INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **47**:693–705
Published online 31 January 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.912

Particle methods

J. U. Brackbill*,†

Department of Mathematics and Statistics, University of New Mexico, Albuquerque, NM, U.S.A.

SUMMARY

Three, very different particle methods, PIC, SPH and vortex-blob methods, are compared. Results on consistency and stability are collected. A PIC instability, its properties, and its resolution by stability in the energy norm is discussed at some length. Examples are given of the mutual exchange between computational fluid dynamics and plasma modelling. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: particle methods; ALE

1. INTRODUCTION

A particle method models a continuum using point-set data. Convolution of the mass-point data with a particle shape function, S, for example, yields a mass density ρ

$$\rho(x) = \sum_p m_p \delta(x_p) * S \tag{1}$$

from a discontinuous set of data points. The density has the continuity and differentiability properties of S. $\int S(x)\,dx = 1$, and typically S has bounded support.

The particle-in-cell (PIC) method was invented by Harlow for compressible flow problems [1]. It combines particles to follow material motion with a grid to solve the equations of motion. In PIC, the particle co-ordinates are advanced by an area-weighted mean of the velocities from neighbouring grid points. The area-weights are computed from the overlap of a piecewise constant particle shape function

$$S^{(0)}(x) = \begin{bmatrix} \frac{1}{h} & |x| \leqslant \frac{h}{2} \\ 0 & |x| > \frac{h}{2} \end{bmatrix} \tag{2}$$

*Correspondence to: J. U. Brackbill, Department of Mathematics and Statistics, University of New Mexico, Albuquerque, NM 87131, U.S.A.
†E-mail: jubrackbill@comcast.net

Received 27 April 2004
Revised 12 December 2004
Accepted 16 December 2004

with grid cells. Both the support of $S^{(0)}$ and the grid spacing are equal to h. The overlap area A of a particle p's shape function with cell c is given by the convolution of particle and cell shape functions

$$A = S^{(0)}(x_c) * S^{(0)}(x_p) \tag{3}$$

where x_p and x_c are the particle and cell centre co-ordinates, respectively. Grid cells exchange conserved variables, such as mass, in proportion to the number of particles that cross their common boundaries. The nearest grid point (NGP) function assigns δ function particles to the cells. One computes, for example, the density in cell c from,

$$\rho_c = \sum_p m_p S^{(0)}(x_c - x_p) \tag{4}$$

The advantages of PIC are its ability to model highly distorted flows and interfaces, especially hydrodynamically unstable interfaces. The method has had some notable successes, but its low accuracy and apparent instability in stagnating flows caused it to become obsolete.

The PIC method was reinvented as the cloud-in-cell (CIC) method for plasma simulation [2]. The CIC algorithm models the self-consistent motion of charged particles in electric and magnetic fields. In contrast to PIC, CIC computes both particle charge contributions to grid cells and forces acting on particles by area-weighting, i.e. by a convolution of piecewise constant shape functions. Thus, for example, the charge density in cell c, ρ_{qc}, is given by an area-weighted sum of the particle charges, q_p,

$$\rho_{qc} = \sum_p q_p S^{(1)}(x_c - x_p), \quad S^{(1)} = \begin{bmatrix} \frac{1}{h}\left(1 - \frac{|x|}{h}\right) & |x| \leqslant h \\ 0 & |x| > h \end{bmatrix} \tag{5}$$

from the recursion relation for b-splines,

$$S^{(l+1)} = S^{(l)} * S^{(0)} \tag{6}$$

Equations for the self-consistent electric and magnetic fields are solved on the grid. CIC 'clouds' model collision-less plasmas very well, and such models continue to dominate the field of plasma simulation.

The vortex-in-cell method [3] applies CIC methodology to incompressible, inviscid flow. The vortex method tracks the motion of vortex blobs in the flow they induce. Both particle vorticity contributions to the grid and particle motion are computed by area-weighting. However, the apparent diffusion in vortex-in-cell results, and the prospect of applying grid-free particle vortex methods to singular boundary layers caused Rosenhead's point vortex method to be more widely adopted. In the point vortex method, a particle vortex weighted convolution of δ-function particle shape functions with a Green's function yields the velocity. The singularity of the Green's function with point particles, which causes an apparent lack of convergence [4], lead to the regularization of point vortices as in Equation (1), but typically using higher order functions

$$p = 2, \quad S_h = \frac{\mathrm{e}^{-r^2/2h^2}}{2\pi h^2}$$

$$p = 4, \quad S_h = \frac{\mathrm{e}^{-r^2/h^2} - \frac{1}{2}\,\mathrm{e}^{-r^2/2h^2}}{\pi h^2}$$

Proofs of stability and convergence quickly followed, and recipes for particle shape functions yielding high order accuracy followed [5]. However, accuracy is conditional on adequate coverage of the domain by particles [6]. In addition, high accuracy integration of the Green's function can be sustained only so long as the particles retain some spatial order.

