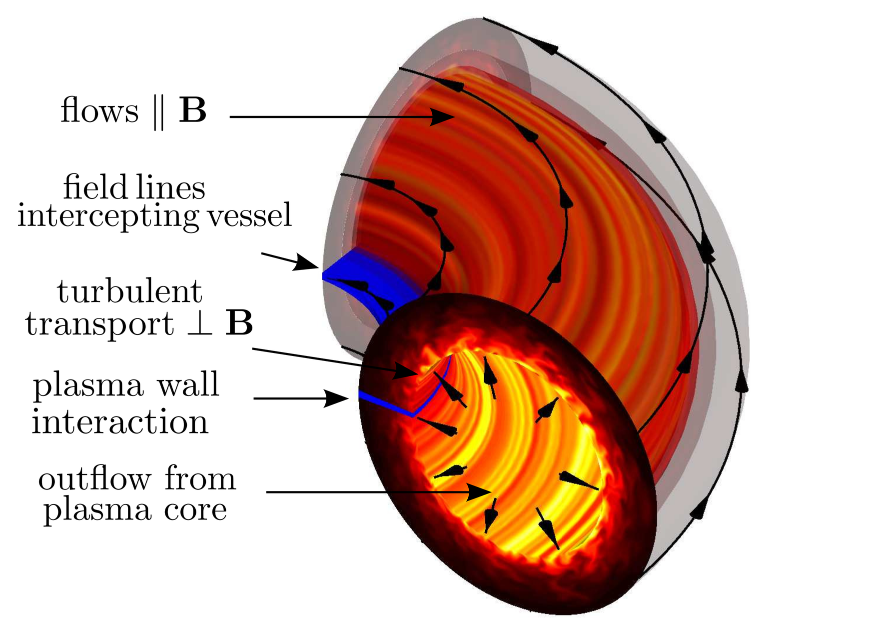
Plasma refuelling at the SOL simulated with the GBS code

A. Coroado, P. Ricci, C. Beadle, P. Paruta, F. Riva, C. Wersal

École Polytechnique Fédérale de Lausanne (EPFL), Swiss Plasma Center, CH-1015 Lausanne, Switzerland

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE | SWISS PLASMA CENTER

Introduction



- In tokamaks Scrape-Off Layer (SOL), **magnetic field lines intersect the walls** of the fusion device
- **Heat and particles** flow along magnetic field lines and are **exhausted to the vessel**
- **Turbulence** amplitude and size **comparable** to **steady-state** values
- **Neutral** particles interact with the plasma
- SOL plays a key role on determining the **refuelling** of the plasma

The **Global Braginskii Solver (GBS) code:**
a 3D, flux-driven, global turbulence code in limited geometry
used to study **plasma turbulence in the SOL**

- GBS is a simulation code to evolve plasma turbulence in the edge of fusion devices. [Halpern *et al.*, JCP 2016], [Ricci *et al.*, PPCF 2012]
- GBS solves 3D **fluid equations for electrons and ions**, Poisson's and Ampere's equations, and a **kinetic equation for neutral atoms**.

The Global Braginskii Solver (GBS) code

Two fluid drift-reduced Braginskii equations, $k_\perp^2 \gg k_\parallel^2$, $d/dt \ll \omega_{ci}$

$$\frac{\partial n}{\partial t} = -\frac{\rho_*^{-1}}{B}[\phi, n] + \frac{2}{B}[C(p_e) - nC(\phi)] - \nabla\cdot(n v_{\parallel e}\mathbf{b}) + \mathcal{D}_n(n) + S_n + n_n\nu_{iz} - n\nu_{rec} \quad (1)$$

