Hybrid Kinetic-MHD Simulations in General Geometry

Charlson C. Kim [a] Carl R. Sovinec [a] Scott E. Parker [b]

[a] *University of Wisconsin, Madison, CPTC*

[b] *University of Colorado, Boulder, CIPS*

The NIMROD Team

Abstract

We present a hybrid kinetic-MHD model consisting of 3 species, the bulk fluid ions and electrons, and a kinetic minority hot particle species. The 3 species equations are derived from moments of the Vlasov Equation and then reduced using the usual hot particle assumption of $n_h \ll n_0$, $\beta_h \sim \beta_0$ to formulate the hybrid kinetic MHD model. The 3 species equations reproduce the usual MHD equations with the addition of a hot particle pressure in the momemtum equation. In the limit $n_h \to 0$, the MHD equations are recovered. These model equations are implemented and examined in the NIMROD code [C.R. Sovinec, et al, "Nonlinear Magnetohydrodynamic Simulations Using Higher-Order Finite Elements", submitted to JCP] which solves three dimensional magnetohydrodynamic initial-value problems using the finite element method(FEM). The finite elements allow the represention of highly shaped geometries, but the particle-in-cell (PIC) method is complicated by the irregular grid. The associated complications are a nontrivial shape function, a more complex search algorithm, parallelization. We present our implementation of PIC in a FEM simulation and some preliminary results and performance measurements.

1 Introduction

It has long been known that the MHD model is limited in its ability to model plasma phenomena. The hybrid kinetic-MHD model[1,2] captures kinetic effects that are not possible to simulate with fluid MHD simulation alone. The addition of kinetic particle effects captures the wave-particle interactions lost in the conventional MHD approximation. Wave-particle interactions are known to strongly affect various MHD instabilities[3] such as internal kink stabilization, generation of sawtooth and fish bone instabilities, and toroidal Alfven eigenmodes.

To study the effects of a hot minority species on MHD modes in realistic geometry, we have developed δf particle-in-cell(PIC) method for quadrilateral finite element grid. This is implemented in the NIMROD code[4]. We assume that the kinetic particles are an energetic minority species, i.e. the kinetic particle density is small compared to the bulk plasma density, but the kinetic particle pressure is comparable to the bulk plasma pressure. The kinetic particles are evolved in the MHD fields using the drift kinetic equations of motion. A perturbed pressure tensor is calculated from velocity moments of the kinetic particles using the δf method. The δf PIC method has better particle noise properties for a slowly evolving distribution function than conventional PIC[5]. This hot particle pressure tensor is added to the momentum equation.

The particle simulation capabilities in NIMROD can be extended to simulate various phenomena such as neutral beam injection, ion cyclotron resonance heating, and anomalous loss mechanisms and lays the foundations for future work in a kinetic closures of the MHD equations.

2 The Hybrid Kinetic-MHD Equations

The hybrid kinetic-MHD equations are constructed by considering the three species consisting of an ions species, an electrons species, and a hot particle species and summing their governing equations:

$$\frac{\partial n_s}{\partial t} + \nabla \cdot (n_s \mathbf{V}_s) = 0 \tag{1}$$

$$\frac{\partial m_s n_s \mathbf{V}_s}{\partial t} + \nabla \cdot \underline{\mathbf{P}}_s = n_s \langle \mathbf{F}_s \rangle \tag{2}$$

$$\frac{1}{2}\frac{\partial n_s m_s \langle v_s^2 \rangle}{\partial t} + \nabla \cdot \left(\frac{n_s m_s}{2} \left\langle \mathbf{v}_s \, v_s^2 \right\rangle \right) = n \langle \mathbf{F}_s \cdot \mathbf{v}_s \rangle \tag{3}$$

where $\langle A \rangle = \frac{\int A f d\mathbf{v}}{n}$, where f is the phase space distribution function and n is the density, $\mathbf{V}_s = \langle \mathbf{v}_s \rangle$ the flow velocity, $\underline{\mathbf{P}}_s = m_s n_s \langle \mathbf{v}_s \mathbf{v}_s \rangle$ the stress tensor, $\mathbf{F}_s$ the Lorentz force, and the subscript $s = \{i, e, h\}$ denotes the species. Multiplying each continuity equation by species mass m_s and summing, gives the mass continuity equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \tag{4}$$

