1994ApJ...425..171T

THE ASTROPHYSICAL JOURNAL, 425:171–194, 1994 April 10

# NUMERICAL STUDY OF TWO-FLUID C-TYPE SHOCK WAVES

GÁBOR TÓTH
Princeton University Observatory, Princeton, NJ 08544
*Received 1993 August 18; accepted 1993 October 18*

## ABSTRACT

C-type magnetohydrodynamic shocks in the partially ionized interstellar medium are studied in the two-fluid approximation. The ionized fluid can move along the magnetic field lines, while it interacts with the neutral fluid via ion-neutral elastic scattering. I use an explicitly flux-conserving two-dimensional Eulerian flux-corrected transport code to study the dynamics of two-fluid shocks. A numerical instability intrinsic to two-fluid problems was discovered, and a solution to the problem is proposed. The code can successfully simulate C-type shocks. The results of the linear stability analysis by Wardle are confirmed, and the nonlinear behavior of the instability is explored.

*Subject headings:* methods: numerical — MHD — shock waves

## 1. INTRODUCTION

There are several mechanisms driving shocks into the interstellar medium (ISM): supernova explosions, strong stellar winds, and collisions of molecular clouds. The partially ionizd ISM can be treated as two distinct fluids coupled by the ion-neutral friction. When the fractional ionization is small, the neutral fluid carries most of the density and inertia, but only the ion fluid interacts with the dynamically important magnetic field. If the shock speed $v_s$ is lower than the ion Alfvén speed $v_A^{(i)} \equiv B/(4\pi\rho^{(i)})^{1/2}$, where $B$ is the magnetic field strength and $\rho^{(i)}$ is the ion density, the ions can build up a magnetic precursor ahead of the shock (Draine 1980), and the ion flow remains continuous. There is an ion-neutral slip (ambipolar diffusion) throughout the shock front, thus—in the "standard shock frame," where the shock front is at rest—the ions are decelerating the neutral fluid running into the shock at a speed of $v_s$. If the neutral fluid remains cold due to the weakness of the shock, or because of effective cooling, the neutral flow will be supersonic everywhere and the flow variables will vary continuously, hence the shock is called C-type. If the neutral fluid reaches a temperature so that the flow becomes subsonic and (as it cools down behind the shock) supersonic again, the shock will be either J-type, containing a jump in the neutral flow variables at the supersonic-subsonic transition, or C*-type (Roberge & Draine 1990), having two continuous sonic points in the neutral flow. I will focus on numerical studies of C-type shocks; the methods presented here may well work for the other types too, but they are beyond the scope of this paper. Draine, Roberge, & Dalgarno (1983) conclude that shocks with $v_s \lesssim 25$ km s$^{-1}$ in a gas with $10^2$ cm$^{-3}$ ambient density and $10^{-4}$ fractional ionization are C-type. The maximum velocity rises to around 50 km s$^{-1}$ in denser gas with lower ionization (Smith & Brand 1990).

The physical and observational significance of C-type shocks is discussed in many papers; I refer the reader to references given by Wardle (1990, 1991a, b) and an excellent review article by Draine & McKee (1993). It was recognized by Wardle that while the models of the chemistry and physics of C-type shocks assumed a steady state planar shock front, in reality all but the weakest shocks are subject to an instability analogous to the Parker instability. The magnetic field lines can buckle across the shock front and the ions flow along the field lines due to the force of ion-neutral friction. The ion density increases in the troughs (i.e., in the bends closer to the downstream flow), thus the neutral drag will be stronger at these points than at the crests of the magnetic field, and the field lines will bend further. The linear analysis caried out by Wardle showed that there will be growing modes with wavelength on the order of the shock thickness $L_{\rm flow}$ and with an $e$-folding time that can be much shorter than the flow time $t_{\rm flow}$ through the shock. Therefore, the steady state models are of dubious validity, and a fully dynamical simulation is called for to model the physics of C-type shocks which fulfill the instability criterion. Another consequence of the instability is the formation of high-density lumps of the neutral fluid, a possible place for low-mass $\sim 0.1\ M_\odot$ star formation. One needs, however, to know the nonlinear behavior of the instability to check whether these speculations are correct. There is, of course, a possibility that the instability saturates at a low amplitude, or that the necessary approximations in the linear analysis make the analytic results very different from the solutions of the exact equations; these questions give further motivation for numerical simulations.

I set out to study numerically the dynamics of two-fluid shocks. The present paper concentrates on the numerical methods and problems related to the integration of the two-fluid differential equations. A few test cases will be presented, both to confirm the results of the linear analysis and to show that the code can simulate cases unavailable for analytic calculations. A more thorough exploration of the parameter space will be presented in a subsequent paper in the near future. Here I restrict myself to study the stability of isothermal shocks with plane-parallel initial conditions. The plane-parallel approximation depends on $L_{\rm flow} \sim 0.01$ pc being much smaller than the curvature radius of the shock front, which is often satisfied. For the initially cold gas, isothermality is equivalent to the assumption that the neutral pressure is negligible compared to the magnetic pressure throughout the shock front due to the effective cooling. These simplifications are made to reduce the number of free parameters and to facilitate comparisons with the analytical results, although the numerical model is not limited by them. On the other hand, the code neglects dynamical effects of grains, chemical reactions, and ionization.

1994ApJ...425..171T

Ionization is negligible at $v_s < 50$ km s$^{-1}$, while chemical reactions probably do not change the dynamics appreciably. Inclusion of chemistry would become unavoidable if the purpose was to predict and compare line emissions or column densities to observations. The dynamical role of grains is controversial; much depends on the size distribution of grains, which is a poorly known parameter (Elmegreen & Fiebig 1993). For fractional ionization $x_e \gtrsim 10^{-4}$, as in the simulations presented in this paper, grains are unimportant in any case. For the sake of simplicity these effects are omitted from our numerical model.

Below I briefly review the analytic results regarding the steady state solution and linear perturbation of C-type shocks (Wardle 1990, 1991a, b and references therein). The main physical parameters characterizing the structure of the shock are the shock speed $v_s$; the neutral density $\rho^{(n)}$; the fractional ionization $x_e \equiv n^{(i)}/n_{\rm H}$, where $n^{(i)}$ and $n_{\rm H}$ are the ion and hydrogen number densities, respectively; the rate coefficient for ion-neutral elastic scattering $\langle \sigma v \rangle \approx 2 \times 10^{-9}$ cm$^3$ s$^{-1}$; and the strength and direction of the magnetic field $\boldsymbol{B}$ in the ambient medium. A shock is called perpendicular when the angle between the magnetic field and the normal to the shock front is $\theta_s = 90°$ upstream, and it is oblique when $\theta_s < 90°$. The steady state solution for a perpendicular shock involves motions perpendicular to the shock front only, while oblique shocks with velocity components parallel to the shock front are more complex due to their lower degree of symmetry. The temperature of the ambient medium $T \approx 20$ K has no influence on the dynamics because of the isothermal assumption. Assuming that the helium number density is 10% of the hydrogen number density, $\rho^{(n)} = 1.4 M_{\rm H} n_{\rm H}$, where $M_{\rm H}$ is the mass of the hydrogen atom. Typical values in the ISM are $n_{\rm H} = 10^2$–$10^6$ cm$^{-3}$, $B = 10$–$10^3$ $\mu$G, and $x_e = 10^{-4}$ to $10^{-8}$, with scalings $n_{\rm H}/{\rm cm}^{-3} \approx (B/\mu{\rm G})^2 \approx 10^{-10} x_e^{-2}$ except for the lowest density "diffuse" medium, where $x_e \approx 10^{-4}$. The scaling laws imply $L_{\rm flow} \approx 1.4 \times 10^{17} (10^{-7}/x_e)$ cm with geometric factors of order unity for oblique shocks and $t_{\rm flow} \approx 5 \times 10^{11} (10^{-7}/x_e)$ s. The neutral Alfvén speed in the unshocked medium is $v_{\rm A}^{(n)} \equiv B/(4\pi\rho^{(n)})^{1/2} \approx 2$ km s$^{-1}$, while the ion Alfvén speed is $v_{\rm A}^{(i)} \approx 1200(10^{-7}/x_e)^{1/2}$ km s$^{-1}$, and the compression ratio is $r_f \approx \sqrt{2} v_s / v_{\rm A}^{(n)}$ from the balance of neutral ram pressure upstream and magnetic pressure downstream.

In the limit of very small fractional ionization there are two dimensionless numbers that determine the stability of the shock: the neutral Alfvén number $A^{(n)} \equiv v_s / v_{\rm A}^{(n)}$ in the ambient medium, and the angle $\theta_s$. For perpendicular shocks Wardle predicts instability if $A^{(n)} \gtrsim 5$, and he finds that oblique shocks with the same $A^{(n)}$ are even less stable than their perpendicular counterparts. Note that $\langle \sigma v \rangle$ does not appear in the stability condition; it determines the length and timescales of the solution only. In the linear analysis Wardle assumes that ion pressure and ion inertia are negligible, which holds if $\rho^{(i)} (v^{(i)})^2 \ll B^2/(4\pi)$, i.e., the ion Alfvén number $A^{(i)} \equiv v_s / v_{\rm A}^{(i)} \ll 1$. In the diffuse medium with $n_{\rm H} \approx 10^2$ cm$^{-3}$, $x_e \approx 10^{-4}$, and a weak magnetic field $B \approx 5$ $\mu$G, the ion Alfvén speed is $v_{\rm A}^{(i)} \approx 20$ km s$^{-1}$; thus the condition is only marginally satisfied. We shall see, however, that the numerical results are very close to the analytical predictions.

The rest of the paper is organized as follows: In § 2 the differential equations of two-fluid magnetohydrodynamics are presented with discussion of various algebraic approximations. Section 3 describes the numerical algorithms used to solve the fluid equations. Several improvements are proposed to the flux-conserving implementation of the flux-corrected transport (FCT) finite-difference scheme. Section 4 discusses a few tests of the code, the setting up of initial conditions, and the results of sample simulations of C-type shocks. The results and conclusions are summarized in § 5.

## 2. EQUATIONS

The fluid equations are written in terms of the conservative variables: $\rho^{(f)}$, $m_j^{(f)}$, $e^{(f)}$, and $B_j$, the mass momentum, and energy densities of the fluid $f$, and the magnetic field strength, respectively. The superscript $f$ is either $i$ for ions or $n$ for neutrals, while the lower index $j$ denotes any of the three spatial variables $x$, $y$, and $z$. The momenta are simply $m_j^{(f)} \equiv \rho^{(f)} v_j^{(f)}$. The energy densities contain kinetic, thermal, and magnetic contributions:

$$e^{(n)} \equiv \frac{1}{2}\, \rho^{(n)} (v^{(n)})^2 + \frac{1}{\gamma^{(n)} - 1}\, p^{(n)} , \tag{2.1a}$$

$$e^{(i)} \equiv \frac{1}{2}\, \rho^{(i)} (v^{(i)})^2 + \frac{1}{\gamma^{(i)} - 1}\, p^{(i)} + \frac{1}{8\pi}\, B^2 , \tag{2.1b}$$

where $p^{(f)}$ and $\gamma^{(f)}$ are the thermal pressures and adiabatic indices, respectively. The conservation equations for mass, momentum, and energy, and the Maxwell equation for the magnetic field, are

$$\partial_t \rho^{(i)} + \partial_k [m_k^{(i)}] = 0 , \tag{2.2a}$$

$$\partial_t m_j^{(i)} + \partial_k [v_k^{(i)} m_j^{(i)} - (1/4\pi) B_k B_j] + \partial_j [p_{\rm tot}^{(i)}] = \mathscr{F}_j^{(i)} , \tag{2.2b}$$

$$\partial_t e^{(i)} + \partial_k [v_k^{(i)} (e^{(i)} + p_{\rm tot}^{(i)}) - (1/4\pi) B_k B_j v_j^{(i)}] = \mathscr{E}^{(i)} , \tag{2.2c}$$

$$\partial_t B_j + \partial_k [B_j v_k^{(i)} - B_k v_j^{(i)}] = 0 , \tag{2.2d}$$

$$\partial_t \rho^{(n)} + \partial_k [m_k^{(n)}] = 0 , \tag{2.2e}$$

$$\partial_t m_j^{(n)} + \partial_k [v_k^{(n)} m_j^{(n)}] + \partial_j [p^{(n)}] = -\mathscr{F}_j^{(i)} , \tag{2.2f}$$

$$\partial_t e^{(n)} + \partial_k [v_k^{(n)} (e^{(n)} + p^{(n)})] = -\Lambda(\rho^{(n)}, p^{(n)}) - \mathscr{E}^{(i)} , \tag{2.2g}$$

where summation over repeated indices is implied. In the numerical calculations the velocities and the pressures are derived from the conservative variables rather than the other way around; thus the velocities are $v_j^{(f)} \equiv m_j^{(f)} / \rho^{(f)}$, while the pressures are defined by the inverse of equations (2.1) as

$$p^{(i)} \equiv (\gamma^{(i)} - 1) \left( e^{(i)} - \frac{1}{2}\, \rho^{(i)} v_j^{(i)} v_j^{(i)} - \frac{1}{8\pi}\, B_j B_j \right) , \tag{2.3a}$$

$$p_{\rm tot}^{(i)} \equiv p^{(i)} + \frac{1}{8\pi}\, B_j B_j , \tag{2.3b}$$

$$p^{(n)} \equiv (\gamma^{(n)} - 1) \left( e^{(n)} - \frac{1}{2}\, \rho^{(n)} v_j^{(n)} v_j^{(n)} \right) . \tag{2.3c}$$

The source terms $\mathscr{F}_j^{(i)}$, $\mathscr{E}^{(i)}$, and $\Lambda$ are the drag force and energy transfer between the two fluids, and radiative cooling, respectively. Conservation of momentum and energy implies that $\mathscr{F}_j^{(n)} = -\mathscr{F}_j^{(i)}$ and $\mathscr{E}^{(n)} = -\mathscr{E}^{(i)}$, as it is explicitly used above. I note that $\mathscr{G}^{(f)} \equiv \mathscr{E}^{(f)} - \mathscr{F}_j^{(f)} v_j^{(f)}$ is the frictional heating plus heat exchange through elastic scattering. Although the code can handle any form of cooling function, in this paper I shall use $\gamma^{(n)} = 1.001$ instead, which effectively implies an isothermal neutral gas, i.e., a very efficient cooling. This simplification allows easier comparison with the analytic results as well as a reduction of the number of free parameters. I take $\gamma^{(i)} = 5/3$ for the ions. The drag force and energy transfer are

$$\mathscr{F}_j^{(i)} = \alpha\rho^{(i)}\rho^{(n)}(v_j^{(n)} - v_j^{(i)}) , \tag{2.4a}$$

$$\mathscr{E}^{(i)} = \frac{\alpha}{M^{(i)} + M^{(n)}} [\rho^{(i)}\rho^{(n)}(v_j^{(n)} - v_j^{(i)})(v_j^{(n)}M^{(n)} + v_j^{(i)}M^{(i)}) + 3(p^{(n)}\rho^{(i)}M^{(n)} - p^{(i)}\rho^{(n)}M^{(i)})] , \tag{2.4b}$$

where

$$\alpha \equiv \frac{\langle\sigma v\rangle}{M^{(i)} + M^{(n)}} \approx 3.7 \times 10^{13}\ \mathrm{cm^3\ s^{-1}\ g^{-1}} \tag{2.4c}$$

is the coupling constant and $M^{(i)} \approx 30M_{\mathrm{H}}$ and $M^{(n)} = (7/3)M_{\mathrm{H}}$ are the mean ion and neutral particle masses, assuming that $n_{\mathrm{He}}/n_{\mathrm{H}} = 10\%$ and the hydrogen is fully molecular. In diffuse molecular clouds, where the dominant ions are $\mathrm{C}^+$ and $\mathrm{H}^+$ (Draine & Katz 1986), $M^{(i)} \approx 10M_{\mathrm{H}}$ and $\alpha \approx 9.7 \times 10^{13}$ cm$^3$ s$^{-1}$ g$^{-1}$.

In some cases it proved to be useful for the numerical stability of the calculations to replace the differential equation (2.2c) for the ion energy density by an approximate algebraic equation. As Chernoff (1987) pointed out, the ion heat capacity is low, thus the heating rate must be $\mathscr{G}^{(i)} \approx 0$, i.e., the ions are heated to an equilibrium temperature

$$T^{(i)} = T^{(n)} + \frac{1}{3k_{\mathrm{B}}} M^{(n)}(v^{(i)} - v^{(n)})^2 , \tag{2.5}$$

where $k_{\mathrm{B}}$ is the Boltzmann constant. The ion energy density is therefore determined from $p^{(i)} = (k_{\mathrm{B}}/M^{(i)})\rho^{(i)}T^{(i)}$ and equation (2.1b). The energy transfer, needed for the neutral energy density equation, simplifies to $\mathscr{E}^{(i)} = \mathscr{F}_j^{(i)}v_j^{(i)}$. Numerical tests confirmed that the approximation hardly changes the other flow variables, while the temperature remains smoother than if it was integrated from the differential equation. I used the algebraic approximation in the simulations presented in this paper, but several check runs were done solving the full system of differential equations.

A further algebraic approximation is possible for the ion momentum equation. For low fractional ionization the dominant terms in equation (2.2b) are the gradients of the magnetic field and the drag force; ion pressure and momentum are negligible if $A^{(i)} \gg 1$ (Wardle 1991b). The balance between the dominant forces is

$$\mathscr{F}_j^{(i)} = \partial_j\left[\frac{1}{8\pi} B_k B_k\right] - \partial_k\left[\frac{1}{4\pi} B_k B_j\right] . \tag{2.6}$$

Ion velocity can be expressed as $v_j^{(i)} = v_j^{(n)} - \mathscr{F}_j^{(i)}/(\alpha\rho^{(i)}\rho^{(n)})$, and $m_j^{(i)} = \rho^{(i)}v_j^{(i)}$. While this simplification turned out to be useful for the linear analysis, and in principle could reduce the amount of computation in a numerical simulation, I found that the approximation produced large numerical errors, probably due to the infinite ion Alfvén speed implied by the tight coupling of $B_j$ and $v_j^{(i)}$; thus, at least with our numerical code, I could not make use of it.