$$\frac{\partial \Omega}{\partial t} = -\frac{\rho_*^{-1}}{B}\nabla_\perp\cdot[\phi,\omega] - \nabla_\perp\cdot\left[\nabla_\parallel\left(v_{\parallel i}\omega\right)\right] + B^2\nabla\cdot(j_\parallel\mathbf{b}) + 2BC(p) + \frac{B}{3}C(G_i) + \mathcal{D}_\Omega(\Omega) - \frac{n_n}{n}\nu_{cx}\Omega \quad (2)$$

$$\frac{\partial U_{\parallel e}}{\partial t} = -\frac{\rho_*^{-1}}{B}[\phi, v_{\parallel e}] - v_{\parallel e}\nabla_\parallel v_{\parallel e} + \frac{m_i}{m_e}\left[\frac{\nu j_\parallel}{n} + \nabla_\parallel\phi - \frac{\nabla_\parallel p_e}{n} - 0.71\nabla_\parallel T_e - \frac{2}{3n}\nabla_\parallel G_e\right] + \mathcal{D}_{v_{\parallel e}}(v_{\parallel e}) + \frac{n_n}{n}(\nu_{en} + 2\nu_{iz})(v_{\parallel n} - v_{\parallel e}) \quad (3)$$

$$\frac{\partial v_{\parallel i}}{\partial t} = -\frac{\rho_*^{-1}}{B}[\phi, v_{\parallel i}] - v_{\parallel i}\nabla_\parallel v_{\parallel i} - \frac{\nabla_\parallel p}{n} - \frac{2}{3n}\nabla_\parallel G_i + \mathcal{D}_{v_{\parallel i}}(v_{\parallel i}) + \frac{n_n}{n}(\nu_{iz} + \nu_{cx})(v_{\parallel n} - v_{\parallel i}) \quad (4)$$

$$\frac{\partial T_e}{\partial t} = -\frac{\rho_*^{-1}}{B}[\phi, T_e] - v_{\parallel e}\nabla_\parallel T_e + \frac{4T_e}{3B}\left[\frac{C(p_e)}{n} + \frac{5}{2}C(T_e) - C(\phi)\right] + \frac{2T_e}{3n}\left[0.71\nabla\cdot(j_\parallel\mathbf{b}) - n\nabla\cdot(v_{\parallel e}\mathbf{b})\right] + \mathcal{D}_{T_e}(T_e) + \mathcal{D}^\parallel_{T_e}(T_e) + S_{T_e} + \frac{n_n}{n}\nu_{iz}\left[-\frac{2}{3}E_{iz} - T_e + \frac{m_e}{m_i}v_{\parallel e}\left(v_{\parallel e} - \frac{4}{3}v_{\parallel n}\right)\right] - \frac{n_n}{n}\nu_{en}\frac{m_e}{m_i}\frac{2}{3}v_{\parallel e}(v_{\parallel n} - v_{\parallel e}) \quad (5)$$

$$\frac{\partial T_i}{\partial t} = -\frac{\rho_*^{-1}}{B}[\phi, T_i] - v_{\parallel i}\nabla_\parallel T_i + \frac{4T_i}{3B}\left[\frac{C(p_e)}{n} - \frac{5}{2}\tau C(T_i) - C(\phi)\right] + \frac{2T_i}{3n}\left[\nabla\cdot(j_\parallel\mathbf{b}) - n\nabla\cdot(v_{\parallel i}\mathbf{b})\right] + \mathcal{D}_{T_i}(T_i) + \mathcal{D}^\parallel_{T_i}(T_i) + S_{T_i} + \frac{n_n}{n}(\nu_{iz} + \nu_{cx})\left[\tau^{-1}T_n - T_i + \frac{1}{3\tau}(v_{\parallel n} - v_{\parallel i})^2\right] \quad (6)$$

$$\rho_\star = \rho_s/R_0, \quad \mathbf{b} = \frac{\mathbf{B}}{B}, \quad [A,B] = b\cdot(\nabla A\times\nabla B), \quad C(A) = \frac{B}{2}[\nabla\times(\frac{b)}{B}]\cdot\nabla A, \quad \nabla_\parallel f = \mathbf{b}_0\cdot\nabla f + \frac{\beta_{e0}}{2}\frac{\rho_*^{-1}}{B}[\psi, f]$$

$$p = n(T_e + \tau T_i), \quad U_{\parallel e} = v_{\parallel e} + \frac{\beta_{e0}}{2}\frac{m_i}{m_e}\psi, \quad \Omega = \nabla\cdot\omega = \nabla\cdot(n\nabla_\perp\phi + \tau\nabla_\perp p_i)$$

