JOURNAL OF COMPUTATIONAL PHYSICS **117**, 194–204 (1995)

Particle-Mesh Schemes for Advection Dominated Flows

WAYNE ARTER AND JAMES W. EASTWOOD

Culham Laboratory, Abingdon, Oxon OX14 3DB, United Kingdom

Received July 8, 1991; revised August 22, 1994

Extended particle-in-cell (EPIC) schemes are considered with a view to applications in electrostatic drift-wave turbulence and ordinary hydrodynamical turbulence, where periodic boundary conditions are inappropriate. We treat issues relating to the dual particle-mesh representation and to the need to follow particle orbits accurately. A successful application of EPIC to an advection dominated flow is demonstrated. The errors are quantified so that choosing suitable numerical parameters to obtain a result of a given accuracy is straightforward. © 1995 Academic Press, Inc.

---

1. INTRODUCTION

We shall be concerned with fluid flow problems where the convective term is dominant. The particular application we wish to make is to drift-wave turbulence in plasma confinement devices, especially tokamaks. Our governing equations, derived by simplifying the two-fluid equations (the plasma electrons constitute one fluid, the ions the second), retain many features of the originals, e.g., convective nonlinearities and diffusive terms (see the review by Horton [1]).

It is natural to introduce particles to treat advective terms, while retaining a mesh for other physical effects. The particle-in-cell or PIC scheme [2] was one of the earliest methods, of any type, to be used for hydrodynamical calculations. In two space dimensions and for most drift wave problems a vorticity formulation is possible. The particles then carry vorticity, leading to the vortex methods (see the review by Leonard [3]). There are three principal types: (1) the point-vortex method, where the exact (singular) Green's function is used to obtain the flow from the velocity, (2) the vortex blob method, where the Green's function kernel is smoothed, and (3) the vortex-in-cell method. The first two are essentially meshless, making the inclusion of extra drift wave effects too difficult. Christiansen [4] was the first to study vortex-in-cell, and his code was also used for simple drift calculations [5]. Recent examples of vortex-in-cell calculations are provided by Refs. [6, 7]. In the early 1980s it was realised that treating the mesh from the finite element standpoint could be very profitable. Bardos *et al.* [8] is an early effort in this direction, apparently abandoned in favour of schemes where the particles are generated and removed at each timestep [9]. These, and indeed nearly all schemes using Lagrangian ideas, can be simply treated within the EPIC framework [10, 11], where the "E" stands for "ephemeral" or "extended," depending on whether or not particles retain their identities over more than one timestep.

The extended PIC scheme upon which this paper will focus, involves the idea of a dual basis: a field is discretised both by particles bearing some attribute, e.g, vorticity, and by nodal values of a finite element representation. The two representations can of course be equivalent only in the weak or inner product sense. An example of this kind of equivalence is provided by the charge-assignment step in particle methods [12]. For a finite element basis consisting of chapeau functions, the resulting schemes closely resemble vortex-in-cell when applied to the Euler equation, but differ for drift wave systems.

An important question concerns the treatment of the diffusive terms. The simplest technique is to introduce a random walk into the particle motions [13], but this suffers from a lack of determinism. The other meshless methods are either not applicable, e.g., Leonard's expanding vortex cores [3], or at least as involved [14, 15] as using the mesh to calculate the diffusion, the strategy that we adopt. Hence the particle attributes change as a result of the dissipation and, indeed, due to other physical processes.

The paper is organised as follows: Section 2 sets out the kind of equations we are trying to solve and indicates that for demonstration purposes we can specialise to the Hasegawa–Mima equation. Section 3 outlines the EPIC scheme we shall use. Sections 4 and 5 discusses how to pose a test problem for a scheme involving a dual particle–mesh representation of fields, with Section 5 treating the difficulties of initialising a computation. Results of test calculations and a further analysis of EPIC are given in Section 6. A summary is provided by Section 7.

2. GOVERNING EQUATIONS

It is not the purpose of this paper to dwell on the derivation of these equations (see, e.g., Horton [1]). A representative set is provided by Biskamp and Walter [16], viz.,

$$\frac{d}{dt}(\Phi - \nabla_\perp^2\Phi + \delta_0 D\Phi) = -\delta_n\frac{\partial\Phi}{\partial x} - (\mathbf{b}\cdot\nabla)v_\| - \nu\nabla_\perp^4\Phi, \qquad (1)$$

$$\frac{d}{dt}v_\| = -(\mathbf{b}\cdot\nabla)\Phi + \mu_\|(\mathbf{b}\cdot\nabla)^2 v_\| + \mu_\perp\nabla_\perp^2 v_\|, \qquad (2)$$

0021-9991/95 $6.00

where

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla, \quad \mathbf{u} = \nabla\Phi_{\wedge}\hat{z}, \tag{3}$$

$$\nabla_{\perp} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, 0\right) \quad \text{and} \quad \mathbf{b} = (-\delta_s y, 0, 1). \tag{4}$$

Φ is the electric potential, doubling as a stream function, and $v_{\parallel}$ is the ion velocity parallel to a magnetic field imposed in the direction of $\mathbf{b}$; $\delta_0 D$ is the so-called nonadiabatic operator, δ_n and δ_s correspond to gradients of density and magnetic field, respectively, and ν, $\mu_{\parallel}$, and $\mu_{\perp}$ represent assorted diffusive processes.

The physical implications of (1)–(2) become clearer if we neglect the $(\mathbf{b} \cdot \nabla)$, or shear terms. Equation (2) is then a pure advection–diffusion equation for $v_{\parallel}$. In the adiabatic limit and neglecting spatial variation with z, (1) becomes

$$\frac{d\rho}{dt} = -\delta_n \frac{\delta\Phi}{\delta x} - \nu\nabla^4\Phi, \quad \rho = \Phi - \nabla^2\Phi. \tag{5}$$

In the case $|\nabla^2\Phi| \gg |\Phi|$ and $\delta_n = 0$ we recognise this as the Navier–Stokes equation for vorticity $\omega = -\nabla^2\Phi$. In oceanography and meteorology (5) with $\nu = 0$ is known as the *equivalent barotropic equation*, or the *vorticity equation on the β-plane*. It is customary, though, to make the oceanographic equations dimensionless so that a parameter γ^2 appears, i.e., $\rho = \gamma^2\Phi - \nabla^2\Phi$; γ^{-1} is the horizontal radius of deformation or the Rossby–Obukhov radius, which corresponds to the ion Larmor radius ρ_s in the drift wave case. Since ρ_s is traditionally set to unity in drift wave studies the role of γ^{-1} is played by the domain size L. In plasma physics (5) with $\nu = 0$ is known as the Hasegawa–Mima equation.

The derivation of drift wave equations is complex and involves simplifying assumptions, so that there is always some doubt as to what constitutes an adequate model (e.g., [17]). This complexity has tended to obscure a difficult issue, namely the choice of boundary conditions at the radial boundaries $y = 0, L_y$. Equations (1)–(2) are derived for slab geometry; that is, the intent is to model only a small fraction in minor radius (y) of a toroidal device. x and z correspond to angular co-ordinates; hence, periodic boundary conditions are widely used in those directions. The explicit y-dependence of the shear makes periodicity in y difficult to justify for (1)–(2), although it is acceptable for (5).

In the latter case, in the inviscid limit the most important invariants of the Hasegawa–Mima equations are "momentum,"

$$M = \int \Phi \, dx \, dy; \tag{6}$$

"energy,"

$$E = \frac{1}{2}\int [\Phi^2 + (\nabla_{\perp}\Phi)^2] \, dx \, dy = \frac{1}{2}\int \rho\Phi \, dx \, dy; \tag{7}$$

and "enstrophy,"

$$N = \frac{1}{2}\int [\Phi^2 + 2(\nabla_{\perp}\Phi)^2 + (\nabla_{\perp}^2\Phi)^2] \, dx \, dy = \frac{1}{2}\int \rho^2 \, dx \, dy. \tag{8}$$

M, E, and N are useful for monitoring the accuracy of computations. However, although for hydrodynamical applications the limit of small ν is of interest, the "ergodic" $\nu = 0$ case has a distinctly different energy spectrum [18].