## 3. NUMERICAL METHOD

There are several methods in use to solve the hydrodynamic equations (e.g., Woodward & Colella 1984). To choose the best option, one needs to consider the specific application. Our simulations involve shocks, magnetic fields, two fluids interacting via ion-neutral scattering, and two spatial dimensions. Lagrangian methods were excluded, because they are difficult to implement in two dimensions. The existence of shocks prohibits the use of the simplest methods, like the Lax-Wendroff scheme (Lax & Wendroff 1960), since in some cases even artificial viscosity (Lapidus 1967) is unable to make the calculations numerically stable (see Brio & Wu 1988, p. 414, or Woodward & Colella 1984, p. 146). Many of the most advanced methods are based on solving the nonlinear Riemann problem exactly (Godunov 1959) or approximately (Roe 1981) for each cell. These are well studied and developed for hydrodynamical simulations, but the first paper on extending the approximate Riemann solver method to magnetohydrodynamics has appeared only recently (Brio & Wu 1988). There seems to be no easy way to generalize the exact Riemann solution to the magnetic case, and especially not in two dimensions. The usual operator splitting approximation may sacrifice some of the benefits one hoped to gain from a highly advanced and complicated scheme. Operator splitting makes it especially difficult to keep the magnetic field divergence-free, since changes in one sweep should be canceled exactly by the other sweep in the other coordinate direction.

The method of choice was the FCT scheme (Book, Boris, & Zalesak 1981), which offers an explicitly two-dimensional Eulerian difference scheme on an optionally nonuniform rectangular grid. It is known to work well for magnetohydrodynamic shock waves, and has a version that can conserve the divergence of the magnetic field to the accuracy of numerical truncation errors (DeVore 1991). It is relatively easy to code but sufficiently powerful for our purposes. One disadvantage in comparison with some other methods is the appearance of small oscillations ("ringing") behind sharp features like shock waves. Fortunately the C-type shocks of primary interest here contain no sharp jumps in any variable. The original realization of the magnetic flux-conserving constrained transport (CT) algorithm by Evans & Hawley (1988), which uses the van Leer (1979) monotonic method, seems to perform somewhat worse than the implementation of the CT approach to FCT (DeVore 1991). A very recent approach (Stone & Norman 1992), method of characteristics–constrained transport (MOC-CT), may be comparable to or better than DeVore's FCT. The well-known alternative to the CT-type algorithms is the use of the vector potential as a flow variable, and obtaining a divergence-free magnetic field from it. Although it is an elegant and simple idea, it requires accurate calculation of the second derivatives of the potential (needed for the first derivatives of the field), thereby essentially reducing the order of accuracy by 1.

### 3.1. *The FCT Method*

The FCT method is based on the idea of using a high-accuracy (second or higher order) difference scheme when it is possible, but applying a first-order diffusive solution where the higher order method is bound to break down, for example at shock waves, where the flow variables change dramatically from cell to cell and second-order approximations fail. Actually, the prescription is more sophisticated than that: the difference between the high-order–accurate and the low-order–diffusive solutions, the "antidiffusive flux," is multiplied by a coefficient between 1 and 0 before being added to the low-order solution. Where the antidiffusive flux is fully applied, the final solution will be second-order or higher order accurate, while at other places where the antidiffusive fluxes are "corrected" (i.e., the coefficients are less than unity), the result is a more diffusive but stable solution. The correction method (Zalesak 1979) aims to preserve the stability and positivity of the diffusive solution, so that no new minima and maxima may be created by the application of the antidiffusive flux. This criterion proved to be the most powerful in deciding whether the second-order corrections make the solution more accurate or introduce huge errors. Note that the diffusive solution may show some "ringing," but the flux correction stops further amplification of this finite amplitude and stable error.

The FCT method represents shock waves with steep gradients of the flow variables over a few cells; typically the shocks are smeared over three to five cells. Although this is clearly different from the exact step functions, and the error at a given cell near the shock front can be large, the overall jump conditions and the propagation speed of the shock wave will faithfully reproduce the analytical solution. The key is that FCT, like most of the popular hydrodynamic algorithms, explicitly conserves mass, momentum, and energy. The fluid equations are rewritten in terms of mass, momentum, and energy densities, and these conserved flow variables are stored for each cell in the simulation. Numerical fluxes of these variables are calculated at the cell boundaries and added to and subtracted from the flow variables of the two adjacent cells; thus the sum is conserved to the accuracy of machine round-off errors. The artificially introduced diffusive and antidiffusive fluxes are applied the same way. Generally there can be source terms in the equations as well: cooling and heating that change the energy density, or the drag force between the two fluids, which acts like a source of momentum density in the momentum equations for each fluid. Source terms are represented in the cell centers and are added to the variables of the same cell.

In two dimensions the differential equation for a flow variable $w$ (which can be $\rho^{(f)}$, $m_j^{(f)}$, $e^{(f)}$, or $B_z$) has the following form:

$$\frac{\partial w}{\partial t} + \frac{\partial F_x[w]}{\partial x} + \frac{\partial F_y[w]}{\partial y} = S[w] . \tag{3.1}$$

In FCT first the fluxes and sources are applied (transported stage):

$$\begin{aligned} w^T[i, j] = w[i, j] + \Delta t(F_x[i - \tfrac{1}{2}, j] - F_x[i + \tfrac{1}{2}, j] \\ + F_y[i, j - \tfrac{1}{2}] - F_y[i, j + \tfrac{1}{2}] + S[i, j]) . \end{aligned} \tag{3.2}$$

The edge-centered fluxes are calculated from the flow variables of the neighboring cells as described in the next section. For the sake of simplicity I assume a uniform grid with $\Delta x = \Delta y = 1$ cell. Next the diffusive fluxes are calculated from the flow velocities and the gradients of $w$ at the cell boundaries (diffusive stage) and added to the solution:

$$\begin{aligned} w^D[i, j] = w^T[i, j] + F_x^D[i - \tfrac{1}{2}, j] - F_x^D[i + \tfrac{1}{2}, j] \\ + F_y^D[i, j - \tfrac{1}{2}] - F_y^D[i, j + \tfrac{1}{2}] . \end{aligned} \tag{3.3}$$

Finally the antidiffusive fluxes are calculated ($F_x^A$ and $F_y^A$), then corrected according to the monotonicity criterion ($F_x^C$ and $F_y^C$), and added to $w^D$:

$$\begin{aligned} w^{\text{new}}[i, j] = w^D[i, j] + F_x^C[i - \tfrac{1}{2}, j] - F_x^C[i + \tfrac{1}{2}, j] \\ + F_y^C[i, j - \tfrac{1}{2}] - F_y^C[i, j + \tfrac{1}{2}] . \end{aligned} \tag{3.4}$$

The calculations of the fluxes $F_x$ and $F_y$ are analogous to their physical equivalents in the differential equation. The diffusive and uncorrected antidiffusive fluxes are defined as

$$F_x^D[i + \tfrac{1}{2}, j] = +(\tfrac{1}{6} + \epsilon_x^2[i + \tfrac{1}{2}, j]/3)(w[i + 1, j] - w[i, j]) , \tag{3.5a}$$

$$F_x^A[i + \tfrac{1}{2}, j] = -(\tfrac{1}{6} - \epsilon_x^2[i + \tfrac{1}{2}, j]/6)(w^T[i + 1, j] - w^T[i, j]) , \tag{3.5b}$$

where $\epsilon_x[i + \frac{1}{2}, j] = v_x[i + \frac{1}{2}, j](\Delta t/\Delta x)$, and analogous expressions apply to the $y$ components. Note that the artificial diffusive and antidiffusive fluxes are, unlike the physical fluxes, not multiplied by $\Delta t$ when applied, but their total effect approaches zero for $\Delta t \to 0$ as expected.

The one-step FCT algorithm can easily be modified to a two-step, time-centered procedure. In the first half-step the flow variables are advanced by $\Delta t/2$, and fluxes and source terms are calculated from them. Next, starting from the original flow variables, a full $\Delta t$ step is made, but this time the time-centered fluxes are used. Formally:

$$\begin{aligned} \{w[t], F[t], S[t]\} \to \{w[t + \Delta t/2], F[t + \Delta t/2], S[t + \Delta t/2]\} , \\ \{w[t], F[t + \Delta t/2], S[t + \Delta t/2]\} \to \{w[t + \Delta t]\} . \end{aligned} \tag{3.6}$$

The maximum size of the time step $\Delta t$ is determined by the Courant condition, which basically says that no signal should propagate more than about one cell in one time step. In MHD the fast magnetosonic waves have the highest speed, thus

$$\Delta t \le \Delta t_C \equiv C \min\left(\frac{\Delta x}{|v_x| + c_{\text{fast}}}, \frac{\Delta y}{|v_y| + c_{\text{fast}}}\right) , \tag{3.7}$$

where $c_{\text{fast}}^2 = c_{\text{sound}}^2 + v_A^2$. In the one-step method the Courant number has to be $C < 0.5$; usually 0.3–0.4 is taken. In the time-centered version $C = 0.6$ may be used safely.

### 3.2. *Modifications to the Original FCT Algorithm*

In the original descriptions of FCT the edge centered fluxes are calculated from edge centered flow variables,

$$F_x[i + \tfrac{1}{2}, j] = F_x\left(\frac{w[i + 1, j] + w[i, j]}{2}\right) . \tag{3.8}$$

This requires the centering of all fluid variables in both the $x$ and $y$ directions at every time step, since they are given and calculated at the cell centers only. Ryu (1991) has invented and thoroughly tested a simple modification that uses less CPU time and memory while having the same accuracy. First notice that one can calculate the fluxes at the cell centers and interpolate to the cell edges afterward:

$$F_x[i + \tfrac{1}{2}, j] = \frac{F_x(w[i + 1, j]) + F_x(w[i, j])}{2} . \tag{3.9}$$

The real gain is that there is no need for the centering at all, since only the gradients of the fluxes are needed in equation (3.2), and they can be obtained as

$$\begin{aligned} F_x[i + \tfrac{1}{2}, j] - F_x[i - \tfrac{1}{2}, j] \\ = \frac{F_x(w[i + 1, j]) - F_x(w[i - 1, j])}{2} . \end{aligned} \tag{3.10}$$

This is a second-order approximation to $\partial_x F_x(x[i])$, just like the original formula. Note that this simplification cannot and should not be used for the diffusive and antidiffusive fluxes.

Another change due to Ryu is the replacement of the transported $w^T$ by the diffused $w^D$ flow variable in the definition of the uncorrected antidiffusive fluxes in equation (3.5b). Numerical experiments show a rather slight difference in performance, and probably in favor of Ryu's method. With the modification it becomes unnecessary to calculate and store the transported flow variable $w^T$ separately; one can add the diffusive fluxes right away in equation (3.2). The high-order solution, when no flux correction is done, differs only by a term fourth order in $\Delta x$ and $\Delta y$ from the original method.

### 3.3. *Solving for the Magnetic Fields in FCT*

The $x$ and $y$ components of the magnetic field can in principle be handled the same way as the other flow variables. The results achieved this way can be satisfactory, but the presence of divergence in the calculated magnetic field is a disturbing sign. A detailed analysis of this problem can be found in the description of the constrained transport method by Evans & Hawley (1988). The solution outlined in their paper relies on the idea of placing the magnetic fields to the cell edges rather than to the cell centers. Fluxes, the $(\boldsymbol{v} \times \boldsymbol{B})$ terms, are calculated at the cell corners and are added to and subtracted from the $B_x$ and $B_y$ components sitting on the four edges that meet at the corner. The numerical equivalent of the divergence of the magnetic field at the cell center is

$$\nabla \cdot \boldsymbol{B} = \frac{B_x[i+\frac{1}{2}, j] - B_x[i-\frac{1}{2}, j]}{\Delta x} + \frac{B_y[i, j+\frac{1}{2}] - B_y[i, j-\frac{1}{2}]}{\Delta y} . \tag{3.11}$$

The fluxes, $F_x[B_x] = F_y[B_y] = 0$ and $F \equiv F_y[B_x] = -F_x[B_y] = v_y B_x - v_x B_y$ (see eq. [2.2d]), are applied as

$$B_x^T[i+\tfrac{1}{2}, j] = B_x + \Delta t(F[i+\tfrac{1}{2}, j-\tfrac{1}{2}] - F[i+\tfrac{1}{2}, j+\tfrac{1}{2}]) ,$$
$$B_y^T[i, j+\tfrac{1}{2}] = B_y - \Delta t(F[i-\tfrac{1}{2}, j+\tfrac{1}{2}] - F[i+\tfrac{1}{2}, j+\tfrac{1}{2}]) . \tag{3.12}$$

By substituting equation (3.12) in equation (3.11), it can be easily shown that $\nabla \cdot \boldsymbol{B}^T = \nabla \cdot \boldsymbol{B}$; thus the numerical divergence is conserved if the (physical as well as diffusive and antidiffusive) fluxes are all added as described here.

Numerical experience showed that straightforward application of the CT-type magnetic field and flux centering to the original FCT algorithm results in an excessively diffusive solution. DeVore invented a modified version of FCT to deal with the magnetic fields. The basic idea is to evolve the two components $B_x$ and $B_y$ together through the transport, diffusive, and antidiffusive stages, while keeping track of the flux contributions from the two fields separately, i.e., the fluxes are split into two partial fluxes: $F = F^{B_x} + F^{B_y}$. The partial antidiffusive fluxes are corrected separately, thus the diffusion is reduced. Even though the monotonicity is enforced less strictly this way, numerical tests by DeVore show that the solution remains well behaved.

As in the case of other flow variables, I found it beneficial to use the diffused rather than the transported magnetic field as a basis for the uncorrected antidiffusive flux. I also want to point to a minor error in DeVore's paper: the definition following equation (A9) has wrong superscripts and should probably read

$$B^{cy}_{x\,ij+1/2} = \tfrac{1}{4}(B^{cy}_{x\,i-1/2j} + B^{cy}_{x\,i-1/2j+1} + B^{cy}_{x\,i+1/2j} + B^{cy}_{x\,i+1/2j+1}) . \tag{3.13}$$

The equal weighting of the $i - \frac{1}{2}$ and $i + \frac{1}{2}$ centered terms is not really accurate for a nonuniform grid, since the distance from the $i$ center to the $i - \frac{1}{2}$ edge is not necessarily equal to the distance to the $i + \frac{1}{2}$ edge (the edges, on the other hand, exactly bisect the distances between cell centers). This inaccuracy is, however, not serious, since cell sizes should not change more than a few percent from cell to cell, and the terms involving $B^{cy}_{x\,ij+1/2}$ have small coefficients and are of second or higher order anyway.

### 3.4. *Numerical Instability of Two-Fluid Hydrodynamics*

There is an intrinsic numerical instability related to the friction between the neutral and ion fluid. To demonstrate the mechanism for this instability, let us take a simple one-dimensional steady state flow with both fluids at constant densities and temperatures, and at the same constant velocity. The magnetic field can be taken to be homogeneous too.

Imagine that there is a slight staggered perturbation of the ion velocity, i.e., in every second cell the ion momentum is slightly higher, and in every other cell it is slightly lower than the equilibrium value. The neutral momentum is changed accordingly but with opposite sign; thus the total momentum is conserved. The perturbation has no effect on the flux gradients, since they are calculated as differences (eq. [3.2] or eq. [3.10]). Only the source terms dependent on the velocities change, most importantly the drag forces in the momentum conservation equations for the ion and neutral fluids:

$$\frac{\partial}{\partial t} m^{(i)} + \frac{\partial}{\partial x} F[m^{(i)}] = \alpha \rho^{(i)} \rho^{(n)} (v^{(n)} - v^{(i)}) . \tag{3.14a}$$

$$\frac{\partial}{\partial t} m^{(n)} + \frac{\partial}{\partial x} F[m^{(n)}] = \alpha \rho^{(i)} \rho^{(i)} (v^{(i)} - v^{(n)}) . \tag{3.14b}$$

Dropping the spatial derivatives and rewriting the right-hand-sides, I obtain a differential equation for the $\delta m^{(i)} = -\delta m^{(n)}$ perturbation:

$$\frac{\partial}{\partial t} \delta m^{(i)} = -\alpha(\rho^{(i)} + \rho^{(n)}) \delta m^{(i)} . \tag{3.15}$$

The analytical solution is, of course, an exponential decay, but with an explicit finite-difference method the numerical solution may easily be overstable, if the time step is not chosen carefully. The critical dimensionless parameter is

$$D \equiv \alpha(\rho^{(i)} + \rho^{(n)}) \Delta t . \tag{3.16}$$

The situation is similar for any two-fluid flow with the velocities perturbed in either a one- or a two-dimensional staggered pattern, as shown in Figure 1. The amplification of the error by the two-step FCT algorithm is plotted as a function of $D$ in Figure 2 (see Appendix A for derivation). The difference scheme is stable for $D < D_{\text{stable}} \approx 1.3$. However, $D < D_{\text{damp}} \approx 0.8$ is required for damping, and in the range from $D_{\text{damp}}$ to $D_{\text{stable}}$ the perturbation is neither amplified nor damped. Strong gradients introduce large errors which can trigger the instability; thus damping is preferred to marginal stability. $D_{\text{damp}} \approx 0.95$ for the one-dimensional and $\approx 0.75$ for the two-dimensional instability in case of the modified algorithm (Ryu 1991). The original FCT algorithm gives $D_{\text{damp}} \approx 1.2$ in the one-dimensional case and $D_{\text{damp}} \approx 0.85$ in two-dimensions.

