Modified Lattice Boltzmann method for compressible fluid simulations

F. L. Hinton, M. N. Rosenbluth, S. K. Wong, Y. R. Lin-Liu, and R. L. Miller*
General Atomics, San Diego, California 92186-9784
(Received 22 September 2000; published 29 May 2001)

A modified lattice Boltzmann algorithm is shown to have much better stability to growing temperature perturbations, when compared with the standard lattice Boltzmann algorithm. The damping rates of long-wavelength waves, which determine stability, are derived using a collisional equilibrium distribution function which has the property that the Euler equations are obtained exactly in the limit of zero time step. Using this equilibrium distribution function, we show that our algorithm has inherent positive hyperviscosity and hyperdiffusivity, for very small values of viscosity and thermal diffusivity, which are lacking in the standard algorithm. Short-wavelength modes are shown to be stable for temperatures greater than a lower limit. Results from a computer code are used to compare these algorithms, and to confirm the damping rate predictions made analytically. Finite amplitude sound waves in the simulated fluid steepen, as expected from gas dynamic theory.

DOI: 10.1103/PhysRevE.63.061212 PACS number(s): 47.11.+j

I. INTRODUCTION

The lattice Boltzmann method [1] has the potential of providing fast algorithms for fluid simulations, but it tends to be numerically unstable when temperature variations are allowed [2–5]. This is because of the lack of an H theorem for the thermal lattice Boltzmann method [6]. In this paper, we present a modification of the standard lattice Boltzmann (LB) algorithm, which we demonstrate to be much more stable in tests using a computer code. Our analysis of the algorithm in terms of wave damping shows why this is expected. Our algorithm uses overrelaxation in the advection step, while the standard algorithm uses overrelaxation in the collision step [7]. We refer to our algorithm as the AOR algorithm (for advection with overrelaxation). Advection with overrelaxation is implemented using a two time advection scheme, with the relative weights of the two times determined by a parameter. With the correct choice of the overrelaxation parameter, the second-order dissipation coefficients (viscosity and thermal diffusivity) can be made arbitrarily small, while maintaining positive fourth-order dissipation coefficients (hyperviscosity and hyperdiffusivity). By contrast, in the LB algorithm, the fourth-order dissipation coefficients also go to zero when the second-order coefficients go to zero. The improved stability is clearest when the temperature is allowed to vary because the thermal mode (defined in Sec. VIII) tends to be less stable than shear modes, but is stabilized by hyperdiffusivity. Short-wavelength modes generated by nonlinearity are thus effectively suppressed with the AOR algorithm.

In lattice-based fluid simulations, fluid variables are time advanced using a picture of molecules with velocities which lie on a lattice, and which move between points on a spatial lattice and then collide with each other. The fluid variables are sums, over velocity lattice points, of the populations of these velocities, appropriately weighted. The basic premise [1] is that the complexity of hydrodynamics can be well described by a drastically simplified version of molecular dynamics.

The distribution function f is defined such that $f(\vec{x},\vec{c},t)d^3x$ is the number of molecules in the spatial volume d^3x at position $\vec{x}$, with velocity $\vec{c}$, at time t. The velocity $\vec{c}$ is one of the points of a lattice in velocity space; we shall use the four-dimensional face-centered hypercube lattice, because of its symmetry properties [8]. After a time interval Δt, a particle at position $\vec{x}$ will move to $\vec{x}+\vec{c}\Delta t$, which defines a spatial lattice. The fluid variables—the density ρ, the flow velocity $\vec{u}$, and the average energy per particle ε—are defined as moments of the distribution function, which are sums over the lattice velocities:

$$\rho=\sum_{\vec{c}} f, \quad \rho\vec{u}=\sum_{\vec{c}} \vec{c}f, \quad \rho\varepsilon=\sum_{\vec{c}} e(\vec{c})f \tag{1}$$

where $e(\vec{c})=c^2/2$ is the energy of a molecule with velocity $\vec{c}$. (The particle mass is chosen to be 1 in our units.) The simulation follows the fluid variables in time, for given initial and boundary conditions, by time advancing the distribution function f and evaluating these moments. Advancing f consists basically of particle advection and collisional equilibration.

Fluid equations are obtained from the AOR algorithm by using an expansion in the discreteness. Using an appropriate choice of the equilibrium distribution function, given in Sec. IV the fluid equations are shown to be the Euler equations for a perfect fluid, plus discretization errors. These errors are shown to cause wave damping, and are identified with viscosity, thermal diffusivity, etc. Although the leading-order fluid equations have the property of Gallilean invariance, the same is not true for the discretization errors. In fact, the wave damping rates become negative for flow speeds larger than critical values, as discussed in Secs. VI and VIII, and this limits the allowable flow speeds.

A complete discussion of the stability properties of the AOR algorithm would need to include all of the wave vectors allowed on the spatial lattice, not just the longest wave-

*Present address: Archimedes Technology Group, 5405 Oberlin Dr., San Diego, CA 92121.

lengths. A first step in this direction is taken in Sec. X, where the stability of the shortest allowable waves is considered. For a special case, stability is shown to restrict the allowed temperature range somewhat.

II. ADVECTION WITH OVERRELAXATION

Given the fluid variables (density, flow velocity, and energy density) at two previous times, and assuming the distribution function (DF) at the prevous times has been set equal to the collisional equilibrium DF determined by the fluid variables at those times, the fluid variables are determined at the present time by using the following three steps.

(1) The advected DF is obtained from the DF at the previous times consistent with advection of particles from other spatial positions.

(2) The fluid variables are obtained as moments of the advected DF.

(3) The new DF is given by the collisional equilibrium DF determined by the fluid variables.