3. NUMERICAL TECHNIQUES

3.1. *Outline*

Our ultimate goal is to solve Eqs. (1)–(2), rather than Eq. (5). Hence, although we shall use the Hasegawa–Mima equation as a test problem, we exclude at once the fully spectral schemes such as those used by Fyfe and Montgomery [18] and McWilliams [19]. In any event, we do not wish to be tied to a restricted set of boundary conditions. In these circumstances, it has become customary in drift-wave calculations to use finite differencing in radius, together with a Fourier representation for any angle-like co-ordinates [16, 20, 21].

An anisotropic representation of this kind is not desirable, if the slab model is to have any kind of validity. The argument for it is apparently that spectral methods are "optimal," so it should be used wherever possible. This claim requires careful examination; in fact spectral methods are best when approximating smooth functions. Advection-dominated flows are good at generating boundary and internal layers, i.e., fields with exactly the kinds of functional dependence that are not well represented by, e.g., Fourier harmonics or Chebychev polynomials. Even in the case of doubly periodic flow, Browning and Kreiss [22] have shown that fourth-order finite differences are equivalent to a spectral method, at least for the viscous Euler equation.

Advection tends to generate fine structure, but shear has a length-scale approaching device size; hence we expect to perform calculations with features at the limits of grid resolution. When working at this limit, we take the view that it is best to be guided by the physics we are seeking to represent. This makes the use of Lagrangian, or particle schemes, very compelling.

The extended PIC scheme can be illustrated by considering (5), rewritten in Lagrangian form as far as possible [23], i.e.,

$$\frac{dF}{dt} = S, \tag{9}$$

where

$$F = \rho + \delta_n y, \quad \rho = \Phi - \nabla^2\Phi, \quad S = -\nu\nabla_\perp^4\Phi. \tag{10}$$

When $S = 0$, F is conserved following a particle trajectory $\dot{\mathbf{x}} = \mathbf{u}$. However, to advance the particles we need to calculate the stream function Φ. We invoke the dual representation

$$\rho(\mathbf{x}) = \begin{cases} \sum_i \rho_i \delta(\mathbf{x} - \mathbf{x}_i) \\ \sum_{\mathbf{p}} \rho_{\mathbf{p}} \psi_{\mathbf{p}}(\mathbf{x}), \end{cases} \tag{11}$$

where $\mathbf{x}_i$, $i = 1, ..., P$, are the positions of the particles, $\delta(\mathbf{x})$ is the Dirac delta function, and $\psi_{\mathbf{p}}(\mathbf{x})$ is the basis function corresponding to node $\mathbf{p}$. Taking the inner product of (11) with $\psi_{\mathbf{q}}(\mathbf{x})$ and using the middle expression from (10) yields

$$\sum_{\mathbf{q}} \Phi_{\mathbf{q}}(\langle \psi_{\mathbf{q}}, \psi_{\mathbf{p}}\rangle + \langle \nabla_\perp \psi_{\mathbf{q}}, \nabla_\perp \psi_{\mathbf{p}}\rangle) = \sum_i \rho_i \psi_{\mathbf{p}}(\mathbf{x}_i). \tag{12}$$

In finite element language $(MM)_{\mathbf{pq}} = \langle \psi_{\mathbf{p}}, \psi_{\mathbf{q}}\rangle$ is the the mass matrix. Since ρ_i may be straightforwardly calculated from F_i, (12) is an equation for the unknown quantity $\Phi_{\mathbf{q}}$. S is taken into account by allowing F_i to vary; integrating (9) over a particle path from time level n to $n + 1$ yields

$$F_i^{n+1} - F_i^n = \int_{t_n}^{t_{n+1}} S(\mathbf{x}_i(t), t)\, dt \tag{13}$$

and the integral may be approximated in any convenient manner.

Solution of (9) by EPIC therefore consists of the following stages:

(A) Initialisation, where particles are introduced at positions $\mathbf{x}_i^0$ with attributes F_i^0 derived from a specified potential $\Phi(\mathbf{x})$, see the discussion in Section 5,

(B1) The flow $\mathbf{u}^n$ is calculated from $\Phi_{\mathbf{p}}^n$ and the particles moved to new positions $\mathbf{x}_i^{n+1}$,

(B2) F_i^{n+1} is calculated using a discrete version of (13), hence ρ_i^{n+1},

(B3) The right-hand side of (12) is generated, in the so-called charge assignment step, and

(B4) Eq. (12) is solved for $\Phi_{\mathbf{q}}^{n+1}$.

The timestep loop (B) is repeated as often as needed.

3.2. *Implementation of EPIC*

At each stage of the timestep loop, several options are available. We shall be guided by the philosophy of choosing the simplest, consistent with producing accurate results. Hence we take $\psi_{\mathbf{p}}$ to be chapeau functions, i.e., piecewise linear functions with nonvanishing gradient, since this is the lowest order conforming element for (12).

If the element nodes are taken to lie on a uniform, two-dimensional rectilinear mesh with spacing h, then $\mathbf{p} = (p, q)$ and (12) may be written in stencil form as

$$\left\{ \frac{h^2}{36} \begin{pmatrix} 1 & 4 & 1 \\ 4 & 16 & 4 \\ 1 & 4 & 1 \end{pmatrix} + \frac{1}{3} \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix} \right\} \Phi_{p,q} = h^2 \rho'_{p,q}, \tag{14}$$

where $\rho'_{\mathbf{p}}$ is defined as the right-hand side of (12); i.e., it is obtained from the ρ_i by the area weighting rule if the $\psi_{\mathbf{p}}$ are chapeau functions. Equation (14) is an elliptic equation that is solved by standard techniques, viz., Fourier transforming in the x direction to reduce the discrete operator to block tridiagonal form and then using the Thomas or factorization algorithm for each tridiagonal system. General three-point conditions at the y = constant boundaries are, with this approach, little harder to impose than zero or periodic boundary conditions.

We note that we could have invoked a dual basis representation for F; cf. Eq. (11). At first sight this is attractive, since the right-hand side of (12) becomes

$$\sum_i F_i \psi_{\mathbf{p}}(\mathbf{x}_i) - \tilde{\delta}_n \sum_{\mathbf{p}} \langle y, \psi_{\mathbf{p}}\rangle, \tag{15}$$

where $\tilde{\delta}_n = \delta_n / n_{pn}$ and the quantity n_{pn} is the number of particles per node. Summing (15) over all the nodes, leads, when the F_i are invariant, to a discrete conservation law for ‘‘momentum.’’ However, on a node-by-node basis, there is no compensation in the second term for fluctuations in the number of particles per cell that affect the first term. Preliminary calculations show the harmful effects of an increased level of noise.

Consistent with our ‘‘simplest first’’ policy, the integrated source term is approximated as

$$\{\tfrac{3}{2} S(\mathbf{x}_i^c, t^n) - \tfrac{1}{2} S(\mathbf{x}_i^c, t^{n-1})\}\, \Delta t, \tag{16}$$

where $\mathbf{x}_i^c = \frac{1}{2}(\mathbf{x}_i^n + \mathbf{x}_i^{n+1})$ is the mid-point of the particle trajectory. Clearly, the chapeau function does not conform for the fourth-order operator, so we replace it by two successive applications of the $\nabla_\perp^2$ stencil viz.,

$$\frac{1}{3h^2} \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix}. \tag{17}$$

Equation (16) is explicit, so we expect a maximum stable timestep scaling as $1/\nu$, but we imagine ν so small that this limit need not concern us.