The time step has to be the same for all cells; thus the condition for the stability of the FCT algorithm becomes

$$\Delta t \le \Delta t_D \equiv \frac{D_{\max}}{\alpha \max(\rho^{(i)} + \rho^{(n)})} , \tag{3.17}$$

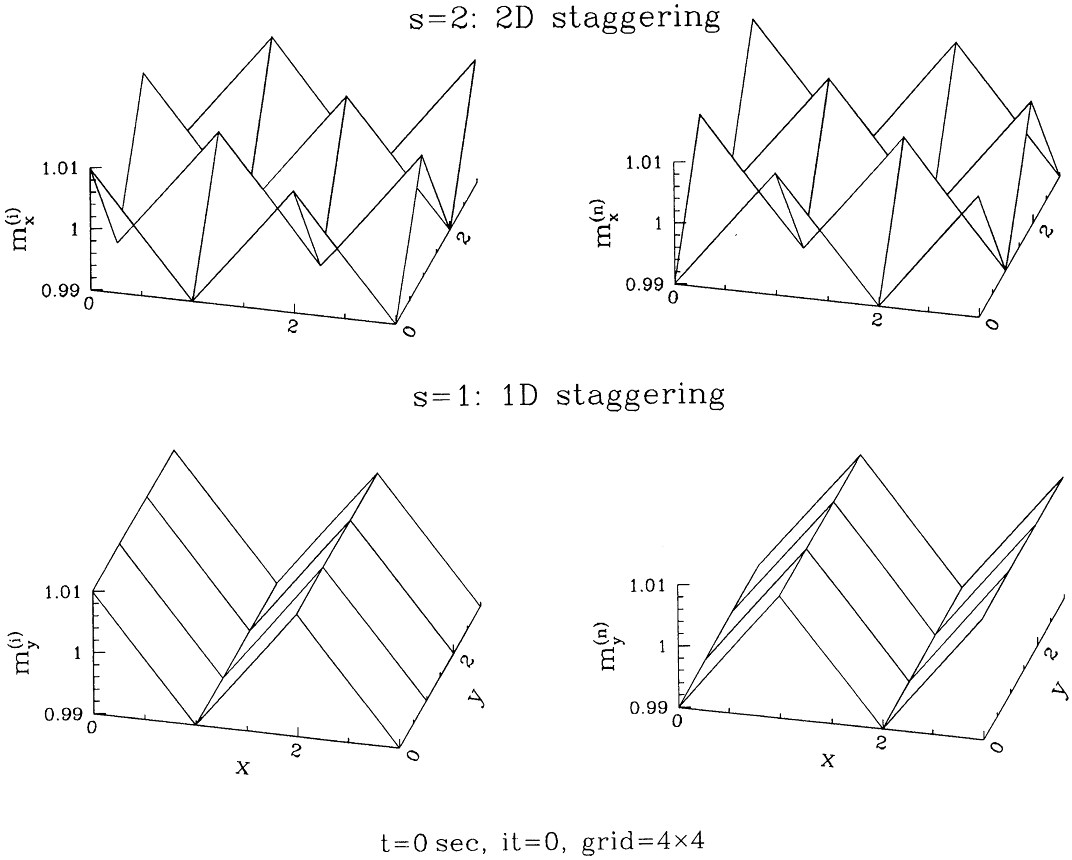


FIG. 1.—Staggered instability arises if the computational time step is too long compared to the ion-neutral scattering time. A one-dimensional (*bottom*) or two-dimensional (*top*) staggered perturbation with equal amplitudes but opposite signs for the ion (*left*) and neutral (*right*) momenta can grow exponentially. Other variables, like density, energy density, or the magnetic field, remain constant. The $x$, $y$, and $z$ components of the momenta are subject to the instability independently of each other. The figure shows a specially prepared test for the FCT code; in actual simulations the instability starts to grow at sharp edges, where the errors have small wavelength components, then spreads over the whole area where the conditions for its growth are satisfied.

where the maximum is taken over all cells in the simulation, and $D_{\max}$ is analogous to the $C$ parameter for the Courant condition (eq. [3.7]). The time step is set to the smaller of $\Delta t_C$ and $\Delta t_D$.

It should be noted that in two dimensions the modified algorithm can in principle be overstable for $0 < D < 0.1$ in the extreme case, when both $v_x^{(i)}$ and $v_y^{(i)}$ reach the maximum value allowed by the Courant condition with $C = 0.6$ and $c_{\text{fast}} = 0$, and the total density is less than one-tenth of the maximum density. This results from the overshooting by the artificial diffusive flux as explained in Appendix A. In the simulations presented in this paper the fast magnetosonic speed always exceeds the ion flow speed; thus $v_x$ and $v_y$ are at most half of the maximum value, e.g., $v_x < 0.3\Delta x/\Delta t$. This fact itself ensures marginal stability at the low-density regions, but in typical cases, where $v_x$ and/or $v_y$ is small, the instability will be damped everywhere.

### 3.5. *Improving the Numerical Stabiliity of the FCT Algorithm*

A small but effective change in the FCT algorithm largely eliminates the staggered instability. Notice that the problem arises from the different methods used for calculating the flux gradients and the source terms. The flux gradients are computed from the two neighboring cells (or four in two dimensions), while the source terms are limited to the cell itself, for which the equation is being solved. Thus the staggered perturbation cancels for the flux terms but affects the source terms in the equations.

I propose to calculate the source terms as a weighted average with the inclusion of neighboring cells. In one dimension

$$\langle S[i] \rangle = \tfrac{1}{4}S[i-1] + \tfrac{1}{2}S[i] + \tfrac{1}{4}S[i+1] \tag{3.18}$$

will cancel the effect of the staggered instability, while in two dimensions

$$\begin{aligned}\langle S[i,j] \rangle = {} & \tfrac{1}{4}S[i,j] \\ & + \tfrac{1}{8}(S[i-1,j] + S[i+1,j] \\ & + S[i,j-1] + S[i,j+1]) \\ & + \tfrac{1}{16}(S[i-1,j-1] + S[i+1,j-1] \\ & + S[i+1,j+1] + S[i-1,j+1])\end{aligned} \tag{3.19}$$

is the only nine-point weighting scheme (see Appendix B for a proof) that eliminates the effects of both the one- and the two-dimensional staggered perturbations.

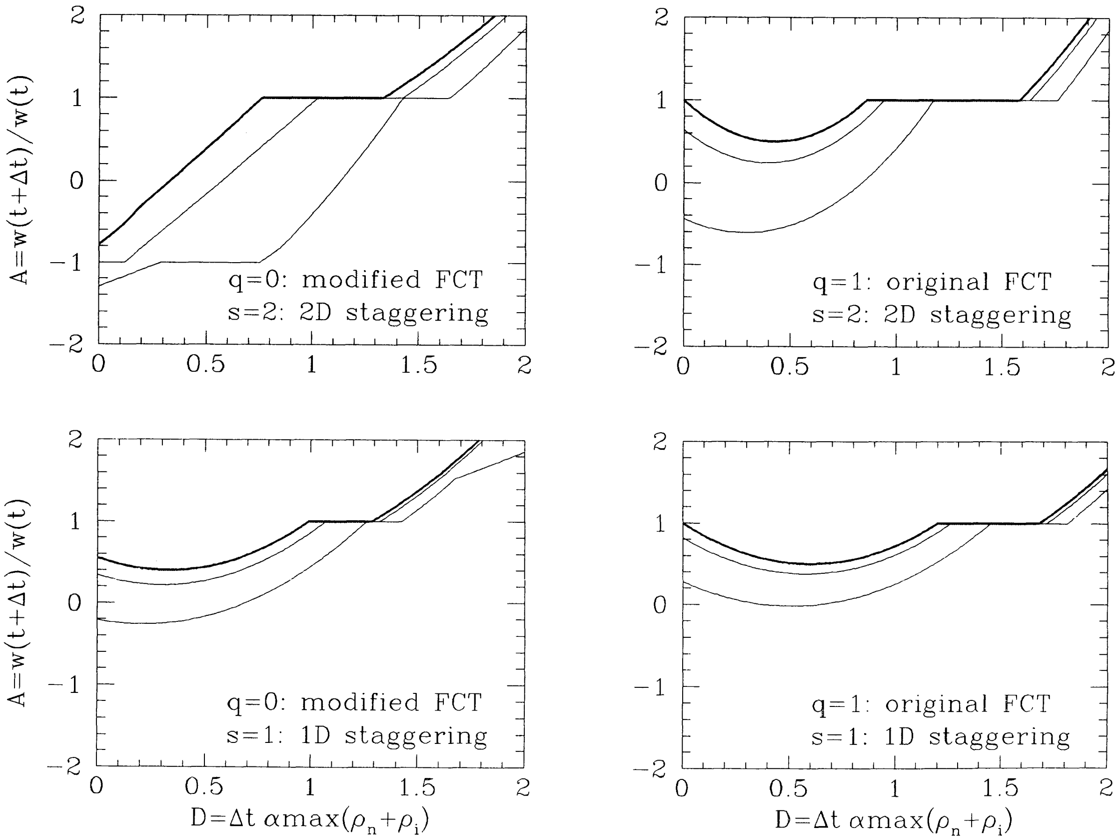


FIG. 2.—Amplification factor of the staggered instability by one time-centered FCT time step, shown as a function of the dimensionless critical parameter $D$. The difference scheme is stable if $|A| < 1$, marginally stable for $|A| = 1$, and unstable/overstable if $|A| > 1$. The four plots correspond to the combinations of the two versions of the FCT algorithm with the two possible staggering patterns. In one-dimensional simulations only the plots at the bottom are relevant, while in two-dimensional simulations all of them are relevant. In each plot the top (*bold*), middle, and lowest curves correspond to $\delta = 0$, 0.015, and 0.06, respectively, where $\delta \propto (v\Delta t/\Delta x)^2$. The amplification is a monotonic function of $\delta$: the top and bottom curves represent theoretical upper and lower bounds on $A$, provided that $C \leq 0.6$ for the Courant condition. In our simulations $c_{\mathrm{fast}} > |v|$, thus $\delta < 0.015$, and the amplification is further limited to the areas between the top and middle curves; hence $D < 0.75$ ensures damping even without the weighting scheme.

The numerical experiments show that using $\langle S \rangle$ instead of $S$ allows bigger time steps, while the results do not change perceivably. The diffusion associated with $(S - \langle S \rangle)$ is second order in $\Delta x$ and $\Delta y$. The weighting scheme cannot cancel the less unstable perturbation patterns with longer wavelengths; thus there is a limit on increasing $D_{\max}$ in equation (3.17) above $D_{\mathrm{damp}} \approx 0.8$.

### 3.6. *Boundary Conditions*

Boundaries are represented as an extra layer of two rows of cells around the $n_x \times n_y$ computational grid at ($i = -1$, 0, $n_x + 1$, $n_x + 2$; $j = 1, \ldots, n_y$) and ($i = -1, \ldots, n_x + 2$; $j = -1$, 0, $n_y + 1$, $n_y + 2$). In the calculations the boundary cells play the role of first and second neighbors to the cells close to the edge of the grid. Below, I shall refer to the two rows of boundary cells as the inner and outer boundary layers, while I call the cells at ($i = 2$, $n_x - 1$; $j = 1, \ldots, n_y$) and ($i = -1, \ldots, n_x + 2$; $j = 2$, $n_y$) the inner edge, and the cells at ($i = 1$, $n_x$; $j = 1, \ldots, n_y$) and ($i = -1, \ldots, n_x + 2$; $j = 1$, $n_y$) the outer edge. The following boundary conditions are allowed in our code:

1. *Continuous.*—The flow variables are copied from the outer edge to the two boundary layers next to them, therefore all the gradients vanish at the boundaries, which keeps the reflectivity low—a good approximation of the infinitely distant or free boundaries.

2. *Fixed.*—In some cases I want to fix the boundary conditions for the flow variables. The initial outer edges are stored in the adjacent boundary layers and are kept fixed for the rest of the computation.

3. *Reflective.*—A solid wall or piston may be represented by copying the flow variables from the outer edges to the inner layer of boundary cells, while the outer boundary cells are updated from the corresponding inner edges. The component of the momentum orthogonal to the reflective boundary changes signs.

4. *Periodic.*—The inner boundary cell values are taken from the opposite inner edge, and the outer boundary layer is copied from the opposite outer edge. This assumes that the simulation is invariant with a translation by $n_x$ (or $n_y$) cells.

5. *Shifted periodic.*—The procedure is similar to the previous one, except that there is a shift along the edge; thus the translation is $(i; j) \rightarrow (i + n_s; j + n_y)$, where $n_s < n_x$ is an integer. The typical use of this boundary condition is for testing another run which is periodic in the $y$ direction and has a fixed or continuous $x$ boundary. By rotating the initial condition in the $x$-$y$ plane, and applying the corresponding shifted periodic boundary condition, one can rerun the simulation and verify

that the results are independent of the grid orientation. Not all boundary cells are determined by this translational symmetry, e.g., the ($i = 1, \ldots, n_s; j = n_y + 1$) cells have no corresponding section at the $j = 1$ edge. I fix this problem by extending the fixed or continuous boundary condition along the $x$ boundaries to the unmatched portions of the $y$ boundary layers; thus I can use the content of the corner cells from the opposite edges. The shifted periodic boundary condition should not be combined with reflective or periodic boundaries along the other sides, and the grid has to be uniform too.

Our code allows the $i = 0$ and $i = n_x + 1$ boundaries to be either of the continuous, fixed, reflective, or periodic types, while the $j = 0$ and $j = n_y + 1$ boundaries are periodic with an optional shift. First the $x$ boundary, then the $y$ boundary layers are determined. In the above description cell-centered flow variables are assumed, but the CT method introduces edge-centered $B_x$ and $B_y$ variables. It is easy to generalize the above rules for edge-centered variables, only the reflective boundary condition is somewhat undetermined at $i = \frac{1}{2}, n_x + \frac{1}{2}$.

## 4. RUNNING THE CODE

### 4.1. *Tests*

An excellent test suite for magnetohydrodynamical simulations was published by Stone et al. (1992). Since the FCT algorithm has been thoroughly tested in the past, I will make no attempt here to evaluate the performance of the code on all the test problems. Advection tests show that FCT is quite diffusive at contact discontinuities of the flow variables, e.g., when a square wave of $B_y$ is advected with a constant speed, although it is relatively good at preserving the sharpness of shocks, where the velocity has a singularity as well, as shown by the shock-tube problem in Brio & Wu (1988). The modifications in the FCT algorithm by Ryu make no noticeable difference in either cases. DeVore (1991) presents two tests for the flux-conserving FCT: the rigid rotation of a current-carrying cylinder and the self-similar spherical expansion of a strong shock wave and trailing magnetic bubble out of the potential well of a star.

To my knowledge there is no test for two-fluid problems in the literature; thus I chose to use the steady state solution for the C-type shock as a test problem. It will be evident from the sample runs that the code can accurately maintain the steady state solution for many thousands of time steps. It is a bigger challenge to build up a C-type shock by driving a piston into a uniform medium. This is a one-dimensional problem, due to the slab symmetry of the steady state solution. The piston is represented by a reflective boundary at $x = 0$, and initially all the cells contain the undisturbed ambient medium with

$$\begin{aligned}
\rho^{(n)} &= 2.338 \times 10^{-22} \text{ g cm}^{-3} , \\
\rho^{(i)} &= 5.010 \times 10^{-25} \text{ g cm}^{-3} , \\
v_x^{(f)} &= -2.2605 \text{ km s}^{-1} , \\
T^{(f)} &= 20 \text{ K} , \\
B_y &= 5 \ \mu\text{G} ,
\end{aligned} \tag{4.1}$$

which is equivalent to $n_{\rm H} = 100 \text{ cm}^{-3}$ and $x_e = 10^{-4}$. All other flow variables are zero. The velocity $v_x^{(f)}$ is the difference between the upstream and downstream velocities of a 3 km s$^{-1}$ shock; thus the piston is expected to build up a shock structure identical to the 3 km s$^{-1}$ steady state solution, except that it is moving by the downstream velocity to the right. The right-hand boundary condition is set to continuous; the grid size is $n_x = 400$. The Courant number is $C = 0.6$, but the time step has to be limited by the drag instability condition. The early transients are extremely violent, thus $D_{\rm max} = 1.5$ is needed (without the weighting scheme $D_{\rm max} = 0.95$ would be required).

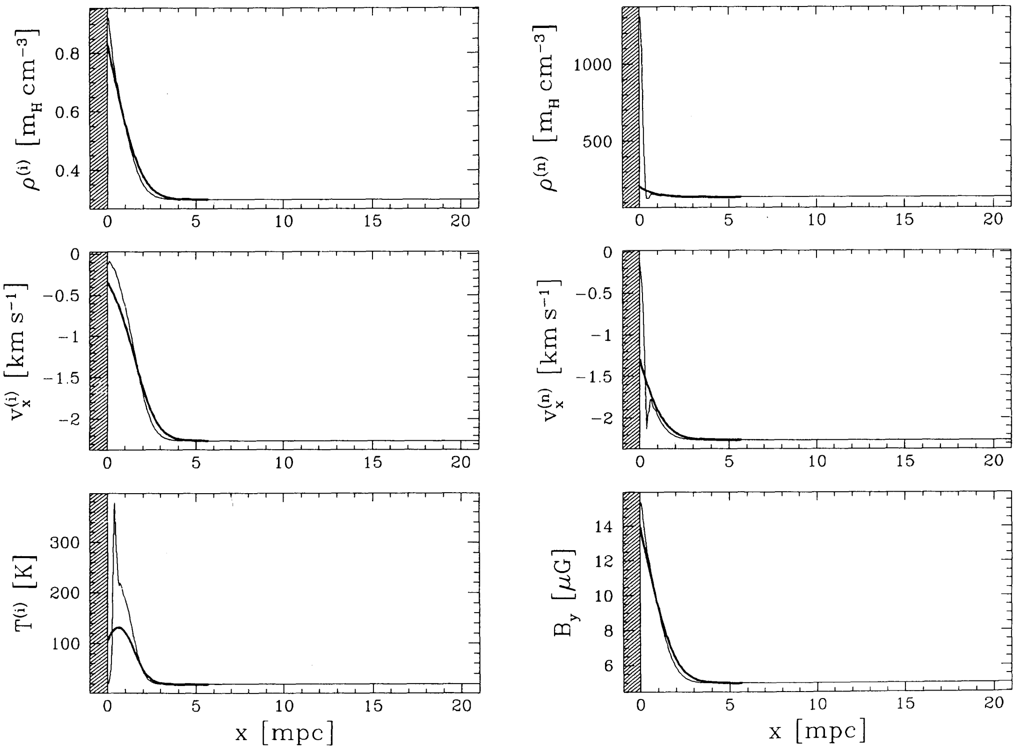
The flow hits the reflective boundary with full speed, producing heat and compression of both fluids and the magnetic field. The magnetic pressure increases until it can balance the ram pressure. First the ions slow down, then the ion-neutral friction helps the neutral gas to move against the incoming flow. Finally the C-type shock is built up perfectly, with a damping transient at the piston. Figure 3 shows the perfect agreement between the final stage and the analytic solution for the steady state shock. The average relative errors are less than 0.1% for all the flow variables in the shock transition region. The transient, which leads to demagnetization of the gas next to the piston, is erased at late times in the isothermal simulation presented here; note, however, that in an adiabatic simulation it would leave a permanent "imprint" on the gas adjacent to the piston (see Fig. 3*c*).