The advected DF is given by

$$f_*(\vec{x},\vec{c},t)=\alpha f(\vec{x}-\vec{c}\Delta t,\vec{c},t-\Delta t) + (1-\alpha)f(\vec{x}-2\vec{c}\Delta t,\vec{c},t-2\Delta t), \tag{2}$$

where α is the overrelaxation parameter. The result of many collisions is assumed to be that the DF at time t is equal to the equilibrium DF

$$f(\vec{x},\vec{c},t)=f^{\mathrm{eq}}(\vec{x},\vec{c},t), \tag{3}$$

where $f^{\mathrm{eq}}(\vec{x},\vec{c},t)$ is given by an assumed function of $\vec{c}$ chosen such that the particle density, momentum density, and energy density are the same as given by the advected DF:

$$\sum_{\vec{c}} \{1,\vec{c},e(\vec{c})\}f^{\mathrm{eq}}=\sum_{\vec{c}} \{1,\vec{c},e(\vec{c})\}f_* \tag{4}$$

We have written a computer code which implements Eqs. (1), (2), and (3), and the results will be given below.

III. FLUID LIMIT

As with a real gas, we expect fluid behavior in the limit of small time intervals between collisions. The velocities $\vec{c}$ are now assumed to be normalized, so that the smallest nonzero component along any coordinate axis is unity. The spatial grid separation is then $\Delta x=\Delta t$. Distance is normalized to the length of the simulation. We assume that Δt is small enough for good resolution of the flows of interest, and expand the advected DF in a Taylor series; from Eq. (2), we find

$$f_*\simeq f-(2-\alpha)\Delta t\left(\frac{\partial}{\partial t}+\vec{c}\cdot\vec{\nabla}\right)f + \frac{1}{2}(4-3\alpha)\Delta t^2\left(\frac{\partial}{\partial t}+\vec{c}\cdot\vec{\nabla}\right)^2 f, \tag{5}$$

where $f=f^{\mathrm{eq}}$ is the collisional equilibrium DF.

Denoting one of the quantities $\{1,\vec{c},e(\vec{c})\}$ by ψ, we multiply this equation by ψ and sum over lattice velocities $\vec{c}$, using Eqs. (3) and (4), and obtain

$$(2-\alpha)\Delta t\sum_{\vec{c}} \psi\left(\frac{\partial}{\partial t}+\vec{c}\cdot\vec{\nabla}\right)f-\frac{1}{2}(4-3\alpha)\Delta t^2 \times\sum_{\vec{c}} \psi\left(\frac{\partial}{\partial t}+\vec{c}\cdot\vec{\nabla}\right)^2 f=\sum_{\vec{c}} \psi(f-f_*)=0. \tag{6}$$

We thus obtain fluid equations, with discretization errors of order Δt:

$$\frac{\partial}{\partial t}\left(\sum_{\vec{c}} \psi f\right)+\vec{\nabla}\cdot\left(\sum_{\vec{c}} \vec{c}\psi f\right) =\frac{(2-3\alpha/2)}{(2-\alpha)}\Delta t\sum_{\vec{c}} \psi\left(\frac{\partial}{\partial t}+\vec{c}\cdot\vec{\nabla}\right)^2 f. \tag{7}$$

The first set of sums on the left-hand side of Eq. (7) consists of the lowest moments of the equilibrium DF, the density, flow velocity, and energy density, which are defined by Eq. (1). The second set of sums on the left hand side of Eq. (7) includes the momentum flux tensor

$$P_{ij}=\sum_{\vec{c}} c_i c_j f \tag{8}$$

and the energy flux vector

$$\rho\zeta_i=\sum_{\vec{c}} c_i e(\vec{c}) f. \tag{9}$$

The traceless momentum flux is defined by

$$\rho\pi_{ij}=\sum_{\vec{c}} \left(c_i c_j-\frac{c^2}{4}\delta_{ij}\right)f \tag{10}$$

(the trace of δ_{ij} is 4, in the four-dimensional velocity space.)

Thus the fluid equations obtained in the limit $\Delta t\to 0$ are

$$\frac{\partial\rho}{\partial t}+\frac{\partial}{\partial x_\mu}(\rho u_\mu)=0, \tag{11}$$

$$\frac{\partial(\rho u_i)}{\partial t}+\frac{\partial}{\partial x_i}(\rho\varepsilon/2)+\frac{\partial}{\partial x_\mu}(\rho\pi_{i\mu})=0, \tag{12}$$

$$\frac{\partial(\rho\varepsilon)}{\partial t}+\frac{\partial}{\partial x_\mu}(\rho\zeta_\mu)=0, \tag{13}$$

where we have used the fact that the trace of the momentum flux is $P_{\mu\mu}=2\rho\varepsilon$.

For later use, we give the result of carrying out the expansion in Δt to higher order,

$$\frac{\partial}{\partial t}\left(\sum_{\vec{c}} \psi f\right)+\vec{\nabla}\cdot\left(\sum_{\vec{c}} \vec{c}\psi f\right)\simeq\sum_{p=2}^{4}\beta_p\sum_{\vec{c}} \psi\left(\frac{\partial}{\partial t}+\vec{c}\cdot\vec{\nabla}\right)^p f, \tag{14}$$

neglecting terms of order Δt^4, where

$$\beta_2=\frac{(2-3\alpha/2)}{(2-\alpha)}\Delta t,$$

$$\beta_3=-\frac{(4/3-7\alpha/6)}{(2-\alpha)}\Delta t^2, \tag{15}$$

$$\beta_4=\frac{(2/3-5\alpha/8)}{(2-\alpha)}\Delta t^3.$$

The $O(\Delta t)$ discretization error is zero if $\alpha=\frac{4}{3}$, giving a first-order accurate algorithm for the ideal fluid equations. Alternatively, choosing $\alpha<\frac{4}{3}$, the term proportional to β_2 can be identified with viscous stress, giving a first-order accurate algorithm for the viscous fluid equations. (It is easily shown that the error in the mass conservation equation is of order Δt^2 in either case.)