3.3. *Computing Particle Orbits*

As we shall see, the form of particle pusher in step (B1) is critical. There are two main issues to resolve: (a) how to calculate **u** given Φ, and (b) how to update **x**, given **u**.

Since $\psi_{\mathbf{p}}$ conforms for first derivatives, it is natural to evaluate, e.g., $u_y = \partial\phi/\partial x$ as

$$u_y(\mathbf{x}) = \sum_{\mathbf{p}} \Phi_{\mathbf{p}} \partial\psi_{\mathbf{p}}/\partial x. \tag{18}$$

The representation used by Christiansen [4] has greater spatial dependence, i.e.,

$$u_y(\mathbf{x}) = \sum_{\mathbf{p}} u_{\mathbf{p}} \psi_{\mathbf{p}}(\mathbf{x}), \tag{19}$$

where $u_{\mathbf{p}}$ is formed by differencing $\Phi_{\mathbf{p}}$. This, however, has the useful property of vanishing "self-velocity," corresponding to vanishing self-force in particle codes [12, Section 5-2-3].

(b) is a problem in the stability theory of ordinary differential equations, e.g., Lambert [24]; we have to consider approximately solutions to $\dot{y} = -\lambda y$, where

$$\lambda^2 = \left(\frac{\partial^2\Phi}{\partial x \partial y}\right)^2 - \frac{\partial^2\Phi}{\partial x^2}\frac{\partial^2\Phi}{\partial y^2}. \tag{20}$$

Ideally, explicit formulae should be available to keep the cost down. The possible schemes are tightly constrained by storage requirements (see, e.g., [12, Section 4-7-3, 25, Section 4.3]) that rule out all but the low order multistep schemes, viz., simple Euler and the associated predictor–corrector methods. The relevant amplification factors r are respectively

$$r_{\mathrm{E}} = 1 + \bar{h}, \quad r_{\mathrm{PC}} = 1 + \bar{h} + \bar{h}^2, \tag{21}$$

where $\bar{h} = \lambda\ \Delta t$, and they can crudely be identified with the Courant number (see Section 6). Hence, near a stagnation point with closed stream-lines ($\lambda^2 < 0$), we expect particles to spiral outwards for Euler ($|r_{\mathrm{E}}| > 1$), and inwards for predictor–corrector ($|r_{\mathrm{PC}}| < 1$ for $-1 < i\bar{h} < 1$). Early calculations with these two schemes showed this behaviour and an associated lack of energy conservation. An intriguing possibility is to alternate the two schemes, giving an overall r such that $|r|^2 - 1 = |\bar{h}|^6$ for $\lambda^2 < 0$. Turning to second-order schemes, we have the Runge–Kutta schemes with

$$r = 1 + \bar{h} + \tfrac{1}{2}\bar{h}^2, \tag{22}$$

so that for $\lambda^2 < 0$,

$$|r|^2 - 1 = \tfrac{1}{4}|\bar{h}|^4. \tag{23}$$

High order, low storage, Runge–Kutta schemes have been derived, but are not pursued here on grounds of complexity.

All the above schemes suffer as soon as $|\bar{h}|$, effectively the Courant number, gets large (and if $\bar{h} < -2$ they completely misrepresent the flow near an X-type stagnation point). The trapezoidal rule, at first sight implicit, has the ideal property that $|r| \le 1$ for any $\bar{h}$ and $|r| = 1$ when $\lambda^2 < 0$. Further, it is also symplectic, in the sense that for Φ independent of time, it defines a mapping $(x_n, y_n) \to (x_{n+1}, y_{n+1})$ such that

$$\frac{\partial(x_{n+1}, y_{n+1})}{\partial(x_n, y_n)} = 1; \tag{24}$$

cf. the leapfrog rule for the equation $\ddot{\mathbf{x}} = \mathbf{a}$. Other sympletic schemes are possible [26] but are timestep limited. It is unnecessary to consider them when (18) involves chapeau functions, since the trapezoidal rule may be rearranged to give explicit formulae for (x_{n+1}, y_{n+1}) within each cell. The trapezoidal rule is the method of choice in these rather restricted circumstances, provided the "self-velocity" is not important.

4. CHOICE OF TEST PROBLEM

4.1. *General Issues*

We consider the damped Hasegawa–Mima equation, since it is the simplest advection-dominated system of relevance to drift-waves. We concentrate on the nonlinear case, since a von Neumann stability analysis is easily performed when the particles are fixed and EPIC then behaves like a good second-order scheme. Unfortunately there is a lack of analytic stable nonlinear solutions for testing purposes and no code comparison exercise has been published.

When $\nu = 0$, the modons (see Meiss and Horton [27] and references therein) have little advantage for testing purposes over the simple solution

$$\Phi(x, y, t) = \Phi_0 \sin\frac{2\pi k}{L_x}(x - v_D t)\sin\frac{2\pi l}{L_y}y, \tag{25}$$

where

$$v_D = \delta_n/[1 + (2\pi k/Lx)^2 + (2\pi l/L_y)^2], \tag{26}$$

since both have vanishing nonlinearity. Equation (25) can also be viewed as the linear mode for the case of zero or periodic boundary conditions in y, whose zero growth rate is to be verified. Unfortunately, (25) is unstable, so it only provides a crude check on v_D when Φ is $\mathcal{O}(1)$. It is natural therefore to concentrate on checking the conservation laws.

We also have to decide which parameters, not only mesh-spacing h and time-step Δt, but also number of particles per node n_{pn}, give the best accuracy for the least computational cost. Regrettably, since we have a dual representation for the fields, there are several possible discrete analogues of the definitions of M, E, and N; hence there is uncertainty in how we measure accuracy.

A second approach would be to make a "definitive" computation using very small Δt and h, and compare. We defer this to later work [28].

4.2. *Diagnostics*

Since the accuracy to which the quantities momentum M, energy E, and enstrophy N are conserved is the factor discriminating among the various numerical methods, it is necessary to discuss how these should be computed. In one sense the particle method does better than a spectral scheme, for it has an infinity of invariants, namely $\sum_i F_i^Q$ for any power Q corresponding to the quantities

$$\int (\rho + \delta_n y)^Q \, dx \, dy, \tag{27}$$

preserved when there is no normal flow at $y = 0, L_y$. However, we want to be able to analyse the spatial distributions of conserved quantities, requiring mesh-based definitions. Moreover, there is no analogue of E in the above hierarchy.

The most obvious option is to use finite differences to calculate the gradients in E and N and to sum the resulting mesh quantities. It might be wondered why this is not acceptable, since in the limit of vanishing h, all consistent measures must be the same. But, of course, carrying this reasoning to its logical conclusion, all consistent numerical schemes are the same. The point is that a particle-mesh computation contains more information than just the mesh represents.

Treating momentum M first, most information is used in the definition if we choose

$$M_D = h^2 \sum_{i,\mathbf{q}} \rho_i \psi_{\mathbf{q}}(\mathbf{x}_i). \tag{28}$$

The mesh analogue of (6) is $h^2 \sum_{\mathbf{p}} \Phi_{\mathbf{p}}$, and summing (12) over $\mathbf{p}$ establishes its equivalence with (28), since, excluding surface terms,

$$\sum_{\mathbf{p},\mathbf{q}} \Phi_{\mathbf{p}} \langle \psi_{\mathbf{p}}, \psi_{\mathbf{q}} \rangle = \sum_{\mathbf{p}} \Phi_{\mathbf{p}}. \tag{29}$$

Here, the lumped density $\rho'_{\mathbf{p}}$ plays the role of momentum. Equation (28) implies that

$$M_D = h^2 \sum_i \rho_i. \tag{30}$$

Since each F_i is conserved, we have

$$\rho_i^{n+1} + \tilde{\delta}_n y_i^{n+1} = \rho_i^n + \tilde{\delta}_n y_i^n; \tag{31}$$

therefore

$$M_D^{n+1} - M_D^n = -\tilde{\delta}_n h^2 \left(\sum_i y_i^{n+1} - \sum_i y_i^n \right); \tag{32}$$

i.e., M_D varies because the centroid of the particle distribution moves a distance Δy, and this is an error for zero or periodic y boundary conditions. M_D is thus more revealing than $\sum_i F_i$. Indeed, we have

$$M_D^{n+1} - M_D^n = -\delta_n L_x L_y \Delta y; \tag{33}$$

so in the test problem an error in M_D^n of 0.1 translates to a tiny error in mean position $\Delta y = 10^{-3}$.