- Equations implemented in GBS, a **flux-driven** plasma turbulence code with limited geometry to study SOL heat and particle transport
- System completed with **first-principles boundary conditions** applicable at the magnetic pre-sheath entrance where the magnetic field lines intersect the limiter [Loizu *et al.*, PoP 2012]
- Parallelized using domain decomposition, **excellent parallel scalability** up to $\sim$ 10000 cores
- Gradients and curvature discretized using **finite differences**, Poisson Brackets using Arakawa scheme, integration in time using **Runge Kutta method**
- Code **fully verified** using method of manufactured solutions [Riva *et al.*, PoP 2014]
- Note: $L_\perp \to \rho_s$, $L_\parallel \to R_0$, $t \to R_0/c_s$, $\nu = ne^2R_0/(m_i\sigma_\parallel c_s)$ normalization

The Poisson and Ampere equations

- **Generalized Poisson equation**, $\nabla\cdot(n\nabla_\perp\phi) = \Omega - \tau\nabla_\perp^2 p_i$
- **Ampere's equation** from Ohm's law, $\left(\nabla_\perp^2 - \frac{\beta_{e0}}{2}\frac{m_i}{m_e}n\right)v_{\parallel e} = \nabla_\perp^2 U_{\parallel e} - \frac{\beta_{e0}}{2}\frac{m_i}{m_e}nv_{\parallel i}$
- Stencil based **parallel multigrid** implemented in GBS
- Elliptic equations separable in parallel direction allow for **independent 2D solutions** for each x-y plane

The kinetic neutral atoms equation

$$\frac{\partial f_n}{\partial t} + \vec{v}\cdot\frac{\partial f_n}{\partial\vec{x}} = -\nu_{iz}f_n - \nu_{cx}n_n\left(\frac{f_n}{n_n} - \frac{f_i}{n_i}\right) + \nu_{rec}f_i \quad (7)$$

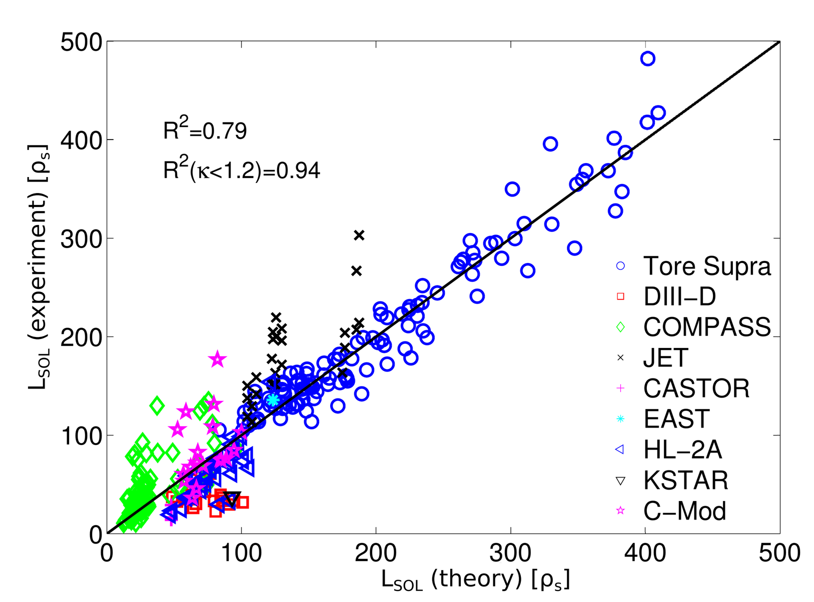
- **Method of characteristics** to obtain the formal solution of f_n [Wersal *et al.*, NF 2015]
- **Two assumptions**, $\tau_{\text{neutral losses}} < \tau_{\text{turbulence}}$ and $\lambda_{\text{mfp, neutrals}} \ll L_{\parallel,\text{plasma}}$, leading to a 2D steady state system for each x-y plane
- **Linear integral equation** for neutral density obtained by integrating f_n over $\vec{v}$
- **Spatial discretization** leading to a linear system of equations

