INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 22, 691–712 (1996)

ELAFINT: A MIXED EULERIAN–LAGRANGIAN METHOD FOR FLUID FLOWS WITH COMPLEX AND MOVING BOUNDARIES

H. S. UDAYKUMAR, W. SHYY AND M. M. RAO

Department of Aerospace Engineering, Mechanics and Engineering Science, University of Florida, Gainesville, FL 32611, U.S.A.

SUMMARY

In this work a mixed Eulerian–Lagrangian technique is devised, hereinafter abbreviated as ELAFINT (Eulerian–Lagrangian Algorithm For INterface Tracking). The method is capable of handling fluid flows in the presence of both irregularly shaped solid boundaries and moving/free phase boundaries. The position and shape of the boundary are tracked explicitly by the Lagrangian translation of marker particles. The field equations are solved on an underlying fixed grid as in Eulerian methods. The interface passes through the grid lay-out and details regarding the treatment of the cut cells so formed are provided. The issues involved in treating the internal boundaries are dealt with, with particular attention to conservation and consistency in the vicinity of the interface. The method is tested by comparing with solutions from well-tested body-fitted co-ordinate methods. Test cases pertaining to forced and natural convection in irregular geometries and moving phase boundaries with melt convection are presented. The capability developed here can be beneficial in solving difficult flow problems involving moving and geometrically complex boundaries.

KEY WORDS: ELAFINT; interface tracking; solidification

1. INTRODUCTION

Many fluid dynamics problems of practical interest involve irregularly shaped boundaries. In some cases the boundaries also move and change shape.[1,2] Considerable efforts have been directed in the past towards simulating fluid flows in the presence of such boundaries, which often separate distinct phases. For example, algorithms based on body-fitted co-ordinates have been extensively developed in the last two decades.[3] When the interface is not highly deformed, a single boundary–conforming grid arrangement is convenient for obtaining solutions to the transport equations.[4,5] However, the efficacy of the boundary-fitted grid deteriorates when the interface becomes highly distorted, owing to difficulty in obtaining smooth grid distributions. The situation can be further complicated when there are moving boundaries. The method experiences serious difficulties when interfaces merge or fragment. For highly deformed interface shapes, purely Eulerian methods[2,6–10] have been found to be useful, since in such methods the grid is completely independent of the interface and is usually of fixed Cartesian form. However, when details of the interface shape are to be explicitly tracked, the Eulerian methods are not suitable, since in such methods the interface is deduced based on a computed fluid fraction field. In tracking highly distorted interfaces, a combination of the strengths of Eulerian and Lagrangian methods will be desirable. Some efforts have been devoted recently to developing such methods.[11–17] Such methods employing fixed structured grids afford simplicity and availability of well-tested, economical field solvers. The only moving component is the interface. The presence of the internal boundary that passes through the fixed grid lay-out calls for special treatment, since some of the control volumes in the domain are fragmented by the interface. Udaykumar and Shyy[11] have described

CCC 0271–2091/96/080691–22

Received 20 June 1994
Revised 30 May 1995

some procedures designed to deal with the pure conduction system including the cut cells. Quirk[14] has detailed some of the procedures involved in dealing with cell fragments in the framework of compressible, inviscid flows around stationary obstacles. However, the calculation of fluxes is not clear from that source. Zeeuw and Powell[15] and Bayyuk *et al.*[16] employ a Cartesian grid to track the motion of solid objects through an inviscid compressible fluid. Miyata[12] has employed a fixed grid formulation for the simulation of wave breaking. The treatment of the free/moving surface in that work is not crucial to the physical phenomena under study and hence the treatment there is not very detailed. The physical problems discussed here are strongly affected by viscous effects and thus diffusion terms are to be included. Thus the full Navier–Stokes equations for incompressible flow including phase change are solved, with particular focus on the conservation of fluxes at the interface.