The definition of energy E follows the usual practice in particle codes, viz.,

$$E_D = \frac{1}{2} h^2 \sum_i \rho_i \Phi_{\mathbf{p}}(\mathbf{x}_i). \tag{34}$$

Enstrophy N has many more plausible definitions. It is natural to base it upon $\int \rho^2 \, dx \, dy$, viz.,

$$N_{D1} = \frac{1}{2} h^2 n_{pn} \sum_{i,\mathbf{p}} \rho_i^2 \psi_{\mathbf{p}}(\mathbf{x}_i), \tag{35}$$

$$N_{D2} = \frac{1}{2} h^2 \sum_{i,\mathbf{p}} [\rho_i \psi_{\mathbf{p}}(\mathbf{x}_i)]^2, \tag{36}$$

$$N_{D3} = \frac{1}{2} h^2 \sum_{i,\mathbf{p}} \rho_{\mathbf{p}} \rho_i \psi_{\mathbf{p}}(\mathbf{x}_i). \tag{37}$$

The last is not obvious, but it possesses a satisfying symmetry. Remember the dual representation; Eq. (11) implies that

$$\sum_{\mathbf{q}} \rho_{\mathbf{q}} \langle \psi_{\mathbf{q}}, \psi_{\mathbf{p}} \rangle = \frac{1}{2} h^2 \sum_i \rho_i \psi_{\mathbf{p}}(\mathbf{x}_i). \tag{38}$$

Now

$$\begin{aligned} \frac{1}{2} \int \rho^2 \, dx \, dy &\approx \frac{1}{2} \int \sum_{\mathbf{p}} \rho_{\mathbf{p}} \psi_{\mathbf{p}}(\mathbf{x}) \sum_{\mathbf{q}} \rho_{\mathbf{p}} \psi_{\mathbf{q}}(\mathbf{x}) \, dx \, dy \\ &= \frac{1}{2} \sum_{\mathbf{p}} \rho_{\mathbf{p}} \sum_{\mathbf{q}} \rho_{\mathbf{q}} \langle \psi_{\mathbf{q}}, \psi_{\mathbf{p}} \rangle \end{aligned} \tag{39}$$

which, by Eq. (37),

$$\begin{aligned} &= \frac{1}{2} h^2 \sum_{i,\mathbf{p}} \rho_{\mathbf{p}} \rho_i \psi_{\mathbf{p}}(\mathbf{x}_i) \\ &= \frac{1}{2} \int \sum_{\mathbf{p}} \rho_{\mathbf{p}} \psi_{\mathbf{p}}(\mathbf{x}) \sum_i \rho_i \delta(\mathbf{x} - \mathbf{x}_i) \, dx \, dy. \end{aligned} \tag{40}$$

So N_{D3} is equivalent to both a mesh-based and a mixed particle–mesh definition, and is adopted on these grounds.

Last, we remark on the efficient evaluation of M, E, and N by the above formulae. It is natural to treat them as sums over particles, but we observe that for each particle we have to form ρ_i from F_i and interpolate Φ to the particle position. If the same procedure were to be used to calculate total source terms, e.g., from viscosity, the extra computational cost could reach that of the principal timestep loop. Instead we can rearrange the definitions in terms of $\rho'_{\mathbf{p}}$ (a quantity that has to be calculated anyway; see Eq. (14)) to

$$M_D = h^2 \sum_{\mathbf{p}} \rho'_{\mathbf{p}}, \tag{41}$$

$$E_D = \frac{h^2}{2} \sum_{\mathbf{p}} \Phi_{\mathbf{p}} \sum_i \rho_i \psi_{\mathbf{p}}(\mathbf{x}_i) = \frac{h^2}{2} \sum_{\mathbf{p}} \Phi_{\mathbf{p}} \rho'_{\mathbf{p}}, \tag{42}$$

$$N_{D3} = \frac{h^2}{2} \sum_{\mathbf{p}} \rho_{\mathbf{p}} \rho'_{\mathbf{p}}. \tag{43}$$

Hence, the operation count is reduced by an order of magnitude, at the expense of a small increase in store.

5. INITIAL CONDITIONS

5.1 *Starting Procedure*

It is not always appreciated that this is often the most arduous part of writing a particle code. Careful attention has to be paid to this stage, as the test problem selected (Section 6) is inviscid; therefore even the discretised fields will have long memories. Second-order accuracy is required for the particle positions. To minimise further the effect of the starting procedure, we compute backwards to time $t = -\Delta t$.

Euler is the simplest scheme to model the particle motion and can be made to yield second-order accuracy by means of Richardson extrapolation; viz., we compute particle positions $\mathbf{x}_i^{-E}$ and $\mathbf{x}_i^{-F}$ at $t = -\Delta t$ by two steps of $-\frac{1}{2}\,\Delta t$ and a single step of $-\Delta t$, respectively. The quantity

$$\mathbf{x}_i^{-1} = 2\mathbf{x}_i^{-E} - \mathbf{x}_i^{-F} \tag{44}$$

contains no second-order errors.

The resulting algorithm is as follows:

(A1) calculate $\mathbf{x}_i^{-E/2}$ using $\Phi_{\mathbf{p}}$ (given),
(A2) evaluate $\Phi_{\mathbf{p}}^{-E/2}$ from $\rho_i^{-E/2} = F_i - \tilde{\delta}_n\, y_i^{-E/2}$,
(A3) calculate $\mathbf{x}_i^{-E}$ using $\Phi_{\mathbf{p}}^{-E/2}$ and $\mathbf{x}_i^{-E/2}$,
(A4) calculate $\mathbf{x}_i^{-F}$ using $\Phi_{\mathbf{p}}$ (given) and simultaneously form $\mathbf{x}_i^{-1}$,
(A5) evaluate $\Phi_{\mathbf{p}}^{-1}$ from $\rho_i^{-1} = F_i - \tilde{\delta}_n\, y_i^{-1}$.

There is a choice to be made for the initial particle distribution; it is natural to make it spatially uniform, but its alignment with respect to the nodes still has to be specified. The obvious possibilities are *AN*—particles at nodes—and *MW*—nodes midway between particles. *MW* seems preferable for an element-based approach to particle tracking (i.e., the trapezoidal rule), since for *AN* some of the particles sit initially in anomalous positions, viz., exactly on the element boundaries.

We have to specify how the F_i are to be initialised for a given, usually analytic, $\Phi(\mathbf{x})$, which is the most natural way to start a calculation. Thereafter, Φ will be approximated from knowledge of the F_i so the first step is all-important. The field to focus upon is ρ (rather than F), which can obviously be formed by turning (14) around and then inverting the mass matrix to give $\rho_{\mathbf{p}}$. The problem reduces to one of assigning particle values ρ_i, given mesh values $\rho_{\mathbf{p}}$; and there is an obvious rule for this, namely, area weighting in the case of chapeau functions, or generally,

$$\rho_i = \frac{1}{n_{pn}} \sum_{\mathbf{p}} \rho_{\mathbf{p}} \psi_{\mathbf{p}}(\mathbf{x}_i). \tag{45}$$

Problems arise when we determine $\Phi_{\mathbf{p}}$ from the above ρ_i. It turns out that these $\Phi_{\mathbf{p}}$ are not the same as $\Phi(\mathbf{x}_{\mathbf{p}})$, in general. Operator notation makes the problem clearer—we have formed

$$\Phi_{\mathbf{p}} = O^{-1} \cdot AM \cdot AP \cdot MM^{-1} \cdot O\Phi(\mathbf{x}_p), \tag{46}$$

where O is the operator in (14), giving ρ' from Φ, MM is the mass matrix, AP is the operator assigning particles, and AM is the operator assigning back to the mesh. $\Phi_p \neq \Phi(\mathbf{x}_{\mathbf{p}})$, since $AM \cdot AP \neq MM$; in fact,

$$AM \cdot AP = \frac{1}{n_{\mathrm{pn}}} \sum_i \psi_{\mathbf{p}}(\mathbf{x}_i) \psi_{\mathbf{q}}(\mathbf{x}_i). \tag{47}$$

In default of operators for which $AM \cdot AP = MM$ (see Section 5.2) we avoid the discrepancy by discarding $\Phi(\mathbf{x}_{\mathbf{p}})$ in favour of Φ_p.