$$\begin{bmatrix} n_n \\ \Gamma_{out}\end{bmatrix} = \begin{bmatrix} K_{p\to p} & K_{b\to p} \\ K_{p\to b} & K_{b\to b}\end{bmatrix}\cdot\begin{bmatrix} n_n \\ \Gamma_{out}\end{bmatrix} + \begin{bmatrix} n_{n,rec} \\ \Gamma_{out,rec} + \Gamma_{out,i}\end{bmatrix} \quad (8)$$

- This system is solved for neutral density, n_n, and neutral particle flux at the boundaries, Γ_{out}, with the threaded LAPACK or MUMPS (serial or parallel) solvers.

Past achievements of GBS

- Characterization of **non-linear turbulent regimes** in the SOL
- **SOL width scaling** as a function of dimensionless / engineering plasma parameters
- Origin and nature of **intrinsic toroidal plasma rotation** in the SOL
- Mechanisms regulating the SOL **equilibrium electrostatic potential**



Moving towards a density-conserving model

- Current version of the GBS code does not conserve charged particle density since:
  - the inverse aspect ratio $\epsilon = \frac{r}{R_0}$ is taken constant over the simulation domain, $\epsilon_0 = \frac{a_0}{R_0}$
  - parallel gradient components of Poisson brackets and curvature operators neglected
- Studying the plasma refuellling requires a density-conserving model to be implemented in GBS
- GBS must conserve the total sum of the ion+neutral density over the whole simulation domain
- This is important to address refuelling and Greenwald density limit physics
- Continuity equation must compute the exact variation of the ion density
- To make the model density-conserving, we implemented in GBS:
  - **Radially variable inverse aspect ratio** $\epsilon = \frac{r}{R_0}$ to take into account **curvilinear geometry**
  - **Parallel gradient terms** included in Poisson brackets and curvature operators

$$[\phi, A] = P_{yx}[\phi, A]_{yx} + \mathbf{P}_{\mathbf{x}\parallel}[\phi, \mathbf{A}]_{\mathbf{x}\parallel} + \mathbf{P}_{\parallel\mathbf{y}}[\phi, \mathbf{A}]_{\parallel\mathbf{y}}, \quad C(A) = C^x\frac{dA}{dx} + C^y\frac{dA}{dy} + \mathbf{C}^\parallel\nabla_\parallel\mathbf{A}$$

$$[\phi, A]_{uv} = \frac{d\phi}{du}\frac{dA}{dv} - \frac{d\phi}{dv}\frac{dA}{du}, \quad P_{yx} = \frac{a}{Jb^\varphi}, \quad \mathbf{P}_{\mathbf{x}\parallel} = \frac{\mathbf{b}_{\theta^*}}{\mathbf{J}\mathbf{b}^\varphi}, \quad \mathbf{P}_{\parallel\mathbf{y}} = \frac{\mathbf{a}\mathbf{b}_\mathbf{r}}{\mathbf{J}\mathbf{b}^\varphi}$$

$$C^x = -\frac{2B}{J}\frac{dc_\varphi}{d\theta^*}, \quad C^y = \frac{aB}{2J}\left[\frac{dc_\varphi}{dr} + \frac{1}{q}\left(\frac{dc_{\theta^*}}{dr} - \frac{dc_r}{d\theta^*}\right)\right], \quad \mathbf{C}^\parallel = \frac{\mathbf{B}}{2\mathbf{J}\mathbf{b}^\varphi}\left(\frac{\mathbf{dc_r}}{\mathbf{d}\theta^*} - \frac{\mathbf{dc}_{\theta^*}}{\mathbf{dr}}\right)$$