The smoothed particle method (SPH), is a grid-less method for compressible flow. Hydrodynamic forces are short-range and there are constants of the motion to be preserved. In SPH, the shape function support h defines the interaction distance between particles. Conservation is achieved by carefully preserving the reciprocity of interactions between pairs of particles [7]. For example, velocity gradients are computed by differentiating

$$\rho\nabla\cdot\mathbf{v}=\nabla\cdot(\rho\mathbf{v})-\mathbf{v}\cdot\nabla\rho \tag{7}$$

rather than Equation (1). The result is

$$\nabla\cdot\mathbf{v}=\frac{\sum_{p'}(\mathbf{v}_{p'}-\mathbf{v}_p)\cdot\nabla_p S(\mathbf{x}_p-\mathbf{x}_{p'})}{\rho} \tag{8}$$

A typical shape function for SPH is given by

$$S_h=\frac{1}{\pi^{3/2}h^3}\left(\frac{5}{2}-r^2\right)\mathrm{e}^{-r^2/h^2} \tag{9}$$

with $\int S(r)\mathrm{d}r=1$ and $\int r^2S(r)\,\mathrm{d}r=0$.

Explicit time integration limits the domain of dependence to particles whose support overlaps, so that individual particle motions are computed within a moving window. The flow velocity is not single-valued, in general, but viscous interaction between particles reduces multistreaming. The form of the derivative implied by this viscosity is used in a particle exchange method for transport [8].

A hybrid fluid and kinetic plasma simulation algorithm, the implicit moment method, solves kinetic equations on hydrodynamic time scales [9]. (This is an extremely long time scale for a plasma, which has many high frequency modes.) One needs only the charge and current densities to solve Maxwell's equations, and these are given by the $k=0$ and $k=1$ moments of the velocity distribution of the particles, $M^k=\int v^k f(x,v,t)\,\mathrm{d}^3v$. Since there are more particles to move than moment equations to solve, it is less challenging to solve implicitly differenced moment equations self-consistently with Maxwell's equations. The moment equation

$$\frac{\partial M^{(k)}}{\partial t}+\frac{\partial M^{(k+1)}}{\partial x}=kF(x,t)M^{(k-1)} \tag{10}$$

where $F(x,t)$ is the force, do not form a closed system. However, the pressure, $M^{(2)}$, can be evaluated from the particles and treated explicitly in time, because the stability constraint it imposes is on the hydrodynamic time scale. Similar ideas are currently being explored for flows in micro-devices, where the Froude number may be $O(1)$ [10].

To complete the circle from Harlow's PIC through plasma simulation and the implicit moment method, the fluid-implicit-PIC method (FLIP) can be derived by taking the collisional limit of the implicit moment equations for neutral particles [11]. The particle shape functions are defined as functions of logical co-ordinates, and the computation cycle is divided into a Lagrangian phase, during which the logical co-ordinates and the particle weights are constant, and an 'Eulerian' or remap phase, during which the particles are stationary and the grid is

moved or replaced. Since all data are carried by particles and essentially none by the grid, the motion of the grid is arbitrary. Implicit in time differencing, extends PIC's applicability to low-speed flows, and together with staggered mesh difference equations on the mesh, gives momentum and energy conservation. A variant of FLIP, called the material point method (MPM), evaluates derivatives analytically [12]. For example the gradient of the density is computed from Equation (1),

$$\nabla\rho(x) = \sum_p \int m_p \delta(x' - x_p) \nabla_x S(x - x')\, \mathrm{d}x' \tag{11}$$

2. ON THE STABILITY AND ACCURACY OF PARTICLE METHODS

2.1. Accuracy

A study of the vortex-blob method by Perlman compares theoretical estimates of the consistency error, which is defined as the distance between the exact flow velocity and the discrete velocity computed from a distribution of vortex blobs, with the results of numerical computations [13]. The error estimate is given by

$$\varepsilon \leqslant C_1 \left(\frac{\delta}{h}\right)^L h^{-1-\gamma} + C_2 h^p \tag{12}$$

where δ is the separation of particles, h is the support of the particle shape function, L is a measure of the smoothness of the particle shape function, $\gamma < 1$, and C_1 and C_2 depend on the smoothness of the solution but not on h and δ. Perlman's results suggest that the high order of accuracy obtained with high order kernels is lost as particles are disordered by the evolving flow. However, if each blob always overlaps several others, second order accuracy persists, in which case, '*the smoothing error should be larger than the discretization error*' [13, p. 220].

An analysis of the accuracy of the PIC method yields a similar conclusion [14]. With particle shape function, $S^{(0)}$, the error is bounded by

$$\varepsilon \leqslant C_1 \left(\frac{\delta}{h}\right)^2 + C_2 h^2 \tag{13}$$

where δ is the separation of particles, h is the mesh spacing, and C_1 and C_2 are independent of δ and h, and depend only on the smoothness of the data. In PIC, the ratio, δ/h, is the number of particles per cell.

A convergence analysis for SPH shows that the method is convergent, with convergence rate $\varepsilon \leqslant Ch^{1/4}$ [15].