The one-dimensional version of (47) is instructive. $MM = \langle \psi_p(x), \psi_q(x) \rangle$ yields the well-known stencil $(\frac{1}{6}\ \frac{2}{3}\ \frac{1}{6})$ when ψ is a chapeau function, whereas (54) evaluates to $(\frac{1}{8}\ \frac{3}{4}\ \frac{1}{8})$ for the *AN* case and $(\frac{3}{16}\ \frac{5}{8}\ \frac{3}{16})$ for *MW* if $n_{\mathrm{pn}} = 2$. This is another advantage for *MW*, since $\frac{3}{16}$ is within 13% of $\frac{1}{6}$, whereas $\frac{1}{8}$ is 25% smaller.

5.2. *A Refinement*

The clumsiness of the above procedures ultimately arises because $AM \cdot AP \neq MM$. Since there is considerable freedom in defining AP, it is possible to find an $AP = AP_e$ so that equality holds, as we shall demonstrate.

The derivation for the multi-dimensional case is simplified by noting that AM and MM both factorise into products of one-dimensional operators, so for most purposes it is adequate to consider the problem illustrated in Fig. 1a. $\alpha_1, \ldots, \alpha_J$ are the unknown weights in the case $n_{\mathrm{pn}} = J$, so that particle j is assigned a "mass" $\alpha_j \rho_L + \alpha_{J-j} \rho_C$. Let us assume the ψ_p to be

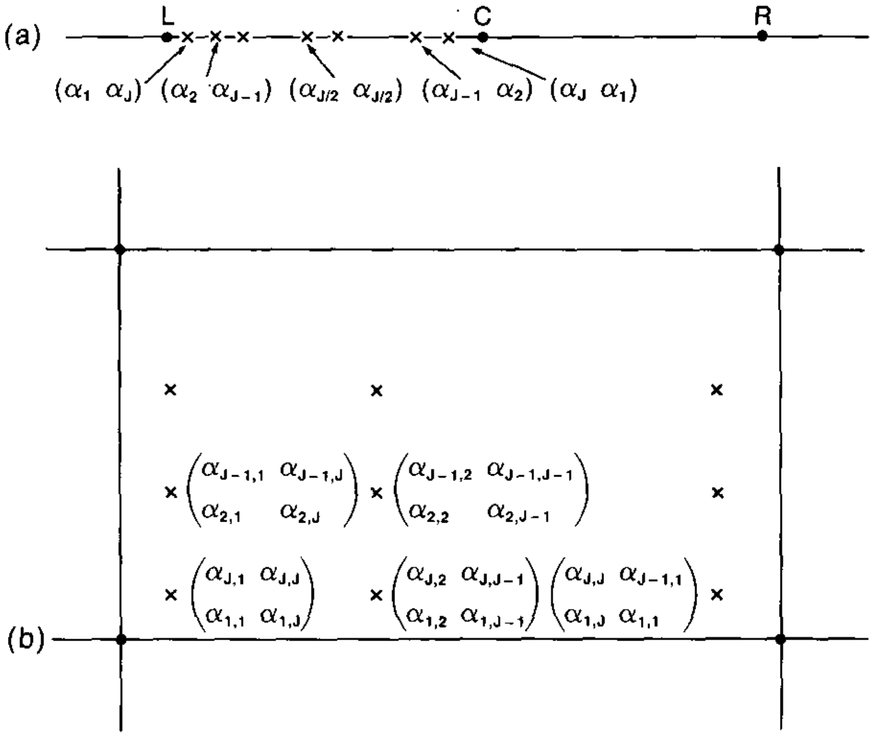

FIG. 1. Assignment of mesh-based information to particles; particle positions are marked with crosses, dots indicate nodes: (a) one-dimensional case, (b) two-dimensional case.

chapeau functions; then assignment back to the mesh is just area-weighting. The mesh value at C receives contributions from particles to either sides; in the MW positions the total turns out to be:

$$\frac{(\rho_L+\rho_R)}{J}\sum_{j=1}^{J}\alpha_j\left(j-\frac{1}{2}\right) + \frac{\rho_C}{J}\left\{\sum_{j=1}^{J}\alpha_j\left(J-j+\frac{1}{2}\right)+\alpha_{J-j+1}\left(j-\frac{1}{2}\right)\right\}. \tag{48}$$

Comparison with $MM\rho$ gives the relations

$$\begin{aligned} -\frac{1}{2}\sum_{j=1}^{J}\alpha_j+\sum_{j=1}^{J}j\alpha_j &= \frac{1}{6}J,\\ 2\left(J+\frac{1}{2}\right)\sum_{j=1}^{J}\alpha_j-2\sum_{j=1}^{J}j\alpha_j &= \frac{2}{3}J. \end{aligned} \tag{49}$$

Evidently any α_j satisfying

$$\sum\alpha_j=\frac{1}{2},\quad \sum j\alpha_j=\frac{J}{6}+\frac{1}{4} \tag{50}$$

will be acceptable and there are similar relations for the AN case. (For ψ_p that are more complicated than chapeau functions, it may help to note that if weights w_i are added to (47), its equality with $\langle\psi_p,\ \psi_q\rangle$ for ψ_p products of a given order is a standard, solved problem in the theory of numerical quadrature.) For the case $n_{pn}=2$, the weights are unique, viz., $\alpha_1=\frac{5}{12}$, $\alpha_2=\frac{1}{12}$. For $n_{pn}=3$, we suggest $\alpha_1=\frac{7}{24}$, $\alpha_2=\frac{1}{6}$, and $\alpha_3=\frac{1}{24}$, since these have the property that for equal values at the two nodes, equal masses are assigned to the three particles.

The two-dimensional case opens up more possibilities than just using the product rule $\alpha_{ij}=\alpha_i\alpha_j$ (see Fig. 1b). Restricting attention to the chapeau ψ_p, there is little freedom in the case $n_{pn}=4$. When $n_{pn}=9$, lengthy but straightforward algebra shows the additional constraint of locally $\sum\rho_i^2=\sum\rho_i\rho_p'$ cannot be met and that the schemes that come closest to meeting it have some weights that are negative. Thus, again, the product rule is to be preferred.