We choose for investigation the solidification phenomenon, where the interface is in motion and there is transport of momentum and energy across the interface. In various solidification processing arrangements, conditions are encountered under which the interface separating the solid and liquid phases needs to be accurately resolved in order to simulate the transport and phase change processes.[1,18,19] At the microscopic level, solid–melt interfaces experience morphological instabilities and break up into convoluted structures such as dendrites and cells.[18] At the macroscopic level, convective transport also causes the interface to display highly complicated characteristics in time and space.[20,21] In this work we develop a numerical simulation technique called ELAFINT (Eulerian–Lagrangian Algorithm For INterface Tracking). The technique can handle the conditions that are encountered during a typical solidification process.

(a) The solid–liquid interface is a phase discontinuity; boundary conditions in each phase have to be applied for the fluid flow equations on this moving internal boundary. The interface is a source of heat, mass and momentum and can be arbitrarily distorted.
(b) The method developed should be able to handle the solidification processes at the macro- and microscales.
(c) Other demands on the solution procedure are similar to those on the pure conduction system.[11]

It is noted that the method developed here is applied to the solidification problem on account of the interesting and challenging interface behaviour involved; it is, however, not restricted to solid–liquid interfaces. The formulation and numerical technique are easily applied to other interfacial phenomena as well, with modifications for the boundary conditions.

The numerical simulation of solidification phenomena requires the solution of the transport equations in each phase. In the liquid phase the Navier–Stokes equations are solved subject to the Boussinesq approximation as given below:

$$\text{continuity,} \quad \nabla \cdot \vec{u} = 0; \tag{1}$$

$$\text{momentum,} \quad \frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} = \nu \nabla^2 \vec{u} - \frac{\nabla p}{\rho_0} - \beta (T - T_0) g \vec{j}; \tag{2}$$

$$\text{energy,} \quad \frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T = \alpha_l \nabla^2 T. \tag{3}$$

If necessary, the Boussinesq approximation can be removed and the full Navier–Stokes equations solved in the melt. In the solid phase

$$\vec{u} = 0 \tag{4}$$

and the energy equation reads

$$\frac{\partial T}{\partial t} = \alpha_s \nabla^2 T. \tag{5}$$

In the above equations, $\vec{u}$ is the velocity vector, p, T and ρ are the pressure, temperature and density respectively, α, ν and β are the thermal diffusivity, kinematic viscosity and thermal expansion coefficient respectively, subscript 'o' implies a reference value and 'l' and 's' represent liquid and solid phases respectively.

The field equations are solved subject to the following boundary condition on the moving interface:[18]

$$T_{\text{interface}} = T_{\text{m}}\left(1 - \frac{\gamma}{L}\kappa\right) \quad \text{(Gibbs–Thomson condition)}, \tag{6}$$

where T_{m} is the melting temperature, γ is the surface tension, L is the latent heat of fusion and κ is the local curvature of the interface. The fluid flow velocity boundary conditions are

$$\vec{u} \cdot \vec{t} = 0 \quad \text{(no slip)}, \tag{7}$$

$$\rho_{\text{L}} \vec{u} \cdot \vec{n} = (\rho_{\text{s}} - \rho_{\text{L}}) V_{\text{N}} \quad \text{(mass conservation)}, \tag{8}$$

where $\vec{t}$ and $\vec{n}$ are the tangent and normal respectively and V_{N} is the normal velocity of the interface, given by

$$\rho_{\text{s}} L V_{\text{N}} = k_{\text{s}} \frac{\partial T_{\text{s}}}{\partial n} - k_{\text{l}} \frac{\partial T_{\text{l}}}{\partial n} \quad \text{(Stefan condition)}. \tag{9}$$