Recalling that $f=f^{\rm eq}$, it is now clear that the above fluid equations will be identical to the ideal fluid equations in the limit $\Delta t\to 0$, provided that we choose the equilibrium DF to have the correct momentum flux and energy flux moments. We define the temperature T in terms of the average particle energy by

$$\varepsilon=\frac{u^2}{2}+c_v T, \tag{16}$$

where c_v is the specific heat at constant volume. Since the velocity space is four dimensional, there are four degrees of freedom, and so $c_v=2$. (We use units where the gas constant is equal to unity). The correct specific heat for a diatomic gas ($c_v=\frac{5}{2}$) can be achieved by using a somewhat more complicated equilibrium DF, which will be the subject of another paper.

The traceless momentum flux and energy flux must have the following forms, to be consistent with the ideal fluid equations:

$$\pi_{ij}=u_i u_j-\frac{u^2}{4}\delta_{ij} \tag{17}$$

and

$$\zeta_i=(\varepsilon+T)u_i. \tag{18}$$

By substituting into Eqs. (12) and (13), we obtain the ideal fluid (Euler) equations, as required.

IV. CHOICE OF EQUILIBRIUM DISTRIBUTION FUNCTION

There are many choices for the equilibrium DF which satisfy the requirements of Eqs. (16), (17), and (18). The choice we make here is mainly to illustrate the stability properties of the AOR algorithm, and to enable us to compare with the LB algorithm. We choose an analytical form for the equilibrium DF so that stability properties can be derived explicitly, by analytical means.

We write the equilibrium DF in the form

$$f^{\rm eq}(\vec{c})=\rho F(c^2/2)[1+g(\vec{c})], \tag{19}$$

where F is determined by minimizing the sum over lattice velocities $\Sigma F\ln F$ subject to the constraints

$$\sum_{\vec{c}} F=1, \quad \sum_{\vec{c}} e(\vec{c})F=\varepsilon \tag{20}$$

and g is determined by minimizing ΣFg^2 subject to the constraints

$$\sum_{\vec{c}} Fg=0, \quad \sum_{\vec{c}} e(\vec{c})Fg=0,$$

$$\sum_{\vec{c}} Fgc_i=u_i, \quad \sum_{\vec{c}} e(\vec{c})Fgc_i=\zeta_i,$$

and

$$\sum_{\vec{c}} Fg\left(c_i c_j-\frac{c^2}{4}\delta_{ij}\right)=\pi_{ij},$$

where ζ_i and π_{ij} are assumed given and are related to $\vec{u}$ and ε by Eqs. (18) and (17). This approach is similar to that discussed in Ref. [9].

Our choice of the four-dimensional face-centered hypercubic (FCHC) lattice simplifies the solution of these equations because of its symmetry properties [8]. We shall use the smallest number of energy states for which a solution exists, which is three, i.e., energies $n\equiv c^2/2=0$, 1, and 2. Denoting by d_n the number of velocities for an energy n, we have $d_1=d_2=24$ for the FCHC lattice. The energy-1 velocities are the permutations of $(\pm 1, \pm 1, 0, 0)$. The energy-2 velocities are the permutations of $(\pm 1, \pm 1, \pm 1, \pm 1)$ and the permutations of $(\pm 2, 0, 0, 0)$. The number of $n=0$ states may be arbitrarily assigned, and we choose $d_0=6$, which also simplifies the solution.

We then obtain

$$F(0)=\frac{1}{4d_0}(2-\varepsilon)^2, \quad F(1)=\frac{1}{2d_1}\varepsilon(2-\varepsilon),$$

$$F(2)=\frac{1}{4d_2}\varepsilon^2 \tag{21}$$

and

$$g(\vec{c})B(n)c_\mu u_\mu+D(n)c_\mu\zeta_\mu+Ac_\mu c_\nu\pi_{\mu\nu} \tag{22}$$

(using the summation convention on repeated indices), where

$$B(n)=\frac{8\left[\left(1+\frac{3}{2}\varepsilon\right)-n\left(1+\frac{1}{2}\varepsilon\right)\right]}{\varepsilon^2(2-\varepsilon)}, \tag{23}$$

$$D(n)=\frac{8\left[n-\left(1+\frac{1}{2}\varepsilon\right)\right]}{\varepsilon^2(2-\varepsilon)}, \tag{24}$$

and

$$A=\frac{3}{\varepsilon\left(1+\frac{1}{2}\varepsilon\right)}. \tag{25}$$

The equilibrium exists for $0<\varepsilon<2$, that is, $0<T<1$ and $u^2<4(1-T)$.

V. LONG-WAVELENGTH STABILITY

We now consider long-wavelength small amplitude plane waves in a uniform gas, in order to identify the viscosity, hyperviscosity, thermal diffusivity, and hyperdiffusivity from the wave damping. The conditions for numerical stability to long-wavelength perturbations, which requires the damping to be positive, will then be obtained.

We use the dissipative fluid equations [Eq. (14)] with $f=f_0+f_1$, where $f_0=\rho_0 F_0(1+g_0)$ corresponds to a time-independent uniform state with $\rho=\rho_0$, $\vec{u}=\vec{u}_0$, and $\varepsilon=\varepsilon_0$, which are all constants, and

$$f_1=(\rho_0 F_1+\rho_1 F_0)(1+g_0)+\rho_0 F_0 g_1, \tag{26}$$

which contains the perturbed fluid variables ρ_1, $\vec{u}_1$, and ε_1. The fluid perturbations are assumed to have the form of plane waves, $\propto\exp(i\vec{k}\cdot\vec{x}-i\omega t)$, so that the perturbed fluid equations become

$$\sum_{\vec{c}}\ \psi(\omega-\vec{k}\cdot\vec{c})f_1=\sum_{p=2}^{4}\ (-i)^{p-1}\beta_p\sum_{\vec{c}}\ \psi(\omega-\vec{k}\cdot\vec{c})^p f_1, \tag{27}$$

where the β_p's are given by Eq. (15). We now choose ψ to be one of the quantities $\{1,\vec{c}-\vec{u}_0,c^2/2-\mathcal{E}_0\}$. We determine the wave frequencies in the form $\omega=\omega_1+\omega_2+\omega_3+\omega_4$, where $\omega_1\propto k$, $\omega_2\propto\beta_2 k^2$, $\omega_3\propto\beta_3 k^3$, and $\omega_4\propto\beta_4 k^4$, where the wave number k is assumed to be small.