Field-aligned right-handed coordinates set: (θ^*, r, φ)

θ^* defined by $b^\varphi = qb^{\theta^*}$ (with q the safety factor) $\quad c_i = \frac{b_i}{B} \quad J = rR_0\frac{\left(1-\epsilon^2\right)^{3/2}}{(1-\epsilon cos(\theta^*))^2}$

Converts to (y, x, z) coordinates set with: $y = a\theta^*$, $x = r$, $z = R_0\varphi$

- Continuity equation is now density-conserving
- Gauss Theorem can be used when taking time and volume integration of the continuity equation, expressing volume-integrated density variation in terms of the fluxes across the volume's boundary.

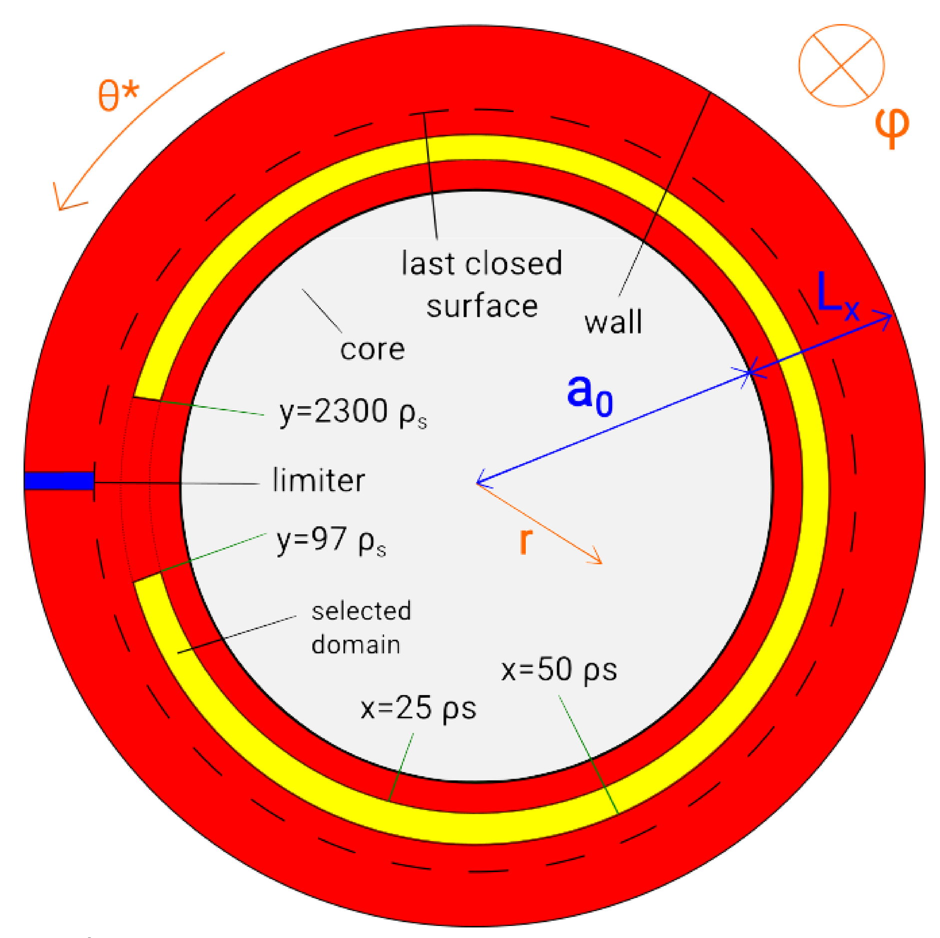
$$\int dt\int\frac{dn}{dt}dV = -\int dt\int(n\mathbf{v}_{de} + n\mathbf{v}_{\mathbf{E}\times\mathbf{B}} + nv_{\parallel e}\mathbf{b})\cdot d\mathbf{S} + \int dt\int(n_n\nu_{iz})dV \quad (9)$$

