JOURNAL OF COMPUTATIONAL PHYSICS 17, 311–331 (1975)

A Generalized Arbitrary Lagrangian–Eulerian Method for Incompressible Flows with Sharp Interfaces

ROBERT K.-C. CHAN

Science Applications, Inc., La Jolla, California 92037

Received June 20, 1974; revised December 2, 1974

A generalized mixed Lagrangian–Eulerian computing technique for incompressible fluid flows is presented. The present method combines the advantages of two existing techniques such that conservation of both linear and angular momentums is insured. New features include a damping formula, which effectively prevents the occurrence of alternating errors, and the formulation of discretized boundary conditions for a sharp fluid–fluid interface. Several examples are given to demonstrate the usefulness of the present method.

I. INTRODUCTION

Lagrangian or mixed Lagrangian–Eulerian computing techniques have been used by several authors in treating time-dependent incompressible flow problems in which free surface and/or material interfaces are present. The ALE (Arbitrary Lagrangian–Eulerian) method, as used by Hirt [1] for example, provides a tool for the study of fairly complex flow configurations where pure Eulerian or Lagrangian methods cannot be applied. Each cycle of the ALE method consists of two phases: In phase 1, the calculation is purely Lagrangian, similar to the treatment in the LINC [2] method. In phase 2, a rezone procedure is used so that a desirable mesh configuration can be maintained. The phase 1 calculation is based on a finite-difference representation of a set of conservation equations written in integral form. As in many other incompressible computing techniques, a simultaneous solution of the discretized pressure field is required to permit the stepwise integration of the flow field with respect to time.

Brennen and Whitney [3], on the other hand, presented a pure Lagrangian scheme in which explicit use of the pressure is avoided by describing the fluid dynamics in terms of transport equations for the velocity circulation. At a given instant of time, a relaxation technique is used to solve the velocity field so that a negligibly small velocity divergence and the correct amount of circulation are obtained for each computing cell.

There are both similarities and differences between phase 1 of the ALE method and the method of Brennen and Whitney, which shall be referred to as the BW

method hereafter. For example, it can be shown that the iterative solution of the velocity field in the BW method is somewhat similar to a simultaneous relaxation of pressure and the velocity components in the ALE method. However, we have found the relaxation procedure in ALE to be more efficient than that in the BW method whenever the finite-difference mesh contains cells of various sizes and aspect ratios. On the other hand, to our knowledge, no finite-difference approximations for the resultant pressure force $\int_S p\mathbf{n}\, dS$ have been derived such that $\nabla \times \nabla p = 0$ is rigorously satisfied in the ALE framework. Consequently, there is no guarantee that an irrotational fluid motion will remain irrotational in the absence of vorticity production. An example in which the velocity circulation grows unboundedly is described in Section 3.1. This difficulty is circumvented in the BW method by obtaining, for any time step, the correct velocity circulation for each cell and then solve the velocity field to satisfy the correct circulation and zero divergence. In addition, the way various boundary conditions are formulated in the BW method eliminates the ambiguity in choosing a satisfactory integration path for $\int_S p\mathbf{n}\, dS$ at mesh points that lie on the boundaries of the fluid domain. On the other hand, the advantage of keeping track of the pressure field is two-fold: (1) It can be used to "predict" the velocity field for the next time step, thus speeding up the subsequent relaxation procedure; and (2) for some types of problems the pressure field is part of the solution being sought.

This paper describes a generalized arbitrary Lagrangian–Eulerian (GALE) method which has evolved from the two methods mentioned above and represents an improvement over its predecessors in several respects. In what follows we show how the desirable features of these methods are combined to advantage. A generalization of the BW method allows velocity circulation to be generated by density variations and/or by tangential stresses. Furthermore, a procedure has been developed which treats an inviscid (free-slip) material interface. This method can handle any density ratio or discontinuous tangential velocity with the correct stress conditions coupling the two separate fluid regions.

A numerical instability which has been found in the LINC, ALE and BW methods is the "alternating error," a growing nonphysical disturbance whose wavelength is equal to two finite-difference mesh spacings. Previous attempts [4, 5] to introduce artificial damping produced excessive damping of the solution to some types of problems. In this paper an effective alternative procedure is described, which yields negligible amount of nonphysical damping.

II. The GALE Method

In the following subsections we first discuss the basic procedure of the GALE technique and then treat separately in more detail several important aspects. We

limit our present discussion to 2-dimensional problems in plane, Cartesian coordinates.

2.1. *Basic Computational Procedure*

To implement a finite-difference representation of the flow, the fluid domain is divided into a number of quadrilaterals or cells (Fig. 1) in such a way that physical boundaries such as the interface between two fluids, the boundary between a fluid and a rigid body, free surfaces, etc., coincide with the cell boundaries. The vertices of these cells are designated by the (i, j) suffix system (Fig. 2); and associated with each vertex are its spatial coordinates $(x_{i,j}, y_{i,j})$, velocity components $(u_{i,j}, v_{i,j})$ and density $\rho_{i,j}$. A special arrangement is needed when an interface of discontinuity is present. In this case, each fluid is regarded as a separate region and the interface can be represented by two lines, each belonging to the fluid adjacent to it. At corresponding boundary points, such as F_1 and F_2 in Fig. 1, boundary conditions to be described later must be satisfied.

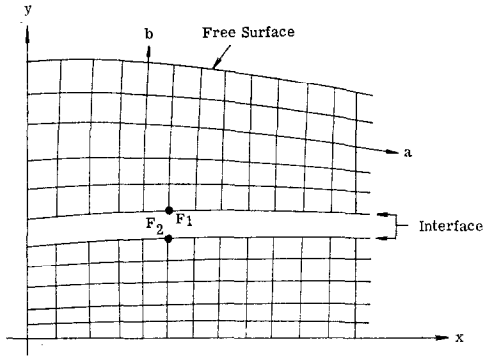

FIG. 1. General mesh set-up.

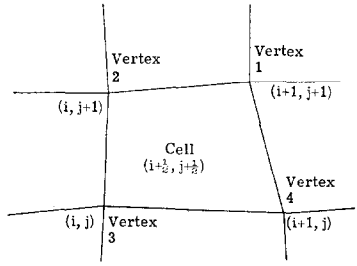

FIG. 2. A typical cell and suffix system.

A set of initial conditions on (x, y), (u, v) and ρ at each vertex is needed to begin a calculation. Then the vertices are moved to their new positions according to their instantaneous velocities through an increment in time δt:

$$\begin{aligned} x_{i,j}^{n+1} &= x_{i,j}^{n} + (\delta t/2)(u_{i,j}^{n} + u_{i,j}^{n+1}), \\ y_{i,j}^{n+1} &= y_{i,j}^{n} + (\delta t/2)(v_{i,j}^{n} + v_{i,j}^{n+1}), \end{aligned} \tag{1}$$

where the superscript n refers to the nth increment in the time. Note that in Eq. (1) the advanced-time velocities $u_{i,j}^{n+1}$ and $v_{i,j}^{n+1}$ are needed, so that a set of cell equations of motion and appropriate boundary conditions can be imposed.

