35th EPS Conference on Plasma Phys. Hersonissos, 9 - 13 June 2008 ECA Vol.**32D**, P-2.049 (2008)

Simulating drift kinetic electron-ion equation with collisions in complex geometry

J M Reynolds[1], D López-Bruna[2], J Guasp[2], J L Velasco[1], A Tarancón[1]

[1] *BIFI: Instituto de Biocomputación y Física de Sistemas Complejos, Zaragoza, Spain*

[2] *Laboratorio Nacional de Fusión, Asociación Euratom-Ciemat, Madrid, Spain*

Abstract

We introduce a new code of plasma transport based on evolving the Boltzmann equation in guiding center approximation. The aim is describing convection and collision dynamics with self consistent electric fields. The spatial geometry is discretized using high order elements in space and a moment expansion in velocity space. Time advance can be made both implicit or explicit. First calculations with non-evolving electric field agree with the particle code ISDEP

Keywords: guiding center, 5D velocity space, collisional, magnetic geometry mesh adapted, high order, discontinuous spectral elements, conservative, parallelized, time explicit/implicit.

Theoretical Model

From Guiding Center Hamiltonian dynamics for currentless confinement, we can obtain the equation for every plasma species, in pitch (λ) and velocity (v) coordinates:

$$\begin{aligned}\frac{\partial f_a(\vec{r},v^2,\lambda)}{\partial t} &+ \vec{\nabla}\cdot\left(\frac{d\vec{r}}{dt} f_a(\vec{r},v^2,\lambda)\right) + \frac{\partial}{\partial\lambda}\left(\frac{d\lambda}{dt} f_a(\vec{r},v^2,\lambda)\right) + \frac{1}{v}\frac{\partial}{\partial v^2}\left(v\frac{dv^2}{dt} f_a(\vec{r},v^2,\lambda)\right)\\ &= \sum_a C_{ab}(f_a(\vec{r},v^2,\lambda), f_b(\vec{r},v^2,\lambda)) \qquad (1)\end{aligned}$$

where $\frac{d\vec{r}}{dt}$, $\vec{v}_d$, μ, $\frac{d\lambda}{dt}$ and $\frac{dv^2}{dt}$ can be found in [1].

Mode Expansion in Velocity Space

We express the velocity space dependencies of the distribution function in an orthonormal basis:

$$f(\vec{r},v',\lambda) = \sum_{i=0,j=0}^{N_i N_j} C_{i,j}(\vec{r}) J_{i,j}(v',\lambda) e^{-v'^2},$$

where the members of the basis are:

$$J_{i,j}(v',\lambda) = k_{i,j} v'^j P_j(\lambda) L_i^{j+\frac{1}{2}}(v'^2)$$

being P and L Legendre and Laguerre polynomials respectively.

An important concern is v', the dimensionless velocity variable. The normalization velocity depends on local temperature and will be selected independently in every spatial hexahedral of the numerical mesh. The number of modes needed to describe f properly depends on the adjustement of the basis to the Maxwellian distribution, so v' can be viewed as a *local grid adjustement*. For a Maxwellian distribution and a correct v', only one mode is necesary. This is the key idea of the present code, so the distribution function is assumed not to be too far from Maxwellian. Thus, different hexahedrals have diferent basis expansion and when we need to operate data among domains, the distribution function must be transformed:

$$f(\vec{r}, v'_2, \lambda) = \sum_{i,j}^{N_{v'}, N_\lambda} C_{i,j}(\vec{r}) J_{i,j}(v'_2 \frac{v_{th_2}}{v_{th_1}}, \lambda) e^{-{v'_2}^2 \left(\frac{v_{th_2}}{v_{th_1}}\right)^2},$$

where $v'_1 = v'_2 \frac{v_{th2}}{v_{th1}}$. This can be seen as the projection of f onto the new local basis, better adapted to the different thermal velocity:

$$f^*(\vec{r}, v'_2, \lambda) = \sum_{i,j}^{N_{v'}, N_\lambda} C^*_{i,j}(\vec{r}) J_{i,j}(v'_2, \lambda) {v'_2}^2 e^{-{v'_2}^2}.$$

As for the collision operator in Eq. 1, a linealized Coullomb collision has been used [2]. Considering, as mentioned previously, that the distribution function is not far from Maxwellian, this is a good aproximation for a correct normalization v'.