Appropriate Neumann and Dirichlet conditions are applied at the boundaries of the computational domain. Boundary conditions for the equations in *each phase* are applied on this boundary, which requires that the curvature and slopes of the interface be obtained accurately. Equation (9) is a statement of heat conservation across the phase discontinuity and inaccuracies in computing heat fluxes can lead to erroneous values of interface velocities. An efficient field equation solver needs to be developed which incorporates the information regarding interface shape and location. The boundary conditions are to be applied at the *exact location of the interface*. Conventional Lagrangian approaches employing moving boundary-conforming grids fail when the interface becomes highly contorted and suffers topological changes. The Eulerian methods solve for fluid fraction variables on stationary grids and reconstruct the interface based on the fluid fraction data in each cell, a process that involves several logical operations. In order for the interfaces obtained to be accurate, highly refined grids may be required. The computational procedure developed seeks to surmount the difficulties associated with conventional approaches by combining features of both methods. With this in view, Udaykumar and Shyy[22] developed an interface tracking procedure that can handle highly distorted fronts and their interactions. In Reference 22 a systematic account is given to address the various aspects involved in the marker advancing scheme, including the orientation of the cut cell, topological changes of the interface and assessment of the interface shape and curvature evaluations. In this work we develop a complete algorithm called ELAFINT (Eulerian–Lagrangian Algorithm for INterface Tracking) capable of solving fluid flow problems bounded by complex configurations and moving boundaries, arising in phenomena such as materials solidification.

To solve the field equations, a *fixed Cartesian grid* is used to perform the computations. The irregularly shaped interface is tracked over this grid and represented using *marker particles*. The markers are connected by piecewise circular arcs to obtain interfacial shape information such as normals and curvatures. The interface is advanced in time by the Lagrangian translation of the marker particles, the normal velocity being computed from equation (9). The new interfacial shape is obtained by joining these updated markers. The use of an Eulerian grid facilitates the execution of merger/break-up procedures at the interface, a drawback conventionally attached to pure Lagrangian methods. The Lagrangian component of the present method allows the interface to be accurately captured, which is a significant improvement over the conventional Eulerian methods. In conjunction with

the interface tracking procedure a pressure-based control volume formulation is designed to solve the field equations. Since the interface cuts through the grid, fragmented control volumes may arise in the cells containing the interface. The present algorithm uses integration formulae consistent with the underlying discretization scheme to handle such cases. The basic features of the methodology presented here were developed in the case of pure conduction in previous work by Udaykumar and Shyy.[11] The ability to handle highly distorted interfaces and to follow the interfacial evolution accurately over long time durations was demonstrated in that work.

In the following we describe for a general conserved variable ϕ the application of a control volume formulation and the discretization scheme for a two-dimensional geometry. In Section 2 we clarify the treatment of the individual terms for each of the flow variables, explain the various issues involved and address the conservation and consistency aspects in the computational notation of the pressure-based methodology. We then present some test cases and comparisons to validate the solution procedure.

2. THE CONTROL VOLUME FORMULATION FOR A TRANSPORT VARIABLE ϕ

2.1. Discretization

Consider the conservation law for the variable ϕ, defined to be (a) unity for the continuity equation, (b) u and v for the x- and y-momentum equations respectively and (c) specific enthalpy for the energy equation. In the general case we have

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho\vec{u}\phi) = (\nabla \cdot \Gamma\nabla\phi) + S. \tag{10}$$

Consider the interface that passes through a Cartesian grid at a given instant. The control volumes in the vicinity of the interface are irregularly shaped and in the most general case can assume a five-sided shape as shown in Figure 1.

