JOURNAL OF COMPUTATIONAL PHYSICS 116, 52–68 (1995)

Free Surface Simulations Using a Conservative 3D Code

T. G. THOMAS, D. C. LESLIE, AND J. J. R. WILLIAMS

Turbulence Unit, Queen Mary and Westfield College, University of London, Mile End Road, London E1 4NS, United Kingdom

Received October 19, 1993; revised June 2, 1994

This paper describes the development and testing of a 3D finite difference code written specifically to model turbulence in an open channel with a moving free surface. The code has been developed so that either a full simulation or a large eddy simulation (LES) of the turbulence may be performed. The free surface may undergo arbitrarily large deformations but the slope may not exceed a limit related to the aspect ratio of the mesh and so the possibility of breaking waves is excluded. The LES application demands numerical approximations which conserve mass, momentum, and total energy with high precision, and it permits wave motion with very little numerical dispersion or dissipation. We describe a novel numerical method for tracking the free surface using a split–merge technique which combines the volume of fluid and height function methods in a way that is conservative.

INTRODUCTION

The turbulent motion of a liquid at a free liquid surface and the physical processes involved there have an important practical role in the dispersion of pollutants in rivers and coastal waters as well as being interesting in their own right. However, our understanding of these processes is limited: partly by the experimental difficulties of obtaining reliable and detailed turbulence measurements close to a moving surface, and partly due to the lack of a satisfactory numerical surface boundary condition for k-ε or algebraic models.

It is now possible to use a supercomputer to integrate the Navier–Stokes equations for turbulent flow without making any modelling assumptions and, hence, to calculate any desired flow quantity. However, this approach which is known as full simulation (FS) or direct simulation (DS) is feasible only for flows at relatively low Reynolds number and in regions of simple shape. The drawback to the method is that the computational effort required to model flows of engineering importance (and, hence, large Reynolds numbers) lies beyond our reach for the foreseeable future.

An alternative approach closely related to FS is large eddy simulation (LES) in which only the large eddies or grid scales (GS) are represented explicitly on a finite difference grid. The interactions of the GS with the unrepresented small eddies or sub-grid scales (SGS) are represented by a sub-grid model (SGM). This method is therefore dependent on a model, but experience has shown that the sensitivity of the results to the details of the model is quite weak, which contrasts with the much greater sensitivity of k-ε and algebraic stress models of turbulence to their model constants.

1. NUMERICAL METHOD

The code is based on the Queen Mary and Westfield College finite difference closed channel code ECCLES (see Gavrilakis *et al.* [3]) which uses a conventional staggered grid and is conservative; the flow must be periodic in the x and y directions, but solid walls are allowed in the z direction. The object of the present work is to incorporate a free surface and its associated stress-free boundary condition, and to do this in an energy conserving way. The authors felt that because LESs have long integration times and it is necessary to conserve energy in the finite sense (as well as that for wave propagation) then it was not possible to use the more established techniques such as (MAC), volume of fluid (VOF), or mapping of the domain.

2. GOVERNING EQUATIONS

We use Cartesian co-ordinates (x, y, z) with x streamwise, y spanwise, and z upwards, aligned with the channel which slopes downwards at an angle θ. The flow is maintained by gravity $\mathbf{g} = (g_x, g_y, g_z)$ and we assume that there is no externally applied surface pressure. The velocity $\mathbf{u}$ and pressure p satisfy the Navier–Stokes equations:

$$\partial \mathbf{u}/\partial t + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot \tau + \mathbf{g}, \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2}$$

where τ denotes the viscous and subgrid stress. The elevation of the free surface is given in terms of a single-valued height function h:

$$z = h(x, y, t), \tag{3}$$

$$|\partial h/\partial x| \leq \Delta z/\Delta x, \quad |\partial h/\partial y| \leq \Delta z/\Delta y. \tag{4}$$

The maximum slope restriction—based on finite difference

0021-9991/95 $6.00

mesh size Δx, Δy, Δz, is imposed because it allows a much simplified surface locator. The single-valued nature of h excludes wave breaking. The kinematic free surface boundary condition is given by

$$\partial h/\partial t = (\mathbf{u} \cdot \mathbf{n})\sqrt{S}, \tag{5}$$

$$S = 1 + (\partial h/\partial x)^2 + (\partial h/\partial y)^2, \tag{6}$$

$$\mathbf{n} = (-\partial h/\partial x, -\partial h/\partial y, 1)\sqrt{S^{-1}}. \tag{7}$$

Equation (5) states that the rate of change of surface elevation is proportional to the flux of fluid over the surface; it can be written in the more usual convective form by eliminating $(\mathbf{u} \cdot \mathbf{n})$. The quantity S is simply the ratio of sloping surface area to vertically projected surface area, and $\mathbf{n}$ denotes the surface unit normal vector. The dynamic free surface condition is:

$$\mathbf{n} \cdot (\tau - p\mathbf{I}) = 0 \tag{8}$$

which states that both the total normal stress, including the viscous component, and the tangential stress must be zero; $\mathbf{I}$ denotes the unit tensor.

3. SURFACE LOCATION AND TRACKING

The purpose of the surface locator is to define the position of the free surface and to provide a means of moving it from one place to another. In an LES simulation in which the same fluid mass is recirculating through the computational box we believe that it is essential to conserve mass exactly.

We considered several types of locator: marker particles; VOF methods; and height functions. Chan and Street [1] in their SUMMAC code used marker particles to define the surface in 2D. This method is able to describe breaking waves but it becomes difficult to implement in 3D because, not only do surface elements stretch, meaning that particles must be added/deleted, but they also rotate in the plane of the surface, which means that the nearest neighbours computation is non-trivial.

A direct discretisation of the height function $h(x, y, t)$ and the kinematic boundary condition (cf. Eq. (5)) was used by Chan and Street [2] in a modified version of SUMMAC and by Hirt, Nichols, and Romero [5] in SOLA-SURF. This method is easy to implement but it does not guarantee exact mass conservation over finite time steps, partly because continuity is satisfied approximately (by iteration) and not exactly.

The VOF method of Hirt and Nichols [4] is based in part on a finite control volume principle and does conserve mass exactly; it also extends relatively easily to 3D and can handle breaking waves, but in its original form the free surface could not move smoothly through the mesh.

We have used a combination of VOF and the height function methods; this conserves mass and allows for smooth surface movement but it excludes breaking waves. The complexity of the method follows partly from the staggered grid and the need

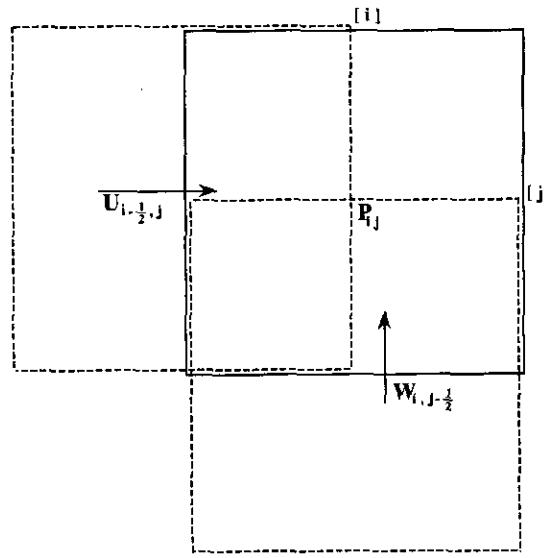

FIG. 1. Staggered mesh.

for the masses in the mutually overlapping cells to be consistent, and partly from the technique adopted to remove the potentially singular behaviour of the momentum equations when applied to a nearly empty cell. The latter is called the split–merge technique and is described in more detail below.

We use a conventional staggered mesh as shown in Fig. 1. The pressure points p_{ijk} are located in the centred or continuity cells whose fluid volume is Fc_{ijk}, and the x-, y-, z-staggered velocity cells contain the u_{ijk}, v_{ijk}, w_{ijk} velocities and the fluid volumes Fx_{ijk}, Fy_{ijk}, Fz_{ijk}, respectively. (The c notation for Fc is optional; its purpose is to simplify expressions involving F or h.) The surface elevations h_{ij} are defined for columns of centred cells, and hx_{ij}, hy_{ij} are defined similarly for staggered cells. The purpose of the surface locator is to provide a means of relating the h_{ji} and F_{ijk}. This relation is constrained in that it must apply equally to the centred and staggered cells, and we have adopted the simple rule:

$$hq_{ijk} = \sum_k Fq_{ij}/(\Delta x\, \Delta y) \quad \text{for all } q \in \{c, x, y\}. \tag{9}$$

This states that the height of a column is equal to the sum of the F-values of the cells composing it and that this is true for both centred and x-, y-staggered columns. In effect this is a simple one-point integration rule, and its apparent crudity is a necessary part of the mass and energy conserving treatment of surface cells.

Clearly, the staggered and centred forms of F (or h) are not independent quantities and must be mutually consistent; we achieve this by using the algebraic relations

$$hx_{ij} = \tfrac{1}{2}(hc_{i,j} + hc_{i-1,j}) \tag{10a}$$

$$hy_{ij} = \tfrac{1}{2}(hc_{i,j} + hc_{i,j-1}) \tag{10b}$$

to define the staggered elevations hx_{ij}, hy_{ij} in terms of the centred elevations hc_{ij}. A consistent set of x-, y-staggered F-values then follows from a backward use of Eq. (9). The z-staggered F-values are found similarly using the centred hc_{ij} and partitioning the column by z-staggered cells rather than centred cells.

The most obvious way of defining cell-based control volumes is to intersect the nominal cell sides with the free surface; this is equivalent to the VOF method and reduces to conventional FD form in the interior. However, this presents at least two problems.

First, it is possible for the volume to become very small, and, hence, a potential singularity of the momentum equations exists as $F \to 0$. The singularity is apparent, rather than real, and can be removed by modifying the intercell forces, but it seems that this cannot be done without destroying the energy conservation property (see later).

Second, the backward use of Eq. (9) is ambiguous for those surface configurations which allow one partly filled cell to stack on another. This arises because the integration rule does not take the surface slope into account. We could have used a higher order rule, but the methods of ensuring energy conservation depend on its essential simplicity.

We avoid these dilemmas by using cell-extension and split–merge techniques in which potentially small cells are combined with their immediate neighbour below, thus avoiding the potential for zero volumes and ensuring that only one partially full cell exists in any one vertical column of cells. The techniques also provide a natural way of allowing smooth movement of the free surface diagonally across the mesh.