In general, analytic solutions are not available for comparison, and one needs to check self-consistency of the results. As Evans & Hawley (1988) point out, the convergence rate can be determined by repeating the simulation at least three times on successively finer grids. $\Delta x$ and $\Delta y$ have to be multiplied by the same factor $h < 1$. The absolute value of the true error is expected to fall as $h^\beta$, where $\beta$ is the rate of convergence. The relative error of the low- and high-resolution grids in the $L_1$ norm is therefore proportional to $1 - h^\beta$. From the two independent relative errors of the three simulations, $\beta$ and the true error can be estimated.

The above procedure deals with the dependence of overall errors on the grid resolution. It is of interest to check whether the orientation or the motion of the grid has a major effect on the results. Operator splitting algorithms are especially susceptible to anisotropy errors, and the advection of the flow is not trivial either. It is very simple to test the Galilean invariance: add a constant velocity to every cell in the initial condition, and repeat the calculation. For periodic boundaries no other change is necessary; for fixed or continuous boundary condition the grid has to be elongated, because the two simulations will contain the same structures at different positions. Of course, this test cannot be done for problems involving reflective boundaries. The results may be transformed back to allow for direct comparison.

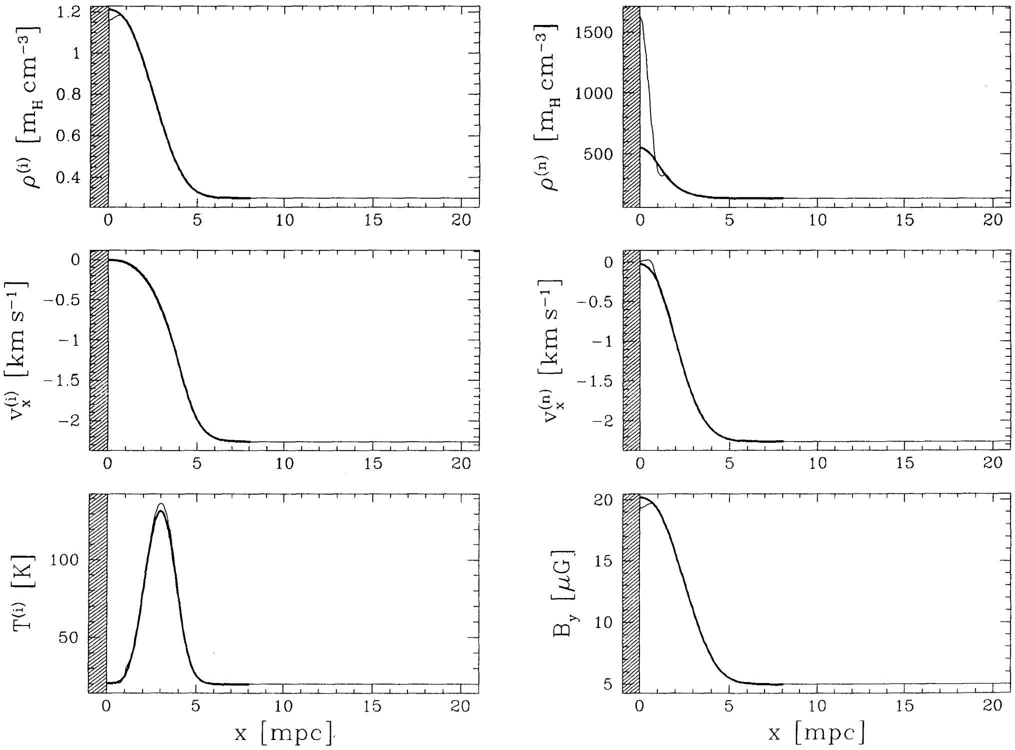
Rotation of the grid is a bit trickier. The flow variables at the rotated cell positions can be obtained by linearly interpolating between the four closest cells of the original grid. The vector variables ($\boldsymbol{v}^{(f)}$ and $\boldsymbol{B}$) have to be rotated accordingly. The rotation angle $\theta_g$ is chosen to make the shift in the shifted periodic condition $n_s = \sin \theta_g (n_y \Delta y / \Delta x)$ an integer, while $\Delta y'$ may need to be slightly different from $\Delta y$ to satisfy $n'_y \Delta y' = n_y \Delta y \cos \theta_g$, where the dimensions of the new grid are given by $n'_x \times n'_y$ (see Fig. 4). The continuous or fixed $x$ boundary condition has to be extended to the sections of the $y$ boundary, which are not determined by the shifted periodicity. To make the test meaningful, the flow variables need to be nearly constant close to the constant or fixed boundaries, otherwise their different shapes in the two grids would make the simulations different. This may require placing the $x$ boundaries farther from the interesting structures. The result of the rotated simulation can be transformed back by a similar procedure.

1994ApJ...425..171T



t=0.883 Kyear

FIG. 3*a*



t=3.598 Kyear

FIG. 3*b*

FIG. 3.—The three snapshots show three stages of the formation of the C-type shock. The ambient gas is moving to the left, hitting the reflective boundary "piston," represented by the shaded areas, at $x = 0$. The thin lines are the actual flow variables in the simulation, and the bold lines show a steady state solution for a 3 km s$^{-1}$ C-type shock for comparison. The steady state equations are integrated to $10^{-8}$ accuracy; then the velocities and positions are shifted by a Galilean transformation to match the ambient velocity of the numerical model and to make the locations where the magnetic fields reach $B_y(x) = 10\ \mu$G coincide for the steady state solution and the piston-driven shock. (*a*) Initially the cold isothermal neutral gas, lacking thermal pressure, passively compresses at the wall, while the ions and the magnetic field are building up the C-shock structure due to the high ion Alfvén speed $v_A^{(i)} \approx 20$ km s$^{-1}$. (*b*) By the time $t = 3.6 \times 10^3$ yr the ions have succeeded in decelerating the neutrals out to $x = 4$ mpc, and the whole flow converges to the steady state solution. (*c*) Finally the shock structure becomes a perfect C-type, with damping residuals of the initial transients at the left boundary. Here and in all subsequent figures the distance, time, and density units are mpc = $10^{-3}$ pc = $3.09 \times 10^{15}$ cm, kyear = $10^3$ yr = $3.16 \times 10^{10}$ s, and $m_H$ cm$^{-3}$ = $1.67 \times 10^{-24}$ g cm$^{-3}$, respectively. Note that 1 mpc/1 kyear = 0.98 km s$^{-1}$.

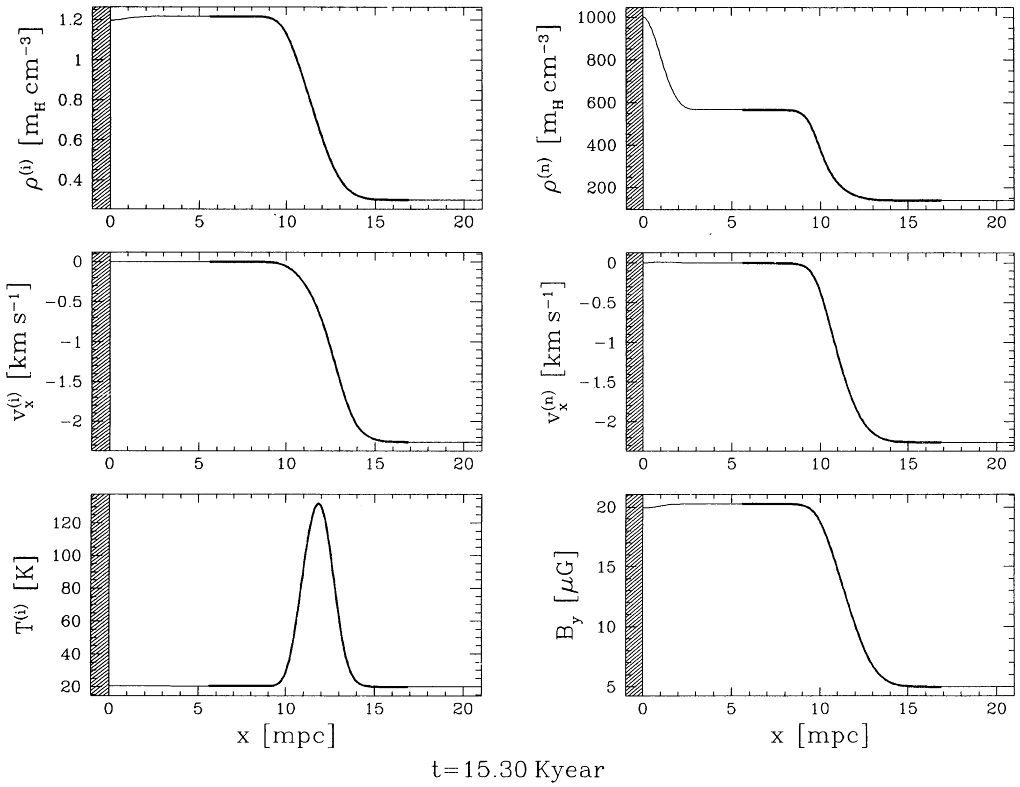


FIG. 3c

These tests require the same size or only slightly bigger grids than the original simulation, so they can be done for the highest resolution "production runs." The comparison may show that some features are results of spurious coincidences, such as alignment of the shock front with the grid, or that numerical errors are accumulating at some location due to the steady position relative to the grid. On the other hand, they may make us more confident if similar results are found with different grid speeds and orientations.

### 4.2. *Initial Conditions for the Wardle Instability*

A slightly perturbed steady state solution is used as the initial condition for studying the development of the Wardle instability. The advantage of this method is that I do not need to wait until the simulation relaxes into the steady state solution; furthermore, the transients of the relaxation process would mix with the growth of the Wardle instability.

First, the steady state equations are solved by setting $\partial_t \equiv \partial_y \equiv \partial_z \equiv 0$ in equation (2.2); thus all flow variables are functions of $x$ alone. Following Wardle & Draine (1987), for perpendicular shocks the standard shock frame is used, where the shock front is in the $y$-$z$ plane, the flow is along the $x$-axis, and the magnetic field is parallel to the $y$-axis; thus $B_x = B_z = v_y^{(f)} = v_z^{(f)} = 0$. For oblique shocks the frame moving by a $v_s \tan \theta_s$ speed parallel to the shock front is the most convenient, since in this frame $\boldsymbol{v}^{(i)} \parallel \boldsymbol{B}$ (and the electric field $\boldsymbol{E}$ vanishes) everywhere. The $E = 0$ frame can be used if $v_s \tan \theta_s \ll c$, where $c$ is the speed of light. For shock speeds of a few times 10 km s$^{-1}$ the condition holds up to $\theta_s < 89^\circ\!.5$ obliqueness, which is hardly distinguishable from a perpendicular shock with $\theta_s = 90°$. In both frames the steady state magnetic field and the fluid densities are simple algebraic expressions from momentum conservation and flux freezing,

$$\rho^{(i)}(x) = \rho^{(i)}(0)v_x^{(i)}(0)/v_x^{(i)}(x) \,, \tag{4.2a}$$

$$\rho^{(n)}(x) = \rho^{(n)}(0)v_x^{(n)}(0)/v_x^{(n)}(x) \,, \tag{4.2b}$$

$$B_x(x) = B_x(0) \,, \tag{4.2c}$$

$$B_y(x) = \begin{cases} \rho^{(i)}(x)B_y(0)/\rho^{(i)}(0) & \text{if } B_x(0) = 0 \text{ (perpendicular)} \,, \\ B_x(x)v_y^{(i)}(x)/v_x^{(i)}(x) & \text{if } B_x(0) \neq 0 \text{ (oblique)} \,. \end{cases} \tag{4.2d}$$

The other variables are integrated from $x = 0$, the downstream boundary, through the shock according to the following set of differential equations:

$$\partial_x v_x^{(i)} = \frac{(\gamma^{(i)} - 1)\mathscr{G}^{(i)} - \mathscr{F}_x^{(i)} v_x^{(i)} + CB_x B_y \mathscr{F}_y^{(i)} v_x^{(i)}}{\gamma^{(i)} p^{(i)} - \rho^{(i)} v_x^{2(i)} + CB_y^2 \rho^{(i)} v_x^{2(i)}} \,, \tag{4.3a}$$

$$\partial_x v_y^{(i)} = C(4\pi \mathscr{F}_y^{(i)} v_x^{(i)} - B_x B_y \partial_x v_x^{(i)}) \,, \tag{4.3b}$$

$$\partial_x p^{(i)} = \mathscr{F}_x^{(i)} - CB_x B_y \mathscr{F}_y^{(i)} - (1 - CB_y^2)\rho^{(i)} v_x^{(i)} \partial_x v_x^{(i)} \,, \tag{4.3c}$$

$$\partial_x v_x^{(n)} = \frac{(\gamma^{(n)} - 1)[\mathscr{G}^{(n)} - \Lambda(\rho^{(n)}, p^{(n)})] + \mathscr{F}_x^{(i)} v_x^{(n)}}{\gamma^{(n)} p^{(n)} - \rho^{(n)} v_x^{2(n)}} \,, \tag{4.3d}$$

$$\partial_x v_y^{(n)} = -\frac{\mathscr{F}_y^{(i)}}{\rho^{(n)} v_x^{(n)}} \,, \tag{4.3e}$$

$$\partial_x p^{(n)} = -\mathscr{F}_x^{(i)} - \rho^{(n)} v_x^{(n)} \partial_x v_x^{(n)} \,, \tag{4.3f}$$

where $C \equiv (4\pi\rho^{(i)} v_x^{2(i)} - B_x^2)^{-1}$. Alternatively, $p^{(i)}$ may be computed from the heating balance (eq. [2.5]). The numerical integration is done by the ODEINT algorithm, an adaptive Runge-Kutta integrator, from Numerical Recipes (Press et al.

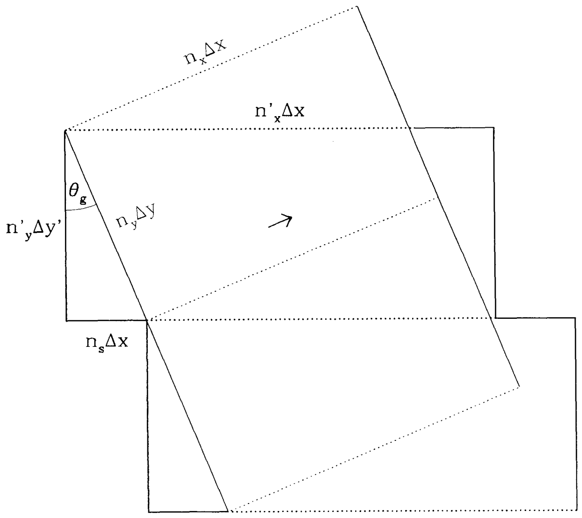


FIG. 4.—Geometry of the rotation of the grid. Both the original $n_x \times n_y$ and the new $n'_x \times n'_y$ grids are duplicated according to their respective translational invariance. The continuous lines are the fixed or continuous boundaries, while the dotted lines are the (shifted) periodic boundaries. The grids are rotated around the upper left-hand corner by $\theta_g$, which causes a shift in the boundary condition by $n_s$ cells, i.e., $n_s\Delta x$ distance. In general $\Delta y'$ cannot be exactly equal to $\Delta y$, but $\Delta x$ is kept fixed, since in the $x$ direction there is only a lower bound for $n'_x\Delta x$. If a vector variable points in the $x$ direction in the original frame, like the arrow, it will have to be rotated by $\theta_g$ relative to the $x'$ axis of the new frame.

1992), with an accuracy of $10^{-8}$. One has to fix $v_x^{(f)}$, $v_y^{(f)}$, $p^{(f)}$, $\rho^{(f)}$, and $B_x$ at $x = 0$. In terms of $v_s$ and $\theta_s$ the upstream velocities are given as $v_y^{(n)}(0) = v_y^{(i)}(0) = v_s \tan\theta_s$ (or 0 for $\theta_s = 90°$), and $v_x^{(n)}(0) = v_x^{(i)}(0) + \Delta v_x = v_s$, where the small velocity slip $\Delta v_x \ll v_s$ sets the distance of the shock front from the boundary. The relations between the $n_{\rm H}$, $x_e$, and $T^{(n)}$ ambient parameters and the upstream numerical flow variables are $\rho^{(n)}(0) = 1.4M_{\rm H}n_{\rm H}$, $\rho^{(i)}(0) = M^{(i)}x_e n_{\rm H}$, and $p^{(n)}(0) = k_B T^{(n)}\rho^{(n)}(0)/M^{(n)}$. It is practical to obtain $p^{(i)}(0)$ from the ion heating balance condition or to simply take $T^{(i)} = T^{(n)}$.

Once the steady state solution is calculated, a perturbation is added to the velocity field upstream of the shock, where the flow variables are almost constant. The velocity perturbation will perturb the densities and the magnetic field as the flow evolves. I chose velocity perturbations rather than density perturbations, since the latter could not excite modes with $v_z^{(f)} \neq 0$. As the perturbation reaches the shock, it may grow, but no further perturbation is coming in at the upstream boundary, thus the growth can be studied in a well-defined environment. A particular wavelength can be triggered by a sine wave perturbation of

$$\delta v_y^{(f)}(x, y) = Vf(x)\cos(k_y y)\,,$$

$$f(x) \equiv \begin{cases} \sin^2[\pi(x - x_f)/w_f] & \text{if } x_f < x < x_f + w_f\,, \\ 0 & \text{otherwise}\,, \end{cases} \qquad (4.4)$$

where $f(x)$ is a smooth masking function with width $w_f$ at position $x_f$, and $V$ is the amplitude in km s$^{-1}$, typically less than 1% of the shock speed. The physical width of the grid in the $y$-direction, $w_y$ should be an integer multiple of the wavelength $\lambda_y = 2\pi/k_y$. I make best use of the CPU by setting them equal, $w_y = \lambda_y$.