6. RESULTS OF THE TEST PROBLEM

Preliminary calculations showed that the initial conditions need to have a small number of nodes per wavelength of Φ in order to generate errors that are easily distinguishable from the fluctuations that are inherent in any particle scheme. The tendency of solutions to the Hasegawa–Mima equation to cascade energy to long wavelengths suggests that a change of scale will take place if we start with several wavelengths in

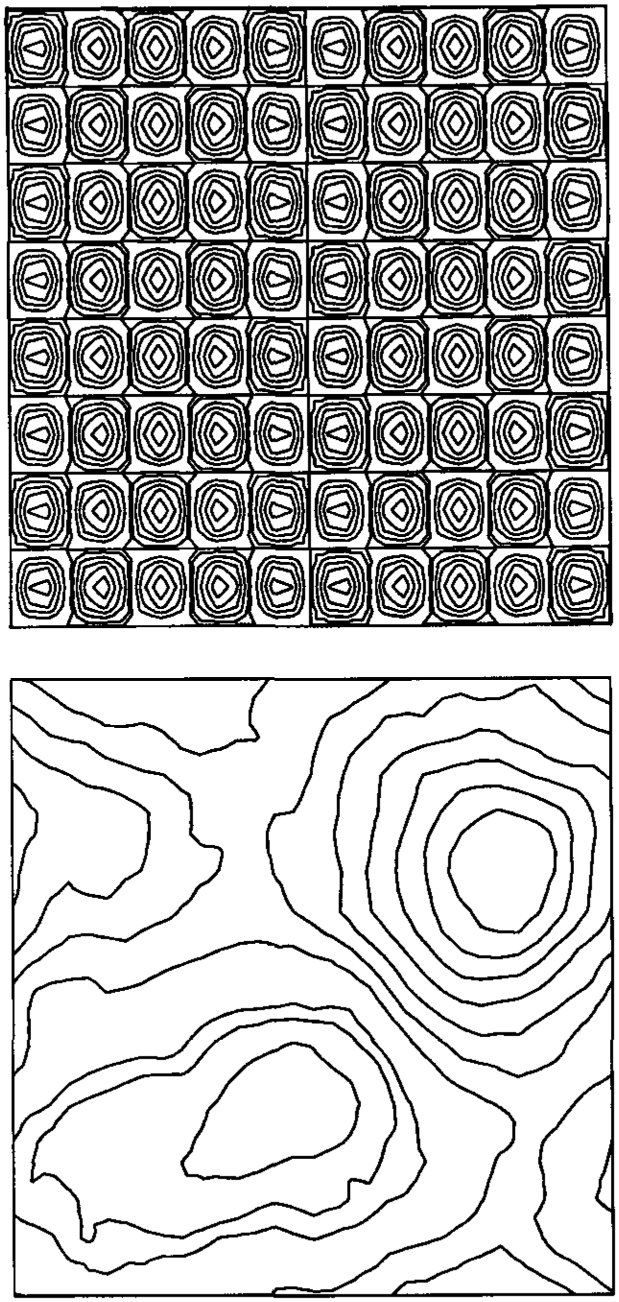

FIG. 2. Potential $\Phi(\mathbf{x}, t)$ for the test problem of Section 6, shown as contours equally spaced in Φ. At top is the initial condition, below lies the computed solution at $t = 0.26$. The parameters used are $n_{\rm pn} = 4$, $h = \frac{1}{32}$, and $c = \frac{1}{2}$, and the particles were tracked using the trapezoidal rule.

our computational domain. We take $L_x = L_y = 1$; i.e., we work in the hydrodynamical limit where $|\nabla^2\Phi| \gg |\Phi|$ and use as initial conditions (25) with $\Phi_0 = 1$, $k = 4$, and $l = 5$. To ensure that $v_D = \mathcal{O}(1)$, we must set $\delta_n = 100$ and, since our main interest attaches to the nonlinear interactions, $\nu = 0$.

The initial conditions and a representative final state at $t = 0.26$ are graphed in Fig. 2. The final $|\Phi|^2$ spectrum has a power law dependence with slope around (-4), as expected [18]. We used the starting procedure described in Section 5.1. The program used the trapezoidal rule for following particle paths. It is worth noting how we calculate Φ at time $t_n + \frac{1}{2}\Delta t$. In effect we have a general multilinear scheme, since Φ can be derived only by extrapolation (in an explicit scheme). The obvious formula is

$$\phi\left(t_n+\frac{1}{2}\Delta t_n\right)=\left(1+\frac{1}{2}\frac{\Delta t_n}{\Delta t_{n-1}}\right)\Phi(t_n)-\frac{1}{2}\frac{\Delta t_n}{\Delta t_{n-1}}\Phi(t_{n-1}), \qquad (51)$$

where $\Delta t_n = t_{n+1} - t_n$ and applying the standard Runge-Kutta error analysis procedure ([24], §4.3) to the whole scheme with $\Delta t =$ constant shows that this gives a fully second-order method. It is, moreover, possible to eliminate error terms $\mathcal{O}(\Delta t^3 \Phi_{ttt})$ by taking (in the case $\Delta t =$ const)

$$\Phi(t+\tfrac{1}{2}\Delta t)=\tfrac{7}{4}\Phi(t)-\Phi(t-\Delta t)+\tfrac{1}{4}\Phi(t-2\Delta t). \qquad (52)$$

This gives a significant improvement in conservation properties at little extra cost. Note that it is computationally far more efficient to extrapolate in time for $\phi(t + \frac{1}{2}\Delta t)$ at the meshpoints, then interpolate Φ at the particle positions.

Applying the trapezoidal rule to $\dot{y} = u_y$ based on (18) with $\psi_{\rm p}$ a chapeau function results in an equation for particle i of the form

$$y_{n+1} - y_n = (a + by_{n+1})\,\Delta t, \qquad (53)$$

where a and b follow from knowledge of $\Phi_{\rm p}$ and y_n and change from one element to another, within an element

$$y_{n+1} = \frac{y_n + a\,\Delta t}{1 - b\,\Delta t}, \qquad (54)$$

and there is a corresponding relation for x_{n+1}. With care at the element boundaries it is possible to arrange it so that $\Phi(x_n, y_n) = \Phi(x_{n+1}, y_{n+1})$ even when particles leap from element to element. In effect the scheme tracks contours of constant $\Phi(t + \frac{1}{2}\Delta t)$.

Figure 3 shows typical time histories of the discrete momentum M_D, energy E_D, and enstrophy N_{D3}. Note that M_D varies irregularly, apparently due to particle noise. E_D changes first quite rapidly at a rate $(\Delta E/t)_1$ and then more slowly at $(\Delta E/t)_2$. N stays almost constant initially; then it drops by about two-thirds.

It must be emphasised that we have arranged it so that there is a gross energy error. Table I indicates the effect of varying the node spacing h, the number of particles per node $n_{\rm pn}$, and the Courant number c on the accuracy of the computation.

The energy error ΔE is expected to increase linearly with time at a rate (see Appendix A for derivation):

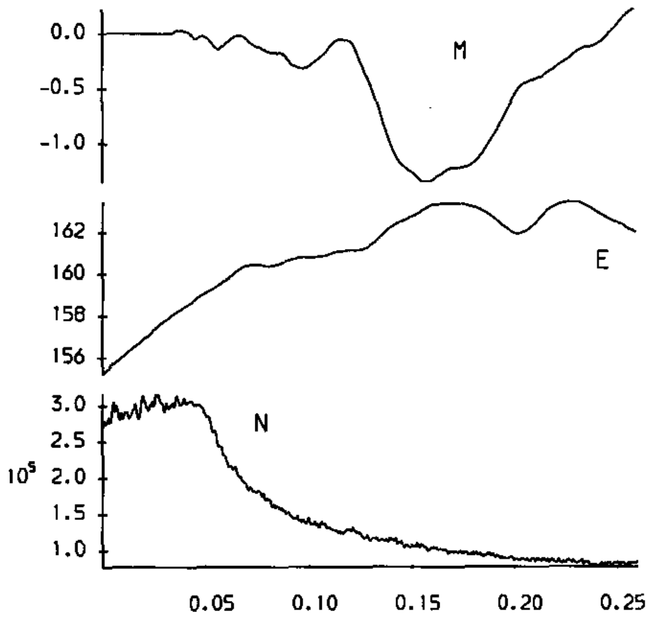

FIG. 3. Plots of momentum, energy, and enstrophy, i.e., M, E, and N as defined by Eqs. (6)–(8), against time for the run illustrated in Fig. 2.

$$\frac{\Delta E}{t}=\frac{2}{3}\frac{\alpha_c}{l}\frac{E}{n_{\mathrm{pn}}}\frac{\overline{\lambda}^3\,\Delta t^3}{h^2}. \tag{55}$$

The computed results fit this formula with $\alpha_c/l \approx 7$. It is worth noting how $|\overline{h}|$ and $c = 2\Delta t \max\{|u_x| + |u_y|\}/h$ are related. Suppose the dominant mode is $\Phi = \sin Kx \sin Ly$—then $|\lambda|_{\max} = |KL|$ and $\max\{|u_x|, |u_y|\} = \max\{K, L\}$. Hence if $K = L$,

$$\frac{c}{|\overline{h}|}=\frac{c}{|\lambda|\,\Delta t}=\frac{2}{Kh}; \tag{56}$$

i.e., their ratio is roughly the number of mesh-points in the dominant wavelength, divided by π. In efficient computations this number will always be of order unity. Equation (55) shows a cubic dependence on c; hence smaller timesteps are favoured. The high power is unfortunate, since it negates a possible asset of EPIC, namely no stability restriction on the maximum c.