$$\mathbf{v}_{de} = \frac{1}{B^2}\nabla p_e\times\mathbf{B}, \quad \mathbf{v}_{\mathbf{E}\times\mathbf{B}} = -\frac{n}{B^2}\nabla\phi\times\mathbf{B}$$

- Diffusion $\mathcal{D}_n(n)$ is neglected at this stage, as well as source terms S_n and $n\nu_{rec}$

Numerical results

- **GBS Simulations** were run for 10 time steps taking the following parameters:
  - $\epsilon_0 = 0.2546$; $R_0 = 1500\rho_s$; circular centered magnetic flux surfaces
  - Simulation of an annular domain with $L_y = 2\pi a_0 = 2400\rho_s$ and $L_x = 150\rho_s$ (while $L_z = 2\pi R_0$)
  - Limited region at $x = 75 - 150\rho_s$
  - CG (coarse grid) with $N_y = 495$, $N_x = 191$, $N_z = 64$ and time step $\Delta t = 3.75\times10^{-6}$s
  - FG (fine grid) with $N_y = 990$, $N_x = 382$, $N_z = 128$ and time step $\Delta t = 1.875\times10^{-6}$s
- First, each of the four terms on the right hand side of (9) was taken separately in the continuity equation in GBS; then, all terms were taken into account



- GBS results were post-processed to obtain $\int dt\int\frac{dn}{dt}dV$ for a space domain and the integral of the right hand side of (9), the relative error between the two being computed.
- Results are presented for a domain inside the **closed flux surfaces region** defined by: $x = 25 - 50\ \rho_s$, $y = 97 - 2300\ \rho_s$, $z = 0.68 - 5.48\ R_0$

Terms considered in the equation	Relative error (%) for CG	Relative error (%) for FG
$n\,\mathbf{v}_{de}$	0.80%	0.12%
$n\,\mathbf{v}_{\mathbf{E}\times\mathbf{B}}$	0.020%	0.11%
$n\,v_{\parallel e}\,\mathbf{b}$	4.1%	6.0%
$n_n\,\nu_{iz}$	9.2×10^{-6}%	2.2×10^{-6}%
all terms	0.057%	0.010%

Discussion and conclusions

- Greatest contribution for particle transport in the closed flux surfaces region comes from perpendicular $E\times B$ transport, while the ionization contribution is also important ($\sim$10 times smaller)
- Non-converging relative error values are found for the $n\,v_{\parallel e}\,\mathbf{b}$ and $n\,\mathbf{v}_{\mathbf{E}\times\mathbf{B}}$ terms due to the numerical scheme used in GBS for the parallel gradient computation
- Since $k_\perp^2 \gg k_\parallel^2$ holds, errors arising from the parallel gradient contributions are negligible when taking the whole continuity equation
- The continuity equation in GBS is consistent with density conservation up to errors $<$ 0.1%.

Next step: plasma + neutrals conservation

- Neutral density variation can be obtained from the integral form of the neutral continuity equation:

$$\int dt\int\frac{dn}{dt}dV = -\int dt\int(n_n\mathbf{v}_n)\cdot d\mathbf{S} - \int dt\int dV(n_n\nu_{iz}) \quad (10)$$

- Ion flux obtained by taking the first order moments of f_n considering the contributions from charge-exchange in the plasma and electron-ion recycling at the limiter and walls

EUROfusion

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053.
The views and opinions expressed herein do not necessarily reflect those of the European Commission.