Another interesting perturbation is a Gaussian random field, a sum of sine waves with a power-law distribution of amplitudes and random phases, for each component of the velocity perturbation:

$$\delta v_j^{(n)}(x, y) = Vf(x)\sum_{k_i=0}^{n_x/2}\sum_{k_j=0}^{n_y/2}(k_i^2 + k_j^2)^{\kappa/2} \times \cos\left[2\pi\left(\frac{k_i x}{w_x} + \frac{k_j y}{w_y}\right) + \phi\right], \qquad (4.5)$$

where $\phi$ is a uniform random variable in the $[0, 2\pi]$ interval. The power-law index $\kappa$ is usually set to zero, so that I give all wavelengths the same chance to grow. The ion velocities are not perturbed initially, but they catch up with the neutrals in a few time steps due to the coupling, and because most of the mass and inertia is in the neutral fluid. Note that a periodic boundary condition in the $y$ direction will preferentially select modes with wavelengths $\lambda_k = w_y/n$, where $n$ is a positive integer. One needs to choose $w_y$ according to the wavelengths of interest.

Let us consider now how the physical dimensions and their numerical equivalents relate to each other, and what the constraints are for the upstream boundary conditions from the physics and the numerical limitations. Wardle (1990) characterizes perpendicular shocks with the length scale of the shock along the $x$-axis being $L_{\rm flow} = \sqrt{2}\,v_{\rm A}^{(n)}/[\alpha\rho^{(i)}(0)]$, where $\alpha$ is the coupling constant (eq. [2.4c]). The grid size has to be about $w_x = 3L_{\rm flow}$ to have a sufficiently long upstream and downstream flow. The typical wavelength for growing modes is $\lambda_y = 0.5L_{\rm flow} \lesssim w_y$. I found that $n_x \times n_y = 180 \times 30$ is the smallest grid that has a sufficient resolution, therefore $\Delta x \approx \Delta y \lesssim L_{\rm flow}/60$. The timescale for the flow is $t_{\rm flow} = r_f L_{\rm flow}/(2v_s) = 1/[\alpha\rho^{(i)}(0)]$, where $r_f \approx \sqrt{2}A^{(n)}$ is the compression ratio for the steady state shock. For the instability to grow fast, the neutral Alfvén Mach number should be $A^{(n)} \gtrsim 10$. The fast magnetosonic speed $c_{\rm fast} \gtrsim v_{\rm A}^{(i)}$, is highest downstream where the ion fluid is compressed by $r_f$, thus the time step is estimated by $\Delta t \leq r_f^{-0.5}C\Delta x/v_{\rm A}^{(i)}$ from equation (3.7). The number of time steps needed to run the simulation for one characteristic time is

$$n_t \geq \frac{t_{\rm flow}}{\Delta t_C} \approx 2500\left(\frac{0.6}{C}\right)\left(\frac{L_{\rm flow}}{60\Delta x}\right)\left(\frac{A^{(n)}}{10}\right)^{0.5}\left(\frac{A^{(n)}}{10A^{(i)}}\right). \qquad (4.6)$$

The other constraint on the time step comes from the drag instability (eq. [3.17]) with $\Delta t \leq D_{\rm max}/[\alpha \max(\rho^{(n)})] = D_{\rm max}/[\alpha r_f \rho^{(n)}(0)]$, and

$$n_t \geq \frac{r_f}{D_{\rm max}}\frac{\rho^{(n)}(0)}{\rho^{(i)}(0)} \approx 1500\left(\frac{1}{D_{\rm max}}\right)\left(\frac{A^{(n)}}{10}\right)\left(\frac{A^{(n)}}{10A^{(i)}}\right)^2. \qquad (4.7)$$

With the weighting scheme proposed for source terms, $D_{\rm max}$ may be somewhat greater than unity. In our simulations the two limits on $n_t$ are often comparable. Both estimates, especially the second one, exclude very low fractional ionization, since the resulting high $A^{(n)}/A^{(i)} \approx 0.22x_e^{-0.5}$ ratio would make the simulation prohibitively long. New ideas and innovative numerical methods may be required to simulate the cases with high Alfvén speed for the ions, like the typical dense ISM with very low fractional ionization and strong magnetic field; however, Wardle's analysis and our numerical experiments with varying fractional ionization suggest that the $x_e$ parameter, in the linear regime of the instability at least, only mildly affects the results.

### 4.3. *Simulations of C-Type Shocks*

I present three examples for modeling C-type shocks. The first is a perpendicular shock with $A^{(n)} \approx 13.5$. The physical parameters of the ambient gas are identical to the second example in Wardle's (1990) table: $v_s = 12.5$ km s$^{-1}$, $x_e = 10^{-4}$, $B_y = 5$ $\mu$G, and $T^{(f)} = 20$ K. Wardle predicts that the fastest-growing unstable mode has a wavelength $2\pi/k_y = 0.71 L_{\rm flow}$ and a growth rate $s = 9.1 t_{\rm flow}^{-1}$, where $L_{\rm flow} = 7 \times 10^{15}$ cm and $t_{\rm flow} = 5.4 \times 10^{10}$ s. The purpose of this simulation is to check the linear behavior of the shock. I take a 180 × 30 grid of dimensions $w_x = 3L_{\rm flow}$ and $w_y = \lambda_y = 0.71 L_{\rm flow}$. The steady state $v_y^{(f)}$ velocities are perturbed by a single sine wave of amplitude 0.001 km s$^{-1}$, and the masking function is $w_f = 0.5 L_{\rm flow}$ wide and positioned to start at the third cell from the $x = 0$ boundary, i.e., $x_f = x[3]$. Figure 5 shows the initial conditions. The simulation is run with $C = 0.6$ Courant condition, which itself ensures sufficiently small time steps so that the weighting scheme can stabilize the momentum equations against the staggered instability, while the ion temperature is calculated from the ion heating balance to avoid excessive inaccuracies. Other variables are almost unaffected by the choice of method for obtaining $p^{(i)}$. The $x$ boundaries are continuous, while the $y$ boundaries are periodic to fit the assumed periodicity of the instability. A typical simulation like this needs ~10 hr of CPU time on a Sun 4/50.

The evolution of the flow is shown in Figure 6. It is evident at first glance that Wardle's qualitative picture of the instability is correct: as the velocity perturbation reaches the shock, it starts to grow exponentially. The magnetic field lines bend ($B_x \neq 0$), and both the ion and neutral densities increase in the troughs ($\partial_y B_x < 0$) and decrease at the peaks ($\partial_y B_x > 0$). The wavelength was set by the perturbation and the grid size, but the growth rate offers an opportunity for quantitative comparison. I chose the standard deviation of $B_x$,

$$\sigma^2(B_x) \equiv \sum_{i=1,\, j=1}^{n_x,\, n_y} \frac{\Delta x_i \Delta y_j B_x^2[i, j]}{w_x w_y} - \left( \sum_{i=1,\, j=1}^{n_x,\, n_y} \frac{\Delta x_i \Delta y_j B_x[i, j]}{w_x w_y} \right)^2 , \tag{4.8}$$

as a measure of the growth rate. The steady state solution is $B_x$ = constant for both the perpendicular and oblique shocks, hence the standard deviation is purely a result of the instability

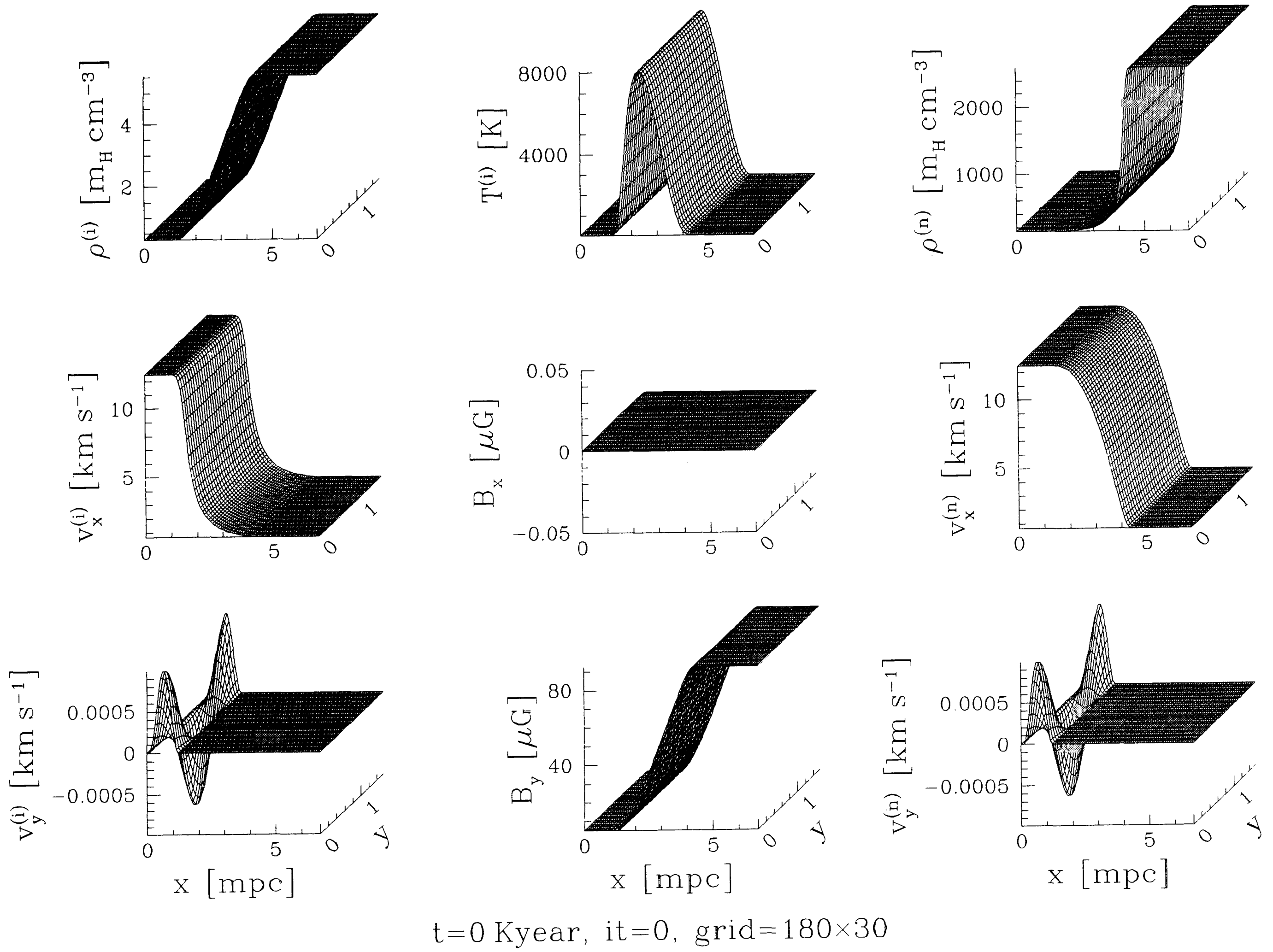


FIG. 5.—Initial condition of an $A^{(n)} \approx 13.5$ orthogonal C-type shock. The steady state solution is perturbed by a 0.001 km s$^{-1}$ amplitude sine wave in the $v_y^{(i)}$ and $v_y^{(n)}$ variables. Only a 90 × 30 submesh is drawn for the sake of clarity.

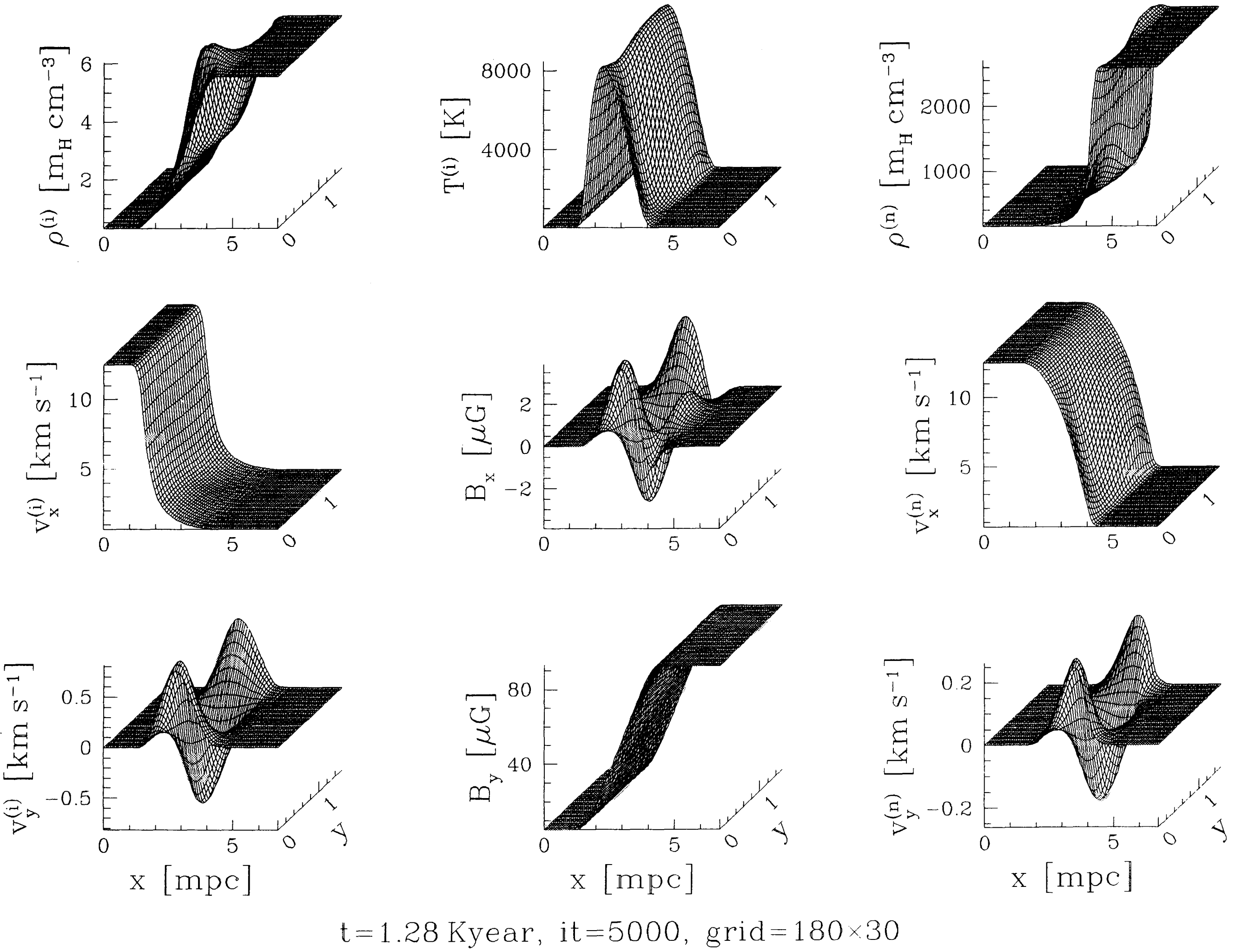


FIG. 6.—$A^{(n)} \approx 13.5$ orthogonal shock at $t = 0.75 t_{\rm flow}$ time, at the end of the linear growth of the instability. The velocity perturbation in $v_y^{(i)}$ increased by almost a factor of 1000. The ion density shows a peak at $y \approx 0.4$ mpc, where $\partial_y B_x < 0$, i.e., where the magnetic field lines buckle over from downstream to upstream direction ("troughs"). The neutral density has an excess at the same position, due to the convergent $v_y^{(n)}$ flow, but there is an elongated maximum at $y \approx 1$ mpc as well, where the depleted ion fluid cannot slow the neutrals down as effectively as in the steady state solution.

in either case. The definition allows for nonuniform grids as well. The code calculates $\sigma(B_x)$ at about every 50th time step and saves it into a log file. The evolution can be readily seen in Figure 7 by plotting log $\sigma(B_x)$ against time. The fast initial growth results from the compression of the ion fluid and the magnetic field by the sine wave velocity perturbation and a transient as the perturbation reaches the shock. The linear growth of the instability corresponds to the next straight section of the curve, which finally bends over as nonlinear effects slow the exponential growth down. The slope of the curve is $s \equiv \partial_t \ln \sigma(B_x) \approx 9.5 t_{\rm flow}^{-1}$, in good agreement with Wardle.

The simulation is repeated on a 216 × 36 and a 150 × 25 grid. The perturbations in the flow variables differ only by a few percent from the ones in Figure 6, with formal convergence rates between 1 and 2 for all the flow variables. The growth rates in the linear stage are essentially identical to the result from the 180 × 30 grid (Fig. 7).

The second problem demonstrates on an oblique shock how the fastest-growing mode can be found when no analytic prediction is available. I chose $\theta_s = 45°$, but otherwise all physical properties of the ambient ISM are identical with the perpendicular shock in the first example. Note that the frame is different, thus $v_x^{(f)} = v_y^{(f)} = v_s = 12.5$ km s$^{-1}$, but $B_x = B_y = 5/\sqrt{2}$ $\mu$G. Oblique shocks are expected to be more unstable and the fastest-growing mode to have shorter wavelength (Wardle 1991b). Oblique shocks have a somewhat smaller thickness, thus $w_x = 2.5 L_{\rm flow}$ is sufficient, while $w_y = L_{\rm flow}$ allows modes with $\lambda_y / L_{\rm flow} = 1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4} \ldots$. Numerical damping selectively suppresses small-wavelength modes, hence a 180 × 90 grid is needed with a finer resolution in the $y$ direction. All components of the steady state neutral velocity are perturbed with a Gaussian noise of $V = 0.001$ km s$^{-1}$ amplitude and $\kappa = 0$ index. The masking function is $w_f = 0.3 L_{\rm flow}$ wide, and it starts at $x_f = x[3]$. See Figure 8 for the initial condition. The simulation is run with $C = 0.6$ and $D_{\rm max} = 1.5$. The boundary conditions and the calculation of $p^{(i)}$ are the same as for the perpendicular case.