Figure 2 shows a column of cells which is intersected by the free surface at H (or H') between the cell centre points B and C, such that H is greater than or equal to B and less than C. The type of cells A, B, C is immaterial because the principles are applied uniformly to centred (these are actually more complicated) and staggered cells alike. The dotted lines show the

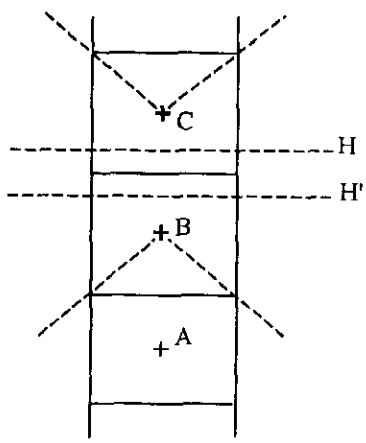

FIG. 2. Surface cell extension.

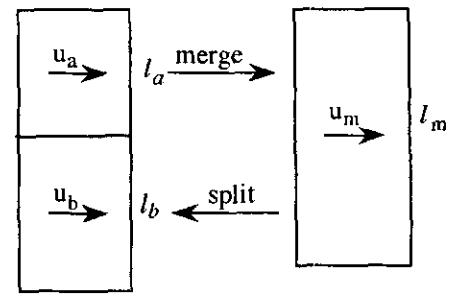

FIG. 3. Velocity cell split/merge.

extreme range of surface positions confined by the slope limit Eq. (4). The lower cell with centre A is an interior cell and is full. The cell B is assumed to extend fully up to the surface and is called an extended or surface cell, and the upper cell C is assumed to be empty and takes no part until it is introduced by a split–merge operation (see later). Thus there is only one surface cell per vertical column and it always has: a complete base, a complete free surface, and at least one immediate neighbour adjoining in each horizontal direction, but not necessarily confined to the same vertical level. The last point implies some considerable programming complexity, but this is manageable because the fundamental definitions are simple and regular.

During a time step the free surface can move within the interval BC without altering the control volume configurations; if it moves outside BC the configuration switches. For example, if H moves past C then the surface cell is split to yield a new surface cell and a new interior cell, and if H moves down past B the old surface cell and the interior cell immediately below are merged together to form a new surface cell. These operations constitute the split–merge technique. Figure 3 illustrates the method for a velocity cell in which we have assumed $\Delta x = \Delta y = 1$ for simplicity. If the upper cell becomes less than half full, i.e., that l_a less than $\frac{1}{2}$, it is combined with the one below and a merged velocity u_m and volume F_m are assigned so that mass and momentum are conserved:

$$l_m = l_a + l_b, \quad u_m = l_a u_a/(l_a + l_b) + l_b u_b/(l_a + l_b). \quad (11)$$

As long as l_m remains greater than or equal to $\frac{1}{2}$ and less than $\frac{3}{2}$ the velocity and volume are advanced using the conventional surface cell equations applied to the extended cell. If l_m exceeds $\frac{3}{2}$ the extended cell is split, mass and momentum are reassigned with both new cells taking the same old velocity. The method deals naturally with the spillage of fluid from one cell to another.

The split–merge process is numerically stable, but it is not strictly conservative. The kinetic energy $\mathrm{KE}_{\mathrm{merge}}$ $(= l_m u_m^2)$ of the merged cells is always less than or equal to the kinetic energy $\mathrm{KE}_{\mathrm{split}}$ $(= l_a u_a^2 + l_b u_b^2)$ of the individual cells. This is clear from the inequality

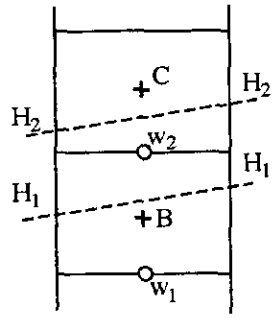

FIG. 4. z-cell extension.

$$\mathrm{KE}_{\mathrm{split}} - \mathrm{KE}_{\mathrm{merge}} = \tfrac{1}{4} l_a l_b (l_a + l_b)(du/dz)^2 \geq 0, \qquad (12)$$

where $du/dz = 2(u_a - u_b)/(l_a + l_b)$.

The merge process dissipates energy while the split process is neutral; thus it is stable overall. Note that the dissipation occurs only at those points in time when the cells merge and not during the time that they remain in any one state. Our early experience with the code suggests that the additional dissipation due to the split–merge technique is not significant.

The cell extension definitions above are applied consistently to the x-, y-, and z-staggered (velocity) cells, but they need an additional modification when applied to the centred (continuity) cells. This is illustrated by Fig. 4 which shows a column of centred cells intersected in the interval BC at H by the free surface. The cell extension definition applied to this column identifies cell B as the surface cell but note its relation to the surface z-staggered (or w-cell)—which is either w_2 or w_1, depending on whether H is greater than or equal to w_2 or not. Thus the pressure point of cell B may or may not have a w-point between it and the free surface and the two situations are fundamentally different. In the first case the pressure point is an internal one determined by continuity equations, whereas in the second case it is in essence external (but not physically) and must be determined by some form of interpolation from the surface boundary condition. We refer to these two distinct types of pressure points as active and passive, respectively.

Consider Fig. 5, showing the kinematics of an active surface cell. The velocity field is treated as constant within a cell (i.e., zeroth order interpolation) and the flux of mass over a plane surface is given by the velocity times the area, and the area of cell sides is given by the one-point integration rule.

The rate of change of volume is given by the flux of mass over the instantaneous free surface,

$$d(Fc_c)/dt = w_c(s) - \tfrac{1}{2}(l_c - l_a)u_b - \tfrac{1}{2}(l_e - l_c)u_d. \qquad (13)$$

This expression is just the kinematic free surface boundary condition Eq. (5) written in finite difference form with $-u(dh/dx)$ approximated by a centred average and Fc in place of h.

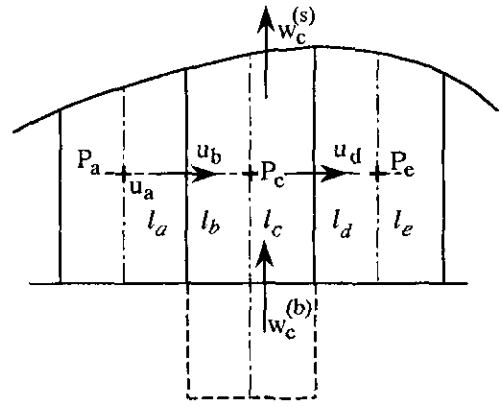

FIG. 5. Surface advancement.

The rate of change of the volume given by the mass flux over the wet sides of the cell is

$$d(Fc_c)/dt = l_b u_b - l_d u_d + w_c(b). \qquad (14)$$

Eliminating dF/dt and using the interpolation rules (10) to eliminate the heights l_b and l_d yields the particularly simple form

$$l_c(u_d - u_b) + (w_c^{(s)} - w_c^{(b)}) = 0 \qquad (15)$$

for the continuity equation in an active surface cell. It reduces to the usual interior form if we set l_c equal to Δz, in this example unity. The distinction between active and passive (centred) surface cells is necessary because the above equations would be degenerate if applied to a passive surface cell.

Consistency also demands that when the surface moves and, hence, when all the F-values change, that these changes are all consistent with the velocity field; this, in fact, constrains the functions that interpolate the velocity from the centres of the staggered cells to their faces. The condition is equivalent to demanding that continuity be satisfied in the staggered cells, but because we do not have an independent equation (e.g., the pressure equation) to achieve this, it must be built into the interpolation equations. With the present surface locator there is only one interpolation allowed, and this is illustrated below.

Consider the surface u-cell shown in Fig. 6, in which, again

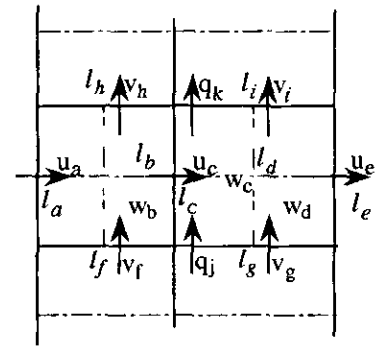

FIG. 6. Plan view of surface cells.

$\Delta x = \Delta y = 1$ and l denotes the vertical length of a cell. The centred cells b, d have volumes of l_b, l_d and the u-cell has volume l_c. Mass conservation in the centred cells, which is imposed explicitly by the pressure correction, implies that

$$d(l_b)/dt = l_a u_a - l_c u_c + l_f v_f - l_h v_h + w_b \tag{16a}$$

$$d(l_d)/dt = l_c u_c - l_e u_e + l_g v_g - l_i v_i + w_d, \tag{16b}$$

where the w denotes the vertical flux through the base of the cell. Mass conservation in the x-staggered (or u-cell) then demands that

$$d(l_c)/dt = q_b - q_d + q_j - q_k + w_c, \tag{17}$$

where q (or w) denotes a mass flux. Because the surface locator computes the volume of the u-cell from the average of the volumes of the centred cells on each side, it follows that Eq. (17) is the average of Eqs. (16), and therefore that

$$\begin{aligned} 2q_b &= l_a u_a + l_c u_c, \quad 2q_d = l_c u_c + l_e u_e, \\ 2q_j &= l_f u_f + l_g u_g, \quad 2q_k = l_h u_h + l_i u_i, \\ 2w_e &= w_b + w_d. \end{aligned} \tag{18}$$

Hence the interpolation is by flux rather then velocity averaging, but it reduces to velocity averaging in the fluid interior.

4. MOMENTUM TERMS

In the interior these are implemented using a standard central difference operator, as in ECCLES [3], which conserves energy. This is modified for surface cells and conserves kinetic energy. This is illustrated for the simple case shown in Fig. 7 in which there are no jumps in the free surface location and the staggered cell has only one immediate neighbour in the x- and y-directions. For simplicity we consider only the 2D case in which $\Delta x = \Delta y = 1$, but the principles are applicable more generally.

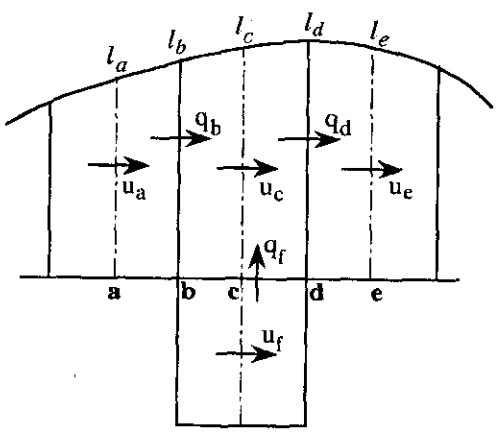

FIG. 7. Momentum flux arrangement in a surface cell.