Meshing Procedure

For the spatial discretization, a mesh adapted to the magnetic geometry is built in a way that follows as best possible the magnetic lines in a hexahedral conforming mesh. Every radial corona is discretized independently to minimize numerical diffusion in the radial direction. A 3D software tool (Fig. 1) has been developed for this purpose using C++ and OpenGL. A low precision solver follows the magnetic field lines and the user needs to find radial positions manually to find low order rationals of their pitch. From here, the nodes are established and then a corona-

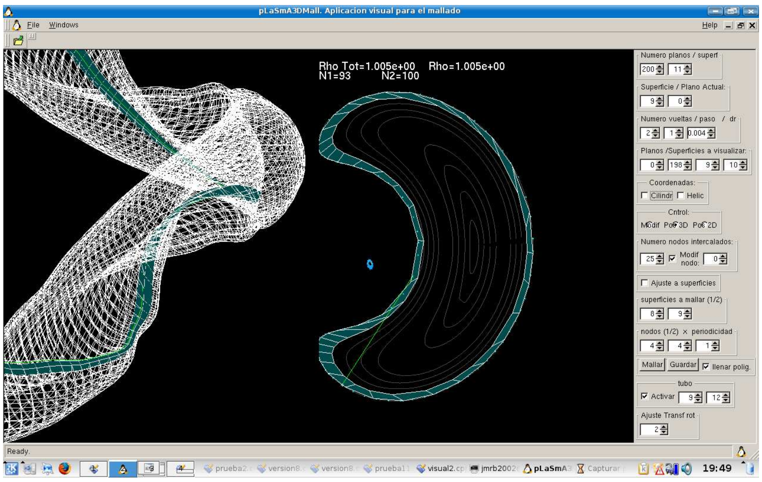

Figure 1: Application for the meshing procedure.

by-corona meshing is done. This is a flexible way to construct meshes of manually adaptable precision that minimizes the cross numerical diffusion of parallel-perpendicular transport.

Spatial and Time Discretization

For the spatial discretization, we have selected a discontinuous spectral elements method: Discontinuous Staggered Multidomain Spectral method [3] (equivalent for the case of hexahedrals to the Spectral Difference Method [4]). It is a high order conservative method. After the spatial meshing procedure the hexahedrals are defined . In every one of them, the system of equations resulting from the spectral modes expansion explained earlier can be expresed as:

$$\frac{\partial Q(\vec{r})}{\partial t} + \vec{\nabla} \cdot \vec{F}(\vec{r}) = S(\vec{r}).$$

The modes/fluxes are expanded in a local polynomial expansion based on the values of the modes/fluxes at the Gauss/Gauss-Lobatto points in a staggered local mesh using Lagrange interpolating polynomials (for more details, see [4]). Now the system can be explicitly solved in time. The fact that flux nodes are in Gauss-Lobatto points permits the interaction between data of diferents hexahedrals. The bi-valuated interface will be solved, as usual in fluids methods, solving a local Riemann flux problem. It is important to note that local normalization velocity is different in every hexahedral, so a transformation must be done as commented previously.

With this convection scheme, the parallelization of the code is near-trivial: one group of hexaedrals on every computing node.

Time can be advanced explicity using a 3rd order Runge-Kutta algorithm or implicitly thanks to SUNDIALS Cvode solver library [5].

Comparison With ISDEP, a Particle Code

ISDEP (Integrator of Differential Stochastic Equations for Plasmas) calculates the kinetic transport by solving the guiding-center trajectories of ions in a TJ-II plasma [6]. Equivalent simulations have been carried out with both codes and good convergence is found for the complex geometry of TJ-II. All simulations have been run with spatial 3x3x3 order polynomials and implicit time advance (tipical simulation time is 1-4 days on 20 CPU's). In all simulations, the initial density profile is linear and the ion temperature profile is flat at 100 eV (as usual in TJ-II ECRH discharges). Some results are depicted in Fig. 2.