The discretization error terms on the right-hand side of this equation proportional to β_2 represent viscosity and heat conduction. We can choose α so that β_2 is small, which will make the viscosity small. Choosing

$$\alpha=\tfrac{4}{3}-\epsilon, \tag{28}$$

where $\epsilon\ll 1$, we have

$$\beta_2\simeq\tfrac{9}{4}\epsilon\Delta t,\quad \beta_3\simeq\tfrac{1}{3}\Delta t^2,\quad \beta_4\simeq-\tfrac{1}{4}\Delta t^3. \tag{29}$$

The hyperviscosity (defined in Sec. VI) is proportional to $-\beta_4$, which remains positive as ϵ (and the viscosity) goes to zero. We formally take ϵ in Eqs. (28) and (29) to be of order Δt, so that $\beta_2\sim\beta_3\sim\Delta t^2$. Corrections to the frequency ω proportional to β_2, β_3, and β_4 can then be neglected on the right hand side of Eq. (27).

The solutions of Eq. (27) correspond to two transverse waves, or shear waves, and three longitudinal waves. Of the latter, two are sound waves and the other is the thermal wave. The shear wave damping will determine the shear viscosity and hyperviscosity, and the thermal wave damping will determine the thermal diffusivity and hyperdiffusivity. The sound wave dispersion relation will determine the anomalous dispersion, proportional to β_3.

VI. VISCOSITY AND HYPERVISCOSITY

We now determine the damping of a shear wave, including terms of order $\beta_2 k^2$ (viscosity) and terms of order $\beta_4 k^4$ (hyperviscosity). Assuming $u_0=0$, $\rho_1=0$, $\hat{k}\cdot\vec{u}_1=0$, and $\varepsilon_1=0$, using $\psi=\vec{c}\cdot\vec{u}_1$ in Eq. (27), the frequency is $\omega=-i\nu_2 k^2$, where

$$\nu_2=\beta_2 T_0. \tag{30}$$

By comparing with the damping rate obtained from the Navier-Stokes equations, we identify ν_2 with the kinematic viscosity. Lattice symmetry ensures isotropy for this result: it is independent of the direction of wave propagation. We note that ν_2 is positive for $\alpha<4/3$. The frequency correction proportional to β_3 is zero.

We now take the wave propagation direction to be along the x axis; then the frequency correction proportional to β_4 is $\omega_4=-i\nu_4 k^4$, where

$$\nu_4=-\beta_4 T_0. \tag{31}$$

We define ν_4 as the hyperviscosity; there is no corresponding term in the Navier-Stokes equations. Lattice symmetry does not ensure isotropy for this result, so it would depend on the direction of wave propagation. We note that ν_4 is positive for $16/15<\alpha<4/3$.

When a shear wave is superimposed on a nonzero background flow with velocity u_0, and we take the direction of wave propagation to be the same as that of the flow, we obtain a flow-velocity-dependent modification of the viscosity:

$$\omega=-i\nu_2 k^2\left[1-\frac{5}{6}\frac{u_0^2}{T_0}\right]. \tag{32}$$

For stability, we must therefore require

$$u_0^2/T_0<6/5. \tag{33}$$

This flow velocity dependence occurs because of the lattice and the particular equilibrium DF we used, and restricts the Mach number to be less than about 0.9, for numerical stability. A more restrictive condition is obtained in Sec. VIII from the stability of thermal waves.

VII. LONGITUDINAL WAVES

Taking $\vec{u}_0$, $\vec{k}$, and $\vec{u}_1$ to be in the x direction, Eq. (27) becomes three equations for the perturbed fluid variables $U=(\rho_1/\rho_0,u_1,\varepsilon_1)$, of the form

$$MU=NU, \tag{34}$$

where MU contains the nondissipative terms in the perturbed fluid equations and NU contains the dissipative terms, the

right hand side of Eq. (27). The lowest order frequencies, keeping only terms of order k, are the roots of $\det M=0$,

$$\omega_1 = k u_0 \quad \text{(thermal wave)} \tag{35}$$

or

$$\omega_1 = k u_0 \pm k c_s \quad \text{(sound waves)}, \tag{36}$$

where $c_s=(\gamma T_0)^{1/2}$ is the sound speed, with the specific heat ratio $\gamma \equiv (c_v+1)/c_v = 3/2$. These results are independent of the direction of wave propagation, and do not depend on the choice of equilibrium DF. The right hand side of Eq. (34) can be included using standard matrix perturbation theory, to obtain the frequency corrections proportional to β_2, β_3, and β_4.

VIII. THERMAL DIFFUSIVITY AND HYPERDIFFUSIVITY

Thermal waves have temperature and density perturbations which are such that the pressure perturbation is zero. By considering the damping of thermal waves with zero background flow, we find $\omega = -i\chi_2 k^2$ where

$$\chi_2 = \tfrac{1}{3}\beta_2 T_0 . \tag{37}$$

By comparison with solutions of the Navier-Stokes equations, we identify χ_2 as the thermal diffusivity. Since $\nu_2/\chi_2 = 3$, the thermal diffusivity is positive whenever the viscosity is positive. This ratio, the Prandtl number, would be closer to realistic values for diatomic gases if a more complicated equilibrium DF were used, which will be the subject of another paper.