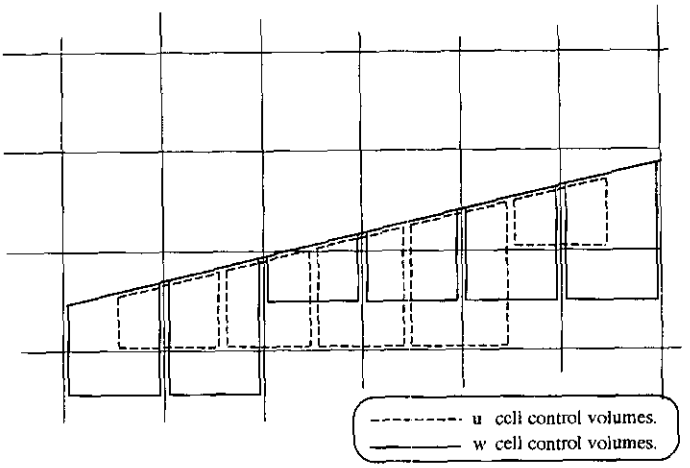

FIG. 8. Discontinuities in pattern of control volumes.

The momentum equation applied to the staggered (u-velocity) cell C is

$$\begin{aligned} d(l_c u_c)/dt = {} & \tfrac{1}{2}(u_a + u_c)q_b + \tfrac{1}{2}(u_f + u_c)q_f \\ & - \tfrac{1}{2}(u_e + u_c)q_d. \end{aligned} \tag{19}$$

The momentum flux across a surface separating two cells is taken as the simple average of the velocity of each cell multiplied by the mass flux between the two cells. In effect, the mass flux term can be regarded as the result of using a flux-average interpolation method for the transporting velocity field. We can demonstrate that Eq. (19) leads to a conservative scheme by invoking the identity

$$d(\tfrac{1}{2}l_c u_c^2)/dt = u_c d(l_c u_c)/dt - \tfrac{1}{2}u_c^2 d(l_c)/dt \tag{20}$$

and, substituting Eqs. (17) and (19) to yield the flux conservative form,

$$d(\tfrac{1}{2}l_c u_c^2)/dt = \tfrac{1}{2}(u_a u_c)q_b + \tfrac{1}{2}(u_f u_c)q_f - \tfrac{1}{2}(u_e u_c)q_d. \tag{21}$$

The convection approximation thus conserves kinetic energy for an arbitrary configuration of the surface. Note that the flux of kinetic energy between two cells is equal to the product of the two velocities and the mass flux. The equations generalise to 3D with the addition of flux terms for the extra dimension. The situation becomes more complicated in 2D, and this is compounded in 3D, when the surface configuration has "jumps."

The new complications introduced when the surface configuration possesses a jump can be illustrated by a 2D example. We will consider this first to establish the key principles and then to consider the additional problems in extending to 3D.

Consider the sloping surface shown in Fig. 8. The cell exten-

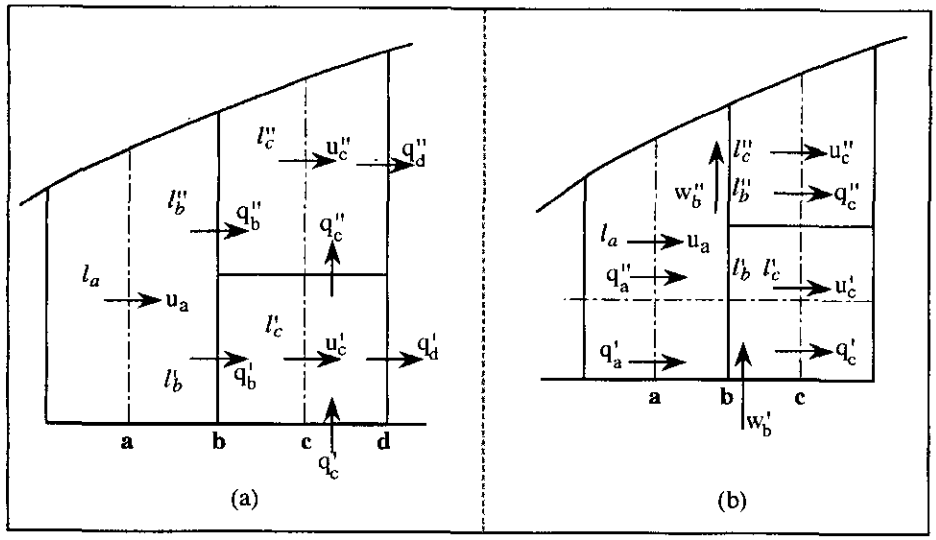

FIG. 9. Flux interpolation at a jump: (a) between u cells; (b) between w cells.

sion definitions when applied to each vertical column lead to discontinuities or jumps in the pattern of FD control volumes. A jump occurs at the boundary of adjacent cells when one cell has more than one immediate neighbour on a side. Jumps can be labelled as UP or DOWN when moving in an increasing co-ordinate direction, or POSITIVE or NEGATIVE, giving the co-ordinate direction in which the extra cells appear. Figure 8 shows examples of jumps in both the u- and w-cells.

The jumps alter the method assigning mass fluxes. In the previous examples without jumps one could use local interpolation rules based on flux averaging to achieve automatic mass conservation in all the staggered cells, provided only that continuity was satisfied in the centred cells. It turns out to be quite difficult to design local interpolation rules with the required properties in 3D with the presence of jumps; the main problem is the complexity of dealing with the configuration of the surrounding cells and the interaction amongst all three sets of staggered cells. We have adopted a compromise in which the mass conservation properties are exactly satisfied but at the expense of introducing a non-local element in the w-flux interpolation.

The method is to calculate the horizontal mass fluxes first, using an appropriate form of interpolation (see later), and then to calculate the vertical fluxes by explicitly enforcing continuity. Of course, the results are only meaningful if mass conservation has been enforced previously in the centred cells.

Typical examples of horizontal fluxes calculations are shown in Figs. 9a and b and we consider first the flux between u-cells. The fluxes q_b'' and q_b' are found by first calculating the average flux Q_b over the whole interface b, using the formula

$$2Q_b = l_a u_a + l_c' u_c' + l_c'' u_c'' \tag{22}$$

and then partitioning Q_b in proportion to the lengths l_b' and l_b' to give

$$(l_b'' + l_b')q_b'' = l_b'' Q_b, \tag{23a}$$

$$(l_b'' + l_b')q_b' = l_b' Q_b. \tag{23b}$$

The horizontal interpolation for Q must be flux-averaged to ensure that the mass of staggered vertical columns is conserved and is an integral part of the surface locator scheme; the only freedom we have is in choosing the partitioning of Q. As we have avoided designing a local interpolation scheme to enforce mass conservation, which would constrain our choice if we found it, we arbitrarily partition assuming constant flux. The fluxes q_d'' and q_d' are found by the same method, and this leaves only the vertical fluxes q_c'' and q_c' (see Fig. 9a). The lower flux is sufficiently far into the fluid interior to be determined by non-surface rules and is found by flux averaging w-velocities; the upper flux q_c'' is found by imposing continuity, i.e., from

$$q_c'' = q_c' + q_b' - q_d'. \tag{24}$$

The result differs from what would have been obtained by flux averaging of the local w-velocities, but it ensures that mass is conserved in the surface staggered cells even in the presence of jumps and that it is consistent with the general order of interpolation used in the surface cells.

The horizontal fluxes between pairs of w-cells are calculated as illustrated in Fig. 9b by supposing that the velocity in the u-cells spanning the interface is constant over the cell. The total flux over the interface is found by summing the individual

velocity contributions multiplied by the surface areas. For example, we have

$$q''_a = (l_a - \tfrac{1}{2})u_a \tag{25a}$$

$$q''_c = l''_c u''_c + \tfrac{1}{2}u'_c. \tag{25b}$$

Note that we have, again, assumed that $\Delta x = \Delta z = 1$. The above rules when applied consistently are sufficient to define all the horizontal fluxes amongst the staggered cells. The vertical fluxes are then defined by using the interior flux at the base of the uppermost full cell in any vertical column and the continuity equation in the same cell. The result is a set of consistent and conservative fluxes for fluid flow amongst all the staggered and centred cells.

The momentum flux terms are implemented using a general transport algorithm for the field variable Φ. Suppose that cell a has volume F_a and Φ_a, and has a number N of neighbouring cells of which cell b is typical. Also, suppose that a mass flux Q_{ab} flows into cell a from cell b. Hence we can write

$$d(F_a)/dt = \sum Q_{ab} \quad (b = 1, N) \tag{26a}$$

$$d(F_a\Phi_a)/dt = \sum \tfrac{1}{2}(\Phi_a + \Phi_b)Q_{ab} \quad (b = 1, N), \tag{26b}$$

where we have used simple averaging of Φ. This is a generalisation of Eq. (19) used above for a surface cell without jumps. Conservation of $F\Phi^2$ follows from using the identity

$$d(\tfrac{1}{2}F_a\Phi_a^2)/dt = \Phi_a d(F_a\Phi_a)/dt - \tfrac{1}{2}\Phi_a^2 d(F_a)/dt \tag{27}$$

and Eqs. (26a) and (26b) to give

$$d(\tfrac{1}{2}F_a\Phi_a^2)/dt = \tfrac{1}{2}\sum (\Phi_a\Phi_b)Q_{ab} \quad (b = 1, N) \tag{28}$$

which is in conservative flux form. Kinetic energy conservation follows if we replace Φ by (u, v, w) and sum over the fluid volume (see Eq. (21)). The terms on the right-hand side of Eq. (28) cancel, leaving only boundary terms which for a solid wall (e.g., at bed) vanish because either Φ or Q is zero. Note that the conservation property is conditional on mass being conserved in the staggered cells, and that we have explicitly enforced this in the presence of jumps.

5. VISCOUS STRESS TERMS

The viscous terms are implemented as in ECCLES [3] for the interior cells and we will describe only the modifications made due to the presence of the surface. The simplest case is when there are no surface jumps and is illustrated below. The complications introduced by jumps are confined to the interfaces between surface cells and the stress calculation for the interfaces.

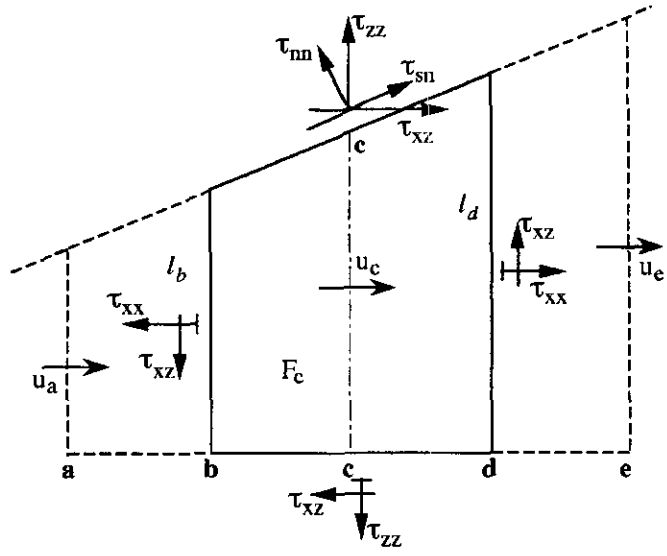

FIG. 10. Viscous stresses in a surface cell.