To those familiar with particle methods, these collected accuracy results may seem unreasonably pessimistic, especially for SPH. Perhaps it is because the analysis emphasizes approximation error, which is high for particles, and ignore fidelity to the dynamical equations, which can be excellent for particle methods [16].

2.2. *Numerical stability properties of particle methods*

The tension instability in SPH is described in Reference [17]. It appears to be due to the density weighting in Equation (8), and causes the fragmentation of solid objects under tension. One proposed remedy is an added artificial pressure to prevent clumping [18].

Harlow noted the instability in PIC calculations of stagnating flows, and called it the ringing instability. An example is shown in Figure 1, where particles from a calculation of a Kelvin–Helmholtz instability (KHI) are shown. Initially, the particles above the midplane are moving to the right relative to those below at Mach number equal to 0.1. A small, mode 1 perturbation in the velocity is imposed initially. The plots in Figure 1 are at equal time intervals. The calculation does not yield the expected growth of the KHI. Rather, it displays small scale, localized disturbances of the flow that cause particle clumping and disordering. One can, and many do, write this off as the natural result of the diffusiveness of a low-accuracy method, and the noisiness of PIC methods. In fact, it is a numerical instability due to aliasing errors, which occur because the grid undersamples particle data [19]. The grid supports modes with wave numbers $k \in [-\pi/\Delta x, \pi/\Delta x]$. However, there are many more particles than grid points, and particles respond to harmonics of the principal modes, $k_q = \pm q\pi/\Delta x$. False resonances from these harmonics cause exponential growth of a broad range of modes in subsonic flow. In supersonic flow, the instability, which is called the finite grid instability, is completely absent. The instability growth can be reduced by using higher order particle kernels, and eliminated by kernels whose support covers the entire domain, e.g. a Gaussian kernel.

The results of a linear dispersion analysis of the finite grid instability are shown in Figure 2, where the growth rates are clearly largest for low-speed flow, but are distributed over all wave numbers. The same KHI problem yields very different results with FLIP, Figure 3. However, because FLIP uses both higher order particle shape functions and implicit differencing in time, the results do not identify unambiguously cause of the differences. For example, while higher order shape functions reduce the growth rate of the FGI, they do not eliminate it [19]. Furthermore, linear dispersion theory attributes little effect to implicit differencing, Figure 2(b).

On the other hand, a comparison of explicit and implicit MPM grain compression results, where the difference equations differ only in the time-advancement, the results are dramatically better with implicit differencing, compare Figure 4(d) with Figure 4(b) [30].

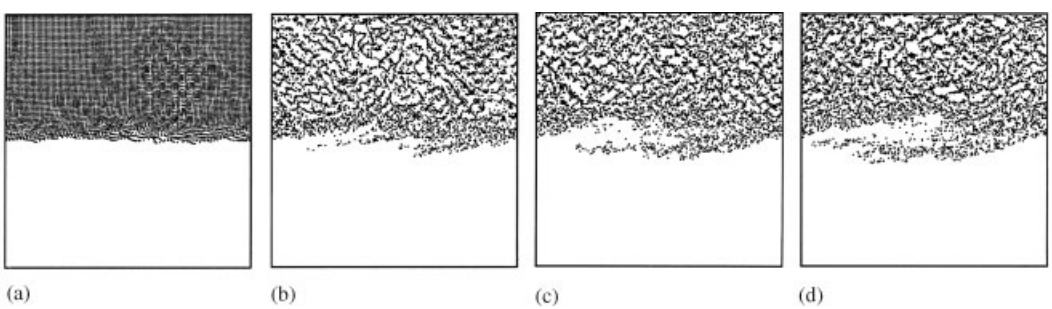

Figure 1. Particle plots from a PIC calculation of a KHI show evidence of a finite grid instability.