Future Work

The code is prepared to evolve electron and ion species. The next step is to find equilibrium profiles from specific heat and particle sources with non evolving electric field. For this purpose, work is being carried on calculating the inverse of the system matrix. After this, an evolving

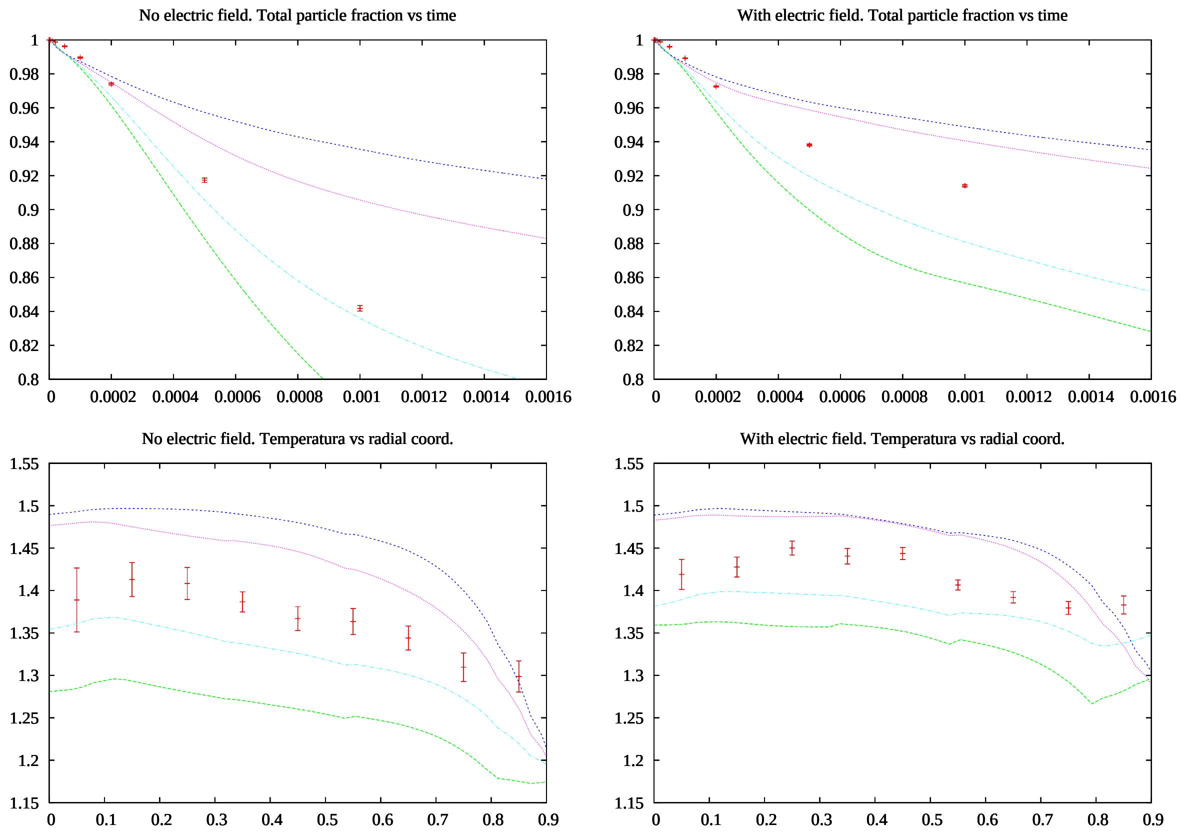

Figure 2: Comparation of code and ISDEP results. Points, ISDEP, green $(4,7)$ modes, blue $(8,7)$ modes, cyan $(4,12)$ modes, magenta $(4,13)$ modes.

radial electric field will be added to find the self-consistent electric field. This is a necessary stage prior to the scientific use of the present code.

Acknowledgments

The authors wish to acknowledge fellowship support from BIFI as well as thanking the CIEMAT supercomputing group for their help on this project.

References

[1] A. J. Brizard, T. S. Hahm, Rev. Mod. Phys. **79**, 421 (2007)

[2] J. Ji, Eric D. Held, Phys. Plasmas **13**, 102103 (2006)

[3] D. A. Kopriva, J. Comp. Phys. **128**, 475 (1996)

[4] Y. Sun, Z. J. Wang, Y. Liu, AIAA-2006-0301

[5] computation.llnl.gov/casc/sundials/main.html

[6] F. Castejón et al, Plasma Phys. Control. Fusion **49**, 753 (2007)