where $\rho \equiv \sum_s \rho_s$, $\rho_s \equiv m_s n_s$, $\rho \mathbf{U} \equiv \sum_s \rho_s \mathbf{V}_s$. Applying the same prescription for the momentum equations and shifting from the lab frame to the center of

mass (COM) frame, $\mathbf{v}_s = \mathbf{U} + \delta\mathbf{v}_s$, we have

$$\rho\left(\frac{\partial \mathbf{U}}{\partial t} + \mathbf{U}\nabla\cdot\mathbf{U}\right) = \sum_s \left(n_s\left\langle \mathbf{F}_s\right\rangle - \nabla\cdot\underline{\mathbf{p}}_s\right) \tag{5}$$

where we have rewritten the stress tensor, $\underline{\mathbf{P}}_s$, in eq.(2) as the pressure tensor, $\underline{\mathbf{p}}_s$, and advective derivative. For our definition of the pressure (defined with respect to the single fluid center of mass velocity $\underline{\mathbf{p}}_s = n_s m_s \left\langle (\mathbf{v}_s - \mathbf{U})\,(\mathbf{v}_s - \mathbf{U})\right\rangle$), this equation is exact. For completeness, we construct the generalized Ohm's law by multiplying equation(2) by $\frac{q_s}{m_s}$ for each species and summing

$$\mathbf{E} = \frac{1}{\omega_p^2}\left[\sum_s \frac{q_s}{m_s}\left(-\mathbf{J}_s\times\mathbf{B} + \nabla\cdot\underline{\mathbf{p}}_s\right) + \frac{\partial \mathbf{J}}{\partial t} + \nabla\cdot\left(\mathbf{UJ} + \mathbf{JU} - \mathbf{UU}\sum_s q_s n_s\right)\right] \tag{6}$$

where $\omega_p^2 = \sum_s \frac{q_s n_s}{m_s}$ is the plasma frequency, $\mathbf{J} = \sum_s \mathbf{J}_s$, and $\mathbf{J}_s = q_s n_s \mathbf{V}_s$. We leave the temperature equation in its species form [1]:

$$\frac{n_s}{\gamma - 1}\left(\frac{\partial T_s}{\partial t} + \mathbf{V}_s\cdot\nabla T_s\right) = -\tilde{p}_s\nabla\cdot\mathbf{V}_s - \nabla\cdot\mathbf{q}_s \tag{7}$$

where $T_s = m_s\left\langle(\mathbf{v}_s - \mathbf{V}_s)\cdot(\mathbf{v}_s - \mathbf{V}_s)\right\rangle$ is the species temperature, γ is the ratio of specific heat, and $\mathbf{q}_s = \frac{\rho_s}{2}\left\langle(\mathbf{v}_s - \mathbf{V}_s)^2\,(\mathbf{v}_s - \mathbf{V}_s)\right\rangle$ is the species heat flux, all defined in the COM frame of the species. We also distinguish the species COM pressure $\tilde{p}_s = p_s - \rho_s(\mathbf{U} - \mathbf{V}_s)^2$.

In the limit of $n_h \ll n_0$, $\beta_h \sim \beta_0$, the momentum equation, eq.(5), reduces to

$$\rho\left(\frac{\partial \mathbf{U}}{\partial t} + \mathbf{U}\nabla\cdot\mathbf{U}\right) = \mathbf{J}\times\mathbf{B} - \nabla\cdot\underline{\mathbf{p}}_b - \nabla\cdot\underline{\mathbf{p}}_h \tag{8}$$

where the subscripts b, p denote the bulk plasma and hot particles. The remaining equations are unchanged, but it must be remembered that the hot particle contributions to the flow are not necessarily negligible[6].

We solve these model equations in a hybrid fashion by coupling particle-in-cell (PIC) techniques for the hot minority species with the higher-order finite element code, NIMROD[4]. We describe the PIC implementation in FEM in the next section.

[1] collisions have not been included but could be added

3 PIC in FEM

Central to any PIC simulation is the gather and scatter. The particle gather and scatter on the finite element grid is done with the use of shape functions as in conventional PIC schemes. Previous implementations of PIC in FEM[7,8] have been limited to linear triangular finite elements. Here we present a PIC method for higher-order quadrilateral finite elements.