A perturbation calculation including the β_3 and β_4 terms in Eq. (34) can be carried out, as with the shear wave. The frequency correction proportional to β_3 is zero. The next frequency correction is $\omega_4 = -i\chi_4 k^4$, where χ_4 is defined as the hyperdiffusivity and is given by

$$\chi_4 = -\frac{5}{3}\beta_4 T_0 (1-T_0), \tag{38}$$

which is positive for $0<T_0<1$.

A thermal wave with nonzero background flow was also considered. Although the expression for the flow-velocity-dependent thermal diffusivity is complicated, the result can be stated simply. In order for the thermal diffusivity to be positive, the Mach number must be less than a critical value M_c, which depends on temperature. For $0.4<T_0<0.8$, for example, the critical Mach number is in the range $0.48 < M_c < 0.62$.

IX. COMPARISON WITH THE LATTICE BOLTZMANN METHOD

The LB algorithm is as follows: given $f(\vec{x},\vec{c},t-\Delta t)$, the advected distribution function is

$$f_*(\vec{x},\vec{c}) = f(\vec{x}-\vec{c}\Delta t,\vec{c},t-\Delta t), \tag{39}$$

and the collision step is given by

$$f(\vec{x},\vec{c},t) = \omega f^{\mathrm{eq}} + (1-\omega) f_*(\vec{x},\vec{c}), \tag{40}$$

where ω is the collisional overrelaxation parameter, and f^{eq} is the equilibrium DF.

We now show how the fluid equations are obtained using the LB algorithm, and derive the parameters β_2, β_3, and β_4 which determine the long-wavelength damping and stability. Using Eq. (40), a Taylor expansion of f_* yields

$$f \simeq f^{\mathrm{eq}} + \frac{(1-\omega)}{\omega}\sum_{p=1}^{3}\frac{(-\Delta t)^p}{p!} d^p f, \tag{41}$$

where $d = \partial/\partial t + \vec{c}\cdot\vec{\nabla}$. By expanding f consistently in powers of Δt, we obtain

$$f = f^{\mathrm{eq}} - \frac{(1-\omega)}{\omega}\Delta t d f^{\mathrm{eq}} + \left[\frac{(1-\omega)}{2\omega} + \frac{(1-\omega)^2}{\omega^2}\right]\Delta t^2 d^2 f^{\mathrm{eq}} - \left[\frac{(1-\omega)}{6\omega} + \frac{(1-\omega)^2}{\omega^2} + \frac{(1-\omega)^3}{\omega^3}\right]\Delta t^3 d^3 f^{\mathrm{eq}}, \tag{42}$$

neglecting terms of order Δt^4.

We then obtain the fluid equations in the form given by Eq. (14) with the definitions

$$\beta_2 = \left(\frac{1}{\omega} - \frac{1}{2}\right)\Delta t, \tag{43}$$

$$\beta_3 = -\left[\left(\frac{1}{\omega}-1\right)^2 + \frac{1}{\omega} - \frac{5}{6}\right]\Delta t^2, \tag{44}$$

$$\beta_4 = \left[\left(\frac{1}{\omega}-1\right)^3 + \frac{3}{2}\left(\frac{1}{\omega}-1\right)^2 + \frac{7}{12}\left(\frac{1}{\omega}-1\right) + \frac{1}{24}\right]\Delta t^3, \tag{45}$$

which can be compared with Eq. (15). From this point, the long-wavelength analysis is the same for both algorithms; in particular, the viscosity is proportional to β_2 and the hyperviscosity is proportional to $-\beta_4$.

The viscosity is positive when $\omega<2$. We can choose ω so that β_2 and the viscosity are small: with

$$\omega = 2-\epsilon, \tag{46}$$

where $\epsilon \ll 1$, we have

$$\beta_2 \simeq \epsilon\Delta t/4, \quad \beta_3 \simeq \Delta t^2/6, \quad \beta_4 \simeq -\epsilon\Delta t^3/24. \tag{47}$$

Note that $-\beta_4$, which is proportional to the hyperviscosity, is also small when the viscosity is small. We take this as an indication that short waves generated by nonlinearity may be unstable using the LB algorithm.

This is the most important difference between the AOR and LB algorithms, since β_4 does not go to zero when β_2 goes to zero using the AOR algorithm; see Eq. (29). When the viscosity and thermal diffusivity are very small, short-wavelength numerical instabilitites will be effectively

damped using the AOR algorithm, but not using the LB algorithm. Thus we expect the AOR algorithm to be numerically more stable than the LB algorithm, and this is evident in the simulation results described in Sec. XI.

X. SHORT-WAVELENGTH STABILITY

We now consider the stability of the basic AOR algorithm, [Eqs. (2)–(4)] to perturbations with short wavelengths, assuming zero unperturbed flow, $u_0=0$, and propagation in the x direction. We use $f=f_0+f_1$ where f_0 is the equilibrium DF for uniform density and temperature, and $f_1(x,\vec{c},t_n)\propto\lambda^n\exp(ikx)$. Stability requires

$$|\lambda|\leqslant 1. \tag{48}$$

Then we have

$$\sum_{\vec{c}}\ \psi(f_1-f_{1*})=0, \tag{49}$$

where the perturbed equilibrium DF f_1 is given by Eq. (26) and the perturbed advected DF is

$$f_{1*}=f_1\left[\frac{\alpha}{\lambda}\exp(-i\Theta c_x)+\frac{(1-\alpha)}{\lambda^2}\exp(-2i\Theta c_x)\right], \tag{50}$$

where $\Theta=k\Delta x$, Δx is the grid spacing, and c_x is now normalized to $\Delta x/\Delta t$, where $\Delta t=t_{n+1}-t_n$ is the time step. In the following, we consider only the shortest possible wavelength, $k\Delta x=\pi$.