One of the two governing equations for each typical cell as shown in Fig. 2 requires that the volume, or the area in 2-dimensional problems, $\mathscr{V}$ of the cell remain unchanged as the calculation proceeds from the nth step to the $(n + 1)$th step. Since in the model the cell vertices are connected with straight lines,

$$\mathscr{V}_{i+1/2,j+1/2}^{n} = \tfrac{1}{2}(x_{13}^{n} y_{24}^{n} - x_{24}^{n} y_{13}^{n}). \tag{2}$$

In most of this paper, the subscripts 1, 2, 3, and 4 are abbreviations for the vertices of the cell $(i + \frac{1}{2}, j + \frac{1}{2})$ (Fig. 2), and the shorthand notations $x_{13}^{n} \equiv x_1{}^{n} - x_3{}^{n}$, $x_{24}^{n} \equiv x_2{}^{n} - x_4{}^{n}$, etc., are used. The time rate of change in $\mathscr{V}$, by using Eqs. (1) and (2), can be expressed as

$$\begin{aligned} 4(\mathscr{V}_{i+1/2,j+1/2}^{n+1} - \mathscr{V}_{i+1/2,j+1/2}^{n})/\delta t = {} & [x_{13}^{n}(v_{24}^{n+1} + v_{24}^{n}) + y_{24}^{n}(u_{13}^{n+1} + u_{13}^{n}) \\ & - x_{24}^{n}(v_{13}^{n+1} + v_{13}^{n}) - y_{13}^{n}(u_{24}^{n+1} + u_{24}^{n})] \\ & + (\delta t/2)[(u_{13}^{n+1} + u_{13}^{n})(v_{24}^{n+1} + v_{24}^{n}) \\ & - (u_{24}^{n+1} + u_{24}^{n})(v_{13}^{n+1} + v_{13}^{n})] = R_1, \end{aligned} \tag{3}$$

where $R_1 = 0$ is the required incompressibility condition.

The second cell equation concerns the rotation of the cell. Let $\Gamma = \oint \mathbf{u} \cdot d\mathbf{s}$ represent the velocity circulation around the boundary of a cell. Assuming linear variation of the velocity components along the line segment between two vertices, we have

$$\Gamma_{i+1/2,j+1/2}^{n} = \tfrac{1}{2}(u_{13}^{n} x_{24}^{n} - u_{24}^{n} x_{13}^{n} + v_{13}^{n} y_{24}^{n} - v_{24}^{n} y_{13}^{n}). \tag{4}$$

For inviscid homogeneous fluid flows, $\Gamma_{i+1/2,j+1/2}^{n+1} = \Gamma_{i+1/2,j+1/2}^{n}$, and Eqs. (1) and (4) lead to

$$\begin{aligned} & u_{13}^{n+1}[x_{24}^{n} + (\delta t/2)(u_{24}^{n+1} + u_{24}^{n})] - u_{24}^{n+1}[x_{13}^{n} + (\delta t/2)(u_{13}^{n+1} + u_{13}^{n})] \\ & \quad + v_{13}^{n+1}[y_{24}^{n} + (\delta t/2)(v_{24}^{n+1} + v_{24}^{n})] - v_{24}^{n+1}[y_{13}^{n} + (\delta t/2)(v_{13}^{n+1} + v_{13}^{n})] \\ & \quad - 2(\Gamma_{i+1/2,j+1/2}^{n}) = R_2, \end{aligned} \tag{5}$$

where $R_2 = 0$ is the required circulation condition. For flows in which the density varies or the shear stresses are present, Eq. (5) must be generalized as described in Section 2.2.

The various boundary conditions, to be described in Section 2.3, and the cell equations (3) and (5) form a system of nonlinear algebraic equations with $(u_{i,j}^{n+1}, v_{i,j}^{n+1})$ as the unknowns, which may be solved by employing a successive over-relaxation procedure. To begin the iteration, the previous-timestep values $(u_{i,j}^{n}, v_{i,j}^{n})$ are used as a first approximation for $(u_{i,j}^{n+1}, v_{i,j}^{n+1})$. After finding u^{n+1} and v^{n+1} for all the vertices, Eq. (1) is used to advance each vertex to its new position in the (x, y) space. The process of solving for (u^{n+1}, v^{n+1}) and moving the vertices can be repeated to obtain the development of flow.

The procedure outlined above employs the Lagrangian description of fluid motion. Its chief advantage lies in that material interfaces are properly maintained and the lack of numerical instability associated with convection terms. Its disadvantage, however, is that the cells can be badly distorted or even inverted in highly strained motions. To remove this difficulty, an automatic rezoning procedure can be employed. The principle behind the rezoning technique is to adjust the position of vertices at the end of each time increment, so that a nearly optimum shape is always maintained for each cell. In making such adjustment, due consideration is given to the amount of flux which results from moving the mesh lines relative to the fluid. In the GALE method, use is made of a rezoning procedure which is operationally easier than the one given in [1]. The details of this technique are covered in Section 2.5.

2.2. *The Generalized Second Cell Equation*

For irrotational flows we simply specify $\Gamma_{i,j}^{n+1} = 0$ for each cell in the basic procedure described above. In general, however, nonzero $\Gamma_{i,j}^{n+1}$, can be produced by density variations or by shear stresses. To derive systematically an equation for computing the time rate of change of Γ, one could in principle obtain a differential expression for $d\Gamma/dt$, and then use finite-difference approximations. However, for internal consistency, we may generalize Eq. (5) using finite-difference expressions from the beginning.

Let (a, b) be a Lagrangian coordinate system, with the coordinate lines coinciding with the curvilinear finite-difference mesh lines (Fig. 1). We assume that the mesh is always set up so that the mapping from the (x, y) plane to the (a, b) plane and vice versa are nonsingular. Furthermore, we require that this mapping transforms every computational cell in the (x, y) plane into a unit square in the (a, b) plane. Note that a and b are continuous variables according to the definition above, but in the finite-difference framework quantities like $x_{i,j}$, $y_{i,j}$, $u_{i,j}$ and $v_{i,j}$ are defined only for integer values, $a = i$ and $b = j$.

The momentum equations can be written as [3]

$$\left(\rho\frac{du}{dt}-\rho g_x-\xi\right)\frac{\partial x}{\partial a}+\left(\rho\frac{dv}{dt}-\rho g_y-\eta\right)\frac{\partial y}{\partial a}=-\frac{\partial\varphi}{\partial a},$$
$$\left(\rho\frac{du}{dt}-\rho g_x-\xi\right)\frac{\partial x}{\partial b}+\left(\rho\frac{dv}{dt}-\rho g_y-\eta\right)\frac{\partial y}{\partial b}=-\frac{\partial\varphi}{\partial b}, \tag{6}$$

where ρ is the density; g_x and g_y are the x-and y-components of gravity, respectively; φ is the isotropic part of the normal stress; ξ and η are defined by

$$\xi \equiv (\partial\tau_{xx}/\partial x)+(\partial\tau_{xy}/\partial y).$$
$$\eta \equiv (\partial\tau_{yx}/\partial x)+(\partial\tau_{yy}/\partial y). \tag{7}$$

The deviatoric stress tensor components τ_{xx}, τ_{xy}, and τ_{yy} have the usual meanings [6].