NIMROD uses a finite element grid in the poloidal plane and a finite Fourier series in the toroidal direction. For the poloidal plane, we make the distinction between the physical coordinates (R, Z), and the logical coordinates (p, q), which is the computational space of the finite element formalism. In the finite element formalism, the logical coordinates are mapped to the physical coordinates

$$R = \sum_{i=1}^{m^2} R_i N_i(p,q), \quad Z = \sum_{i=1}^{m^2} Z_i N_i(p,q), \tag{9}$$

where (R_i, Z_i) are the physical space coordinates of the nodes of the finite element, and $N_i(p, q)$'s are the Lagrange polynomials, which can be quadratic or higher degree in (p, q), associated with node i.

In the PIC algorithm, it is the physical coordinates of the particles $(\hat{R}, \hat{Z})$ that are known. The logical coordinates of the particle (p, q) must be determined in an iterative fashion due to the nonlinear nature of the mapping equations, eq.(9). We use the Newton-Raphson method to solve for (p, q) of the particle given $(\hat{R}, \hat{Z})$. Recall Newton-Raphson:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{f(\mathbf{x}_k)}{f'(\mathbf{x}_k)} \tag{10}$$

where $\mathbf{x} = (p, q)$ and f is the mapping function(eq.9). By substituting in the definitions for the shape functions and putting those expanded equations into the Newton-Raphson formalism we obtain the following iterative system for (p, q):

$$\begin{Bmatrix} p^{k+1} \\ q^{k+1} \end{Bmatrix} = \begin{Bmatrix} p^k \\ q^k \end{Bmatrix} + \begin{pmatrix} \frac{\partial R}{\partial p} & \frac{\partial R}{\partial q} \\ \frac{\partial Z}{\partial p} & \frac{\partial Z}{\partial q} \end{pmatrix}^{-1}_{(p^k, q^k)} \begin{Bmatrix} \hat{R} - \hat{R}^k \\ \hat{Z} - \hat{Z}^k \end{Bmatrix} \tag{11}$$

where k denotes the iteration, $(\hat{R}^k, \hat{Z}^k)$ are the particle coordinate obtained from eq.(9) using (p^k, q^k). We invoked the Inverse Function Theorem to cal-

culate f'^{-1}. We iterate eq.(11) until

$$\sqrt{(\hat{R}-\hat{R}^k)^2+(\hat{Z}-\hat{Z}^k)^2}<\epsilon \tag{12}$$

where ϵ is the exit condition and taken as a small number. The Newton-Raphson method gives rapid quadratic convergence. With the shape functions defined, we may push particles along their trajectories and calculate the perturbed hot particle pressure using the δf method[5].

4 Weight Equation for the Slowing Down Distribution

The hybrid kinetic-MHD model may be used to simulate a high energy, low density, minority species coming from beam injection, a heated minority species, or fusion byproduct (3.5 MeV alpha particles). In each case, the particles are initially (near) monoenergetic but quickly spread in phase space due to collisions with the background. This energetic tail population is modeled with a slowing down distribution function:

$$f_0=\frac{P_0\exp(\frac{P_\zeta}{\psi_0})}{\varepsilon^{3/2}+\varepsilon_0^{3/2}} \tag{13}$$

where ε is the energy of the particle, $P_\zeta = g(\psi_p)\rho_\parallel - \psi_p$ is the canonical poloidal momentum, $g(\psi_p)\rho_\parallel$ is the parallel momentum, ψ_p is the poloidal magnetic flux[9], and $(P_0,\varepsilon_0,\psi_0)$ are parameters that determine the shape of the distribution. The slowing down distribution is commonly used to model an energetic population resulting from Coulomb collisions with background electrons[10].

