2003.
[30] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J.D. Croz, A. Greenbaum, S. Hammarling, A. McKenney, D. Sorensen, LAPACK Users' Guide, third ed., Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1999.
[31] A.M. Dimits, G. Bateman, M.A. Beer, B.I. Cohen, W. Dorland, G.W. Hammett, C. Kim, J.E. Kinsey, M. Kotschenreuter, A.H. Kritz, L.L. Lao, J. Mandrekas, W.M. Nevins, S.E. Parker, A.J. Redd, D.E. Shumaker, R.D. Sydora, J. Weiland, Phys. Plasmas 7 (2000) 969.
[32] S. Jolliet, A. Bottino, P. Angelino, T.M. Tran, B.F. McMillan, R. Hatzky, A.G. Peeters, E. Poli, O. Sauter, L. Villard, Theory of fusion Plasmas, vol. 871, American Institute of Physics, Melville, New York, 2006, p. 124.
[33] X. Lapillonne, S. Brunner, T. Dannert, S. Jolliet, A. Marinoni, L. Villard, T. Göerler, F. Jenko, F. Merz, Phys. Plasmas 16 (2009) 032308.
[34] Y. Nishimura, Z. Lin, J.L.V. Lewandowski, S. Ethier, J. Comput. Phys. 214 (2006) 657.
[35] M.F. Adams, Y. Nishimura, Commun. Comput. Phys. 2 (2007) 881.
[36] M.N. Rosenbluth, F.L. Hinton, Phys. Rev. Lett. 80 (1998) 724.
[37] S. Jolliet, B.F. McMillan, T. Vernay, L. Villard, R. Hatzky, A. Bottino, P. Angelino, Phys. Plasmas 16 (2009) 072309.
[38] L. Villard, A. Bottino, S. Brunner, A. Casati, J. Chowdury, T. Dannert, R. Ganesh, X. Garbet, T. Goerler, V. Grandgirard, R. Hatzky, Y. Idomura, F. Jenko, S. Jolliet, S. Khoshaghdam, X. Lapillonne, G. Latu, B.F. McMillan, F. Merz, Y. Sarazin, T.M. Tran, T. Vernay, Plasma Phys. Control. Fusion 52 (2010) 124038.
[39] R.E. Waltz, J. Candy, M. Fahey, Phys. Plasmas 14 (2007) 056116.
[40] S. Jolliet, Gyrokinetic Particle-In-Cell Global Simulations of Ion-Temperature-Gradient and Collisionless-Trapped-Electron-Mode Turbulence in Tokamaks, PhD thesis no 4326, Ecole Polytechnique Federale de Lausanne, 2009.