Applying the first of Eqs. (6) to the vertices 1 and 2, as shown in Fig. 2, we obtain the finite-difference equation

$$\begin{aligned}\varphi_1^{n+1/2}-\varphi_2^{n+1/2}=\frac{1}{4}\Big[&\xi_1^{n+1/2}+\xi_2^{n+1/2}+(\rho_1+\rho_2)g_x-\rho_1\left(\frac{u_1^{n+1}-u_1{}^n}{\delta t}\right)\\&-\rho_2\left(\frac{u_2^{n+1}-u_2{}^n}{\delta t}\right)\Big]\cdot[(x_1{}^n-x_2{}^n)+(x_1^{n+1}-x_2^{n+1})]\\&+\frac{1}{4}\Big[\eta_1^{n+1/2}+\eta_2^{n+1/2}+(\rho_1+\rho_2)g_y-\rho_1\left(\frac{v_1^{n+1}-v_1{}^n}{\delta t}\right)\\&-\rho_2\left(\frac{v_2^{n+1}-v_2{}^n}{\delta t}\right)\Big]\cdot[(y_1{}^n-y_2{}^n)+(y_1^{n+1}-y_2^{n+1})].\end{aligned} \tag{8}$$

Similar expressions can also be obtained for $\varphi_2^{n+1/2}-\varphi_3^{n+1/2}$, $\varphi_3^{n+1/2}-\varphi_4^{n+1/2}$, and $\varphi_4^{n+1/2}-\varphi_1^{n+1/2}$. Eliminating the four φ's among the four equations thus obtained, and using Eq. (1), we get

$$\begin{aligned}&[\rho_1(u_1^{n+1}-u_1{}^n)-\rho_3(u_3^{n+1}-u_3{}^n)]\,x_{24}^{n+1/2}\\&-[\rho_2(u_2^{n+1}-u_2{}^n)-\rho_4(u_4^{n+1}-u_4{}^n)]\,x_{13}^{n+1/2}\\&+[\rho_1(v_1^{n+1}-v_1{}^n)-\rho_3(v_3^{n+1}-v_3{}^n)]\,y_{24}^{n+1/2}\\&-[\rho_2(v_2^{n+1}-v_2{}^n)-\rho_4(v_4^{n+1}-v_4{}^n)]\,y_{13}^{n+1/2}\\&-\delta t[g_x(\rho_{13}x_{24}^{n+1/2}-\rho_{24}x_{13}^{n+1/2})+g_y(\rho_{13}y_{24}^{n+1/2}-\rho_{24}y_{13}^{n+1/2})]\\&-\delta t[\xi_{13}^{n+1/2}x_{24}^{n+1/2}-\xi_{24}^{n+1/2}x_{13}^{n+1/2}+\eta_{13}^{n+1/2}y_{24}^{n+1/2}-\eta_{24}^{n+1/2}y_{13}^{n+1/2}]=R_2,\end{aligned} \tag{9}$$

where

$$x_{24}^{n+1/2} = x_{24}^{n} + (\delta t/4)(u_{24}^{n+1} + u_{24}^{n}),$$
$$x_{13}^{n+1/2} = x_{13}^{n} + (\delta t/4)(u_{13}^{n+1} + u_{13}^{n}),$$

and similarly for $y_{24}^{n+1/2}$ and $y_{13}^{n+1/2}$. Also, $\rho_{13} = \rho_1 - \rho_3$, $\xi_{13}^{n+1/2} = \xi_1^{n+1/2} - \xi_3^{n+1/2}$ and similarly for ρ_{24} and $\xi_{24}^{n+1/2}$. This is the fundamental equation in the present method. On the left-hand side the terms in the fifth pair of brackets are productions due to gravity and inhomogeneity of the fluid, and terms in the sixth pair of brackets represent the contributions from shear stresses. The gravity contributions have also been derived by Brennen and Whitney [3] in an integral form.

It can be easily shown that when $\rho =$ constant and $\xi = \eta = 0$, Eq. (9) reduces to Eq. (5). The quantities $\xi^{n+1/2}$ and $\eta^{n+1/2}$ are evaluated at the vertices of the mesh. For most applications, except for creeping flows where implicitness is desired to increase δt, using ξ^n and η^n instead of the more complicated $\xi^{n+1/2}$ and $\eta^{n+1/2}$ is quite adequate. A method of computing ξ and η is found in the Appendix.

2.3. *Boundary Conditions*

The conditions at a free surface and at a rigid boundary are the same as in [3]. We simply remark that these boundary conditions can all be cast in the general form

$$Au^{n+1} + Bv^{n+1} + C = R_3, \tag{10}$$

where u^{n+1} and v^{n+1} are the velocity components to be determined at the boundary point under consideration, and $R_3 = 0$ when correct values of u^{n+1} and v^{n+1} are used.

In addition, we describe here a method to treat an inviscid (free-slip) material interface, which occurs in the formulation of many multilayer flow problems. Due to a discontinuity in tangential velocity at the interface, the (u, v) values are in general discontinuous there. Let (u_1, v_1) be the velocity components associated with the vertex F_1 and (u_2, v_2) be those associated with F_2 in Fig. 1. Also, let ρ_1 and ρ_2 be the densities of the upper and the lower fluids, respectively. Then applying the first of Eqs. (6) without the shear-stress terms, we obtain

$$\left(\frac{du_1}{dt} - g_x\right)\frac{\partial x}{\partial a} + \left(\frac{dv_1}{dt} - g_y\right)\frac{\partial y}{\partial a} + \frac{1}{\rho_1}\frac{\partial \varphi_1}{\partial a} = 0 \tag{11}$$

for the upper fluid, and

$$\left(\frac{du_2}{dt} - g_x\right)\frac{\partial x}{\partial a} + \left(\frac{dv_2}{dt} - g_y\right)\frac{\partial y}{\partial a} + \frac{1}{\rho_2}\frac{\partial \varphi_2}{\partial a} = 0 \tag{12}$$

for the lower fluid. In the above equations, a is a Lagrangian coordinate line conforming to the interface (Fig. 1); $\partial x/\partial a$ and $\partial y/\partial a$ are evaluated at the interface.

Since the dynamics require the normal stress to be continuous at the interface, namely $\varphi_1 = \varphi_2$, $\partial\varphi/\partial a$ can be eliminated between Eqs. (11) and (12), giving

$$\left(\rho_1 \frac{du_1}{dt} - \rho_2 \frac{du_2}{dt}\right)\frac{\partial x}{\partial a} + \left(\rho_1 \frac{dv_1}{dt} - \rho_2 \frac{dv_2}{dt}\right)\frac{\partial y}{\partial a} - (\rho_1 - \rho_2)\left(g_x \frac{\partial x}{\partial a} + g_y \frac{\partial y}{\partial a}\right) = 0. \tag{13}$$

Next, continuity of the normal velocity at the interface leads to

$$(u_1 - u_2)(\partial y/\partial a) - (v_1 - v_2)(\partial x/\partial a) = 0. \tag{14}$$

To apply Eqs. (13) and (14) correctly in our finite-difference framework, consider the situation in Fig. 3. A finite displacement Δs of the vertex 1 with respect to the vertex 2 occurs in the time increment δt. Equation (13) can be represented by an "explicit" formula

$$\left(\rho_1 \frac{u_1^{n+1} - u_1^{n}}{\delta t} - \rho_2 \frac{u_2^{n+1} - u_2^{n}}{\delta t}\right)\left(\frac{\partial x}{\partial a}\right)_2^n + \left(\rho_1 \frac{v_1^{n+1} - v_1^{n}}{\delta t} - \rho_2 \frac{v_2^{n+1} - v_2^{n}}{\delta t}\right)\left(\frac{\partial y}{\partial a}\right)_2^n$$
$$- (\rho_1 - \rho_2)\left[g_x \left(\frac{\partial x}{\partial a}\right)_2^n + g_y \left(\frac{\partial y}{\partial a}\right)_2^n\right] = 0, \tag{15}$$

where $(\partial x/\partial a)_2^n$ and $(\partial y/\partial a)_2^n$ are evaluated at point 2 (Fig. 3).