For transverse modes, $\psi=c_y$, and we obtain the dispersion relation

$$\lambda^2-\alpha\mu\lambda+\alpha-1=0, \tag{51}$$

where

$$\mu=\sum_{\vec{c}}\ c_y^2F_0[B_0+3T_0D_0](-1)^{c_x}=1-2T_0, \tag{52}$$

where B_0 and D_0 are given by Eqs. (23) and (24) with $\mathcal{E}=\mathcal{E}_0$. These modes are stable for $0\leqslant T_0\leqslant 1$.

For longitudinal modes, λ is again determined by Eq. (51), where now μ is an eigenvalue of the matrix h, where

$$h_{ij}=\sum_{\vec{c}}\ \psi_i g_j F_0(-1)^{c_x}, \tag{53}$$

with $g_1=1$, $g_2=[B_0+3T_0D_0]c_x$ and $g_3=(\partial F_0/\partial\mathcal{E}_0)/F_0$.

The matrix h has a simple block structure: the eigenvalues which correspond to nonzero ρ_1 and $\mathcal{E}_1$ are stable for $0\leqslant T_0\leqslant 1$. The eigenvalue which corresponds to nonzero u_1 is stable if $T_0\geqslant\frac{1}{3}$. We thus obtain the temperature range for stability;

$$1/3\leqslant T_0\leqslant 1. \tag{54}$$

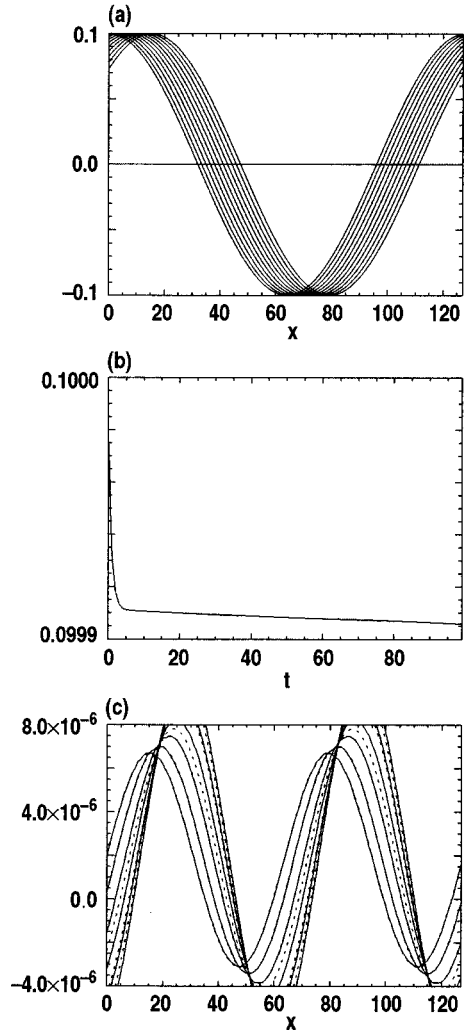

FIG. 1. (a) Transverse velocity as a function of x for a shear wave, using the AOR algorithm. (b) Time dependence of the amplitude of the shear wave. (c) Temperature perturbation due to heating by a shear wave.

XI. SIMULATION RESULTS USING THE LATTICE4 CODE

We have written a computer code, LATTICE4, which uses the AOR algorithm and, optionally, the LB algorithm, in order to verify the stability of the AOR algorithm and to compare the algorithms. The initial conditions for the results given below correspond to the types of waves we have discussed above. Periodic boundary conditions were used. The computed fluid moments are analyzed by fitting them to propagating waves, in order to determine the phase velocities and damping rates. The Mach number is defined in terms of the unperturbed flow velocity and temperature: $M\equiv u_0/(\gamma T_0)^{1/2}$. The overrelaxation parameter α was determined from $\alpha=(4/3)(1-\beta_2)/(1-(2/3)\beta_2)$, with β_2 being an input parameter. The spatial coordinates were normalized so that $\Delta x=1$. For most of the cases discussed below, the wave number was $k=0.049$, for which the wavelength is equal to the length of the simulation, $L=128$.

Figure 1 shows the transverse velocity and the temperature perturbation, obtained using the AOR algorithm, for a sinusoidal shear flow convected by a uniform flow at Mach 0.2, for $T_0=0.4$. We set $\beta_2=0$, so the viscosity should be zero; the wave is damped very slightly by hyperviscosity. The initial rapid damping is due to the fact that only one past time is available for the first time step, so the parameter α must be set equal to unity for the first step; a few time steps are required before the damping rate predicted for the AOR algorithm is achieved. The very small temperature perturbation shown is due to hyperviscous heating. The dotted curves show the fitted sinusoidal waves with twice the wave number, which is consistent with the heating being nonlinear in the velocity derivatives.

When the LB algorithm is used, shear waves are unstable, as shown in Fig. 2, for the same parameters used in Fig. 1. The growing short wave temperature perturbations cause the shear wave to break up, starting at around the last time shown.

Shear waves simulated using the AOR algorithm with different wave numbers were used to fit the damping rate Γ to

$$\Gamma(k)=\nu_2 k^2+\nu_4 k^4 \tag{55}$$

for $T_0=0.4$, $M=0.1$ and two different values of β_2. The viscosity ν_2 and hyperviscosity ν_4 agree reasonably well with the analytical results [Eqs. (30), (31), (32), and (29)]. For $\beta_2=0.10$, the code results are $\nu_2=3.96\times10^{-2}$ and $\nu_4=6.3\times10^{-2}$, to be compared with the analytical results $\nu_2=3.95\times10^{-2}$ and $\nu_4=7.7\times10^{-2}$. For $\beta_2=0.01$, the code results are $\nu_2=3.98\times10^{-3}$ and $\nu_4=9.2\times10^{-2}$, to be compared with the analytical results $\nu_2=3.95\times10^{-3}$ and $\nu_4=9.8\times10^{-2}$.