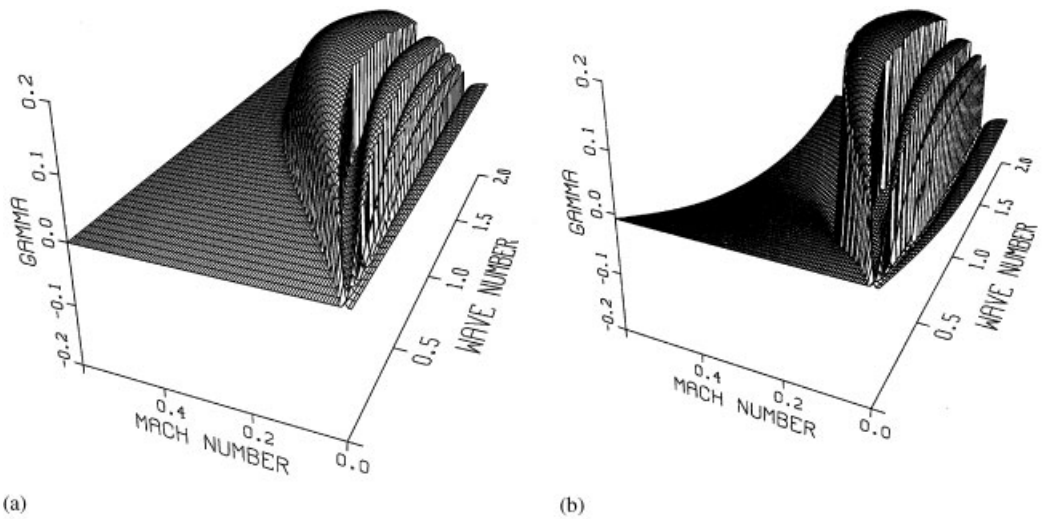

Figure 2. The growth rates of the finite grid instability are plotted as a function of Mach number and wave number. The broadband instability is a feature of low-speed flow. (a) The growth rate for explicit differencing in time; and (b) the growth rate with implicit differencing in time is reduced.

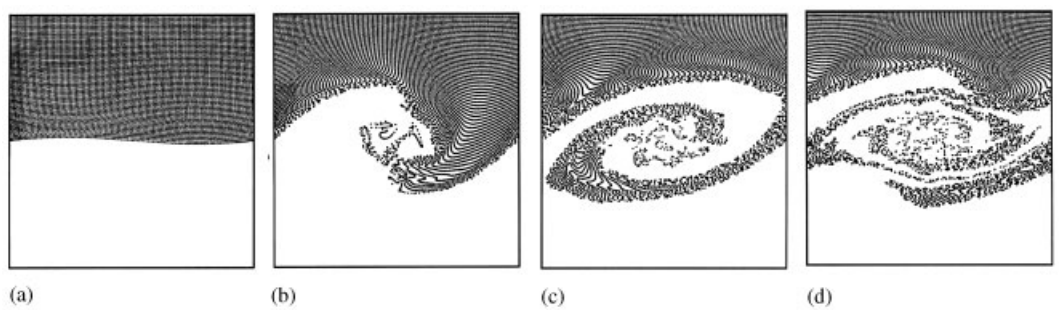

Figure 3. Particle plots from a FLIP calculation of a KHI show no evidence of a finite grid instability, and a much stronger physical instability.

The case of a single grain undergoing repeated collisions with a rigid wall reveals growth in the elastic energy of the grain with an explicit calculation that has no physical cause, and conservation with an implicit calculation, Figure 5.

Lilly shows that linear and quadratic conserving difference schemes eliminate the instabilities caused by aliasing [20]. (He and Arakawa are said to have claimed '*that conservation of* [*quadratic*] *quantities in long-term integrations were more important than accurate phase propagation or prediction of single specific events*' [21, p. 565], a view that is not widely shared at present.) It is said with regard to spectral methods, '*Because* [*the*] *formulation … conserves* [*energy*], *we are assured that aliasing instabilities are not present*' [22, p. 120]. Others make a connection between boundedness in the energy norm, and the Lax-Richtmyer sufficient condition [23].

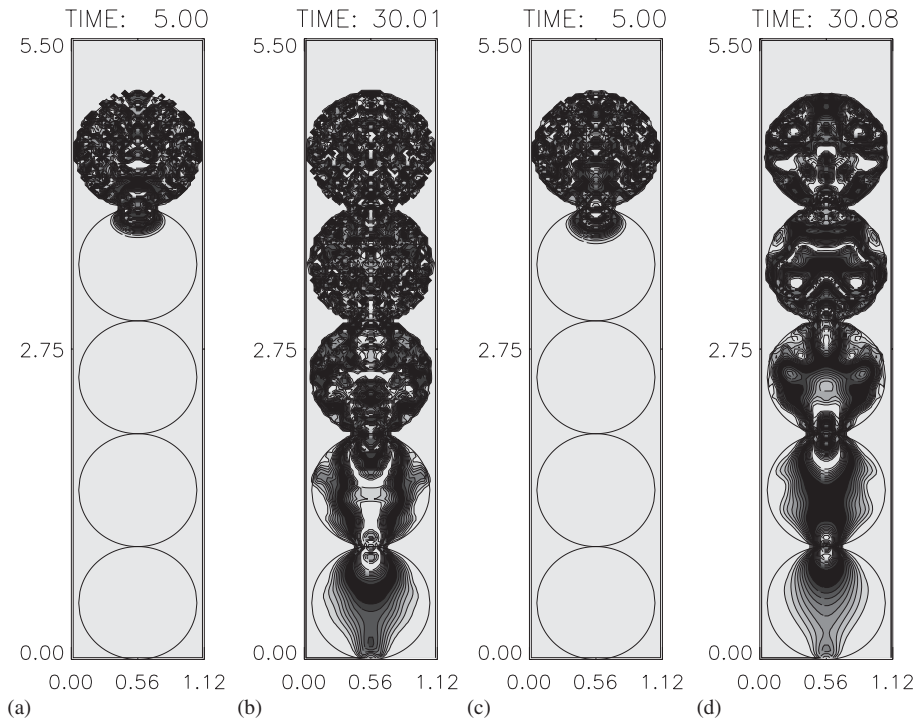

Figure 4. Explicit (a,b), and implicit (c,d) calculations of stress propagation through packed grains are shown. The explicit calculations exhibit noise contributed by the finite grid instability.