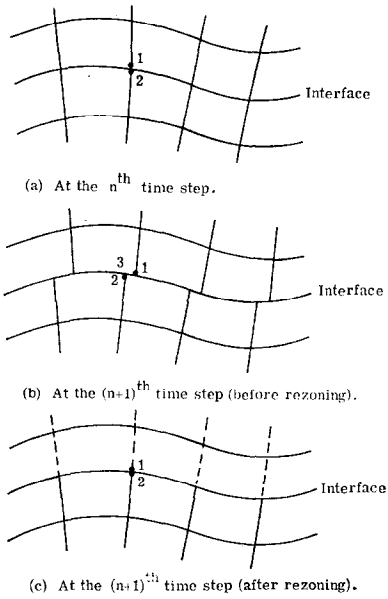

FIG. 3. Boundary conditions at a fluid–fluid interface.

We choose to apply Eq. (14) at the $(n + 1)$th step, also at point 2. However, the counterpart of point 2 in the upper fluid is point 3, rather than point 1 which has now moved away from point 2. To a first approximation, we have

$$\begin{aligned} u_3 &= u_1 - \Delta s(\partial u/\partial s), \\ v_3 &= v_1 - \Delta s(\partial v/\partial s). \end{aligned} \tag{16}$$

Now Eq. (14) becomes

$$\begin{aligned} &[u_1^{n+1} - \Delta s(\partial u/\partial s)_1^{n+1} - u_2^{n+1}](\partial y/\partial a)_2^{n+1} \\ &\quad - [v_1^{n+1} - \Delta s(\partial v/\partial s)_1^{n+1} - v_2^{n+1}](\partial x/\partial a)_2^{n+1} = 0. \end{aligned} \tag{17}$$

Equations (15) and (17) must be satisfied by u_1^{n+1}, v_1^{n+1}, u_2^{n+1}, and v_2^{n+1} associated with a vertex on the fluid–fluid interface. Symbolically, this set of equations may be written as

$$\begin{aligned} Au_1^{n+1} + Bv_1^{n+1} + Cu_2^{n+1} + Dv_2^{n+1} + E &= R_4, \\ A'u_1^{n+1} + B'v_1^{n+1} + C'u_2^{n+1} + D'v_2^{n+1} + E' &= R_5, \end{aligned} \tag{18}$$

where $R_4 = R_5 = 0$ if correct values of the velocity components are used. At the end of each calculational cycle, the upper fluid region is rezoned in such a way that point 1 is moved to the position of point 3 to line up with point 2 in the lower fluid (Fig. 3c).

2.4. *Relaxation Techniques*

To solve Eqs. (3), (5), (10), and (18) simultaneously by an iterative procedure, the values of $u_{i,j}$ and $v_{i,j}$ from the previous time level are used as a first approximation. In general, this set of values do not satisfy the pertinent equations and result in nonzero residuals on the right-hand side of each equation, e.g., R_1, R_2, etc., mentioned above. There are many ways in which the vertex velocities $(u_{i,j}^{n+1}, v_{i,j}^{n+1})$ can be adjusted to dissipate the residuals. We found that a relaxation procedure, evolved from the ALE method, proves to be more efficient than the one used in the BW method.

Referring to Fig. 2, the following formulas for changes in the vertex velocity components are used in the present method to reduce the residuals R_1 and R_2.

$$\begin{aligned} \delta u_1^{n+1} &= \alpha(-y_{24}^n R_1 - x_{24}^n R_2), \\ \delta v_1^{n+1} &= \alpha(x_{24}^n R_1 - y_{24}^n R_2), \\ \delta u_2^{n+1} &= \alpha(y_{13}^n R_1 + x_{13}^n R_2), \\ \delta v_2^{n+1} &= \alpha(-x_{13}^n R_1 + y_{13}^n R_2), \\ \delta u_3^{n+1} &= -\delta u_1^{n+1}, \\ \delta v_3^{n+1} &= -\delta v_1^{n+1}, \\ \delta u_4^{n+1} &= -\delta u_2^{n+1}, \\ \delta v_4^{n+1} &= -\delta v_2^{n+1}, \end{aligned} \tag{19}$$

where

$$\alpha \equiv \omega\{2[(x_{24}^n)^2 + (y_{24}^n)^2] + 2[(x_{13}^n)^2 + (y_{13}^n)^2]\}^{-1},$$

and ω is a relaxation factor ($1 < \omega < 2$).

When very skewed or elongated cells are present, the relaxation procedure employed in the BW method becomes much less efficient because a small value for ω has to be used uniformly across the mesh to avoid instability. This difficulty is avoided by the use of Eqs. (19). Note that the last four of Eqs. (19) imply momentum transfer along the diagonals of a cell. Thus, like the BW method, the present method rigorously conserves both linear and angular momentums through the use of Eqs. (5) and (19).

The relaxation method for correcting $(u_{i,j}^{n+1}, v_{i,j}^{n+1})$ at rigid boundaries and free surfaces as formulated in the BW method has been found to be quite satisfactory. To correct the velocities at a "free-slip" material interface, we multiply each term in Eq. (15) by δt and denote the right side by R_D, the residual in the dynamic interface condition. Also, let R_K, the residual in the kinematic interface condition, be the right-hand side of Eq. (17). The following formulas for adjusting velocities of the interface vertices are used to reduce R_D and R_K.

$$\begin{aligned}
\delta u_1^{n+1} &= (1/\gamma)(-\alpha\, \Delta x\, R_D - (\Delta y/2)\, R_K), \\
\delta v_1^{n+1} &= (1/\gamma)(-\alpha\, \Delta y\, R_D + (\Delta x/2)\, R_K), \\
\delta u_2^{n+1} &= (1/\gamma)(\beta\, \Delta x\, R_D + (\Delta y/2)\, R_K), \\
\delta v_2^{n+1} &= (1/\gamma)(\beta\, \Delta y\, R_D - (\Delta x/2)\, R_K),
\end{aligned} \tag{20}$$

where

$$\Delta x = (\partial x/\partial a)_2^n, \qquad \Delta y = (\partial y/\partial a)_2^n, \qquad \gamma = (\Delta x)^2 + (\Delta y)^2,$$
$$\alpha = \rho_1\rho_2(\rho_1^2 + \rho_2^2)^{-1} \qquad \text{and} \qquad \beta = \rho_2^2(\rho_1^2 + \rho_2^2)^{-1}.$$

The maximum allowable values for the residuals depend on the accuracy desired, and is therefore problem dependent. But a typical value is 0.0005. It usually takes 10 iterations to satisfy this convergence criterion.