Consider the 2D surface example shown in Fig. 10. The u-cell is in the xz-plane and for simplicity we have chosen $\Delta x = \Delta z = 1$. The viscous stress is denoted τ and its components on the co-ordinate axes are τ_{xx}, τ_{xz}. The surface stress components are denoted τ_{nn} and τ_{sn}, where the surface co-ordinates (**s**, **n**) are measured locally in the tangential and normal directions. The unit normal vector **n** located at "c" is given by

$$\mathbf{n} = (n_x, n_z) = S^{-1}(l_d - l_b, 1) \tag{29a}$$

$$S = \sqrt{(1 + (l_d - l_b)^2)}. \tag{29b}$$

The viscous contribution to the u-cell momentum is given by summing the stress contributions from all its sides, i.e.,

$$\begin{aligned} F_c d(u_c)/dt = {} & l_d(\tau_{xx})_d - l_b(\tau_{xx})_b - (\tau_{xz})_c \\ & + (n_z S)(\tau_{sn})_{\text{surface}} + (n_x S)(\tau_{nn})_{\text{surface}}. \end{aligned} \tag{30}$$

The stresses τ_{xx} and τ_{xz} are calculated from the local velocity field, whereas the boundary stresses τ_{nn} and τ_{sn} are applied externally and can be freely chosen. The surface boundary condition requires that the tangential stress component vanish, and so we set

$$(\tau_{sn}) = 0, \tag{31}$$

although we could have prescribed a non-zero value due to, for example, an external wind stress. The normal component is unaffected by the boundary condition and is evaluated from the components of the unit normal vector n_x, n_z and the stresses τ_{xx}, τ_{xz}. Hence we write

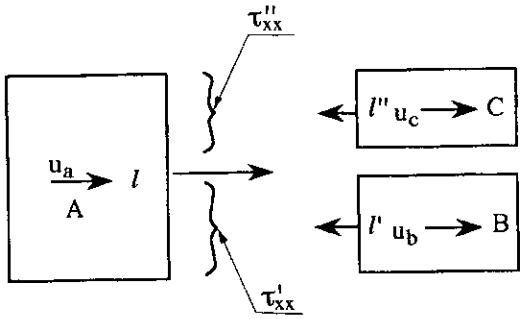

FIG. 11. Situation at the surface jump.

$$(\tau_{nn})_{\text{surface}} = n_x n_x (\tau_{xx})_{\text{surface}} + n_x n_z (\tau_{xz})_{\text{surface}} + n_z n_x (\tau_{zx})_{\text{surface}} + n_z n_z (\tau_{zz})_{\text{surface}} \tag{32}$$

which is also equal to the jump in pressure across the surface when surface tension is neglected. Note that Eq. (32) requires that τ_{xx} be evaluated at "c," whereas it is naturally located at "b" or "d"; the interpolation is done on the whole of the $(n_x n_z \tau_{xz})$ term rather than just on the stress for reasons of computational convenience.

The equations applied to interior cells may be recovered from those above by setting $\mathbf{n} = (0, 1)$, in which case τ_{xx} is not needed, and evaluating τ_{xz} from the velocity field rather than prescribing it. The surface treatment is thus a consistent extension of the interior treatment.

At a surface jump the cell adjoins two cells on a single side; this is illustrated in Fig. 11 for the positive x direction or xx-interface, but the principles involved for the other interfaces are the same. In order to conserve total momentum the total force on the right-hand side of cell, A must exactly balance the sum of the forces on the left-hand side of cells B and C. From the figure, this implies that the rule for connecting cells across a jump must be

$$(l' + l'')\tau_{xx} = l'\tau'_{xx} + l''\tau''_{xx}, \tag{33}$$

where the stresses τ'_{xx} and τ''_{xx} are calculated from the local velocity field and τ_{xx} is the average stress on the whole interface. The possible jumps at other interfaces are simply different expressions of the above example. The method of calculating the stress τ from the velocity field is intimately bound up in the details of the particular interface and leads to many special cases. These will not be enumerated here, but the principle is to evaluate the necessary components of the velocity gradient on the interface and to calculate the components of the viscous stress from the conventional formulae.

6. PRESSURE EQUATION

6.1. *Statement of the Pressure Problem*

For convenience, we shall split the local fluid acceleration $d\mathbf{u}/dt$ into two separate parts: the pressure acceleration $-\nabla p$ and the remainder denoted $\mathbf{H}$ which results from all the other terms in the Navier–Stokes equations. We shall also denote the divergence of $\mathbf{H}$ by s, i.e., $s = \nabla \cdot \mathbf{H}$. The underlying equations associated with the pressure problem can now be written in the convenient form

$$d\mathbf{u}/dt = -\nabla p + \mathbf{H} \tag{34}$$

$$\nabla^2 p = s, \tag{35}$$

where the Poisson equation (35) is simply a re-statement of continuity.

At the surface $z = h(x, y, t)$ the pressure takes on a prescribed value $p_{\text{surf}}(x, y, t)$ equal to the externally applied pressure $p_{\text{ext}}(x, y, t)$ plus the viscous stress component $\mathbf{n} \cdot \tau \cdot \mathbf{n}$; this follows directly by demanding that the total normal stress at the surface vanishes. At the bed $z = 0$ we impose the no-slip condition $w = 0$ at arbitrary time t and, hence, $\partial w/\partial t = 0$. The boundary conditions are, therefore,

$$p = p_{\text{surf}}(x, y, t) \quad \text{on } z = h(x, y, t), \tag{36}$$

$$\partial p/\partial z = H_z \quad \text{on } z = 0, \tag{37}$$

and the pressure is assumed periodic in the x and y directions. In the finite difference implementation we make the arbitrary alteration to $\mathbf{H}$ of setting H_z to zero in the bed w-cells. In effect, we apply a constraining force to prevent any acceleration in the cell. This allows us to solve Eq. (35) with the simplified boundary condition

$$\partial p/\partial z = 0 \quad \text{on } z = 0 \tag{38}$$

but to obtain an identical pressure field. The pressure problem

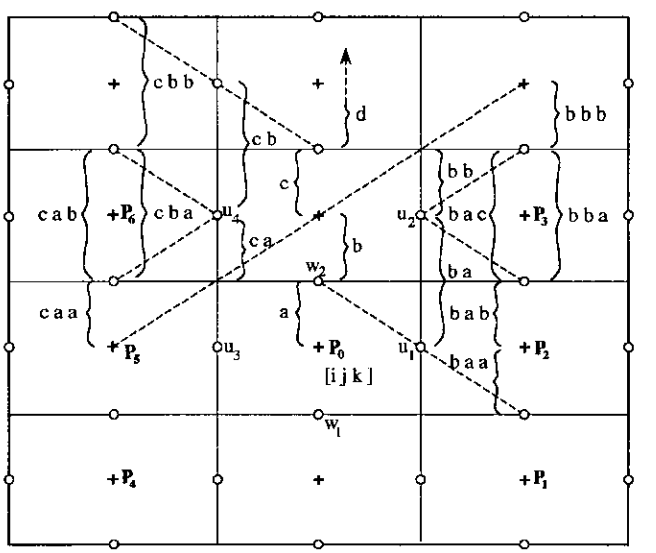

FIG. 12. Logic diagram for FD-star in structure cell.

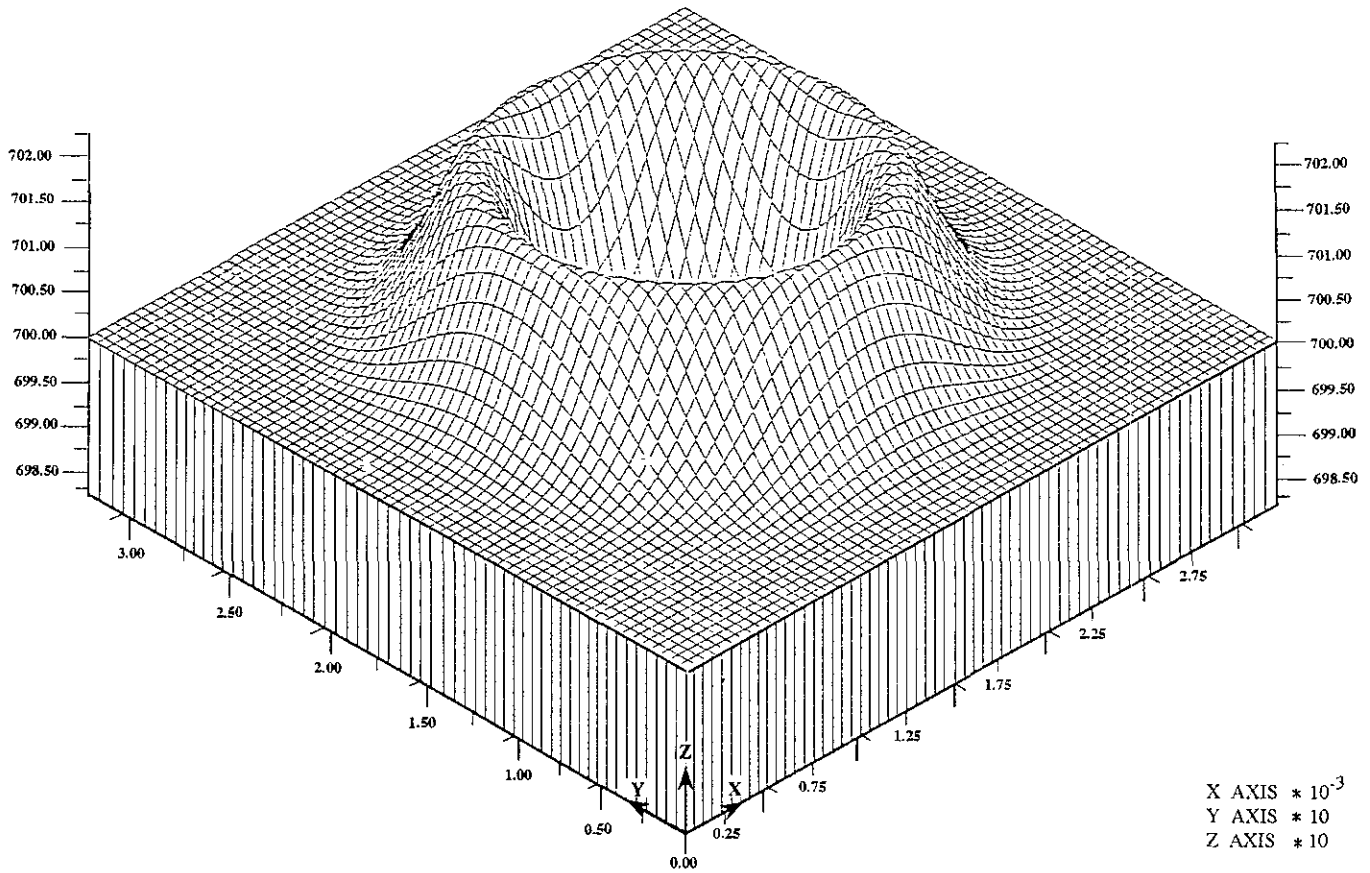

FIG. 13. Gaussian profile.

is now stated in a less general form which efficient numerical solvers can take advantage of. The effect of the pressure field on the total kinetic energy K of the flow can be seen from the energy equation

$$dK/dt = \prod + \int (-p_{\text{surf}})_{\text{surface}}(\mathbf{u} \cdot \mathbf{n})\, dA, \tag{39}$$

where $\mathbf{n}$ denotes the unit normal vector. The first term Π represents the total energy change due to external body forces and viscous dissipation, and the second term represents the pressure-energy contribution from the surface boundary condition (36). We have tried to design the discretisation of the pressure terms in such a way that they satisfy the discrete version of (39) for arbitrary configurations of the surface. Schemes that do not, cannot be conservative. So far, we have been able to satisfy the energy equation (39) only at the expense of accuracy: for example, by reducing the degree of pressure interpolation used in the surface cells. Our early experience indicates that the errors will probably be too large, but this depends very much on the particular flow and we have retained the scheme as a code option. When the pressure is interpolated linearly in the surface cells, the scheme has an energy conservation error which is proportional to the squared vertical pressure gradient $(\partial p/\partial z)^2$ near the surface. We are considering ways of reducing this, for example, by taking the gravity term into the pressure variable and so removing the hydrostatic pressure gradient. Again, the most appropriate technique will depend on the particular flow and we have, therefore, included in the code several alternative ways of treating the gravity terms.