TABLE I

Accuracy ($\Delta E/t$) and Computational Cost (Arbitrary Units for a Given Physical Time) for the Particle–Mesh Scheme as Mesh-Spacing h, Courant Number c, and Number of Particles per Node n_{pn} Are Varied

h	c	n_{pn}	$(\Delta E/t)_1$	$(\Delta E/t)_2$	Cost
$\frac{1}{32}$	2	4	2000	1500	7.1
	1	4	275	150	7.9
	$\frac{1}{2}$	4	83	23	14
$\frac{1}{32}$	1	9	180	55	13
	$\frac{1}{2}$	9	36	$\simeq 0$	22
$\frac{1}{64}$	1	9	88	27	111
	$\frac{1}{2}$	9	17	$\simeq 0$	199

Like all particle codes, the computational cost scales with n_{pn}; i.e., particle-based operations predominate. Hence (55) shows there is no relative cost advantage in increasing n_{pn}; $n_{\mathrm{pn}} = 1$ is too noisy, however, so we are left with $n_{\mathrm{pn}} = 4$. It is important to note that (55) shows an error $\propto h$; i.e., the results are relatively insensitive to grid size, as anticipated in Section 1.

The huge error in N deserves explanation, although it is not particularly important for drift waves in a sheared geometry, since there is then no analogue of N. Ultimately it is a consequence of the enstrophy cascade to small scales. Roughly speaking, enstrophy goes to a lengthscale of order of the particle separation, whereupon mesh-based quantities underestimate it. N_{D2} is the worst definition of N in this respect: to illustrate the point, it was recalculated using nodes with separations h, $h/2$, $h/3$, and $h/4$ for a case with $n_{\mathrm{pn}} = 4$. The initial N were 2.07, 2.52, 3.02, and 6.76 × 10^5, respectively. By the end of the runs N had fallen to 0.57 and 1.6 × 10^5 in the first two cases, slightly risen to 3.2 × 10^5 in the third and fallen to 5.3 × 10^5 in the last. N fluctuates widely as the effective h is decreased, so we should not want to use a $h/3$ mesh, say. The point is that the enstrophy information is there, but we cannot see it on the scale of h. Appendix B shows how for an enstrophy distribution with r particles per wavelength we expect a mesh-based definition to underestimate N by a factor $(1 - C/r)$, where $C = 3$ to 4.

7. SUMMARY

We have produced a code that successfully employs an extended PIC, or EPIC, scheme for a hydrodynamics application. This is only one limit of interest to drift-wave turbulence modellers, and, indeed, [29] shows a different EPIC algorithm to be more appropriate in circumstances where the particle self-velocity causes problems.

As well as resolving a surprisingly large number of issues concerning the dual representation, the paper bears out the contention that it is critical to track the particles accurately. In this respect, we have derived error formulae that should have widespread application. Appendix B contains an illustration of how fluctuations due to the use of particles may affect global quantities. There is thus sufficient information provided here to decide whether a particle–mesh scheme is a suitable choice for a particular application.

We believe that the results of this paper show EPIC to be a powerful method for solving advection-dominated problems. Indeed for incompressible flow, Bermejo [30] has shown an ephemeral PIC scheme to be equivalent to the semi-Lagrange

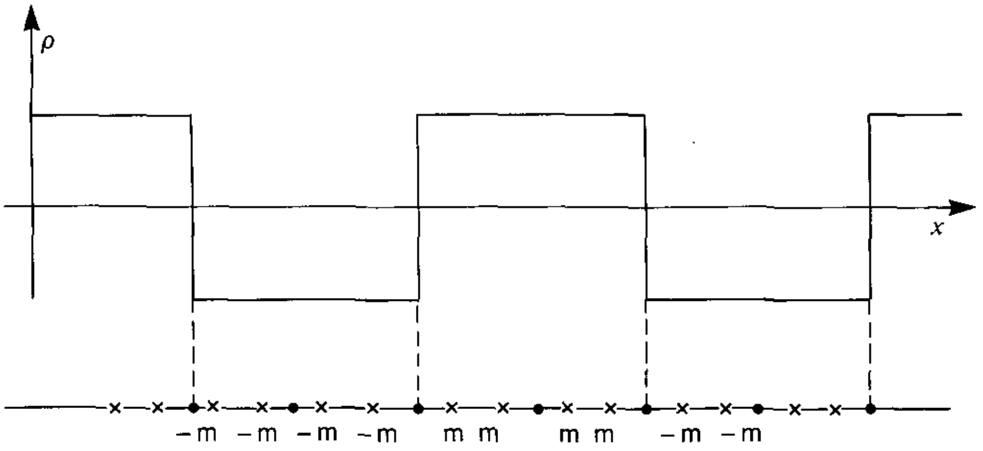

FIG. 4. The periodic piecewise constant density distribution as a function of position x (top). Below it are crosses indicating particle positions and dots marking the nodes, showing the case $s = 4$, *MW(OS)*.

methods that are state-of-the-art in weather forecasting, and, of course, EPIC naturally extends to compressible flow also. Ultimately, of course, only a comparison with other schemes will establish its superiority [28].

APPENDIX A: DERIVATION OF ENERGY ERROR FORMULA

It is relatively easy to estimate the energy error ΔE in the case of scalar advection, i.e., no feedback of the scalar onto the flow, since it is due solely to errors in the particle positions $\Delta\mathbf{x}$. Again, finite element nodes are assumed to lie on a uniform, two-dimensional rectilinear lattice of spacing h. If area-weighting is employed, the change in the mesh value of the lumped scalar ρ'_p at a nearby node is

$$\Delta\rho_p = \rho_i(x\,\Delta y + y\,\Delta x)/h^2; \tag{57}$$

$4n_{pn}$ particles contribute to ρ'_p. The mean absolute value of x and y is $\frac{1}{2}h$; if the mean absolute value of Δx and Δy is ε, then assuming a random distribution for the products $y\,\Delta x$ and $x\,\Delta y$, on average,

$$|\Delta\rho_p| \approx \frac{2\varepsilon\rho_i}{h}\,n_{pn}^{1/2} \approx \frac{2\varepsilon\rho_p}{h n_{pn}^{1/2}}. \tag{58}$$

We use the definition of energy

$$E = \frac{h^2}{2}\sum_p \rho_p\Phi_p; \tag{59}$$

hence,

$$\Delta E = \frac{h^2}{2}\sum_p (\rho_p\,\Delta\Phi_p + \Delta\rho_p\Phi_p) + \mathcal{O}(\Delta^2) \approx h^2\sum_p \Delta\rho_p\Phi_p. \tag{60}$$