2.5. *Rezoning Techniques*

The basic idea behind the rezoning of the finite-difference mesh was discussed in Section 2.1. As shown in Fig. 4, suppose that to maintain a reasonable shape for the mesh we decide to move the vertex 0 to a new location 0′. The problem is to find the velocities (u_0', v_0') and density ρ_0' for the new vertex 0′. Let Q represent either u, v, or ρ. Then, using Taylor's series expansion to the second order in the (a, b) plane,

$$\begin{aligned}
Q_0' = Q_0 &+ \delta a(\partial Q/\partial a)_0 + \delta b(\partial Q/\partial b)_0 \\
&+ \tfrac{1}{2}[\delta a^2(\partial^2 Q/\partial a^2)_0 + 2\delta a\delta b(\partial^2 Q/\partial a\partial b)_0 + \delta b^2(\partial^2 Q/\partial b^2)_0].
\end{aligned} \tag{21}$$

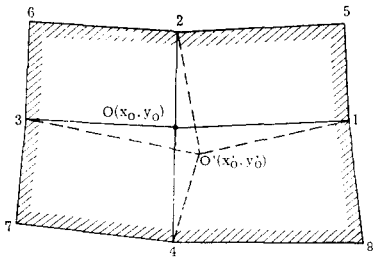

FIG. 4. Definition sketch for the rezoning procedure.

To evaluate δa and δb, we have the following relations

$$\begin{aligned}\delta a &= (\partial a/\partial x)_0(x_0' - x_0) + (\partial a/\partial y)_0(y_0' - y_0) \\ &= (1/J_0)[(\partial y/\partial b)_0(x_0' - x_0) - (\partial x/\partial b)_0(y_0' - y_0)],\end{aligned} \tag{22}$$

and a similar expression can be derived for δb; here

$$J_0 = (\partial x/\partial a)_0\,(\partial y/\partial b)_0 - (\partial x/\partial b)_0\,(\partial y/\partial a)_0$$

is the Jacobian of the transformation. The subscript 0 means evaluation at the

finite difference to complete the calculation of Q_0'. The rezoning procedure described here is the "Eulerian" phase of the GALE algorithm.

2.6. *Numerical Stability and Damping of Short-wavelength Modes*

A rigorous stability analysis is difficult to perform on the system of nonlinear finite-difference equations employed in the present method; hence, heuristic argument and stability criteria for Eulerian methods have been relied upon to obtain an approximate upper bound for the time increment δt.

By considering truncation errors in the free surface condition in the light of a linearized standing wave solution, Brennen and Whitney [3] arrived at a criterion which states that δt should be less than the time taken for a gravity wave to travel one cell length when free surface and gravity are present. Numerical instability due to advection terms does not arise in purely Lagrangian calculations. When the rezoning procedure is used, however, a condition similar to the consideration of advectional stability in Eulerian methods must be satisfied. This is achieved by requiring the rezoning procedure to be an interpolation rather than an extrapolation, namely the point 0′ (Fig. 4) should remain within the shaded area.

The growth of "alternating errors" is an instability which GALE shares with its predecessors. The situation is illustrated in Fig. 5, where the propagation of a

solitary wave in a channel of constant depth (see Section 3.1) is shown. The hourglass-shaped distortion of the computing mesh has resulted from the growth of short-wave disturbances. To trace the origin of alternating errors, it is informative to observe that residuals in the cell equations, viz. R_1 in Eq. (3) and R_2 in Eq. (5), are completely insensitive to errors in u^{n+1} or v^{n+1} that happen to be of the same value along cell diagonals. An example is the case in which u_1^{n+1} and u_3^{n+1} (Fig. 6) contain the same amount of errors.

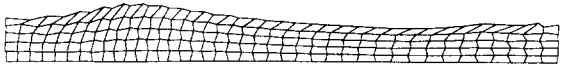

FIG. 5. An example of alternating errors.

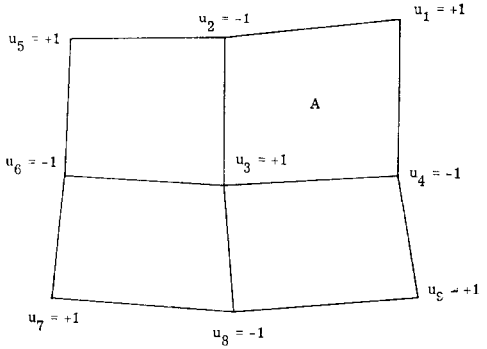

FIG. 6. A typical pattern of alternating errors.

The alternating errors are disturbances with wavelenth equal to two cell spacings in the Lagrangian space. Various attempts have been made to "damp out" the alternating errors. In the BW method, a fourth-order damping along the free surface, i.e., requiring $\partial^4 u/\partial a^4 \approx 0$ and $\partial^4 v/\partial a^4 \approx 0$, has been used with some success. Butler [4], in connection with the LINC method extensions, uses a Hooke's law type force with damping that tends to keep a given interior vertex in the center of mass of its eight surrounding neighbors. Since no explicit finite-difference formula is given in [4], the following formula is found in [5]. The artificial damping force has the following effect.

$$(u_{i,j})_{\text{adjusted}} = u_{i,j} + \beta([(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1})/4] - u_{i,j}), \qquad (23)$$

and a similar formula for $v_{i,j}$. Using a linear stability analysis advanced by von Neumann [7], it can be shown that $0 \leqslant \beta \leqslant 1$ is required to insure numerical stability. The alternating error, as shown in Fig. 6, is completely removed within one timestep when $\beta = 0.5$ is used.

Naturally, a damping mechanism such as Eq. (23) will damp Fourier components of all wavelengths, with the short wavelength components most drastically affected. Thus, it is important not to use too large a value for β. Experience indicates that $\beta = 0.05 \sim 0.10$ is usually required to remove alternating errors. Unfortunately, this can introduce excessive damping of the physically important long-wavelength components.

A more desirable method has been developed, which effectively eliminates the alternating errors but gives very little damping to the meaningful part of the solution. Consider cell A in Fig. 6, and define a discrepancy parameter

$$\psi = (u_1 + u_3) - (u_2 + u_4), \tag{24}$$

which is very sensitive to alternating errors. Now, we wish to add corrections δu_1, δu_2, δu_3, and δu_4 to u_1, u_2, u_3, and u_4, respectively, such that

$$(\delta u_1 + \delta u_3) - (\delta u_2 + \delta u_4) = -k\psi, \tag{25}$$

$$\delta u_1 = \delta u_3 ; \qquad \delta u_2 = \delta u_4 , \tag{26}$$

and

$$\delta u_1 + \delta u_2 + \delta u_3 + \delta u_4 = 0. \tag{27}$$

Equation (25) is imposed to reduce the discrepancy between the average velocities along two cell diagonals, namely $\frac{1}{2}(u_1 + u_3)$ and $\frac{1}{2}(u_2 + u_4)$. Equation (26) insures minimum impact to the cell residuals R_1 and R_2, and Eq. (27) is just conservation of the average u-momentum in cell A. Solving Eqs. (25)-(27) simultaneously,

$$\begin{aligned} \delta u_1 &= \delta u_3 = -k\psi/4, \\ \delta u_2 &= \delta u_4 = k\psi/4. \end{aligned} \tag{28}$$