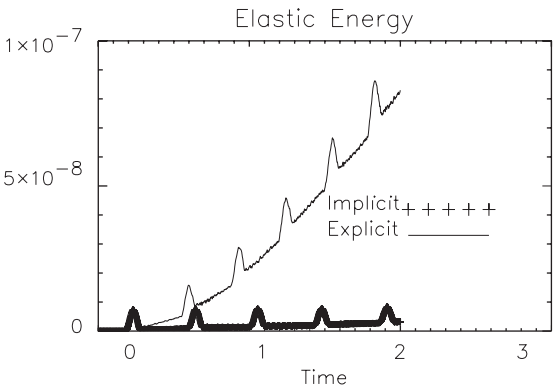

Figure 5. The elastic energy for a single grain experiencing repeated collisions with a rigid wall reveal an unphysical growth of the energy with explicit differencing that is absent in the implicit calculations.

When the energy method is applied to the explicit and implicit calculations above, one finds that implicit solutions not only conserve energy, but also bound all quadratic quantities. The time step dependence of the finite grid instability in explicit calculations, which linear analysis says should not exist, is explained by the increased accuracy of energy conservation

with the leapfrog algorithm with decreased time steps. With smaller time steps, increasingly stringent bounds on quadratic quantities constrains instability growth.

3. PLASMA TURBULENCE AND FLUID INSTABILITY

A much studied plasma equilibrium, the Harris current sheet, is generated by a drifting Maxwellian particle distribution function with number density variation in z, written for species s

$$f_{0s} = n(z)\frac{1}{v_s^3(2\pi)^{3/2}} \cdot \exp\left[-\frac{\left(v_x^2 + (v_y - u_s)^2 + v_z^2\right)}{2v_s^2}\right] \tag{14}$$

where u_s, the drift velocity, and $v_s = \sqrt{kT_s/m_s}$, the thermal velocity for species s, are spatially uniform.

Because the centre of mass velocity is spatially uniform, the Harris equilibrium is hydrodynamically stable. On the other hand, if vorticity were to replace the magnetic field in the equilibrium above

$$\omega_x(z) = -\omega_0 \tanh(z/L) \tag{15}$$

there would result a jet

$$u_y(z) = u_0 \frac{1}{\cosh^2(z/L)} \tag{16}$$

that is unstable to the KHI for all wavelengths of $O(L)$ or longer.

Simulation of the Harris sheet yields a curious result [24].

Early in the simulations, the well-known lower hybrid drift instability (LHDI) grows. In Figure 6(a), where a contour plot of E_y at ($\omega_{ci}t = 7$) is drawn, one can see the characteristic

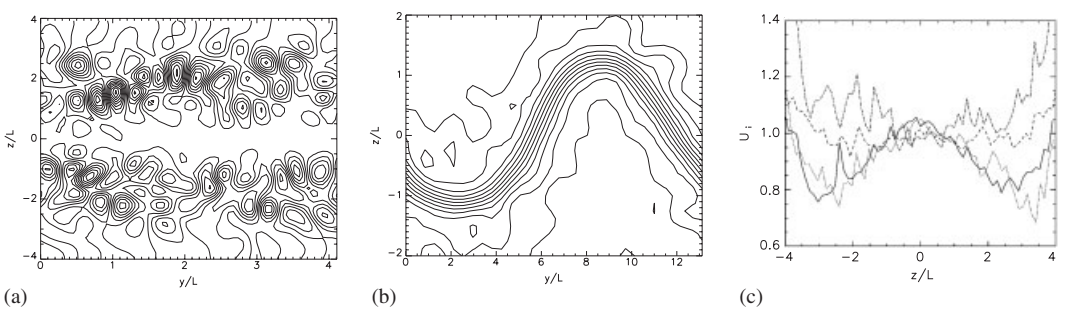

Figure 6. (a) Contours of E_y at $\omega_{ci}t = 7$ with $u_i/v_i = 1$, $T_i/T_e = 4$, $m_i/m_e = 180$, $L_y/L = 4$ reveal the growth of the lower hybrid drift instability; (b) contours $B_x(y,z)$ at a much later time ($\omega_{ci}t = 110$) reveal the growth of a kinking instability; and (c) the shear in the velocity caused by the lower hybrid drift instability explains the subsequent growth of the KHI.

'fishbone' pattern of the LHDI. The LHDI is driven by gradients of the density on the edge of the current sheet, as seen in Figure 6(a). Its growth tends to reduce gradients, which causes a plasma pressure imbalance that is resolved by thinning and intensification of the current. (Induction sustains the total current flow.)

Late in the simulations, a long wavelength mode grows after the LHDI saturates, Figure 6(b), which looks remarkably like a KHI. Shown is a contour plot of B_x at $\omega_{ci}t = 110$. The closely spaced contours lie in the current sheet, and their motion reflects the bulk plasma motion.