The evolution equation for δf along its characteristics[5] is

$$\dot{\delta f}=-\mathbf{v}_1\cdot\nabla f_0-e\mathbf{v}_0\cdot\mathbf{E}\partial_\varepsilon f_0. \tag{14}$$

Inserting the slowing down distribution and taking

$$\mathbf{v}_0=\mathbf{v}_\parallel+\mathbf{v}_D \tag{15}$$

$$\mathbf{v}_D=\frac{m}{eB^3}\left(v_\parallel^2+\frac{v_\perp^2}{2}\right)(\mathbf{B}\times\nabla B)+\frac{\mu_0 m v_\parallel^2}{eB^2}\mathbf{J}_\perp \tag{16}$$

$$\mathbf{v_1}=\frac{\mathbf{E}\times\mathbf{B}}{B^2}+\mathbf{v}_\parallel\cdot\frac{\delta\mathbf{B}}{B} \tag{17}$$

$$\mathbf{B}=\mathbf{B}_0+\delta\mathbf{B} \tag{18}$$

we obtain

$$\dot{\delta f} = f_0 \left\{ \frac{mg}{e\psi_0 B^3} \left[\left(v_\parallel^2 + \frac{v_\perp^2}{2} \right) \delta \mathbf{B} \cdot \nabla B - \mu_0 v_\parallel \mathbf{J} \cdot \mathbf{E} \right] + \frac{\mathbf{v}_1 \cdot \nabla \psi_p}{\psi_0} + \frac{3}{2} \frac{e\varepsilon^{1/2}}{\varepsilon^{3/2} + \varepsilon_0^{3/2}} \mathbf{v}_D \cdot \mathbf{E} \right\} \tag{19}$$

where the first set of terms in the curly brackets comes from the gradient and curvature drifts, the second term arises from the spatial dependence of f_0, and the last term is the work done against the perturbed electric field.

5 Deposition of Particle in FEM

To calculate the perturbed hot particle pressure, we deposit the second velocity moment onto the finite element grid. The perturbed hot particle pressure $\underline{\mathbf{p}}_h(\mathbf{x})$ is

$$\underline{\mathbf{p}}_h(\mathbf{x}) = \int m(\mathbf{v} - \mathbf{U})(\mathbf{v} - \mathbf{U}) \delta f(\mathbf{x}, \mathbf{v}) d^3\mathbf{v} \tag{20}$$

$$= \sum_{i=1}^{N} m(\mathbf{v_i} - \mathbf{U})(\mathbf{v}_i - \mathbf{U}) g_0 w_i \delta^3(\mathbf{x} - \mathbf{x}_i) \tag{21}$$

where the sum is over all particle $i \in [1, N]$, m mass of the particle, $g_0 w_i$ is the normalized weight for particle i[5,11], and and $\mathbf{U}$ is the flow velocity at the particle location (we suppress the subscript h henceforth). We have used the delta function representation of δf

$$\delta f(\mathbf{x}, \mathbf{v}) = \sum_{i=1}^{N} g_0 w_i \delta^3(\mathbf{x} - \mathbf{x}_i) \delta^3(\mathbf{v} - \mathbf{v}_i). \tag{22}$$

to go from the analytic expression(20) to the discrete numeric expression(21). Now, we discretize to the poloidal plane by performing a Fourier transform,

$$\underline{\mathbf{p}}_n(R, Z) = \frac{1}{2\pi} \sum_{i=1}^{N} m(\mathbf{v}_i - \mathbf{U})(\mathbf{v}_i - \mathbf{U}) g_0 w_i \delta(R - R_i) \delta(Z - Z_i) \frac{e^{-in\phi_i}}{R} \tag{23}$$

where $e^{-in\phi_i}$ projects the particle's ϕ_i to the fourier mode, n. This is most efficient for simulations with a low number of toroidal modes. For large n runs, a conventional ϕ deposition with a **FFT** is performed. We may expand

$\underline{\mathbf{p}}_n(R,Z)$ in the finite element basis, $\sum_k \underline{\mathbf{p}}_n^k N^k(p,q)$ and project eq.(23) onto the finite element space by casting it in weak form:

$$\int N^j \underline{\mathbf{p}}_n^k N^k dRdZ = \int N^j \frac{1}{2\pi} \sum_{i=1}^{N} m(\mathbf{v}_i - \mathbf{U})(\mathbf{v}_i - \mathbf{U}) g_0 w_i \delta(R - R_i)\delta(Z - Z_i) \frac{e^{-in\phi_i}}{R} \quad (24)$$