At a given timestep, after the iterative solution for u^{n+1} and v^{n+1} is terminated, the vertex velocities are adjusted according to

$$\begin{aligned} (u_1^{n+1})_{\text{adjusted}} &= u_1^{n+1} - (k\psi/4), \\ (u_2^{n+1})_{\text{adjusted}} &= u_2^{n+1} + (k\psi/4), \\ (u_3^{n+1})_{\text{adjusted}} &= u_3^{n+1} - (k\psi/4), \\ (u_4^{n+1})_{\text{adjusted}} &= u_4^{n+1} + (k\psi/4), \end{aligned} \tag{29}$$

and similar expressions can be derived for the v's. For stability, $0 \leqslant k \leqslant \frac{1}{2}$. The alternating error shown in Fig. 6 is damped out completely when $k = \frac{1}{4}$. It has been found that $k = 0.025$ works very well, as explained in what follows.

To understand the effects of artificial damping, the generation and propagation of a nonlinear solitary wave in a channel of constant depth have been used as a test problem. More details on the solitary wave will be given in Section 3.1; we are only concerned here with how artificial damping affects the accuracy of the solution. As shown in Fig. 7 a solitary wave can be generated by the forward motion of a vertical plate. It has been found that, before the wave crest reaches the wall on the right-hand side of the channel, alternating errors do not occur even without artificial damping. Thus, this early-time solution without damping can be used as a reference against which the effects of artificial damping can be evaluated.

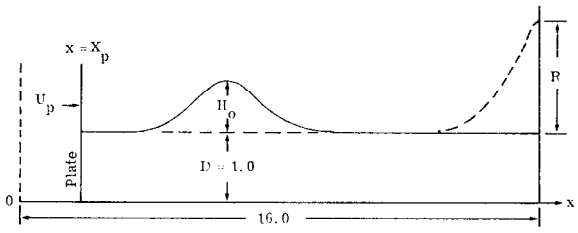

FIG. 7. Definition sketch for a solitary wave.

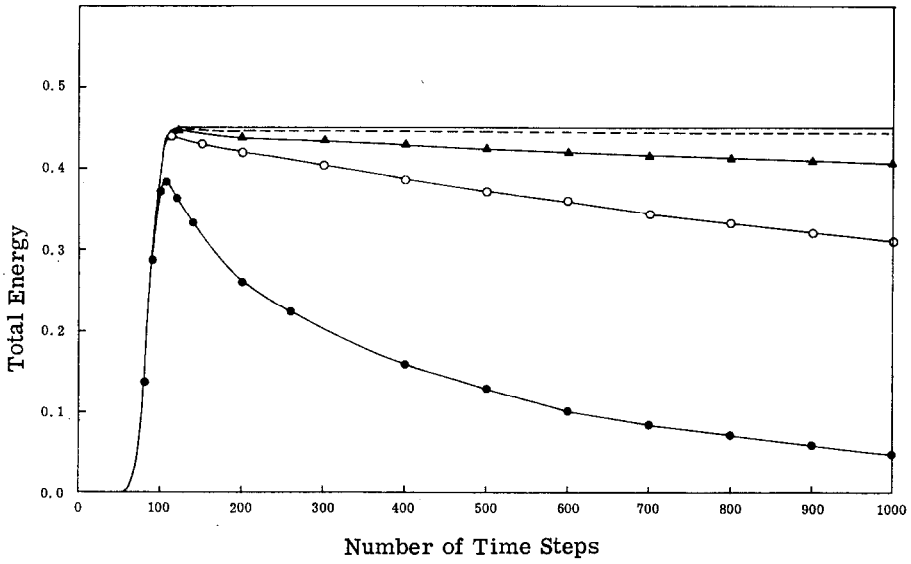

FIG. 8. Effect of artificial damping on the total energy. The solid line is the correct total energy. The dash line is $k = 0.025$, and ▲, ○, ● are $k = 0.25$, $\beta = 0.05$ and $\beta = 0.5$, respectively.

In this reference case, the total energy in the wave tank starts from zero and reaches a maximum value of 0.45 at the end of the plate motion (Fig. 8). Four experimental calculations have been conducted: in two runs Eq. (23) was used, with β=0.05 and 0.5, while in the other two runs Eqs. (29) were used, with $k = 0.025$ and 0.25. Each calculation was continued for 1000 timesteps, which correspond to the time required for the solitary wave to travel across the length of channel eight times. Recall that both the case $\beta = 0.5$ in connection with Eq. (23) and the case $k = 0.25$ in connection with Eqs. (29) completely eliminate the alternating error pattern within one timestep, so these two cases may be compared. In Fig. 8 we see that at the end of the calculation one loses about 90% of the total energy when $\beta = 0.5$ is used in Eq. (23). In contrast, when $k = 0.25$ is used in Eqs. (29), we lose about 10% of the total energy. Now, we have found by experience that $\beta = 0.05$ and $k = 0.025$, both of which reduce the amplitude of the alternating errors by 10% at each timestep, are about the minimum values needed to remove the alternating errors. Using $k = 0.025$ in Eqs. (29), the loss in total energy is only 1.5% while the energy loss is 30% if $\beta = 0.05$ is used in Eq. (23).

III. Examples

3.1. *Generation of Solitary Waves*

With recent advances in computational methods, it becomes possible to study the transient aspects of solitary waves. For example, Chan and Street [8] simulated the maximum run-up R/D of a solitary wave on a vertical wall (Fig. 7) and their results compared favorably with the experiments of Camfield and Street [9]. In their calculations, the initially undisturbed wave profile and the associated velocity field were taken from Laitone's [10] second-order theory and the subsequent development was calculated using the SUMMAC method. Camfield and Street employed a piston-plate, the forward displacement of which is a hyperbolic-tangent function, to generate solitary waves in their experiments. We have utilized the flexibility of the GALE method to simulate their experimental procedure.

In this example, we nondimensionalize lengths by the still water depth, D, gravity by g, velocities by $(gD)^{1/2}$, and the time by $(D/g)^{1/2}$. Shown in Fig. 9 is a time sequence of the configuration of the computational mesh. The vertical plate on the left boundary moves to the right in early times. A well-developed solitary wave appears at about $t = 15$; the wave height for this particular calculation is $H_0 = 0.625$. It then runs up a vertical wall on the right. Note that at $t = 22$ the free surface approaches a sharp angle at the wall. Subsequently the wave is reflected and travels to the left. We have run three cases, $H_0/D = 0.214$, 0.465, and 0.625, and obtained the maximum run-ups $R/D = 0.446$, 1.085, and 1.733, respectively.

These results are in excellent agreement with the experiments of Camfield and Street and with the numerical solution of Chan and Street.