The growth of the KHI is explained by changes in the ion density. When the ion density changes, the ion velocity also changes, Figure 6(c). The plasma physics explanation for this is that the initial (constant) ion velocity is 'diamagnetic',

$$\mathbf{u}_i = \frac{\mathbf{B} \times \nabla p}{neB^2} \tag{17}$$

With the growth of the LHDI in the flanks of the profile, the ion density gradient and ion velocity decrease below their initial values. Figure 6(c) shows this variation of u_y with z. (The significant changes are on either side of the current sheet. The increased u_y near $z = \pm 4$ is in a region of very low plasma density, n.)

There is a more intuitive explanation for the development of a macroscopic shear velocity. The motion of individual ions as they gyrate in a magnetic field cancel each other in the initial equilibrium, but not in the LHDI altered equilibrium. Ions are vortices with finite gyroradii. When L is comparable to the ion gyroradius, modifications of the ion density cause modifications of the gradient of the vorticity, which induces shear flow.

4. THE VORTEX METHOD AND MAGNETOHYDRODYNAMICS

Computation of the development of the KHI in a Harris sheet is examined from the point of view of magnetohydrodynamics (MHD), the non-relativistic, collisional limit of the Maxwell–Boltzmann system for charged particles.

A formal connection is often noted between the transport equation for vorticity

$$\frac{\partial \omega}{\partial t} = \nabla \times \mathbf{v} \times \omega \tag{18}$$

and Faraday's law

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times \mathbf{v} \times \mathbf{B} \tag{19}$$

where $\mathbf{v}$ is the flow velocity, ω is the vorticity, and $\mathbf{B}$ is the magnetic induction. They both describe a vector field that is frozen in to a flow. They differ, of course, in that $\mathbf{v}$ is a function of ω but not of $\mathbf{B}$ [25, p. 102].

It is nevertheless possible to develop particle methods for MHD [26, 27]. One defines **B** from point-set data,

$$\mathbf{B} = \sum_p \mu_p S(x_c - x_p) \tag{20}$$

where μ_p is the particle magnetic moment. One then writes Faraday's law in terms of the material derivative,

$$\frac{\mathrm{d}\mathbf{B}}{\mathrm{d}t} = -\mathbf{B}\nabla\cdot\mathbf{v} + \mathbf{v}\mathbf{B}\cdot\nabla + \nabla\cdot\mathbf{B}\mathbf{v} \tag{21}$$

and substitutes Equation (20) into Equation (21) to derive

$$\frac{\mathrm{d}\mu_p}{\mathrm{d}t} = \mu_p\cdot\nabla\mathbf{v} \tag{22}$$

(It is assumed that S is a symmetric function of its arguments, and has dimensions V^{-1}.)

A **B** computed from μ is not solenoidal, i.e. $\nabla\cdot\mathbf{B}\neq 0$. Since a non-physical force along **B** will result from non-solenoidality, the non-solenoidal part of **B** is removed by projection

$$\mathbf{B}' = \mathbf{B} - \nabla\phi \tag{23}$$

where

$$\nabla\cdot\nabla\phi = \nabla\cdot\mathbf{B} \tag{24}$$

The correction is orthogonal to $\mathbf{B}'$

$$\int \nabla\phi\cdot\mathbf{B}'\,\mathrm{d}V = 0 \tag{25}$$

provided either ϕ or $\hat{\mathbf{n}}\cdot\mathbf{B}' = 0$ on the boundary of the domain.

For the MHD calculation of the KHI, the mass density is constant, and the temperature varies as

$$T(z) = T_s\,\frac{1}{\cosh^2(z/L)} \tag{26}$$

The magnetic field is in the plane of the calculation (the $x-z$ plane), and varies with z as

$$B_x(z) = -B_0\,\tanh(z/L) \tag{27}$$

which results in a thin current sheet with current density $J_y = B_0/(L\,\cosh^2(z/L))$. There is an initial flow with positive x-velocity above the current sheet, and negative x-velocity below. The flow is subsonic, $M = 0.5$ and super-Alfvenic, $A = 10$. A strong instability develops.

In general, an adaptive grid can increase the accuracy of a numerical calculation by equidistributing the error. For difference methods, equidistributing the local truncation error increases accuracy significantly [28]. One writes the error measure in the form