The viscosity was determined for different Mach numbers, for two different values of β_2, for $T_0=0.4$. The Mach number dependence agrees reasonably well with Eq. (32). For $\beta_2=0.10$, we find $\nu_2(M)=0.042(1.0-0.48M^2)$, to be compared with the analytical results $\nu_2(M)=0.040(1.0-1.25M^2)$. For $\beta_2=0.01$, we find $\nu_2(M)=0.0044(1.0-0.45M^2)$, to be compared with the analytical result $\nu_2(M)=0.004(1.0-1.25M^2)$.

The damping rates for thermal waves convected by a uniform flow at Mach 0.1 for $T_0=0.4$, using the AOR algorithm, were calculated for different values of β_2, and fitted to

$$\Gamma(k)=\chi_2 k^2+\chi_4 k^4. \tag{56}$$

The thermal diffusivity χ_2 and hyperdiffusivity χ_4 agree reasonably well with the analytical results neglecting finite M corrections [Eqs. (37) and (38)]. For $\beta_2=0.10$, we find $\chi_2=1.2\times10^{-2}$ and $\chi_4=5.6\times10^{-2}$, to be compared with the analytical results $\chi_2=1.3\times10^{-2}$ and $\chi_4=7.7\times10^{-2}$. For $\beta_2=0.01$, we find $\chi_2=1.1\times10^{-3}$ and $\chi_4=4.6\times10^{-2}$, to be compared with the analytical results $\chi_2=1.3\times10^{-3}$ and $\chi_4=9.8\times10^{-2}$.

When the LB algorithm was used, thermal waves were found to be generally unstable, with the temperature and density perturbations growing in time, especially when the thermal diffusivity is very small, and for shorter wavelengths. This may be the cause of the instability which breaks up shear waves, when using the LB algorithm, as shown in Fig. 2. That is, the temperature perturbation generated by heating is unstable.

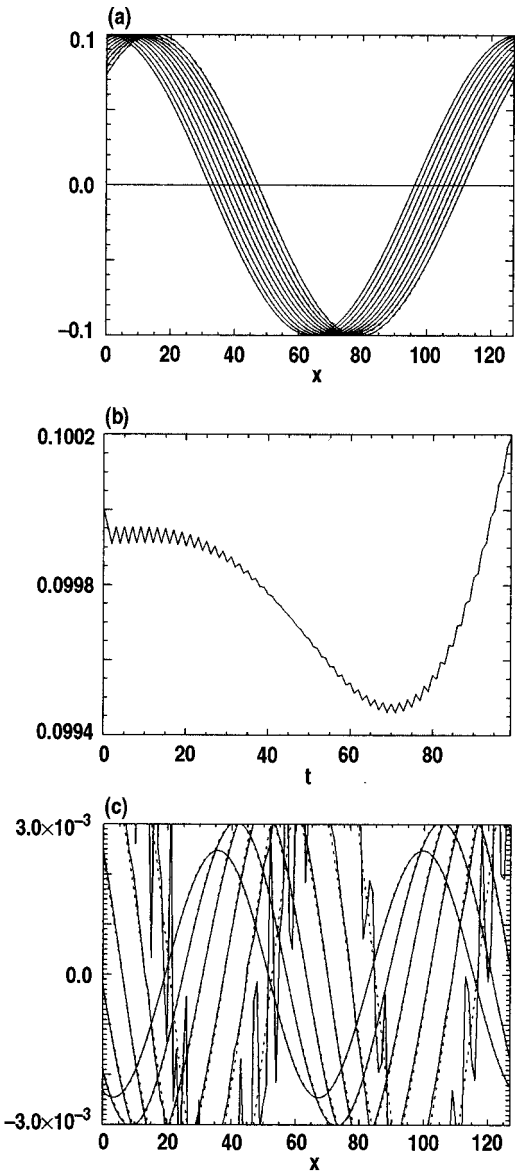

FIG. 2. (a) Transverse velocity as a function of x for a shear wave, using the LB algorithm. (b) Time dependence of the amplitude of the shear wave. (c) Temperature perturbation due to heating by the shear wave, showing the growth of unstable short-wavelength temperature perturbations for later times.

The dispersion curves for small amplitude ($\rho_1/\rho_0=0.01$) sound waves were determined for two different temperatures $T_0=0.35$ and 0.45, and for flow velocities $u_{0x}=-0.15$, 0.0, and 0.15, using the AOR algorithm. The wave frequency was fitted to a function of the wave number,

$$\omega=ku_0+kc_s+\delta_3 k^3, \tag{57}$$

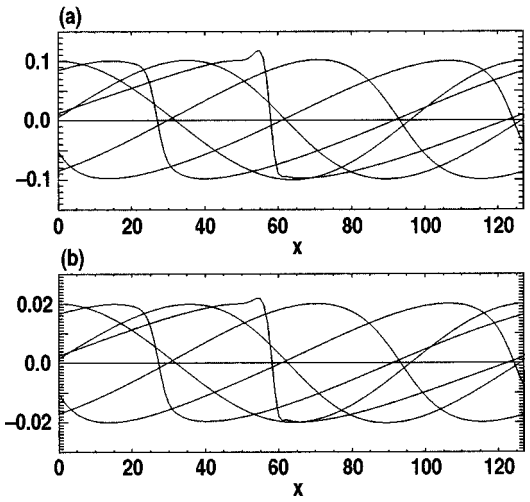

FIG. 3. Steepening of a large amplitude sound wave, showing (a) the density perturbation and (b) the temperature perturbation, as functions of x for different times, up to the predicted steepening time.

where $c_s=(\gamma T_0)^{1/2}$ is the sound speed, and δ_3 is the anomalous dispersion coefficient. The measured specific heat ratio is in excellent agreement with the theoretical value $\gamma=1.5$, with relative errors less than 1.6×10^{-4}. The anomalous dispersion term proportional to δ_3 gives a very small correction to the phase velocity, of about 3×10^{-3}. When the LB algorithm was used, sound waves were found generally to be unstable, for most of the parameters used in these tests.