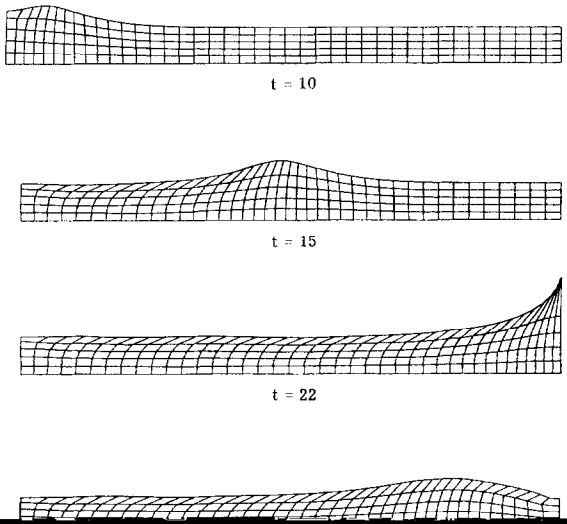

FIG. 9. Generation and reflection of a solitary wave.

Figure 10 shows the growth of rotation at the lower left of the channel when the ALE method is employed, which, as mentioned earlier, does not conserve angular momentum.

FIG. 10. Growth of velocity circulation when R_2 is not controlled.

3.2. *Kelvin–Helmholtz Instability*

The Kelvin–Helmholtz instability of an interface separating two fluids of different densities in a gravity field has been chosen to validate a two-fluid code based on the present method. The problem is formulated as shown in Fig. 11. Here, all lengths are nondimensionalized by the half wavelength $\lambda/2$ of the interface, velocities by $(g\lambda/2)^{1/2}$, gravity by g, and densities by the density of the lower fluid ρ_2.

At $t = 0$, when the wave amplitude $a_0 = 0.01$, the well-known linear solution is used as initial conditions; the subsequent development of the flow field is calculated by the GALE method. To save computer time, periodic boundary conditions

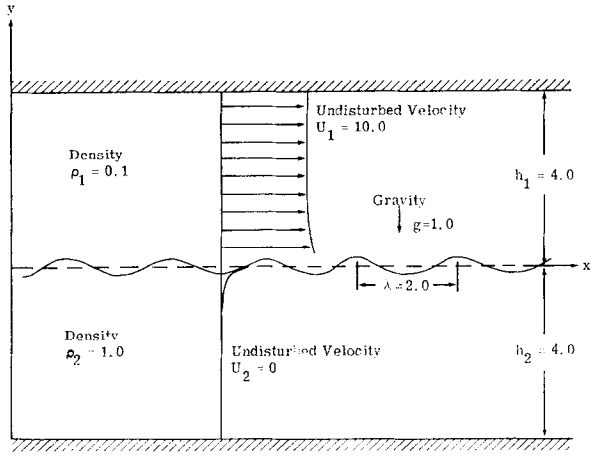

FIG. 11. Definition sketch for the Kelvin–Helmholtz instability problem.

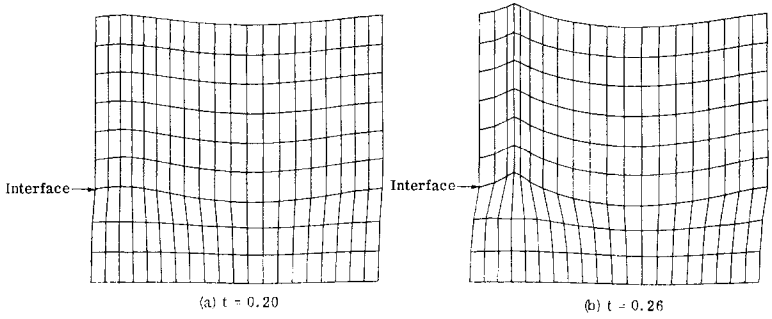

FIG. 12. Kelvin–Helmholtz instability.

have been utilized so that a region of only one wavelength in the x-direction needs to be computed. Figure 12a shows a portion of the computing mesh at $t = 0.20$; the entire mesh contains 20×20 cells in each fluid region, but only six rows above the interface and three rows below it are shown here. At $t = 0.26$, the wave crest forms an angle very close to 120° and the wave height-to-length ratio becomes 0.11. At this point the lower fluid particles start to bunch up at the crest which becomes unstable. Finally, at $t = 0.278$, the calculation went unstable with the height-to-length ratio equal to 0.147. Since analytic solutions for the density ratio $\rho_1/\rho_2 = 0.1$ up to the breaking of the waves are not available, it is attempted here to make some qualitative comparisons with classical solutions for the case $\rho_1/\rho_2 = 0$. Stokes found that when the limiting steepness of progressive waves is reached, the crest

angle is equal to 120°, while Michell and Keulegan found the height-to-length ratio to be 0.143 for progressive waves of maximum height [11]. Thus, the present calculation yields results in good agreement with existing theories.

3.3. *Internal-Surface Wave Interactions*

The GALE method can be easily applied to the interaction of 1-dimensional monochromatic internal waves and surface waves propagating in the same direction (Fig. 13). In the test problem there are four surface waves corresponding to each internal wavelength. The initial group velocity of the surface waves is $(C_g)_0 = 0.1410$ and the phase velocity of the internal waves is $C = 0.1424$. Thus, a near-resonance condition, i.e., $C/(C_g)_0 \approx 1.0$, is simulated. Under this condition, it is possible for the surface waves to extract energy from the internal waves such that amplitude and frequency modulations occur on the surface waves.

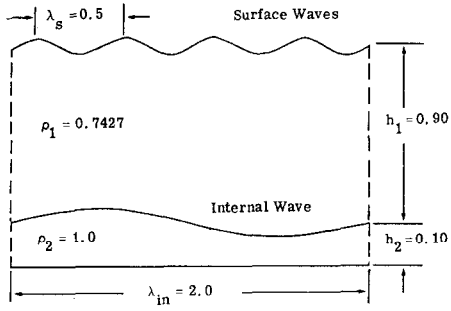

FIG. 13. Definition sketch for the surface–internal wave interaction problem.

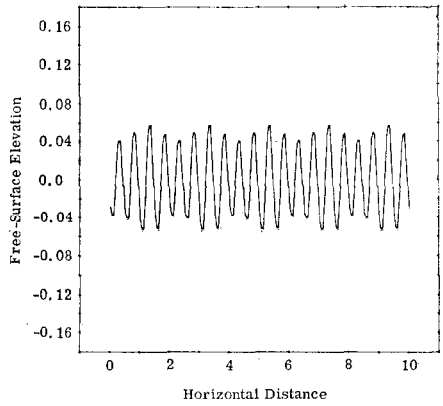

FIG. 14. Surface wave profile at $t = 49.0$.

To avoid calculating the transients which exist naturally in an experimental configuration, periodic boundary conditions are applied to the left and right boundaries of the fluid domain. At $t = 0$, since the wave amplitudes are small, the superposed linear solutions for the surface and internal waves may be used as initial conditions. Thereafter, the calculation is taken over by the nonlinear, full dynamic equations embodied in the GALE method. In this example, all lengths are nondimensionalized by d, the mean distance of the free surface from the bottom of the channel, while all velocities and the time are nondimensionalized by $(gd)^{1/2}$ (g is gravity) and by $(d/g)^{1/2}$, respectively. The amplitude of the surface waves at $t = 49$ is shown in Fig. 14. About 20% modulation of the amplitude can be seen; this agrees fairly well with Ko's theory [12].