The results (Fig. 9) show that the fastest-growing mode is at $\lambda_y \approx \frac{1}{2} L_{\rm flow}$. As concluded by Wardle from the separability of

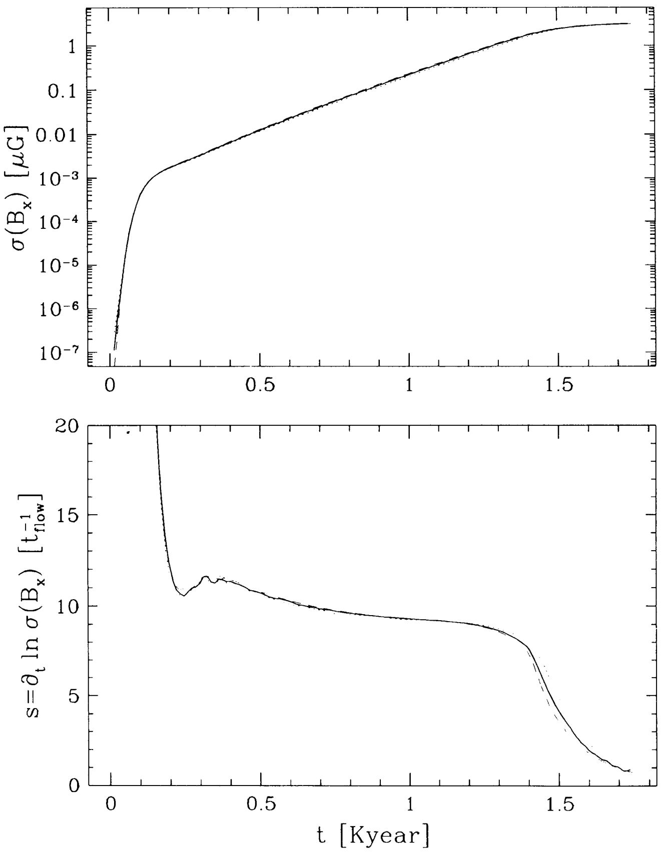


FIG. 7.—Time dependence of $\sigma(B_x)$, a linear measure for the instability, is shown on the top semilog plot. The fast initial increase, as the perturbation reaches the shock, is followed by an exponential growth, and a nonlinear cutoff at $t \approx 1.4 \times 10^3$ yr. The slope can be read from the bottom plot, which is the time derivative of the upper one scaled by $t_{\rm flow}^{-1}$. The linear regime corresponds to the plateau at $s \approx 9.5t_{\rm flow}^{-1}$, which agrees well with Wardle's prediction of a $9.1t_{\rm flow}^{-1}$ growth rate. The short- and long-dashed curves are results from simulations on a lower resolution 150 × 25 grid and a higher resolution 216 × 36 grid, respectively. Note that the relative errors are reduced with increasing grid resolution, a good indication of numerical convergence.

the linearized equations, the $v_z^{(j)}$ components are not amplified; the initial perturbations in the $z$ direction simply propagate in the form of intermediate waves. The $y$ components are, however, growing very fast. The semilog plot in Figure 10 has a slope of $\partial_t \ln \rho(B_x) \approx 12$, which indeed is steeper than the growth curve of the previous perpendicular case, in spite of the contributions from the slower-growing and damping modes. A series of runs with pure sine wave perturbations of different wavelengths would be required to find the accurate wavelength and growth rate for the fastest-growing mode. Results of this more systematic search will be reported in a forthcoming paper.

The third problem aims to explore the nonlinear evolution of the instability. I chose a perpendicular shock of $A^{(n)} \approx 8$ for this purpose, with $v_s = 7.5$ km s$^{-1}$ (the first example in Wardle's 1990 table), and the other parameters are identical to our previous perpendicular shock. The high-resolution grid is 360 × 50 in size, and $w_z$ is $4L_{\rm flow}$ long to give space to the nonlinear structures, which are expected to detach from the shock eventually. The fastest-growing mode is predicted to have $\lambda_y = 1.13L_{\rm flow}$ wavelength; thus $w_y = \lambda_y$. Calculations on a high-resolution grid are quite expensive, and the linear regime is of secondary interest now; therefore, it is appropriate to set the amplitude of the sine wave perturbation to a relatively high $V = 0.05$ km s$^{-1}$ value, while $w_f = 0.5L_{\rm flow}$ and $x_f = x[3]$ as usual. The Courant condition with $C = 0.6$ produces smaller time steps on finer grids, and eliminates the danger of the staggered instability by itself.

The results are shown in Figures 11 and 12. The growth rate during the short linear evolution is about $1.5t_{\rm flow}^{-1}$, close to what is expected. First the ion density becomes nonlinear, then the neutral density and velocity follow. The magnetic field stays very smooth even at late stages of the nonlinear growth. All variables saturate after $t \approx 3t_{\rm flow}$, and the density maxima, both in the ion and the neutral fluid, become more and more elongated in the flow direction. Finally, a high-density clump of the ion fluid detaches from the shock front and is advected downstream. The simulation has to be stopped because of a slowly increasing numerical error in the flow variables due to insufficient grid resolution. The peak, most prominent in the ion temperature in Figure 12*c*, is a result of a slight undershoot in the neutral density at the foot of the steep shock front, which makes the neutral velocity $v_x^{(n)} = m_x^{(n)}/\rho^{(n)}$, and thus the ion-neutral velocity slip responsible for the ion temperature, too high. On a 180 × 30 grid the peak shows up much earlier, confirming its numerical origin.

As a final test of the code, two low-resolution simulations of this same $A^{(n)} \approx 8$ perpendicular shock are shown in Figure 13. One of the runs is done on a 180 × 30 grid with $w_z = 3L_{\rm flow}$ and $w_y = 1.13L_{\rm flow}$, while the other uses a 208 × 29 grid ($w_x' = 3.47L_{\rm flow}$ and $w_y' = 1.08L_{\rm flow}$) *rotated* by $\theta_g = 17.^\circ 1$ relative to the shock normal. The grid rotation corresponds to an $n_s = 20$ shift of the periodic boundary condition. The two simulations are stopped at the same physical time, and the rotated result is transformed back to facilitate comparison. The close agreement is very reassuring, especially considering the fact that the shape of the cells, with an aspect ratio $\Delta x/\Delta y = 0.44$, is quite far from a square.

## 5. CONCLUSIONS

I have implemented a divergence-free version of the FCT finite-difference scheme to solve the coupled differential equations describing the dynamics of the two-fluid system, which contains the neutral gas, the ions, and the magnetic field. The FCT algorithm was slightly modified according to Ryu's (1991) suggestions, which reduce the computational burden and memory usage.

A numerical instability intrinsic to the two-fluid problem was discovered. The instability manifests itself in the exponential growth of the ion and neutral momentum in a staggered pattern with alternating signs. Probably any explicit integration scheme has to take into account the limitation on the time step $\Delta t < D_{\max}/\max(\alpha\rho^{(n)} + \alpha\rho^{(i)})$, where the maximum is taken over the computational grid at every time step, similar to the fashion in which the Courant condition is applied. Physically $(\alpha\rho^{(n)})^{-1} \approx (\langle\sigma v\rangle n^{(n)})^{-1}$ is the average time between ion-neutral scatterings for an ion, assuming that the fractional ionization is small. The numerical coefficient is determined for the time-centered FCT algorithm: $D_{\max} < 0.75$ is required to suppress the two-dimensional staggered instability. The condition can be relaxed if the source terms, in particular the drag force and heat due to ion-neutral friction, are calculated as a weighted average over the eight neighboring cells, rather than using the central cell only as prescribed by the original FCT algorithm.

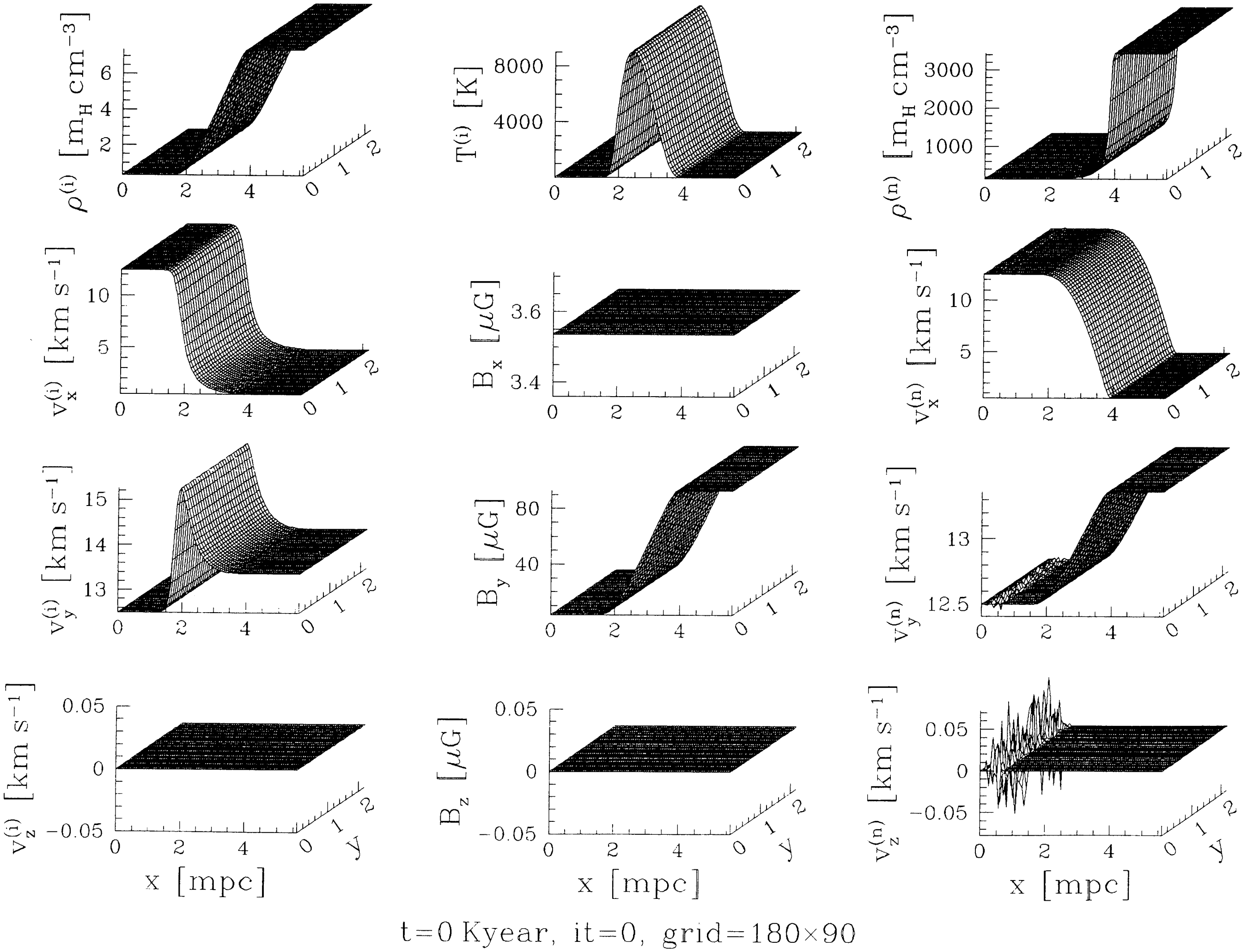


FIG. 8.—Initial condition for an $A^{(n)} \approx 13.5$ and $\theta_s = 45°$ oblique shock with a Gaussian perturbation of $V = 0.001$ km s$^{-1}$ amplitude and $\kappa = 0$ index in the neutral velocity. All components of $v^{(n)}$ are perturbed independently by the same amplitude noise [the perturbations add up to $\approx V(n_x n_y)^{1/2}/2$ from all the waves]; they look different only because of the different scales. Only a 90 × 30 submesh is plotted.

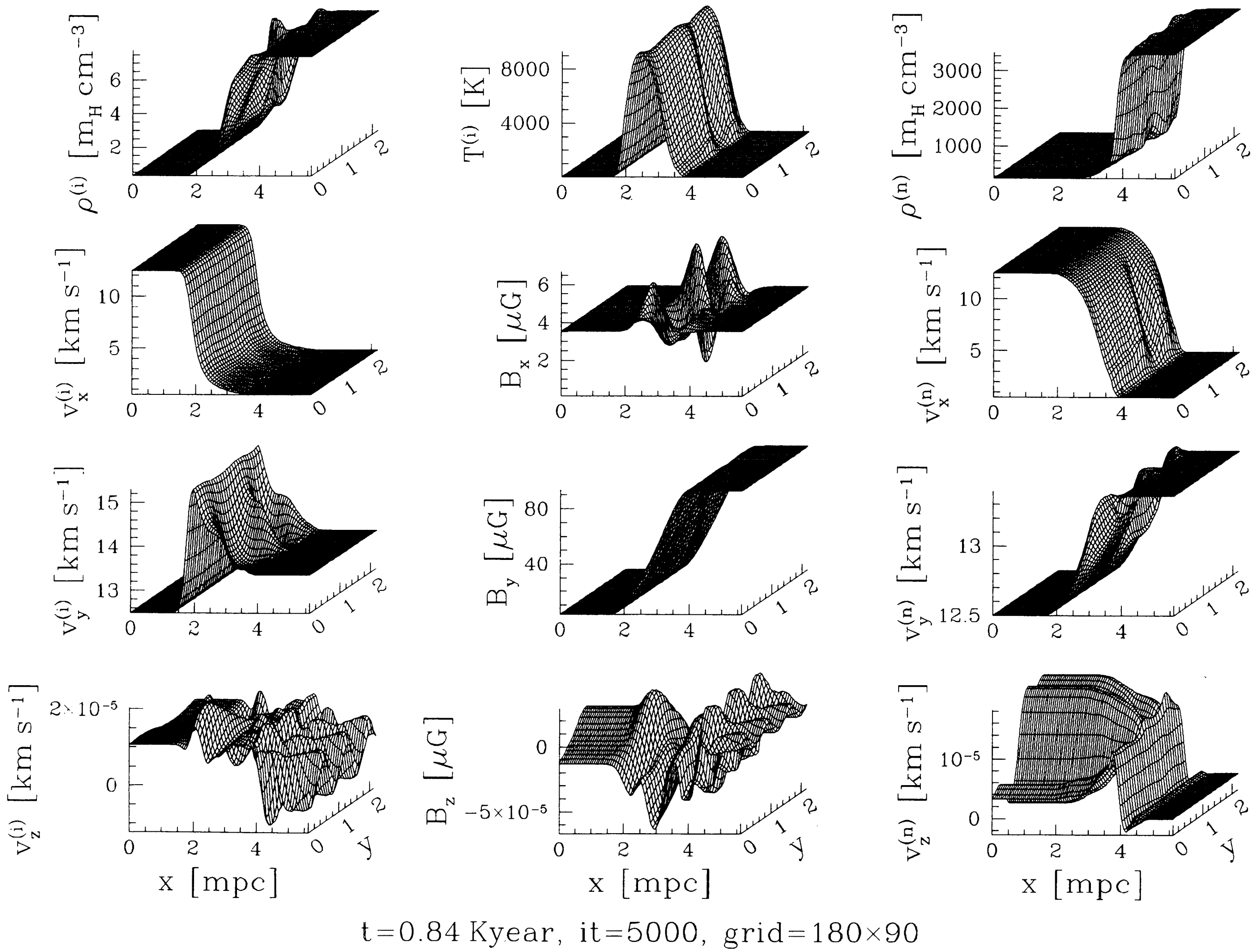


FIG. 9.—Oblique shock in the linear regime. The $x$ and $y$ components and the scalar variables clearly show a fast-growing $\lambda_y = w_y/2 = L_{\rm flow}/2$ wavelength mode. The $z$ components, on the other hand, damped by about a factor of 2000, since they are not participating in the growing mode.

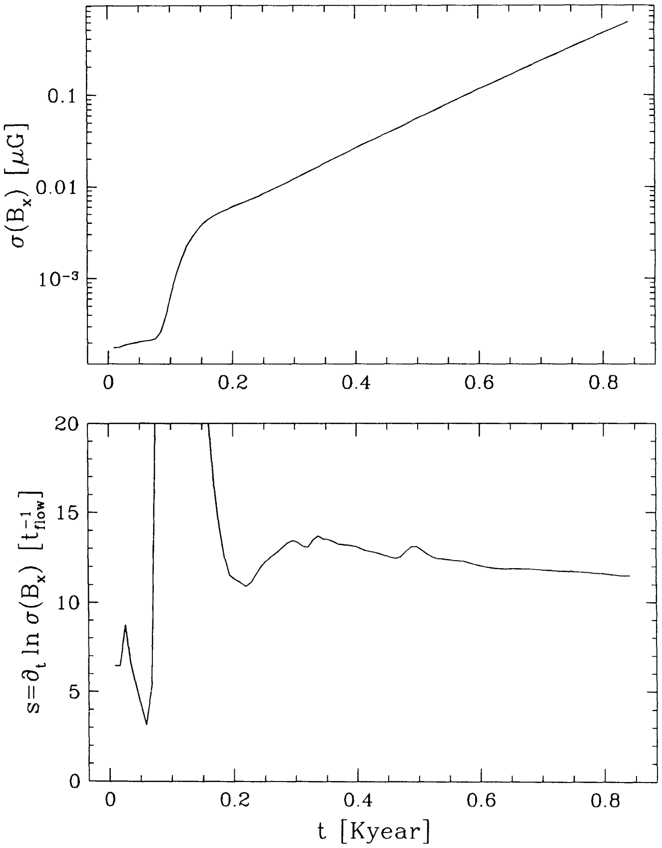


FIG. 10

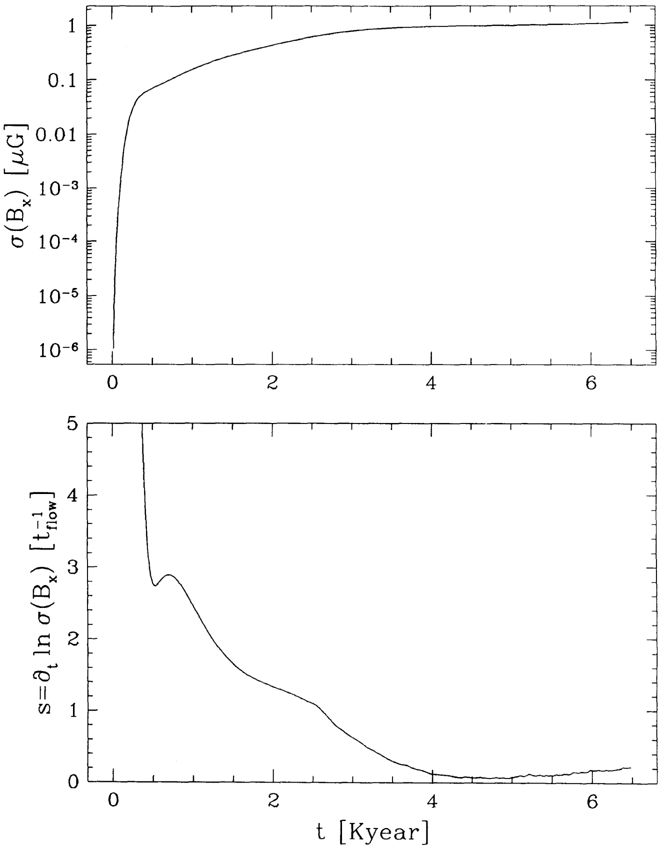


FIG. 11

FIG. 10.—Time dependence of $\sigma(B_x)$ for the oblique shock. The top semilog plot shows a slow increase in the beginning, probably due to the damping of many of the short-wavelength modes. This simulation was stopped in the linear regime; thus the curve does not bend over. The time derivative (*bottom*) is scaled to allow comparison with Fig. 7. The growth rate is $\approx 12$ at the plateau, distinctly higher than in the perpendicular case.