$$e(\mathbf{x}) = w(\mathbf{x})g^k \tag{28}$$

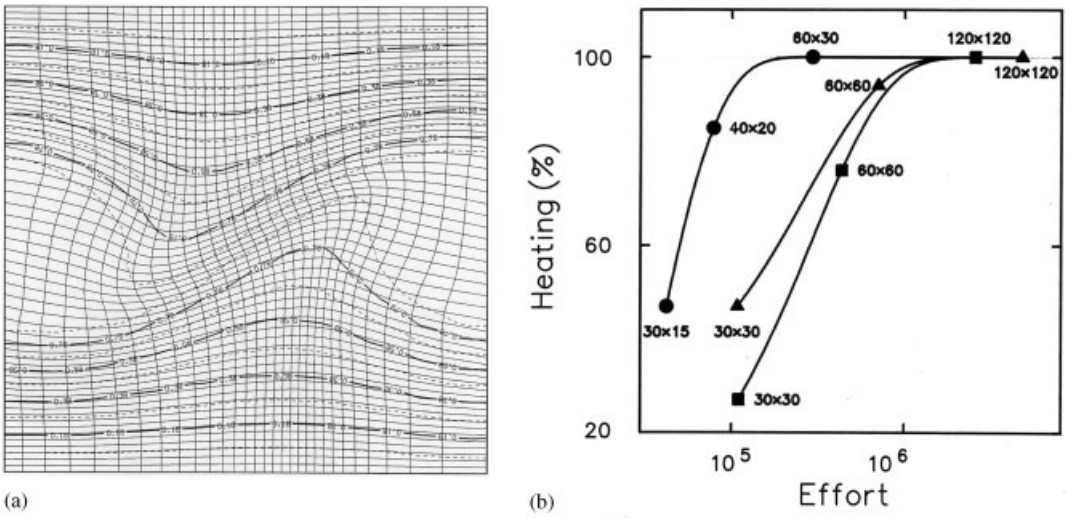

Figure 7. The reconnection caused by a KHI is computed with a PIC MHD model on an adaptive grid. A 60×30 adaptive grid and magnetic field lines are shown (a), and the Ohmic heating that results with various grids is shown (b). The concentration of grid points reflects the weight function, which depends on the current density, J. Convergence is achieved with much less effort with an adaptive grid, circles, than with a uniform grid, squares. The solution on a 60×30 adaptive grid is as accurate as a 120×120 uniform grid.

where g is the determinant of the metric tensor g_{ij} of the transformation from physical to logical co-ordinates, in the variational formulation

$$I = \int w \nabla \xi^{\alpha} \cdot \nabla \xi^{\alpha} \, \mathrm{d}V \tag{29}$$

The solution of this minimization problem yields the equidistribution property expressed in the following theorem.

Theorem 1 (*Error Equidistribution*)
In an optimal grid, defined as a grid that minimizes the local truncation error according to the minimization principle, the product of the local truncation error in any cell i by the cell volume V_i (given by the Jacobian $J = \sqrt{g}$) is constant.

$$e_i V_i = \text{const} \tag{30}$$

For particle methods, there is not only truncation error. There is also consistency error, Equation (13). Thus, a weight function that controls the number of particles per cell, can be as important as truncation error control. In Figure 7 (left) is an overlay of an adaptive computation mesh on a magnetic field line plot. The adaptive mesh is the solution of the generator equations described in Reference [29] with weight function NJ. Here N is the number of particles per cell, and is included in the weight to minimize the consistency error, Equation (13). The heating rates, which are dependent upon resolving the singular currents that develop as a result of the flow, are plotted as a percentage of the correct result in

Figure 7(b). The points are labelled with the number of grid points used, with circles (left curve) denoting results with an adaptive mesh, and squares (right) a uniform mesh.

The truncation error, which is neglected by this choice of weight function, can be expressed as the difference between the linear interpolation of the discretized operator applied to the discretized field, and the exact differential operator applied to the linear interpolation of the discretized field. It would be straightforward to apply to FLIP, where there the recipe above corresponds to comparing FLIP and MPM. Recall that MPM evaluates derivatives analytically Equation (11), and FLIP uses finite differences. Thus, to calculate the truncation error, one simply compares FLIP and MPM computations of the derivatives, e.g. for the computation of the current, J

$$\varepsilon_T = -\sum_p \mu_p \times \nabla_x S(\mathbf{x} - \mathbf{x}_p) + \nabla_x^h \times \sum_p \mu_p S(\mathbf{x} - \mathbf{x}_p) \tag{31}$$

5. CONCLUSIONS

A comparison of PIC, SPH, and vortex-blob methods indicates that the consistency error is dominated by discretization error when initial particle ordering is lost. This error limits accuracy to second order, but can be reduced by increasing the shape function smoothing length at the cost of increased smoothing error. However, the computation of a magnetized, Kelvin–Helmholtz instability gives evidence that control of the local smoothing length and the number of overlapping particles can increase accuracy, even when particles are disordered.

Both PIC and SPH exhibit instability, but for different reasons. SPH is apparently unstable because of the density scaling in the computation of derivatives. PIC is unstable because of the contribution of aliases. Curiously, PIC shares with spectral methods not only its vulnerability to aliasing, but also its stability if boundedness of the solutions in the energy norm is achieved. For PIC, one approach that accomplishes this is implicit differencing in time.

An example of a problem on the boundary of fluid and kinetic behaviour illustrates the potential of particles for related problems in fluid dynamics, such as the computation of flow at extremely small dimensions. Other examples of special capabilities, such as interface tracking, have been successfully exploited.