6.2. *Discretisation*

In the fluid interior, standard centred finite difference formulae are used to represent the ∇^2 and ∇ operators. Again, we will confine the description to the modifications required for the cells at or near to the surface.

The FD star is built up by combining the gradient operator with the continuity equation for a surface cell. This process can be expressed symbolically by the two equations

$$\delta \mathbf{u}_{ijk} = \sum_{lmn} A_{ijk}^{lmn} p_{i+l,j+m,k+n} + \mathbf{H}_{ijk} \tag{40}$$

$$0 = \sum_{a} B_a(\mathbf{u}_a + \delta \mathbf{u}_a), \tag{41}$$

where the notation tries to imply in some consistent way a

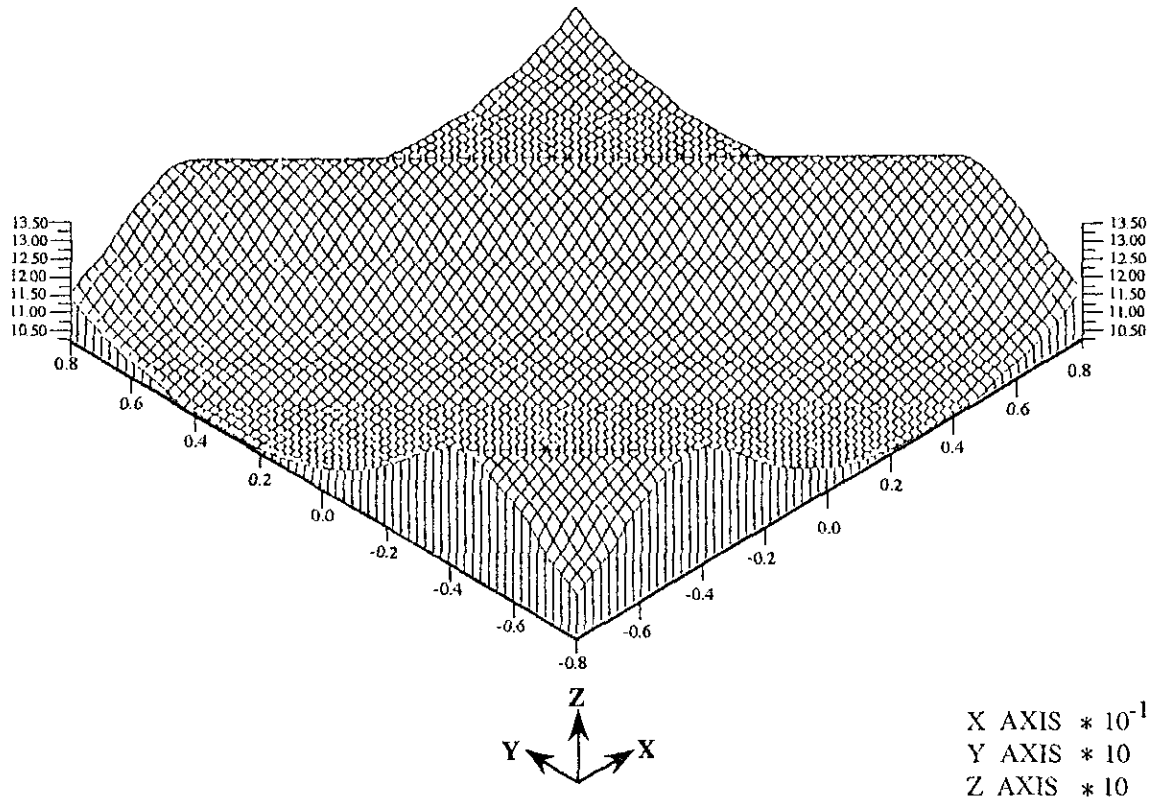

FIG. 14. Diagonal solution run.

summation of the appropriate quantities around the cell. Equation (40) expresses the idea that the cell acceleration depends linearly on the surrounding pressures; Eq. (41) is a statement of continuity applied to a perturbed velocity. In these equations the quantities A are coefficients involving time and inverse lengths, and the quantities B are areas of the faces surrounding cell ijk.

Combining the two equations to eliminate $\delta \mathbf{u}$ and keeping track of the coefficients of pressure leads to the FD star for the surface cell ijk. This is expressed in terms of the star coefficients C_{ijk}^{lmn}, where ijk denotes indices of the pressure cell and lmn denotes the relative indices of the neighbouring cells:

$$0 = \sum_{lmn} C_{ijk}^{lmn} p_{i+l,j+m,k+n} + s_{ijk} \quad \text{for } lmn \in \{-1, 0, +1\}. \quad (42)$$

Of the 27 coefficients implied above only the 15 located in the (xz) and (yz) plane are, in general, non-zero. This is because interpolation of the pressure is confined to the vertical direction, thus linking pressure cells vertically, but it is not performed sideways. The surface boundary condition enters the star coefficients as part of the pressure interpolation and contributes to the source term.

Equation (42) is general enough to represent interpolation formula; the coefficients and source term can be chosen appropriately to implement this. Also, we can apply the same equation to points outside the fluid and recover the null value $p_{ijk} = 0$ if we choose to set C_{ijk}^{000} to unity with all other C_{ijk}^{lmn} and s_{ijk} is set equal to zero. This is convenient because it allows the equations to be defined on a regular mesh enclosing the fluid and their solution by a standard procedure without reference to the position of the free surface. However, more efficient solvers which exploit the structure of the equations and make explicit use of the surface information can be added without difficulty. The coefficients and source term are recalculated at each time step for the two uppermost layers of surface cells only and are stored prior to the solution process.

6.3. *Structure of the Finite Difference Star*

The FD star Eq. (42) is built up according to the logical structure imposed by the cell extension rules. This is illustrated in Fig. 12. The dotted lines indicate the limits on possible surface excursions imposed by the maximum slope condition. We examine the logic concerned with the pressure p_0 at a point known to be inside the fluid. The depth of this point determines whether p_0 is interpolated, or, if not, how the control volume is defined.

For example, if the surface cuts the interval a then p_0 is not spanned by w-cells and the surface continuity equation cannot

TABLE I

		$\nu = 0.0$			$\nu = 10^{-3}$		
$\Delta x, \Delta y, \Delta z$	Δt	PE	KE	TE	PE	KE	TE
$\frac{1}{8}$	T/63	+23.8	−16.0	+4.4	+11.6	−25.2	−6.3
	T/6300	+22.1	−16.7	+3.2	+10.2	−25.7	−7.3
$\frac{1}{16}$	T/125	+8.5	−6.9	+1.0	−5.3	−19.0	−12.1
	T/12500	+8.3	−7.0	+0.9	−5.7	−19.0	−12.1
$\frac{1}{32}$	T/250	−0.1	+0.1	0.0	−11.4	−11.3	−11.4
	T/25000	−0.2	−0.1	−0.1	−11.5	−11.5	−11.5

be applied; instead p_0 must be interpolated from below. This is done by an appropriate modification of the coefficients and source term of Eq. (42). When the surface cuts interval b the existence of the upper w-cell w_2 permits the surface continuity equation to be used, and the control volume is extended up to the surface. If the surface cuts interval c the centred cell associated with p_0 can never be intersected by the surface and thus must be the natural control volume in which the surface continuity equation is applied. The star is guaranteed to be identical to the standard interior one if the surface cuts the open interval d. This is because the u-cells u_2 and u_4 must be the uppermost ones and, therefore, u_1 and u_3 must be standard full cells. In this configuration the surface has no possible effect on the star; it is tested for to provide a shortcut in the calculation and to demonstrate that only two layers of stars need ever be calculated at one time step.

Thus far the star is fully defined unless the surface cuts either b or c; the subsequent conditional branches are shown on the right and left sides of Fig. 12.

First, consider the case of the surface cutting b. The extended control volume is used, and the configuration of the right u-cells depends on whether the surface cuts *ba* or *bb*. If the cut is *ba* there is only one u-cell u_1, whereas if the cut is *bb* there are two u-cells u_1 and u_3 to be considered.

The pressure points p_1, p_2, and p_3 are linked into the star according to the logic governing the vertical interpolation of the pressure acting on the mid-point of the right-hand side of the right u-cells. The number of possible branches is reduced if we note that the p_3 point is governed by the first part of the current logic; i.e., it is known to be interpolated if the surface cuts within half a cell above it. This fact allows a branch in the logic for interpolating past p_3 to be removed.

For the single cell case *ba* there are three distinct surface intersections *baa*, *bab*, *bac*, each of which invokes an interpolation rule for the pressure at the mid-point of the right-hand side of the u-cell, and so it selects the coefficients for p_1, p_2, and p_3. In this case it is easy to show that *baa* selects p_1, *bab* selects p_1 and p_2, and *bac* selects p_2. For the two u-cell case *bb* the lower u-cell is full and therefore always invokes p_2; the upper u-cell invokes either p_2 when the surface cuts *bba*, or p_2 and p_3 when the surface cuts *bbb*. Again, the branching takes account of the interpolation for p_3.

Now consider case when the surface cuts the c interval. The control volume is the standard full cell, and the surface above the left u-cell can cut only in *ca* or *cb*.