FIG. 11.—Growth of the perturbation for the $A^{(n)} \approx 8$ perpendicular shock. The short linear growth with $s \approx 1.5 t_{flow}^{-1}$ is followed by a saturated nonlinear stage. While the perturbation in the magnetic field stops growing, the ion and neutral density fields still evolve.

The weighting does not change the order of accuracy; in fact test runs show no perceivable difference in the results, but there is a gain in speed due to the longer time steps allowed by $D_{max} \approx 1.5$.

Testing a numerical code is a complex problem. The standard magnetohydrodynamical problems, and the convergence tests with refined grids, are well-known methods to evaluate the performance of a numerical code. An additional test can be the verification of Galilean invariance: the grid is either drifted by a constant velocity or rotated by a given angle. I describe in detail how the rotation can be realized by the use of shifted periodic boundary conditions. These tests are useful for any grid-based numerical algorithm, since they can reveal anisotropy or finite resolution-related errors, and there is no need to increase the size of the grid significantly for their execution. As a specific two-fluid test problem, I demonstrate how a C-type shock is built up by a piston driven into the ambient medium. The final stage of this numerical experiment shows a perfect C-type shock, identical to the high-accuracy solution of the steady state equations. The success of this test confirms that the FCT code can accurately calculate the smooth flows of the C-type shocks, provided that the time step is not too long compared to the ion scattering time. The numerical instability was discovered in the course of the first experiments on this simple test problem.

Finally, a few examples for the Wardle instability are examined. This is the first numerical confirmation of the linear analysis. First a perpendicular shock is simulated, with parameters identical to one of Wardle's examples. The shape, the wavelength, and the growth rate of the growing mode are found to be in good agreement with the analytical predictions, even though the numerically accessible parameter space restricts the simulations to relatively low ion Alfvén speeds, where Wardle's analytical approximations are the least accurate. Numerical errors and convergence rates are estimated from comparison with a higher and a lower resolution simulation. An oblique shock of the same shock speed is studied next. As predicted by Wardle, the growth rate of the fastest-growing mode exceeds that of the perpendicular shock, while its wavelength is shorter. In the last example the evolution of the instability for a weak perpendicular shock is followed deep into the nonlinear regime. The perturbation saturates only when the density perturbations become of order unity, the velocity field is changed by about 10%, while the magnetic field lines are still only slightly bent. The density maxima become gradually elongated in the direction of the flow, and finally a density clump

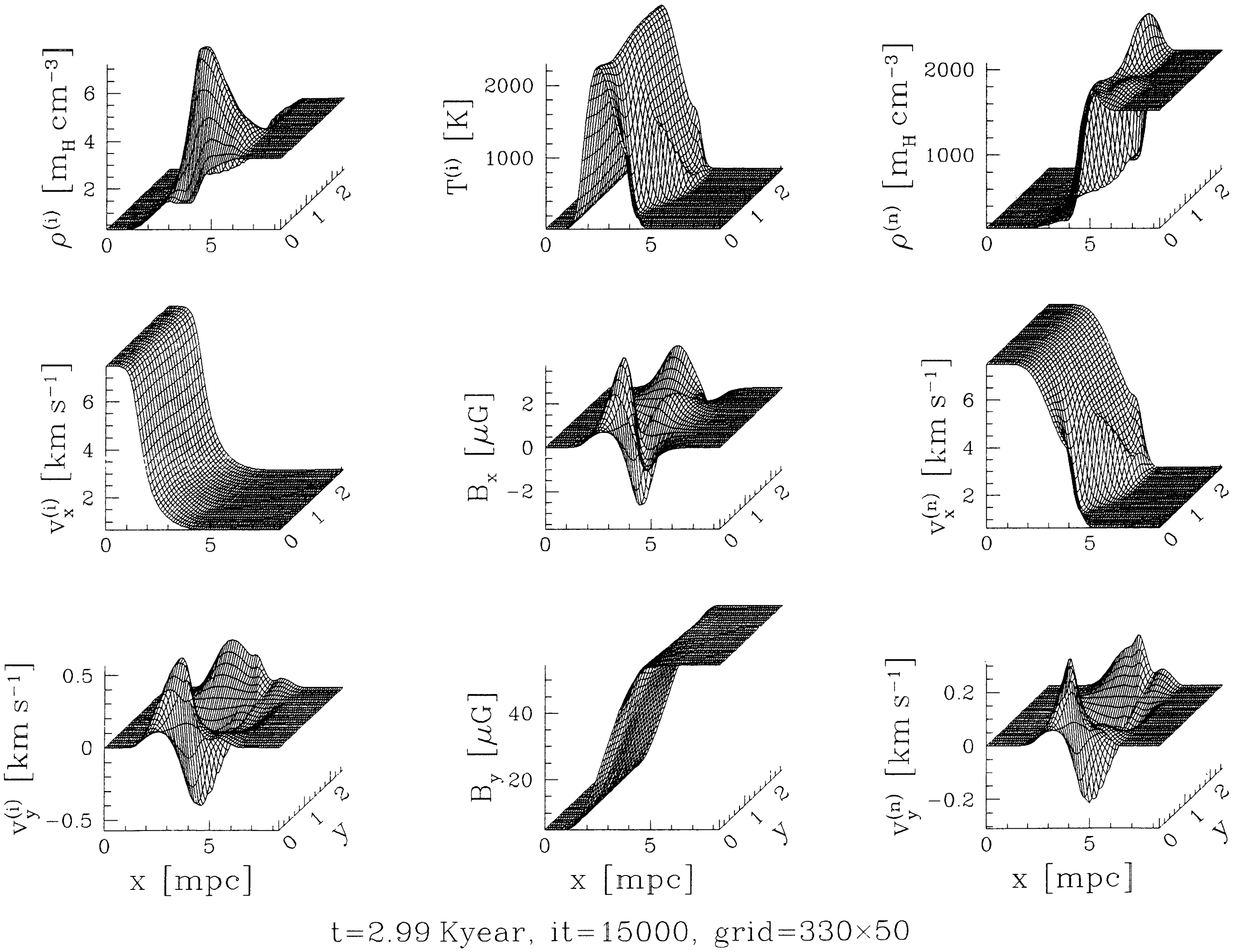
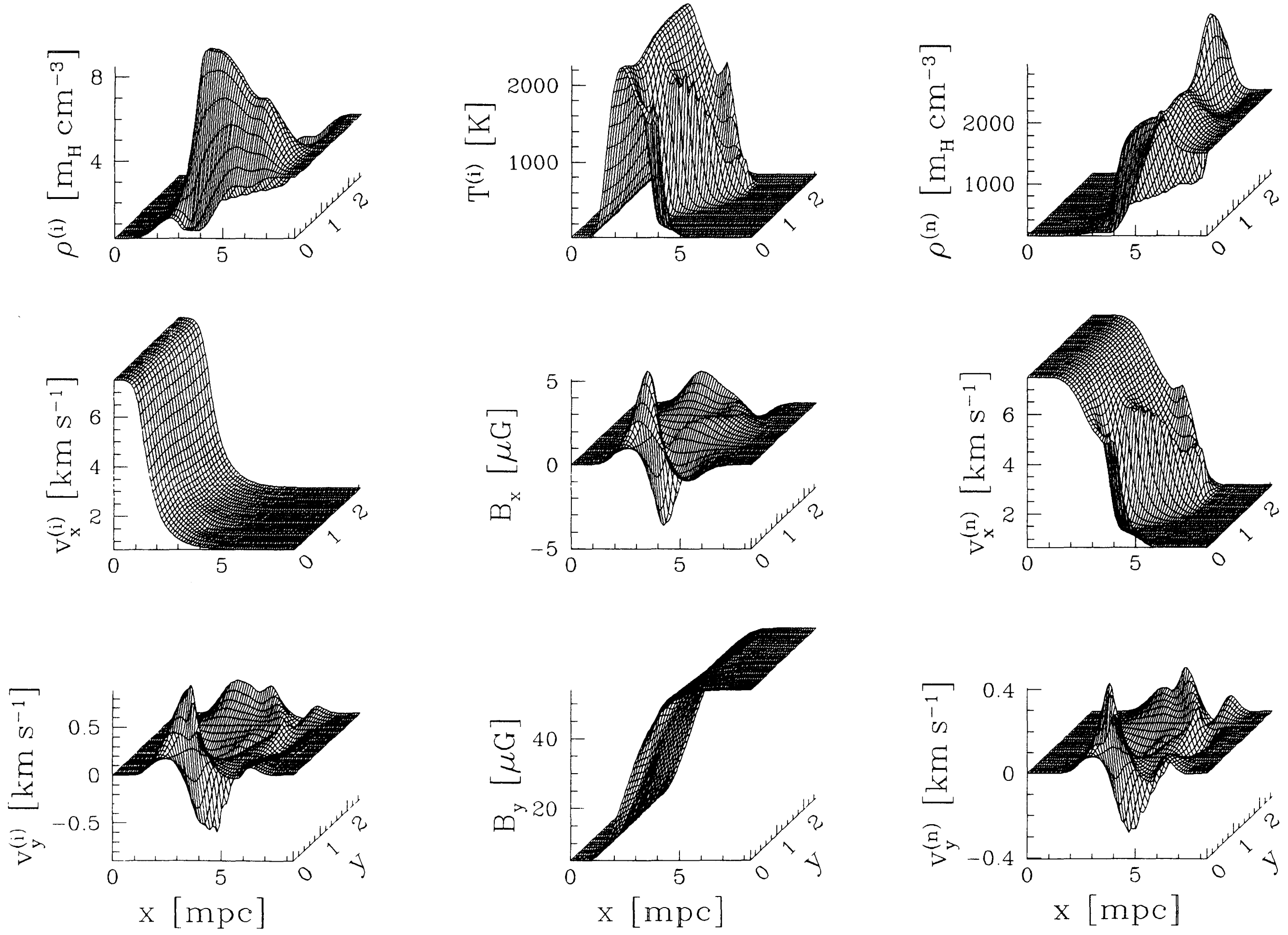


FIG. 12*a*

FIG. 12.—Three snapshots of the nonlinear evolution of the Wardle instability. The 90 × 25 subgrid drawn here shows a ragged appearance at some of the sharp edges, but in reality all features are smooth and resolved. (*a*) The ion density perturbation becomes as high as the full density jump across the steady state C-type shock. $B_x$ is still growing, but not exponentially any more. (*b*) The perturbations in all flow variables have reached their maximum values. The neutral density is slowly increasing ahead of the shock front in the "trough" of the magnetic field (the closer half of the grid), while at the "crest" the compressed neutral gas is pushed downstream: the neutral density gradient is becoming steeper, and a ridge of $\approx 3000\, m_H$ cm$^{-3}$ forms. This propagation can also be observed as an enhancement of the neutral velocity $v_x^{(n)}$ in a V-shaped feature. (*c*) The grid had to be elongated by 30 cells to allow for the continuing downstream drift of density peaks. The only new development compared with the previous figure is the clear separation of a secondary ion density peak from the shock. Unfortunately, the peak in $T^{(i)}$ and $v_x^{(n)}$ is due to numerical error, which forces us to stop the simulation. The downstream flow $v_x^{(f)}(\infty) = 0.66$ km s$^{-1}$ is very slow in any case: in the $3 \times 10^3$ yr elapsed between the first and last snapshot the total drift is only 2 mpc behind the shock.



t=4.85 Kyear, it=30000, grid=330×50

FIG. 12$b$

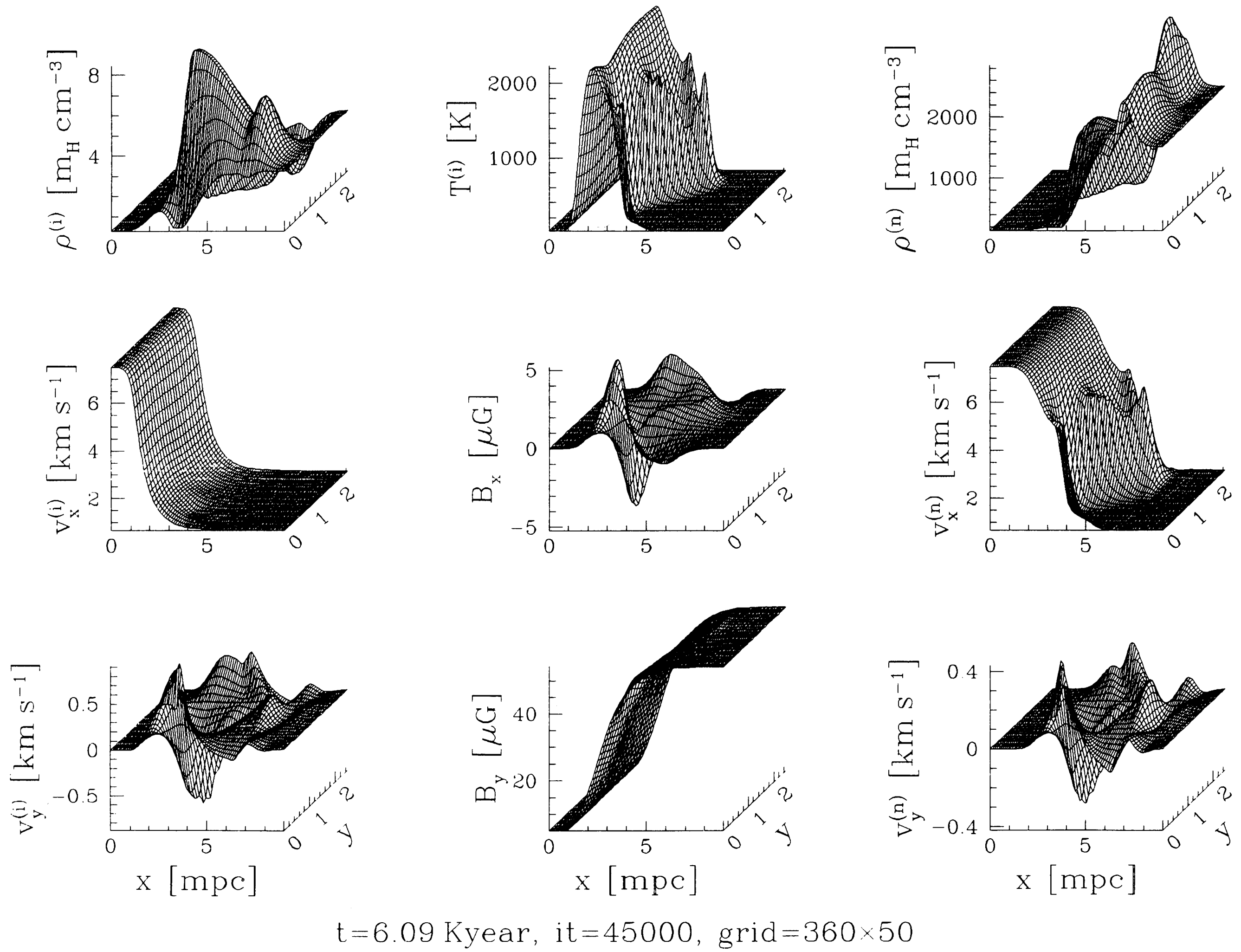

$\rho^{(i)}$ [$m_H$ cm$^{-3}$]
8
4
$T^{(i)}$ [K]
2000
1000
$\rho^{(n)}$ [$m_H$ cm$^{-3}$]
2000
1000
$v_x^{(i)}$ [km s$^{-1}$]
6
4
2
$B_x$ [$\mu$G]
5
0
−5
$v_x^{(n)}$ [km s$^{-1}$]
6
4
2
$v_y^{(i)}$ [km s$^{-1}$]
0.5
0
−0.5
$B_y$ [$\mu$G]
40
20
$v_y^{(n)}$ [km s$^{-1}$]
0.4
0
−0.4
0
5
0
1
2
y
x [mpc]
t=6.09 Kyear, it=45000, grid=360×50