As a further test of the realism of the simulation, a larger amplitude ($\rho_1/\rho_0=0.1$) sound wave propagating upstream through a uniform flow in the $-x$ direction at Mach 0.2 with temperature $T_0=0.4$ was used. It was found to steepen as expected from gas dynamic theory. An estimate of the steepening time, $t_s\sim1/(ku_1)$, is about 250 time steps. In Fig. 3 we show the density and temperature waves every 50 time steps, as calculated by the code, for 250 time steps. The sound wave becomes very steep in that time, as expected from gas dynamic theory.

XII. DISCUSSION AND CONCLUSIONS

We have presented an algorithm for compressible fluid simulations, which is a modification of the standard lattice Boltzmann algorithm [1]. We have used the LATTICE4 code to demonstrate the greatly improved numerical stability obtained with our algorithm, compared with the standard algorithm.

The AOR algorithm is similar to the Chapman-Enskog method for deriving fluid equations from the Boltzmann equation. The Boltzmann equation is not used, however; thus it is different from the LB method, which uses a discretized form of the Boltzmann equation. At each time step the velocity distribution function is set equal to the collisional equilibrium distribution function. With an appropriate choice of the equilibrium distribution function, the fluid variables evolve like solutions of the ideal fluid equations, except for the small effects of nonzero spatial and temporal discreteness. The leading order effects of discreteness appear as viscosity and thermal diffusivity, and are similar to real gas dissipation effects, which are a result of nonzero mean free path. The size of these dissipation effects is controlled by the advection overrelaxation parameter α, and can be made arbitrarily small. Higher-order effects of discreteness appear as anomalous dispersion, and as hyperviscosity and hyperdiffusivity, which have a stabilizing effect in the AOR algorithm which is absent in the LB algorithm, when the viscosity and thermal diffusivity are very small.

The damping rates of shear waves and thermal waves in the simulated fluid were used to determine viscosity, thermal diffusivity, hyperviscosity, and hyperdiffusivity. A reasonably good agreement with analytical predictions was obtained. Also, the anomalous dispersion of sound waves, determined by measurements of the phase velocity, was shown to be very small, and sound waves were shown to steepen, in agreement with gas dynamic theory.

Lattice methods usually simulate a fluid with unrealistic properties resulting from the use of a lattice, i.e., lattice artifacts [10]. In order not to confuse the stability issue, we used a type of collisional equilibrium which has exactly zero artifacts at the Euler level of description, i.e., the Euler equations are obtained exactly in the limit of zero time step. This equilibrium has a simple analytical form which is possible because of the choice of the four-dimensional face-centered hypercubic (FCHC) lattice [8], and which makes possible analytical predictions of stability to long wave perturbations.

Some lack of realism remains, however, because of the choice of this equilibrium and the use of only three energy states: the specific heat ratio is $\frac{3}{2}$, not $\frac{7}{5}$, the viscosity and thermal diffusivity decrease with increasing Mach number, and the Prandtl number is unrealistically large. These unrealistic features can be eliminated by a different choice of collisional equilibrium and the inclusion of more energy states, which will be the subject of a future paper.

The dependence of the wave damping rates, on background temperature, and flow velocity, is a consequence of the particular collisional equilibrium distribution function used. When this is given by Eqs. (19), (21), and (22), the requirement of positive wave damping (the most restrictive being the thermal wave) requires the Mach number to be less than 0.48, for temperatures in the range $0.4<T_0<0.8$, for example. The temperature is only required to be in the range $1/3\leqslant T_0<1$, for short wavelength stability.

Lattice methods should be fast, compared with conventional methods used in computational fluid dynamics, since they involve relatively simple operations. They are highly parallel when implemented using domain decomposition on massively parallel computers, because communication between processors is only required for particle advection near subdomain boundaries. The code LATTICE4, which was used to obtain the simulation results given in this paper, uses domain decomposition and the message-passing interface (MPI) [11], a standard method for interprocessor communication. Figure 3 was made using the data calculated on a Linux Beowulf cluster [12] using 16 processors.

We believe that this technique would be useful in simulating compressible fluid flows under conditions where the effects of viscosity and thermal diffusivity are very small. The most limiting feature is the limitation to Mach numbers less than unity; this will be addressed in future work. In order to show that our algorithm is useful in real engineering problems, it must, of course, be tested in engineering simulations, not just the simple wave tests we have done. This will be the focus of future work.

ACKNOWLEDGMENTS

We have benefitted from discussions with K. Molvig on the LB algorithm. This work was supported by internal GA research funds.

[1] Y. H. Qian, S. Succi, and S. A. Orsag, Annu. Rev. Comput. Phys. **3**, 195 (1995).
[2] G. R. McNamara, A. L. Garcia, and B. J. Alder, J. Stat. Phys. **81**, 395 (1995).
[3] J. D. Sterling and S. Chen, J. Comput. Phys. **123**, 196 (1996).
[4] P. Pavlo, G. Vahala, L. Vahala, and M. Soe, J. Comput. Phys. **139**, 79 (1998).
[5] L. Vahala, D. Wah, G. Vahala, J. Carter, and P. Pavlo, Phys. Rev. E **62**, 507 (2000).
[6] H. Chen and C. Teixeira, Comput. Phys. Commun. **129**, 21 (2000).
[7] H. Chen, C. Teixeira, and K. Molvig, Int. J. Mod. Phys. C **8**, 675 (1997).
[8] S. Wolfram, J. Stat. Phys. **45**, 471 (1986).
[9] I. V. Karlin and S. Succi, Phys. Rev. E **58**, R4053 (1998).
[10] C. Teixeira, Ph.D. thesis, Massachusetts Institute of Technology, 1992, (unpublished).
[11] W. Gropp, E. Lusk, and A. Skjellum, *Using MPI* (MIT Press, Cambridge, MA, 1999).
[12] http://www.beowulf.org.