HYbrid and MHD simulation code (HYM)

The nonlinear 3-D simulation code (HYM) has been developed at PPPL to carry out detailed investigations of the macroscopic stability properties of FRCs. In the HYM code, three different physical models have been implemented: (a) a 3-D nonlinear resistive MHD model; (b) a 3-D nonlinear hybrid scheme with fluid electrons and particle ions; and (c) a 3-D nonlinear hybrid MHD/particle model where a fluid description is used to represent the thermal background plasma, and a kinetic (particle) description is used for the low-density energetic beam ions. The nonlinear delta-f method has been implemented in order to reduce numerical noise and optimize the computational requirements in the particle simulations. The modeling capabilities of the HYM code have recently been extended to include the options of a two-fluid (Hall-MHD) description of the thermal plasma, and the effects of finite electron pressure.

The HYM code is unique in that it employs the delta-f particle simulation method and a full-ion-orbit description in toroidal geometry. The use of the delta-f method has been
found to be crucial for understanding the underlying physics of macroscopic instabilities in the kinetic regime. In this method, the equilibrium ion distribution function f_0 is assumed to be known analytically, and the equation for the perturbed distribution function $\delta f = f - f_0$ is integrated along the particle trajectories. Each simulation particle is assigned a weight $w = \delta f / f$, which is evolved in time using the equation

$$f_0 \frac{dw}{dt} = -(1-w)\frac{df_0}{dt}. \qquad (1)$$

The particle weights are used to calculate the perturbed ion density and current density according to $\delta n(\mathbf{x},t) = \sum_m w_m S(\mathbf{x}_m - \mathbf{x})$, and $\boldsymbol{\delta}\mathbf{J}(\mathbf{x},t) = \sum_m w_m \mathbf{v}_m S(\mathbf{x}_m - \mathbf{x})$, where m is the simulation particle index, and $S(\mathbf{x})$ is a shape function. This method greatly simplifies linearization of the Vlasov equation, and it allows for a detailed study of the linear phase of instability, which otherwise would be completely obscured by the large numerical noise present in conventional particle-in-cell (PIC) simulations. The delta-f method, since it is fully nonlinear, is also essential for the study of weakly nonlinear effects, such as the saturation of instabilities at low

amplitudes. On the other hand, the delta-f method is found to be inaccurate in strongly nonlinear regimes. Therefore, the numerical scheme in the HYM code has been extended to allow for a dynamical switch from the delta-f method to the regular PIC method, when the perturbation amplitude becomes sufficiently large.

The initial equilibrium used in the HYM code is calculated using a Grad-Shafranov solver. The equilibrium solver allows the computation of MHD equilibria including the effects of toroidal flows. In addition, a new version of the Grad-Shafranov solver has recently been developed for kinetic equilibria with a non-Maxwellian and anisotropic ion distribution function.

CODE DESCRIPTIONS

The HYM code is a Fortran 90 code, which is written using a pre-processor language MPPL. The HYM code was written by Elena Belova to be run initially on the Parallel Vector Processor (PVP) cluster at NERSC. Presently the HYM code has been ported to run on the NERSC IBM SP RS/6000, a distributed memory machine with 6,080 compute processors. The HYM code uses the OpenMP application program interface, which supports multi-platform shared-memory parallel programming**.** An MPI (Message Passing Interface) version of the HYM code has been developed, in order to be able to perform calculations using several nodes at the NERSC IBM SP. A typical run requires from 10 to 100 hours.

The Grad-Shafranov equilibrium code, FRCIN (Fortran 90), has been developed for FRC studies. The MPI version of this code has been recently developed, which allows the calculation of FRC kinetic equilibria for a five-dimensional particle phase space. The calculation of the kinetic equilibrium requires typically about 10 minutes of wall-clock time, using 16 processors at Seaborg (NERSC).

Most of the post-processing, data analysis, and visualization in the proposed research will be performed using the IDL-based code, GUI1.