FIG. 12c

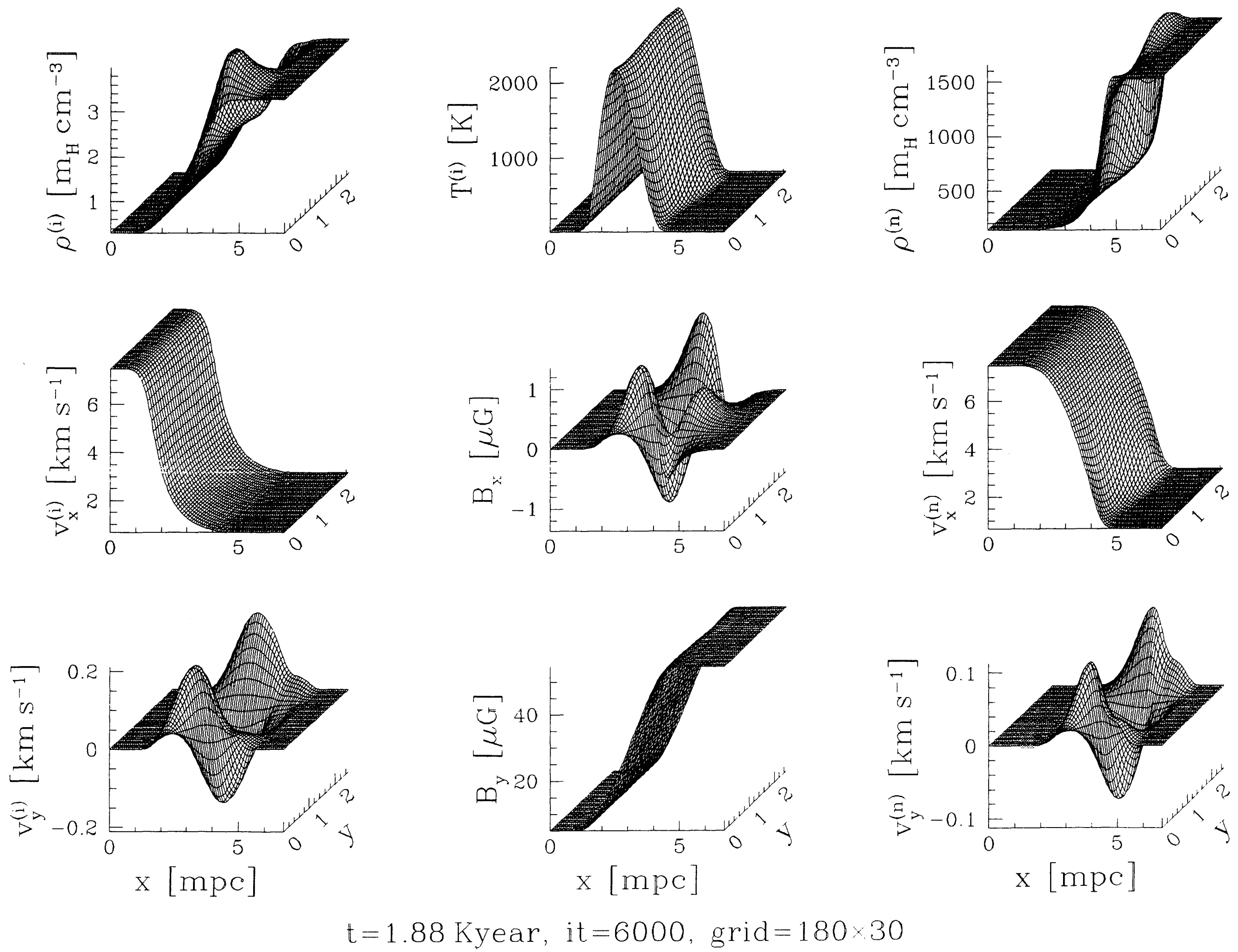
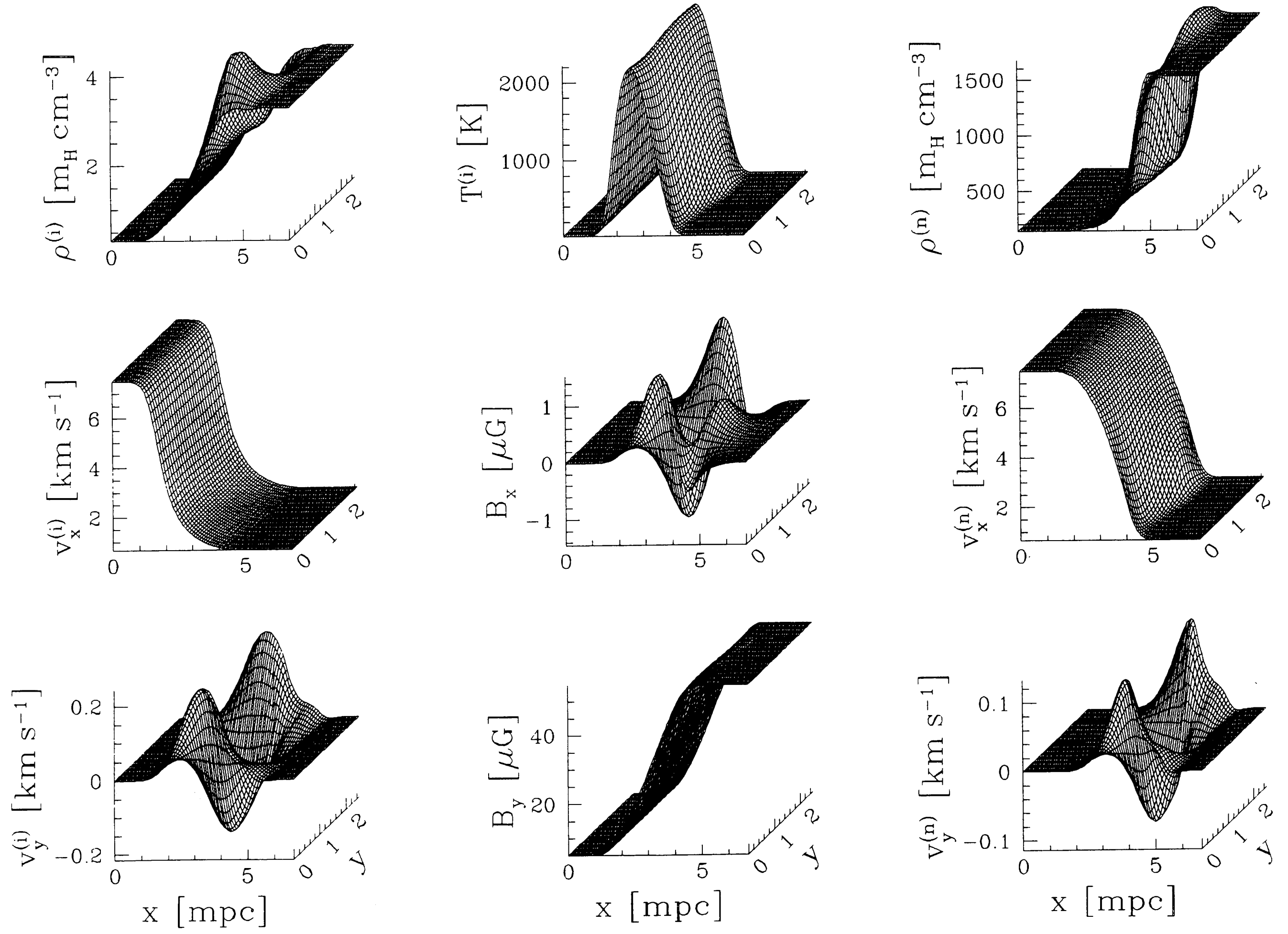


FIG. 13*a*

FIG. 13.—Comparison of two low-resolution simulations for the $A^{(n)} \approx 8$ perpendicular shock. (*a*) A snapshot taken at the beginning of the nonlinear evolution on a 180 × 30 grid. (*b*) This simulation was done on a 209 × 29 grid rotated by $\theta_g = 17\overset{\circ}{.}1$ relative to the shock normal. The periodicity shift was $n_s = 20$ to mimic the original slab symmetry of the problem. The final result is transformed back to the original 180 × 30 grid. The two results are in remarkable agreement, in spite of the strongly elongated cells (with aspect ratio 0.44) that might have made the numerical scheme anisotropic.



t=1.88 Kyear, it=5739, grid=180×30

FIG. 13*b*

detaches from the shock and drifts downstream with the flow. Whether these results are typical or not will be the subject of further investigations.

The two-fluid code presented in this paper is a good tool for studying the dynamics of C-type shocks. The numerical simulations should lead to a physical understanding of the nonlinear evolution of the Wardle instability and other phenomena of two-fluid shocks, hopefully enabling an approximate but general analytical description. With this goal in mind, I am using the code to execute a systematic series of numerical experiments to explore the Wardle instability in the numerically available parameter space; results will be reported in a subsequent paper.

I would like to acknowledge the generous help and encouragement of Bruce T. Draine, who directed my interest to the area of two-fluid shocks. I am indebted to Dongsu Ryu for his help and expertise in numerical hydrodynamics, and also to Robert H. Lupton for his excellent data processing and graphical language SM, which enabled me to develop complex analysis and visualization macros, including the "hidden line" surface drawing algorithm used for most of the figures in this paper. I thank the anonymous referee for his valuable comments. This research was supported in part by NSF grant AST 90 17082 and OTKA, the Hungarian Science Foundation, grant F-4491.

## APPENDIX A

I calculate the amplification of the staggered instability by a time-centered FCT step. A spatially uniform flow is assumed in which the $x$ components of the ion and neutral momenta are perturbed by $\delta m_x^{(i)} = -\delta m_x^{(n)}$ relative to the correct $\bar{m}_x^{(f)}$ values in a staggered pattern, formally written as

$$m_x^{(f)}[i, j] = \bar{m}_x^{(f)} + \delta m_x^{(f)}(-1)^{s_x i + s_y j} . \tag{A1}$$

Here $s_x = 1$ if there is staggering in the $x$ direction, and zero otherwise; $s_y$ is defined analogously. The $\bar{m}_x^{(f)}$ constants cancel in all of the following equations; thus they are omitted. In the first half-step $w \equiv \delta m_x^{(i)}$ is transported by equation (3.2) to

$$w^T = w + \frac{\Delta t}{2} S = \left(1 - \frac{D}{2}\right) w , \tag{A2}$$

since the source term is $S = -wD/\Delta t$ from equations (3.15) and (3.16), while the edge-centered fluxes cancel. The diffusive fluxes are $F_k^D = \pm s_k(\frac{1}{6} + \epsilon_k^2/3)2w$ from equation (3.5a), where $k$ is either $x$ or $y$. The perturbation does not affect the edge-centered $\epsilon_k = v_k(\Delta t/2)/\Delta k$ constants. The diffusion stage (eq. [3.3]) yields

$$w^D = w^T - 4w\left[\frac{(s_x + s_y)}{6} + \frac{(s_x \epsilon_x^2 + s_y \epsilon_y^2)}{3}\right] = \left(N - \frac{D}{2}\right) w , \tag{A3}$$

where $N \equiv 1 - (\frac{2}{3})(s_x + s_y) + (\frac{4}{3})(s_k \epsilon_k^2)$, and summation for $k = x$, $y$ is implied. The uncorrected antidiffusion fluxes are $F_k^A = \mp s_k(\frac{1}{6} - \epsilon_k^2/6)2[qw^T + (1 - q)w^D]$ with $q = 1$ for the original FCT scheme and $q = 0$ for Ryu's version. With the definition of $M \equiv 1 + (\frac{2}{3})(s_x + s_y) - (\frac{2}{3})(s_k \epsilon_k^2)$ the antidiffused perturbation can be written as

$$w^A = w^D + 4[qw^T + (1 - q)w^D]\left[\frac{(s_x + s_y)}{6} - \frac{s_k \epsilon_k^2}{6}\right] = \left(K - M\frac{D}{2}\right) w , \tag{A4}$$

where $K \equiv MN - q(M - 1)(N - 1)$. The half-step is completed by the Zalesak flux correction, which here reduces to a truncation of the magnitude of $w_A$ by the maximum of the absolute values of $w$ and $w^D$, and we obtain the half-step solution

$$w^H = \mathrm{sign}\,(w^A) \min\,[|w^A|, \max\,(|w|, |w^D|)] . \tag{A5}$$

The full step consists of the same stages as the half-step; I use a prime to distinguish the symbols from their half-step equivalents. The time-centered $S' = w^H D/\Delta t$ source term is multiplied by $\Delta t$ in the full step; thus the half-step equations are modified as follows:

$$w^{T'} = w - Dw^H . \tag{A6a}$$

$$w^{D'} = N'w - Dw^H , \tag{A6b}$$

$$w^{A'} = K'w - M'Dw^H , \tag{A6c}$$

$$w^{\mathrm{new}} = \mathrm{sign}\,(w^{A'}) \min\,[|w^{A'}|, \max\,(|w|, |w^{D'}|)] . \tag{A6d}$$

Note that $\epsilon_k' = v_k \Delta t/\Delta k = 2\epsilon_k$ is used in the definitions of $M'$ and $N'$ due to the full-size time step.

The $A \equiv w^{\mathrm{new}}/w$ amplification is a function of $D$, $q$, $s \equiv s_x + s_y$, and $\delta \equiv \frac{2}{3} s_k \epsilon_k^2/s$, since $N = 1 - s(\frac{2}{3} + 2\delta)$ and $M = 1 + s(\frac{2}{3} - \delta)$, and $N'$ and $M'$ can be expressed from $\delta' = 4\delta$ similarly. The $\delta$ parameter is usually small; its range is limited to $0 \leq \delta \leq C^2/6$ by the $|\epsilon_k| < C/2$ Courant condition. It would be rather tedious and useless to calculate the amplification $A(D, \delta; q, s)$ analytically; instead I obtained numerical results for finely spaced values of $D$ and plotted them in Figure 2 for fixed values of $\delta$. The amplification turns out to be a monotonic function of $\delta$; thus $A(D, 0; q, s)$ and $A(D, 0.06; q, s)$ are upper and lower bounds, respectively, in the four cases defined by the dimensionality of the staggering ($s = 2, 1$) and the choice of the original method versus Ryu's modification ($q = 1, 0$). The $A(D, 0.015; q, s)$ curves are shown not just as an intermediate case but as the more typical lower bounds on $A$, since

$\Delta t \leq C\Delta k/(|v_k| + c_{\rm fast})$, and, in our calculations at least, the fast magnetosonic speed exceeds the flow velocities, therefore $|\epsilon_k| \leq C/4$ and $\delta \leq 0.015$.

The results of the above calculations were checked by running the FCT code for a few time steps with initial conditions as in Figure 1 and periodic boundaries on a 4 × 4 grid. The amplitude of the perturbation grew exactly as expected in all cases. The weighting scheme was, of course, not used in these check runs.

## APPENDIX B

The weighting of the source function is required to eliminate perfectly the staggered instability for both the one- and the two-dimensional cases. The most general 9-point weighting scheme is

$$\begin{aligned}\langle S[i,j]\rangle = {} & aS[i,j] \\ & + b_x(S[i-1,j] + S[i+1,j]) + b_y(S[i,j-1] + S[i,j+1]) \\ & + c(S[i-1,j-1] + S[i+1,j-1] + S[i+1,j+1] + S[i-1,j+1])\,,\end{aligned} \tag{B1}$$

where the symmetrically placed corner cells have the same $c$ coefficients, but I let the $b_x$ and $b_y$ weights of the cells on the four sides be different, as may be appropriate if $\Delta x \neq \Delta y$. The sum of the 9 weights must be

$$a + 2b_x + 2b_y + 4c = 1\,. \tag{B2}$$

The staggered perturbation can be written formally as

$$S[i,j] = \bar{S}[i,j) + \delta S(-1)^{s_x i + s_y j}\,, \tag{B3}$$

where $\bar{S}$ is the correct value and $\delta S$ is the numerical error. The $s_x = 0, 1$ and $s_y = 0, 1$ parameters tell whether the perturbation is staggered in the $x$ and $y$ directions, respectively. The three possible patterns corresponding to $(s_x; s_y) = (1; 1), (1; 0), (0; 1)$ and the condition that $\langle S[i,j]\rangle$ should be independent of $\delta S$ yield three more linear equations for the weights:

$$\begin{aligned} a - 2b_x - 2b_y + 4c &= 0\,, \\ a + 2b_x - 2b_y - 4c &= 0\,, \\ a - 2b_x + 2b_y - 4c &= 0\,. \end{aligned} \tag{B4}$$

Combined with equation (B2), the unique solution is $a = \frac{1}{4}$, $b_x = b_y = \frac{1}{8}$, and $c = \frac{1}{16}$, as was indicated in equation (3.19).

## REFERENCES

Book, D. L., Boris, J. P., & Zalesak, S. T. 1981, in Finite-Difference Techniques for Vectorized Fluid Dynamics Calculations, ed. D. L. Book (New York: Springer-Verlag), 29
Brio, M., & Wu, C. C. 1988, J. Comput. Phys., 75, 400
Chernoff, D. F. 1987, ApJ, 312, 143
DeVore, R. C. 1991, J. Comput. Phys., 92, 142
Draine, B. T. 1980, ApJ, 241, 1021
Draine, B. T., & Katz, N. 1986, ApJ, 310, 392
Draine, B. T., & McKee, C. F. 1993, ARA&A, 31, 373
Draine, B. T., Roberge, W. G., & Dalgarno, A. 1983, ApJ, 264, 485
Elmegreen, B. G., & Fiebig, D. 1993, A&A, 270, 397
Evans, C. R., & Hawley, J. F. 1988, ApJ, 332, 659
Godunov, S. K. 1959, Mat. Sbor., 47, 271
Lapidus, A. 1967, J. Comput. Phys., 2, 154
Lax, P. D., & Wendroff, B. 1960, Comm. Pure Appl. Math., 13, 217
Press, W. H., Flannery, B. P., Teukolsky, S. A., & Vetterling, W. T. 1992, Numerical Recipes in Fortran (2d ed.; Cambridge: Cambridge Univ. Press)
Roberge, W. G., & Draine, B. T. 1990, ApJ, 350, 700
Roe, P. L. 1981, J. Comput. Phys., 43, 357
Ryu, D. 1991, private communication
Smith, M. D., & Brand, P. W. J. L. 1990, MNRAS, 242, 495; 244, 384
Stone, J. M., Hawley, J. F., Evans, C. R., & Norman, M. L. 1992, ApJ, 388, 415
Stone, J. M., & Norman, M. L. 1992, ApJS, 80, 753
van Leer, B. 1979, J. Comput. Phys., 32, 101
Wardle, M. 1990, MNRAS, 246, 98
———. 1991a, MNRAS, 250, 523
———. 1991b, MNRAS, 251, 119
Wardle, M., & Draine, B. T. 1987, ApJ, 321, 321
Woodward, P., & Colella, P. 1984, J. Comput. Phys., 54, 115
Zalesak, S. T. 1979, J. Comput. Phys., 31, 335