where $\underline{\mathbf{p}}_n^k$ are finite element pressure coefficients at the node k, and $N^{k(j)}$ is the shape function associated with node $k(j)$. This simplifies to

$$\mathbf{M}\underline{\mathbf{p}}_n^j = \sum_{i=1}^{N} \frac{m(\mathbf{v}_i - \mathbf{U})(\mathbf{v}_i - \mathbf{U}) g_0 w_i e^{-in\phi_i} N^j(p_i, q_i)}{2\pi R_i}. \quad (25)$$

where $\mathbf{M}$ is the finite element mass matrix. This gives our deposition scheme in FEM. The sum is performed only over those particles associated with the node j, which we know from our sort algorithm, described in the following section. This weak form formulation maintains conservation properties expected from PIC simulations.

6 Search Algorithm and Particle Sorting

The search algorithm and particle sorting are intimately tied to the calculation of the logical coordinates. In addition, the sorting is essential to parallelization of the particles. An added benefit of the sorting is reduced memory cache thrashing[12].

The search algorithm is tied to the solution of the logical coordinates. The logical coordinates (p,q) determine whether the particle is associated with a particular finite element. If either (p,q) are not between the values of $(-1,1)$, then the particle is not in this element, and another element is searched. The new element to be searched is determined by the current value of (p,q), left if $p<-1$, right if $p>1$, down if $q<-1$, up if $q>1$, and combinations thereof. This iterative search is carried out on a global search grid.

Once every particle has been located, a bucket sort is performed to sort the particles on the grid[12]. The bucket sort is an efficient means of sorting $\mathbf{n}$ balls into $\mathbf{m}$ buckets. The sorting is $\mathcal{O}(m)$. After the particles are located, there may be particles from one computational domain in a finite element cell of another domain as illustrated in fig. 2. The bucket sort begins with each processor counting the number of particles in each finite element. Each processor counts its own set of particles, but it must be remembered that particles may leave its computational domain, so the counting is done on the global grid. Then a sum is performed over the global grid to obtain the global total count of particles

associated with each finite element. Once the total global count is known, a displacement array can be calculated. The displacement array denotes where in the sorted particle list a particle residing in the **i**th element will go. Using the displacement list and the known location of the particle, the sorted particle list is filled and the displacement count is decremented one. With the particles sorted, passing the particles among processors may be done efficiently.

7 Performance Measurement

Typical calculations are run on massively parallel computers. As such, the particle algorithm was developed to be consistant with the domain decomposition used in NIMROD. Table (1) shows the flop rate for a small calculation on NERSC's IBM SP3. Initial performance measurements show reasonble efficiencies comparable to many large simulations. We note that there is a slow decline in parallel efficiency due to the increase in communications associated with the particle passing. This is illustrated in figure(3). Plotted in figure(3) is the time per timestep versus the number of particles per processor. It illustrates the linear scaling of performance with increasing particle numbers. Each line in the plot corresponds to a particular domain decomposition, denoted in the legend, and number of processors. The two numbers in the legend correspond to concentric decomposition and poloidal decomposition, respectively. The product of the two numbers total the number of processors. It is seen that for the same number of processors, the domain decomposition has a strong effect on the performance. This is due to the increase in domain boundaries and the resulting increase in the amount of particle passing.

8 Applications

Linear simulations of a $(1,1)$ internal kink mode with hot particle effects were done with this extended MHD code. The simulation displayed oscillations with $\gamma\tau_A \sim \omega\tau_A$, in qualitative agreement with Park, et al[13]. Figure (4) shows the contour plot of the $(1,1)$ pressure eigenmode along with the time history of the $(1,1)$ component of the fluid velocity. The oscillation has a real frequency of $.037\omega\tau_A$ and a growth rate of $.077\gamma\tau_A$ for a hot particle core $\beta_h \simeq .07$. This growth rate is a factor of three higher than the ideal kink mode growth rate.