3.4. *Plane Poiseuille Flow*

To validate the generalized cell equation, Eq. (9), the transient development of a 2-dimensional Poiseuille flow has been calculated. The flow in the channel is assumed to have a uniform velocity profile initially. Then the action of viscosity and no-slip boundary conditions start to modify the velocity distribution until it becomes very close to the parabolic steady-state profile. Figure 15 shows the time history of the velocity profile. Periodic boundary conditions and rezoning have been used in this calculation.

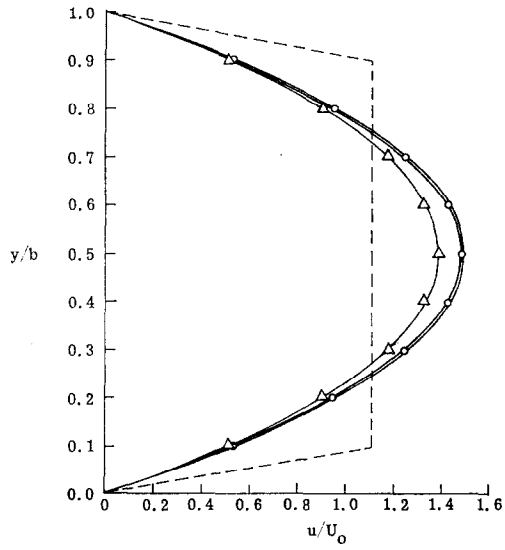

FIG. 15. Velocity profiles of a 2-dimensional Poiseuille flow. The dash line is $t = 0$, and △, ○, — are $t = 5$, 25 and 50, respectively.

IV. Conclusions

The various test problems calculated so far indicate the advantages of using the Lagrangian description of fluid motion whenever interfaces and/or complex boundary geometries are present. To successfully perform Lagrangian calculations, however, it is essential to suppress the alternating errors, with as little distortion on the important parts of the solution as possible, and to use a stable relaxation scheme in solving the velocity field. Furthermore, for highly strained flows, proper rezoning of the computational mesh is required to avoid singular mapping between the physical space and the Lagrangian space. The present GALE method has been addressed to these considerations and represents an improvement over its predecessors.

Extension of the GALE method to cylindrical coordinates for flows with axial symmetry is straightforward, but generalization to three dimensions requires more complicated considerations. Nevertheless, the flexibility of the present technique offers a very useful tool for studying many interesting problems.

Based on the sample problem discussed in Section 3.1, it is found that, using the CDC 7600 System, the computation time required for the application of the GALE method is only about 5% more than that required in the ALE method.

APPENDIX: Method of Calculating ξ and η

In most applications, the stresses τ_{xx}, τ_{xy}, and τ_{yy} are related to the rate of deformation of the fluid element, e.g.,

$$\begin{aligned} \tau_{xx} &= 2\lambda(\partial u/\partial x), \\ \tau_{xy} &= \tau_{yx} = \lambda((\partial u/\partial y) + (\partial v/\partial x)), \\ \tau_{yy} &= 2\lambda(\partial v/\partial y), \end{aligned} \tag{A.1}$$

where λ is taken as the dynamic viscosity μ for laminar flows or it may be set equal to $\rho\,\epsilon$, ϵ being the eddy viscosity computed from a turbulence model, for turbulent flows. Thus, the problem of computing the stress components reduces to that of evaluating the various velocity derivatives.

Regarding $\partial u/\partial x$ and $\partial u/\partial y$ as functions of (a, b), then a formal coordinate transformation results in

$$\begin{aligned} \frac{\partial u}{\partial x}(a, b) &= \left(\frac{\partial y}{\partial b}\frac{\partial u}{\partial a} - \frac{\partial y}{\partial a}\frac{\partial u}{\partial b}\right)\Big/\left(\frac{\partial x}{\partial a}\frac{\partial y}{\partial b} - \frac{\partial x}{\partial b}\frac{\partial y}{\partial a}\right), \\ \frac{\partial u}{\partial y}(a, b) &= -\left(\frac{\partial x}{\partial b}\frac{\partial u}{\partial a} - \frac{\partial x}{\partial a}\frac{\partial u}{\partial b}\right)\Big/\left(\frac{\partial x}{\partial a}\frac{\partial y}{\partial b} - \frac{\partial x}{\partial b}\frac{\partial y}{\partial a}\right). \end{aligned} \tag{A.2}$$

By substituting v for u in Eqs. (A.2), we obtain similar expressions for $\partial v/\partial x$ and $\partial v/\partial y$. The remaining task is to evaluate the first-order derivatives $\partial u/\partial a$, $\partial u/\partial b$, $\partial v/\partial a$, $\partial v/\partial b$, $\partial x/\partial a$, $\partial x/\partial b$, $\partial y/\partial a$ and $\partial y/\partial b$ in the Lagrangian (a, b) plane, which is straightforward in finite differences. To calculate ξ and η according to Eqs. (7), we can again use Eqs. (A.2) by replacing u with τ_{xx}, τ_{xy}, etc.

ACKNOWLEDGMENT

The author appreciates the many stimulating discussions with Dr. C. W. Hirt on the ALE computing technique, which has led to the pursuit of the present study.

This work was sponsored by the Applied Physics Laboratory, Johns Hopkins University, under Contract APL 37 2049. Partial support was provided by the Office of Naval Research under Contract N00014-73-C-0128.

REFERENCES

1. C. W. HIRT, An arbitrary Lagrangian–Eulerian computing technique, *in* "Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics" (M. Holt, Ed.), Lecture Notes in Physics, Springer–Verlag, Berlin, 1971.
2. C. W. HIRT, J. L. COOK, AND T. D. BUTLER, *J. Computational Phys.* **5** (1970), 103.
3. C. B. BRENNEN AND A. K. WHITNEY, Unsteady, free surface flows; solutions employing the Lagrangian description of the motion, *in* "Proceedings of the 8th Symposium on Naval Hydrodynamics," Pasadena, California, 1970.
4. T. D. BUTLER, LINC method extensions, *in* "Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics" (M. Holt, ed.), Lecture Notes in Physics, Springer–Verlag, Berlin, 1971.
5. C. W. HIRT, A. A. AMSDEN, AND J. L. COOK, *J. Computational Phys.* **14** (1974).
6. H. SCHLICHTING, "Boundary-Layer Theory," McGraw–Hill, New York, 1968.
7. J. G. CHARNEY, R. FJÖRTOFT, AND J. VON NEUMANN, *Tellus* **2** (1950).
8. R. K.-C. CHAN AND R. L. STREET, *J. Computational Phys.* **6** (1970), 1.
9. F. E. CAMFIELD AND R. L. STREET, An investigation of the deformation and breaking of solitary waves, Dept. of Civil Engineering Technical Report No. 81, Stanford University, California,
10. E. V. LAITONE, *J. Fluid Mech.* **9** (1960).
11. R. L. WIEGEL, "Oceanographic Engineering," Prentice–Hall, Englewood Cliffs, NJ, 1964.
12. D. R. S. KO, A conservational approach to the interaction of internal wave and surface waves, TRW Technical Report 18202-6001-R0-00, Vol. IV, 1971.