Now (60) has to be evaluated carefully, since the contributions $\Delta\rho_p$ cancel in the case where Φ_p is constant; hence,

$$\Delta E \approx \alpha_C h^2 \sum_p \Delta\rho_p h|\nabla\Phi_p|, \tag{61}$$

$$\approx \alpha_C h^2 \sum_p \frac{2\varepsilon\rho_p}{n_{pn}^{1/2}}\frac{\Phi_p}{l}, \tag{62}$$

where l is the lengthscale of Φ_p and α_C is the correlation between $\nabla\Phi$ and the perturbed particle positions. So

$$\frac{\Delta E}{E} \approx \frac{\alpha_C}{l}\frac{4}{n_{pn}^{1/2}}\varepsilon. \tag{63}$$

TABLE II

Normalised Enstrophy N'_{D2} for Periodic, Piecewise Constant Density Distribution with $r = 2s$ Particles per Wavelength and $n_{pn} = 4$

r	Case	N'_{D2}
2	Any	0
4	*AN*	$\frac{1}{4}$
	MW(OS)	0
	MW(SS)	$\frac{1}{4}$
$2s$, s odd	*AN*	$1 - 3/r$
	MW	$1 - 15/(4r)$
$2s$, s even	*AN*	$1 - 3/r$
	MW(OS)	$1 - 4/r$
	MW(SS)	$1 - 7/(2r)$

The above formula may be too optimistic if ε is taken to be the truncation error of the difference scheme, since it neglects feedback effects; the deflection of a particle by $\Delta \mathbf{x}$ and consequent error $\Delta \rho_\mathrm{p}$ affects the velocity potential. It is then also more plausible that $\alpha_C \neq 0$. Thus an initial error ε leads to a positional error at the next step, such that

$$|\Delta \mathbf{x}| = |\nabla \Delta \Phi_\mathrm{p}| \, \Delta t. \tag{64}$$

Assuming that $\Delta \Phi_\mathrm{p} \propto \Delta \rho_\mathrm{p}$, we estimate an enhancement factor $2\Phi_\mathrm{p}/(h^2 n_\mathrm{pn}^{1/2})$; i.e., if $\Phi_\mathrm{p} = \mathcal{O}(1)$,

$$\frac{\Delta E}{Et} \approx \frac{8\alpha_C}{l} \cdot \frac{\varepsilon}{h^2 n_\mathrm{pn}}. \tag{65}$$

This formula fits the data of Table I if

$$\alpha_C / l \approx 7, \tag{66}$$

since $\varepsilon = \frac{1}{12}\lambda^3 \Delta t^3$ for the trapezoidal rule. If we take $l = \max(|\Phi| / |\nabla \Phi|)$, then $\alpha_C \approx 0.22$, which is reasonable, if a little high.

APPENDIX B: THE ENSTROPHY PROBLEM

Imagine that the density ρ is distributed in a piecewise constant fashion as depicted in Fig. 4. Suppose there are $r = 2s$ particles uniformly distributed in each wavelength and each particle is assigned mass $\pm m$, depending on its position relative to the density field. Let there be two particles per node (corresponding to $n_\mathrm{pn} = 4$ in the two-dimensional case), and use area weighting to assign to the nodes. The total enstrophy or ρ^2 in a domain with n nodes is evidently $n(2m)^2$ if we consider n and s sufficiently large. The case of s extremely small highlights the problem, for if $s = 1$ and the particles are distributed either with one at a node (*AN*) or one at a quarter-node (*MW*), then values assigned to the mesh are zero; i.e., $N = 0$ using definition N_{D2}. When $s = 2$, we can make explicit how N can fluctuate. For the *AN* case, $N/(4nm^2) = N' = \frac{1}{4}$, and likewise for the *MW* case when the nodes are between particles of the same sign (*SS* case). But when the node lies between particles of opposite sign (*OS* case), cancellation again occurs and $N = 0$.

Evidently as s increases there will always be some undersampling, where the density distribution jumps. The dependence on s is worth studying, since any particle quantity measured on a mesh will be affected by undersampling and fluctuations in the same sort of manner, although usually less dramatically. It turns out that the cases s even and s odd need separate treatment, and both *OS* and *SS* cases give the same n when s is odd. Table II lists the results of some straightforward algebra. Finally note that for $n_\mathrm{pn} = 1$, $N' = 1$ for the *AN* case and $N' = 1 - 2/r$ for the case of particles midway between nodes; i.e., N' is on average higher when there are fewer particles per node as the N_{D2} results in Section 6 demonstrate.

ACKNOWLEDGMENTS

We thank Dr. J. W. Connor for stimulating and encouraging this work. Work performed under the UKAEA/Euratom Fusion Association.

REFERENCES

1. W. Horton, *Phys. Rep.* **192,** 1 (1990).
2. F. H. Harlow, *Methods Comput. Phys.* **3,** 319 (1964).
3. A. Leonard, *J. Comput. Phys.* **37,** 289 (1980).
4. J. P. Christiansen, *J. Comput. Phys.* **13,** 363 (1973).
5. J. B. Taylor and B. McNamara, *Phys. Fluids* **14,** 1492 (1972); J. P. Christiansen and J. B. Taylor, *Plasma Phys.* **15,** 585 (1973).
6. P. A. Smith and P. K. Stansby, *J. Fluid Mech.* **194,** 45 (1988).
7. J. M. R. Graham, in *Numerical Methods for Fluid Dynamics III,* edited by K. W. Morton and M. J. Baines (Clarendon Press, Oxford, 1988), p. 310.
8. C. Bardos, M. Bercovier, and O. Pironneau, *Math. Comput.* **36,** 119 (1981).
9. O. Pironneau, *Numer. Mat.* **38,** 309 (1982).
10. J. W Eastwood, in *Astrophysical Radiation Hydrodynamics,* edited by K.-H. A. Winkler and M. L. Norman (Reidel, Dordrecht, 1986), p. 415.
11. J. W. Eastwood and W. Arter, in *Numerical Methods for Fluid Dynamics II,* edited by K. W. Morton and M. J. Baines (Clarendon Press, Oxford, 1986), p. 581.
12. R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (Adam Hilger, Bristol, 1988).
13. A. Chorin, *J. Fluid Mech.* **57,** 785 (1973).
14. P. A. Raviart, in *Numerical Methods for Fluid Dynamics II,* edited by K. W. Morton and M. J. Baines (Clarendon Press, Oxford, 1986), p. 231.
15. P. Degond and S. Mas-Gallic, *Math. Comput.* **53,** 485 (1989).
16. D. Biskamp and M. Walter, *Phys. Lett. A 109,* 34 (1985).
17. K. H. Spatschek, E. W. Laedke, C. Marquardt, S. Musher, and H. Wenk, *Phys. Rev. Lett.* **64,** 3027 (1990).
18. D. Fyfe and D. Montgomery, *Phys. Fluids* **22,** 246 (1979).
19. J. C. McWilliams, *J. Fluid Mech.* **146,** 21 (1984).
20. B. D. Scott, *J. Comput. Phys.* **78,** 114 (1988).
21. R. E. Waltz, *Phys. Fluids* **31,** 1962 (1988).
22. G. L. Browning and H.-O. Kreiss, *Math. Comput.* **52,** 369 (1989).
23. N. J. Zabusky and J. C. McWilliams, *Phys. Fluids* **25,** 2175 (1982).
24. J. D. Lambert, *Computational Methods in Ordinary Differential Equations* (Wiley, London, 1973).
25. C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics* (Springer-Verlag, New York, 1988).
26. A. Thyagaraja and F. A. Haas, *Phys. Fluids* **28,** 1005 (1984).
27. J. D. Meiss and W. Horton, *Phys. Fluids* **26,** 990 (1983).
28. W. Arter, Comparison of schemes for drift-wave turbulence modelling, in preparation, 1991 (unpublished).
29. W. Arter, *Comput. Phys. Commun.* **79,** 381 (1994).
30. R. Bermejo, *Mon. Weather Rev.* **118,** 979 (1990).