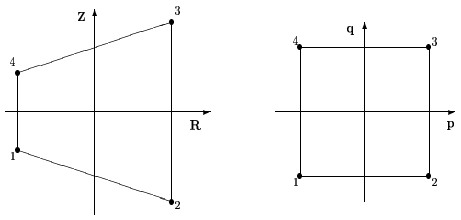
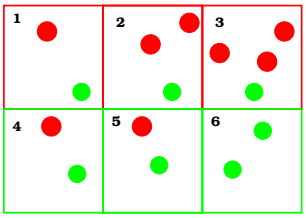


Fig. 1. illustration of sample mapping from irregular Cartesian space to uniform logical space



cell number	1	2	3	4	5	6
red count	1	2	3	1	1	0
green count	1	1	1	1	1	2
total count	2	3	4	2	2	2
local displacement	2	5	9	2	4	6

Fig. 2. illustration and counting of bucket sort. red and green correspond to different domains (i.e. processors)

#proc	16	32	64	128	256
Mflops/s	934.338	1490.952	2660.647	4396.228	6097.852

Table 1
performance measurement on seaborg for 20000 particles per processor

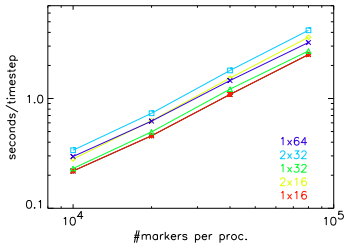


Fig. 3. scaling of time per timestep versus number of particles per processor for varying number of processors and domain decompositions

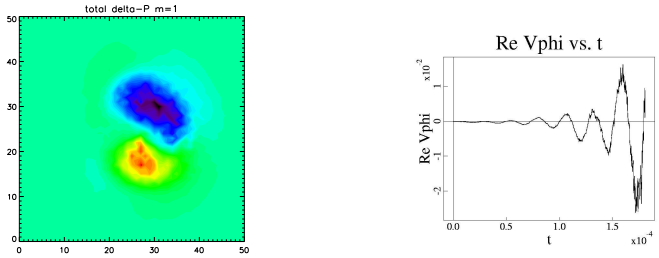


Fig. 4. contour plot of (1,1) p_h and time history of toroidal fluid velocity

9 Conclusion

We have presented a hybrid kinetic-MHD model. These model equations were solved using a δf PIC method applied to NIMROD. To apply the δf PIC method to this higher-order finite element initial value MHD code, we extended the PIC method to higher-order quadrilateral finite elements. The parallel performance on NERSC's IBM SP3 shows reasonable performance and scaling, comparable to many large scale simulations. This hybrid kinetic-MHD model was used to study kinetic effects on the $(1,1)$ internal kink mode. It was observed that the kinetic effects induced an oscillation of the $(1,1)$ mode and increased the growth rate for large hot particle β_h, in agreement with a similar code. Further numerical studies are underway.

References

[1] C. Z. Cheng, J. Geophysical Res. 96 (1991) 21,159.

[2] C. Z. Cheng, J. R. Johnson, J. Geophysical Res. 104 (1999) 413.

[3] W. Park, S. Parker, H. Biglari, M. Chance, L. Chen, C. Z. Cheng, T. S. Hahm, W. W. Lee, R. Kulsrud, D. Monticello, L. Sugiyama, R. White, Phys. Fluids B 4 (1992) 2033.

[4] C. R. Sovinec, A. H. Glasser, T. A. Gianakon, et al, Nonlinear magnetohydrodynamic simulations using higher-order finite elements, J. Comput. Phys. pending publication.

[5] S. Parker, W. Lee, Phys. Fluids B 5 (1993) 77.

[6] S. T. Hsu, D. J. Sigmar, Phys. Fluids B 4 (1992) 1492.

[7] F. Assous, P. Degond, J. Segre, Comput. Phys. Com. 72 (1992) 105.

[8] F. Kazeminezhad, S. Zalesak, D. Spicer, Comput. Phys. Com. 90 (1995) 267.

[9] R. B. White, The Theory of Toroidally Confined Plasmas, Imperial College Press, London, 2001.

[10] R. J. Goldston, P. H. Rutherford, Introduction to Plasma Physics, Institute of Physics Publishing, Philadelphia, 2000.

[11] G. Hu, J. Krommes, Phys. Plasmas 1 (1994) 863.

[12] K. J. Bower, Ph.D. thesis, U.C. Berkeley (2001).

[13] W. Park, E. V. Belova, G. Y. Fu, et al, Phys. Plasmas 6 (1999) 1796.