The first case *ca* produces a single u-cell, and the interpolation is determined by the branch between *caa* and *cab*. The first links in p_4 and p_5, and the second links in p_5 only. The second case *cb* produces two u-cells; the lower cell u_3 links in p_5, and the upper cell u_4 leads to the branches *cba* or *cbb*. The first links in p_5 and the second links in p_5 and p_6.

The logic above is applied independently in each of the positive and negative x and y co-ordinate directions to assemble the star coefficients.

In general, the effect of the cell extension method, apart from complicating the calculation of the FD star, is to make the star span more points: 9 in 2D and 15 in 3D, rather than the usual 5 and 7. In addition, the difference is confined to the top two layers of cells and, therefore, only these need be recalculated at each time step. The surface pressure boundary condition enters the star as part of the interpolation rules and, as it is a constant for the purposes of the pressure solution, it contributes only to the source term.

7. SEQUENCE OF OPERATIONS

The velocity $\mathbf{u}(x, y, z, t)$ and surface elevation $h(x, y, t)$ are advanced over a time step Δt in the following way. A provisional velocity $\mathbf{u}^*$ is calculated from the Navier–Stokes equations, according to

$$\mathbf{u}^* = \mathbf{u}(t) + \tfrac{3}{2}\Delta t\, \mathbf{H}(t) - \tfrac{1}{2}\Delta t\, \mathbf{H}(t - \Delta t). \tag{43}$$

This can be thought of as advancing the velocity but neglecting the pressure contribution. The pressure $p(t)$ is found by solving the Poisson equation

$$\nabla^2 p + (\tfrac{2}{3}\Delta t)\, \nabla \cdot \mathbf{u}^* = 0 \tag{44}$$

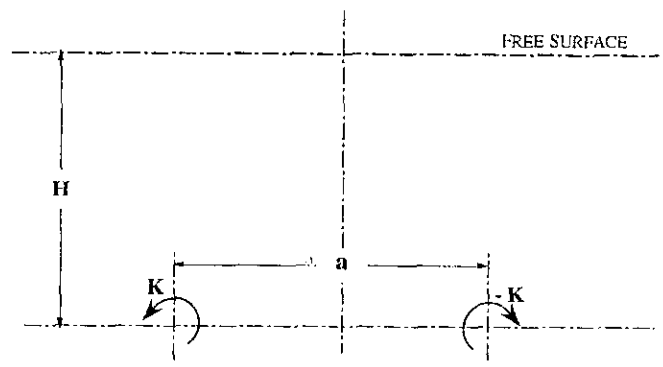

FIG. 15. Initial location of the vortices.

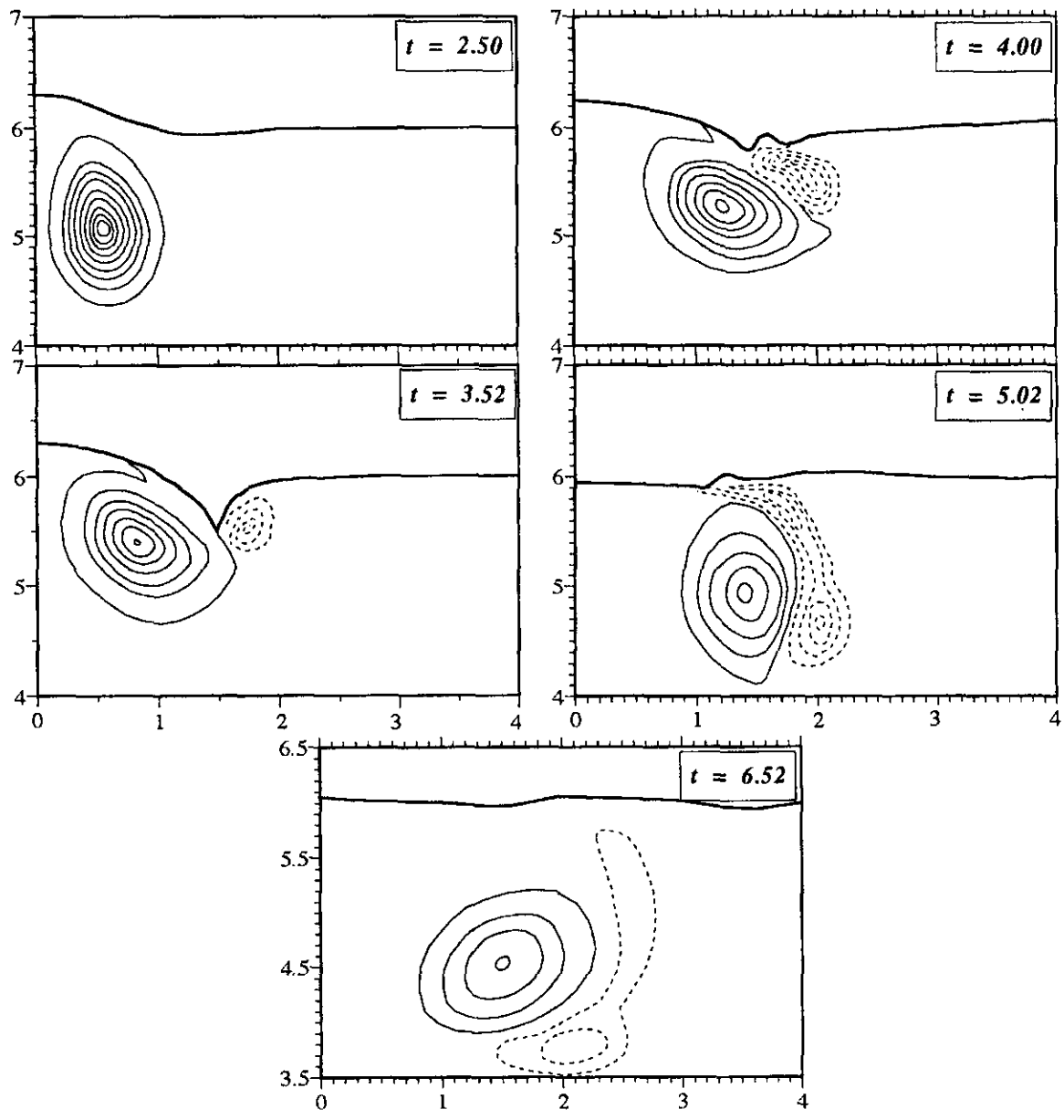

FIG. 16. Ohring and Lugt equivorticity lines: ..., −3, −1, +1, +3, ...

with the appropriate boundary conditions. In calculating the divergence terms for surface cells we have accounted for the moving surface by adopting a stationary surface and allowing the fluid to pass through it. The velocity advancement is completed by

$$\mathbf{u}(t + \Delta t) = \mathbf{u}^* - \tfrac{3}{2}\Delta t\, \nabla p(t) \tag{45a}$$

$$\mathbf{H}(t) = \mathbf{H}(t) + \tfrac{1}{2}\Delta t\, \nabla p(t). \tag{45b}$$

The surface is advanced by calculating the change in volume of a vertical column of cells $\Delta\ \mathrm{col}_{ij}$ as a result of fluid flowing into it during the time step. Thus we calculate

$$\Delta\ \mathrm{col}_{ij} = \Delta t \cdot (\text{Flux into } \mathrm{col}_{ij}), \tag{46}$$

where the flux is based on the old elevations $h(t)$ and the new velocity $u(x, y, z, t)$. The change of elevation follows from the surface locator rules and is given by

$$h_{ij}(t + \Delta t) = h_{ij}(t) + (1/\Delta x\, \Delta y)(\mathrm{col}_{ij}(t) + \mathrm{col}_{ij}(t + \Delta t)). \tag{47}$$

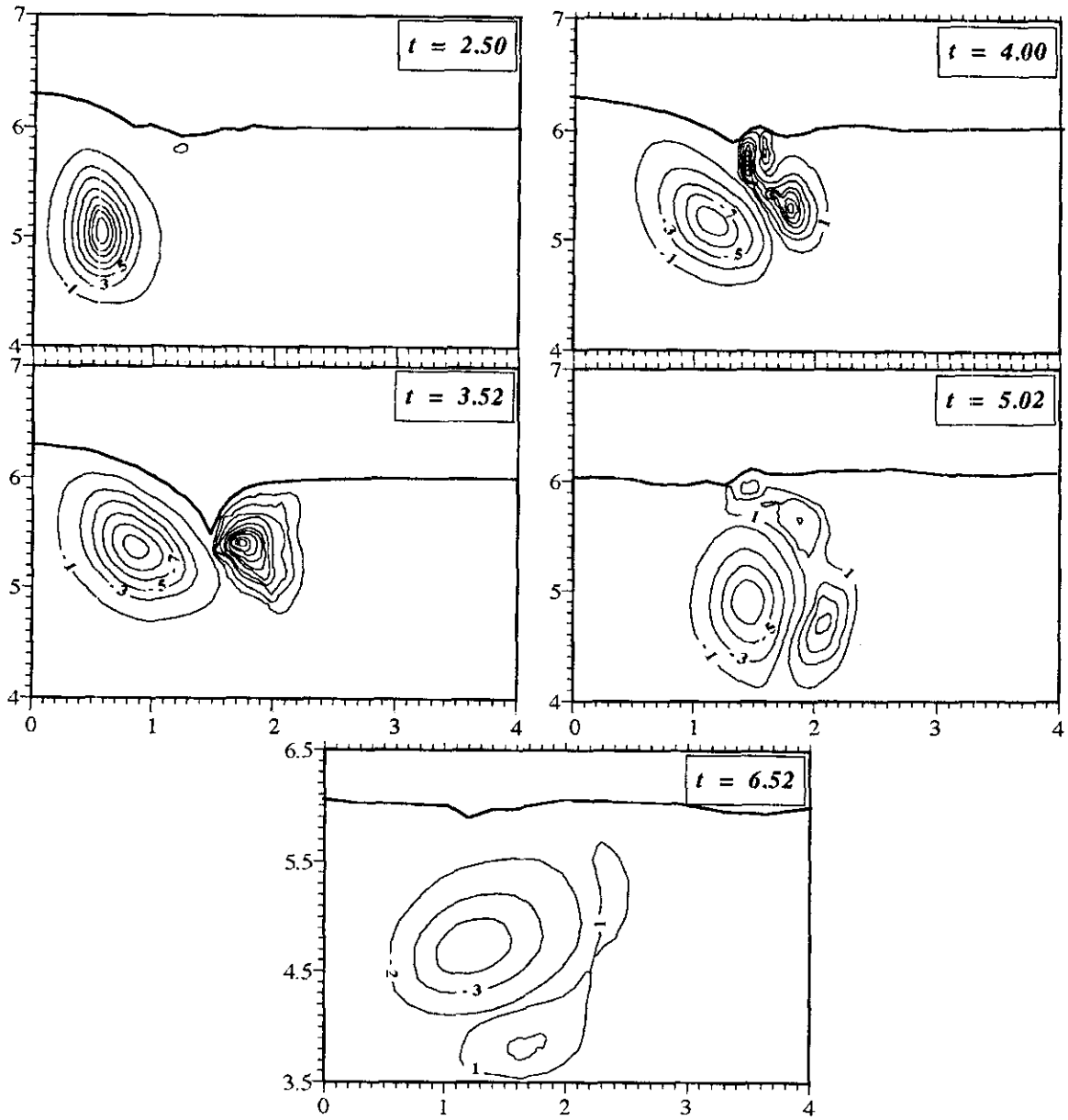

FIG. 17. FRECCLES equivorticity lines: ..., −3, −1, +1, +3, ...