REFERENCES

1. Harlow FH. The particle-in-cell computing method for fluid dynamics. *Methods in Computational Physics* 1963; **3**:319–343.
2. Birdsall CK, Langdon AB, Okuda H. Finite-size particle physics applied to plasma simulation. In *Methods in Computational Physics*, Alder B, Fernbach S, Rotenberg M (eds), vol. 9. Academic Press: San Diego, 1970; 241–258.
3. Christiansen JP. Numerical simulation of hydrodynamics by the method of point vortices. *Journal of Computational Physics* 1973; **13**:363–379.
4. Chorin AJ, Bernard PS. Discretization of a vortex sheet with an example of roll-up. *Journal of Computational Physics* 1973; **13**:423–429.
5. Leonard A. Vortex methods for flow simulation. *Journal of Computational Physics* 1980; **37**:289–335.
6. Raviart PA. An analysis of particle methods. *Lecture Notes in Mathematics* 1985; **1127**:243–324.
7. Monaghan JJ. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics* 1992; **30**:543–574.
8. Degond P, Mas-Gallic S. The weighted particle method for convection–diffusion equations. Part 1: the case of isotropic velocity. *Mathematics of Computation* 1989; **53**:485–507.

9. Brackbill JU, Forslund DW. Simulation of low-frequency, electromagnetic phenomena in plasmas. In *Multiple Time Scales*, Brackbill JU, Cohen BI (eds), vol. 3. Academic Press: San Diego, 1985; 272–310.
10. Sun Q, Boyd ID, Candler GV. A hybrid continuum/particle approach to modeling subsonic, rarefied gas flows. *Journal of Computational Physics* 2004; **194**:256–277.
11. Brackbill JU, Ruppel HM. FLIP: a method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *Journal of Computational Physics* 1986; **65**:314.
12. Sulsky D, Zhou ZJ, Schreyer HL. Application of the particle-in-cell method to solid mechanics. *Computer Physics Communications* 1995; **87**:236.
13. Perlman M. On the accuracy of vortex methods. *Journal of Computational Physics* 1985; **59**:200–223.
14. Vshivkov VA. The approximation properties of the particles-in-cell method. *Computational Mathematics and Mathematical Physics* 1996; **36**:509–515.
15. Lanson N, Vila JP. Convergence des méthodes particulaires renormalisées pour les systèmes de Friedrichs. *Comptes Rendus de l'Académie des Sciences Paris* 2001; **332**:1.
16. Price DJ, Monaghan JJ. Smoothed particle magnetohydrodynamics-II. Variational principles and variable-smoothing-length terms. *Monthly Notices of the Royal Astronomical Society* 2004; **348**:139–152.
17. Swegle JW, Hicks DL, Atttaway SW. Smoothed particle hydrodynamics stability analysis. *Journal of Computational Physics* 1995; **116**:123–134.
18. Monaghan JJ. SPH without a tensile instability. *Journal of Computational Physics* 2000; **159**:290–311.
19. Brackbill JU. The ringing instability of particle-in-cell calculations of low-speed flow. *Journal of Computational Physics* 1988; **75**:469–492.
20. Lilly DK. On the computational stability of numerical solutions of time-dependent non-linear geophysical fluid dynamics problems. *Monthly Weather Review* 1965; **93**:11–26.
21. Mendez-Nunez LR, Carroll JJ. Comparison of leapfrog, Smolarkiewicz, and MacCormack schemes applied to nonlinear equations. *Monthly Weather Review* 1993; **121**:565–578.
22. Canuto C, Hussaini MY, Quarteroni A, Zang TA. *Spectral Methods in Fluid Dynamics*. Springer: New York, 1989.
23. Grasso F, Meola C. Euler and Navier–Stokes equations for compressible flows: finite-volume methods. In *Handbook of Computational Fluid Mechanics*, Peyret R (ed.). Academic Press: San Diego, 1996; 159–282.
24. Lapenta G, Brackbill JU. Nonlinear evolution of the lower-hybrid-drift instability: current sheet thinning and kinking. *Physics of Plasmas* 2002; **9**:1544–1554.
25. Davidson PA. *An Introduction to Magnetohydrodynamics*. Cambridge University Press: Cambridge, UK, 2001.
26. Brackbill JU. FLIP MHD: a particle-in-cell method for magnetohydrodynamics. *Journal of Computational Physics* 1991; **96**:163–192.
27. Price DJ, Monaghan JJ. Smoothed particle magnetohydrodynamics-I. Algorithm and tests in one dimension. *Monthly Notices of the Royal Astronomical Society* 2004; **348**:123–138.
28. Lapenta G. Variational grid adaptation based on the minimization of local truncation error: time-independent problems. *Journal of Computational Physics* 2003; **193**:159–179.
29. Brackbill JU. An adaptive grid with directional control. *Journal of Computational Physics* 1993; **108**:38–50.
30. Cummins SJ, Brackbill JU. An implicit particle-in-cell method for granular materials. *Journal of Computational Physics* 2002; **180**:506–548.