Once the new surface elevation is established the split–merge process is applied to the velocity cells where necessary. Since $\Delta\ \mathrm{col}_{ij}(t + \Delta t)$ is not known until $\mathbf{u}(t + \Delta t)$ is found (46) and (47) are iterated several times. The above scheme is essentially a Adams–Bashforth (AB) step for velocity and a Crank–Nicholson (CN) step for surface elevation.

A linear stability analysis shows these mixed second-order schemes to be very slightly unstable for free surface small amplitude waves but the degree of instability is much less than the damping introduced into the system by viscosity. As a user defined option, though, a neutrally stable free surface long wave system may be chosen which will invoke a forward Euler step for the velocity and a backward Euler step for the surface elevations.

8. FIRST EXAMPLES OF USE

8.1. *Gaussian Hill*

In order to test the three-dimensional characteristics of the code the decay of a small amplitude inviscid radially symmetric wave was modelled. The wave had an initial Gaussian profile following the relationship $0.7 + 0.01\ \exp(-r^2/10)$, where $r^2 = x^2 + y^2$. Figure 13 shows the wave after 200 time steps using $\Delta t = 0.05$ and $\Delta x = \Delta y = \frac{1}{2}$, $\Delta z = \frac{1}{4}$, $g = 1.0$, and with

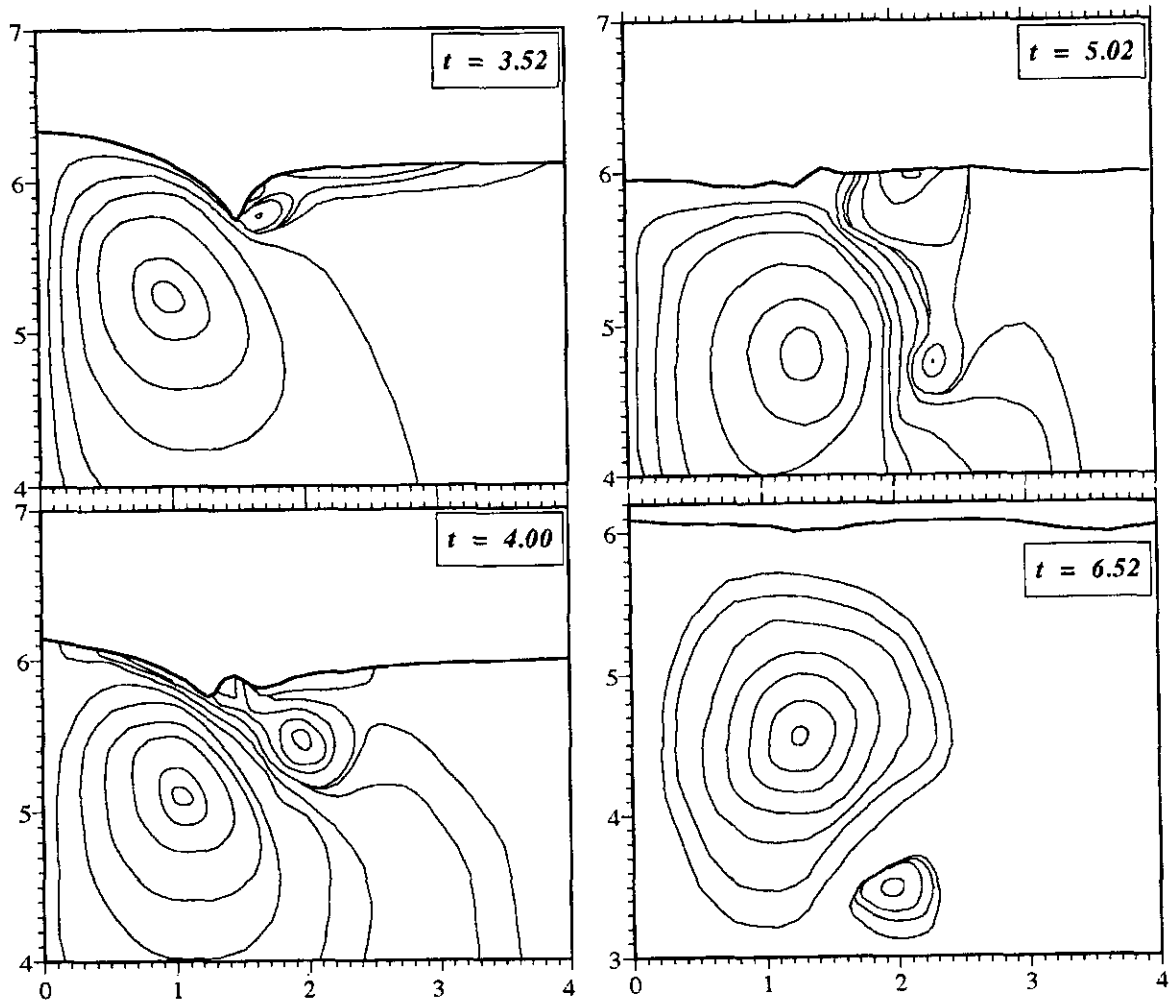

FIG. 18. Ohring and Lugt streamlines.

64 cells in the x and y directions. Euler time-stepping was used for the velocity terms and, for the initial maximum height of 0.71, no split/merge routines were activated. Figure 13 shows the wave to be perfectly symmetrical about its centre and for the peak to be an approximately radial distance of 18 cells, compared with the theoretical value of 17 cells.

8.2. *Soliton Runs*

We have run a train of finite amplitude solitons diagonally across the surface of a square box so as to exercise the split–merge operations and to test the periodic boundary conditions. The waves must be separated by $1/\sqrt{2}$ of the box diameter to satisfy periodicity along the box axes, and the characteristic wavelength $\lambda = 4d^3/3D$ must be significantly smaller than this to keep the waves separate and distinct; here d denotes the fluid depth and D denotes the maximum wave height. The individual solitons were calculated for $D/d = \frac{1}{3}$ using the second-order approximation (see, e.g., Wehausen and Laitone [8]) thus neglecting terms of $O(D/d)^3$. The interactions between successive waves in the train may be neglected, provided that they are sufficiently far apart. The characteristic time for the waves is $T = \sqrt{(d/g)}$, where g denotes the acceleration due to gravity, and we have run the test with $\Delta t/T = \frac{1}{20}$. The box is $16d$ long with 64 cells along each side, so that the horizontal cell size is $\Delta x/d = \Delta y/d = \frac{1}{4}$; the vertical cell size is $\Delta z/d = \frac{1}{10}$. Again, the energy conservative Euler time-stepping scheme was used.

The resulting surface elevation is shown in Fig. 14 after the waves have travelled a distance of just over a box diagonal. It exhibits some small amplitude perturbations on the trailing face of each wave; these seem to be associated with the points where the wave intersects the vertical mesh lines, and may be a side effect of the split–merge process. In this test the split–merge process always acts at the same points on the wave and it is expected that some errors should accumulate there. Overall, the wave shows little effect of having been passed through a mesh of cells.

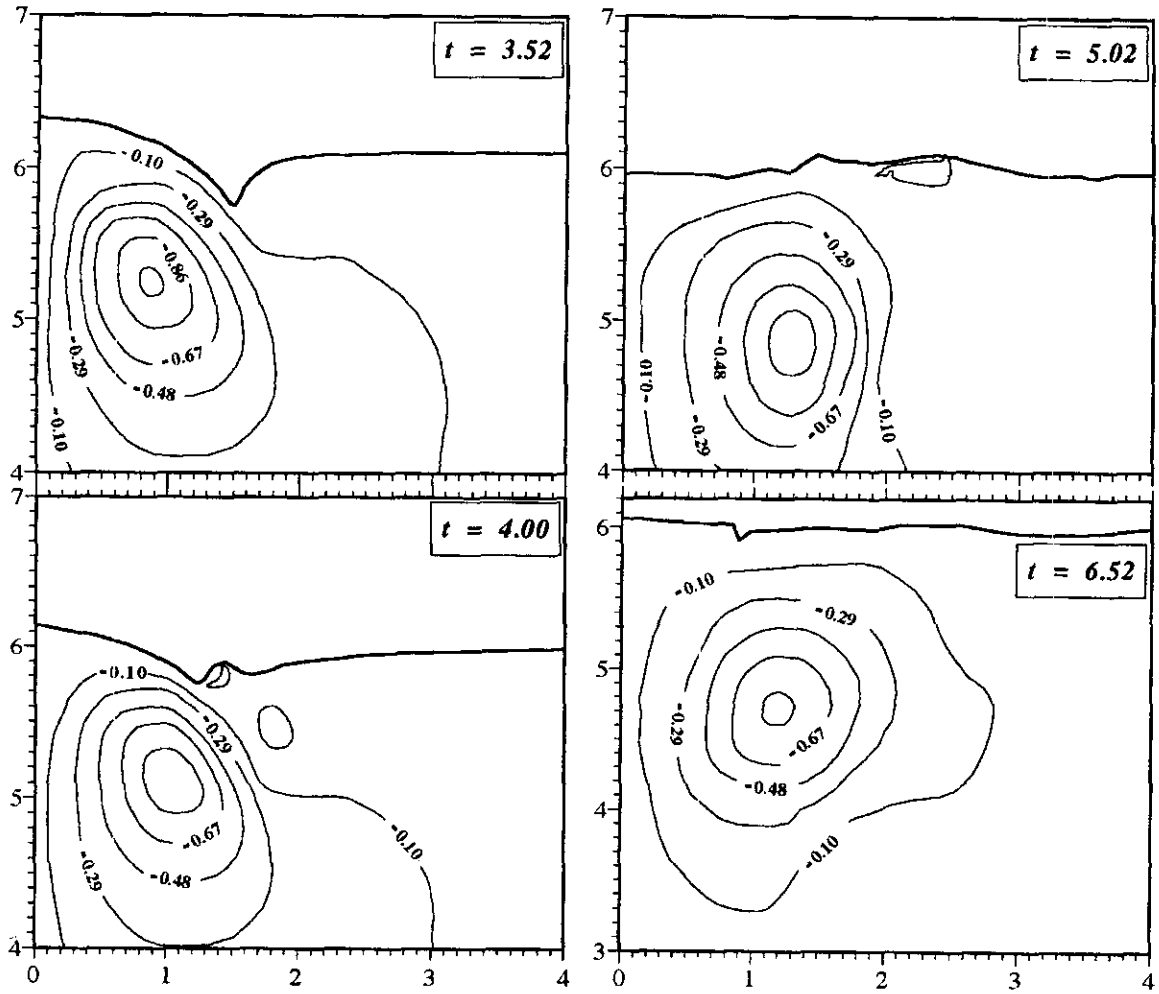

FIG. 19. FRECCLES streamlines.

8.3. *Viscous Channel Flow*

We ran a laminar channel flow test problem to provide a first check on the implementation of the viscous terms. The code was able to reproduce the parabolic velocity profile of a (diagonally) inclined channel flow with the correct zero stress surface boundary condition. In order to further test the momentum terms, the code was run with a point source of momentum given to a w-cell in the body of the fluid. This was achieved by over-writing the w-cell velocity by the value 1.0, and it was calculated to be sufficient to invoke a split/merge in the surface cells. This test enabled the radial symmetry of the viscous stresses to be checked.

8.4. *Viscous Decay of an Airy Wave*

The energy decay of a viscous Airy wave of steepness 0.05 was investigated for varying degrees of spacial and time resolution. The wave length, box length, and fluid depth were each given a value of 1.0 which resulted in a wave celerity of 1.25 with a period (T) of 0.8.

Table I gives values of the change (%) in potential, kinetic, and total energy of the wave after it has travelled for a time equal to one complete wave period. These results are given for two values of viscosity: 0.0 and 10^{-3}. For this later non-zero value the theoretical energy decay was calculated to be 11.9% using the expression given in Lamb [6].

The first line of results for each value of the spacial resolution is for a time step that gives a wave Courant number equal to 0.32 and the results below these are for time steps that are $\frac{1}{100}$th of these values. This second set of results are given to illustrate the convergence properties of the finite difference schemes used, in that the errors tend to zero as Δx and $\Delta t \to 0$. It will also be noted that the computed values of the energy decay converge to the theoretical value as the spacial and time resolution increases. Mass was conserved exactly as the flow was periodic in both horizontal directions.

8.5. *Interaction of a Vortex Pair with a Free Surface*

The test of Ohring and Lugt [7], in which a pair of counter-rotating vortices ascended towards and, subsequently, travelled

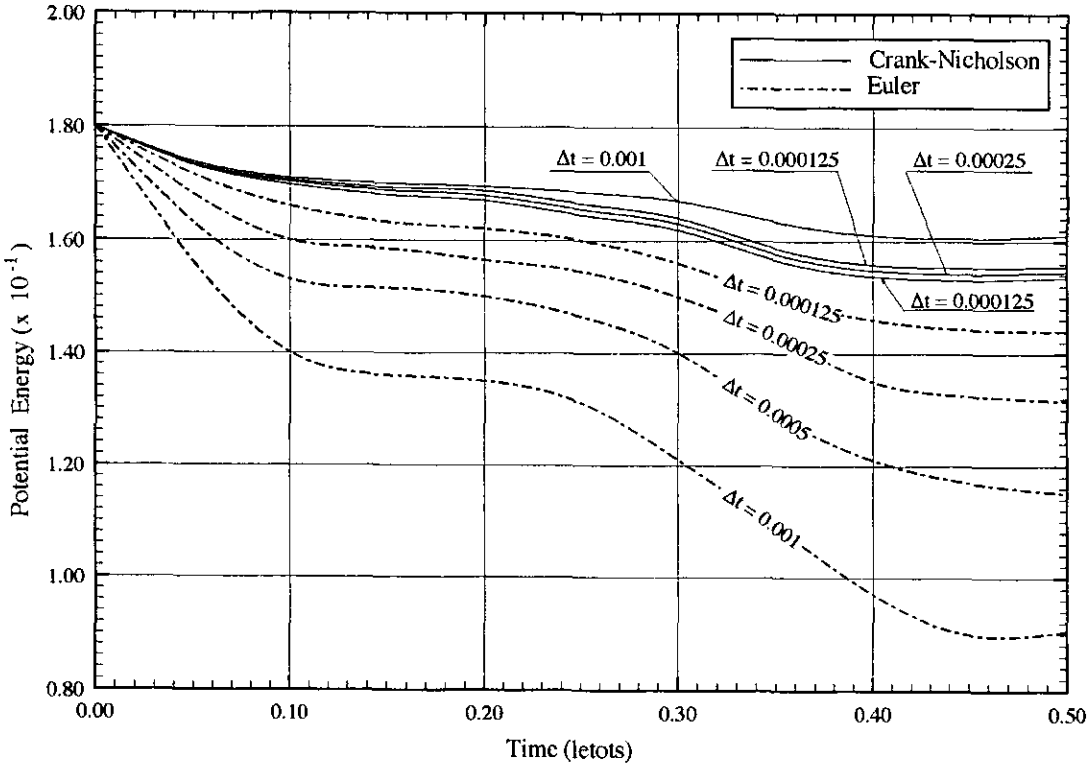

FIG. 20. Potential energy plots.

away from, the free surface of a viscous liquid was investigated. When the code is run to simulate turbulence the Reynolds number will be very much higher (and other properties of the code become important, such as energy conservation) than in this test. However, this test was still thought worthwhile as it provides another rigorous check on the implementation of the viscous stresses in the free surface cells. Figure 15 shows the initial position of the vortices and the test chosen for comparison was Re = 100, Fr = 0.4, and zero surface tension. Which, using the relationships: Re = κ/μ and Fr = $\kappa/\sqrt{(ga^3)}$, gave $\mu = 0.01$ and $g = 6.25$ for $\kappa = a = 1$. H was set to 3 and the depth of liquid to 6.

Ohring and Lugt used an adaptive-grid technique in which the physical domain was mapped onto a uniformly spaced Cartesian mesh. They used 201 points vertically and a horizontal mesh density of 50 grid points per unit length around the central region of computation. This spacing was then stretched towards the boundary positioned at 10.8 length units from the centre of the vortices, where they then used one-sided derivatives as boundary conditions. The present authors used uniformly spaced cells with 60 vertically and 30 per unit length horizontally, a resolution that was primarily dictated by ensuring that surface slopes were within the cell aspect ratio. Periodic boundary conditions were used, together with a time step of 1.666×10^{-3} time units (1 time unit = a^2/κ).

Figure 16 shows vorticity plots for various times taken from the Ohring and Lugt paper, whereas Fig. 17 shows the corresponding plots, obtained using the authors code, FRECCLES. Figure 18 shows Ohring and Lugt values of the stream function for the times t = 3.52, 4.0, 5.02, and 6.52, whereas Fig. 19 shows plots of the values obtained from FRECCLES. It was not possible for the authors to plot all the corresponding streamlines, as contour values are not given in the Ohring and Lugt paper. Close agreement can be seen to have been obtained and any small differences are attributed to the split–merge process at the free surface of the authors code, together with the interpolation procedures used in their plotting routines. An interesting feature of the flow is the appearance of secondary vorticity with an opposite sign to the main vortex produced at the free surface by an interaction with the tangential stress boundary condition. It can be seen that the positions of the vortices (both primary and secondary) and the surface profiles are in close agreement to those of Ohring and Lugt [7]. The production of secondary vorticity, in particular, is dependent upon endorsement of the tangential stress condition.

8.6. *Conservation of the Potential Energy*

Figure 20 shows plots of the potential energy above the mean surface elevation for the computational box against time for the combined *AB* and *CN* schemes and the forward/backward Euler scheme. The horizontal time scale is in letots, i.e., time

normalised by the friction velocity divided by the depth of fluid. The initial conditions for these runs were taken from a dump of the conditioned turbulent velocity field produced by ECCLES [3] running the same problem but with a rigid lid. Any significant deviation in the way the energy parameters varied after the turbulent fields were transferred must be due either to the removal of the lid or to numerical quirks of FREC-CLES. Here we used $g = 1000$, slope $= \frac{1}{1000}$, viscosity $= \frac{1}{192}$, the number of cells in the x, y, and z directions $= 32$, 64, and 16, respectively, the length of the box in the x and y directions $= 2\pi$ and π, respectively, the friction velocity was 1.0, and the depth $= 1.0$. Four different time steps Δt were used: 0.001, 0.0005, 0.00025, 0.000125. From the figure it can be seen that as $\Delta t \rightarrow 0$ all the curves converge as expected and that for moderately large values of Δt the errors in the $AB + CN$ scheme are much less than those of the Euler schemes. Further, the use of a time step of 0.001 for the $AB + CN$ scheme gave acceptable errors in the potential energy. This time step, which is equivalent to that which would be used in our ECCLES [3] code for the equivalent rigid lid problem gave a convective Courant number of 0.1 in the streamwise direction and wave Courant numbers of 0.16 and 0.65 in the streamwise and spanwise directions.

9. CONCLUSIONS

The tests described above, together with detailed checking of the code, were necessary in order to test that the code was performing as it should, i.e., that the stresses, mass fluxes, etc., were being calculated correctly and that the split–merge technique was also working properly. The tests were also necessary to show that the code conserved the important properties of mass, momentum, and energy in the limit as $\Delta t \rightarrow 0$. It was possible to show conservation of this latter property when using a scheme that conserved the energy of small amplitude long waves, but not when a second-order time-stepping scheme (necessary for LES work) such as $AB + CN$ was used. However, it was shown that the combination of AB for the velocity terms and CN for the surface advancement was acceptable for real fluids. The authors intend to proceed with the code development by introducing a sub-grid model. This can then be used with the combined $AB + CN$ scheme above to model turbulence in an open channel.

REFERENCES

1. R. K. C. Chan and R. L. Street, *J. Comput. Phys.* **6,** 68 (1970).
2. R. K. C. Chan and R. L. Street, in *Proceedings, Second Int. Conf. Numer. Methods Fluid Dynamics, Berkeley, 1970,* p. 429 (unpublished).
3. S. Gavrilakis, H. M. Tsai, P. R. Voke, and D. C. Leslie, in *Notes on Numerical Fluid Mechanics,* Vol. 15, edited by U. Schumann and R. Friedrich, (Vieweg, Braunschweig, 1986), p. 105.
4. C. W. Hirt and B. D. Nichols, *J. Comput. Phys.* **39,** 201 (1981).
5. C. W. Hirt, B. D. Nichols, and N. C. Romero, Los Alamos National Laboratory Report LA-5852, April 1975 (unpublished).
6. H. Lamb, *Hydrodynamics* (Cambridge Univ. Press, Cambridge, UK, 1959), p. 624.
7. S. Ohring and H. J. Lugt, *J. Fluid Mech.* **227,** 47 (1991).
8. J. V. Wehausen and E. V. Laitone, in *Encyclopaedia of Physics,* edited by S. Flugge (Springer-Verlag, Berlin, 1960), Vol. 9, p. 446.