In order to evaluate the momentum fluxes through the cell faces, we integrate equation (10) over the control volume[23] and employ the divergence theorem to get

$$\int_V \frac{\partial(\rho\phi)}{\partial t}\,\mathrm{d}V + \int_A (\rho\vec{u}\phi)\cdot\vec{n}\,\mathrm{d}A = \int_A (\Gamma\nabla\phi)\cdot\vec{n}\,\mathrm{d}A + \int_V S\,\mathrm{d}V. \tag{11}$$

In two dimensions we have

$$\int_A \frac{\partial(\rho\phi)}{\partial t}\,\mathrm{d}A + \oint_l (\rho\vec{u}\phi)\cdot\vec{n}\,\mathrm{d}l = \oint_l (\Gamma\nabla\phi)\cdot\vec{n}\,\mathrm{d}l + \int_A S\,\mathrm{d}A, \tag{12}$$

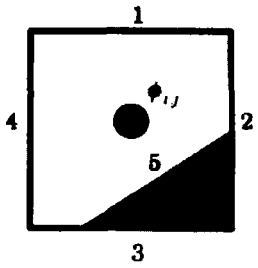


Figure 1. Typical control volume encountered in mixed Eulerian–Lagrangian method

where the second term on the left-hand side is now the line integral of the outward normal convective fluxes and the first term on the right-hand side is the line integral of the outward normal diffusion fluxes through the faces of the control volume. We now proceed to discretize each of the terms in equation (12) for the control volume shown. Thus

$$\int_A \frac{\partial(\rho\phi)}{\partial t} \mathrm{d}A = \frac{\rho_{i,j}^{n+1}\phi_{i,j}^{n+1} - \rho_{i,j}^{n}\phi_{i,j}^{n}}{\delta t} A_{\mathrm{cv}}, \tag{13}$$

where superscripts n and $n+1$ indicate the time levels, δt is the time step size and A_{cv} is the area of the irregular control volume. Next

$$\oint_l (\rho\vec{u}\phi)\cdot\vec{n}\,\mathrm{d}l = \sum_{k=1}^{5}(\rho_k u_k \phi_k)^{n+1}\,\mathrm{d}l_k \tag{14}$$

is the summation of the convective normal effluxes through each cell face of the control volume. The superscript $n+1$ indicates the implicit nature of the scheme. The diffusion fluxes are computed as

$$\oint_l (\Gamma\nabla\phi)\cdot\vec{n}\,\mathrm{d}l = \sum_{k=1}^{5}\left[\Gamma_k\left(\frac{\partial\phi}{\partial n}\right)_k\right]^{n+1}\mathrm{d}l_k. \tag{15}$$

Let us denote the source term as

$$\int_A S\,\mathrm{d}A = \bar{S}. \tag{16}$$

Substituting equations (13)–(16) into equation (12), one obtains the discretized form as

$$\frac{\rho_{i,j}^{n+1}\phi_{i,j}^{n+1} - \rho_{i,j}^{n}\phi_{i,j}^{n}}{\delta t} A_{\mathrm{cv}} + \sum_{k=1}^{5}(\rho_k u_k \phi_k)^{n+1}\mathrm{d}l_k = \sum_{k=1}^{5}\left[\Gamma_k\left(\frac{\partial\phi}{\partial n}\right)_k\right]^{n+1}\mathrm{d}l_k + \bar{S}. \tag{17}$$

In particular the discrete form of the continuity equation can be written as

$$\frac{\rho_{i,j}^{n+1} - \rho_{i,j}^{n}}{\delta t} A_{\mathrm{cv}} + \sum_{k=1}^{5}(\rho u_n)_k\,\mathrm{d}l_k = 0. \tag{18}$$

Now multiplying equation (18) by the value $\phi_{i,j}$ and subtracting from equation (17) gives

$$\frac{\rho_{i,j}^{n}(\phi_{i,j}^{n+1} - \phi_{i,j}^{n})}{\delta t} A_{\mathrm{cv}} + \sum_{k=1}^{5}\rho_k u_k(\phi_k - \phi_{i,j})^{n+1}\mathrm{d}l_k = \sum_{k=1}^{5}\left[\Gamma_k\left(\frac{\partial\phi}{\partial n}\right)_k\right]^{n+1}\mathrm{d}l_k + \bar{S}. \tag{19}$$

2.2. The staggered grid

The solution of the discrete form, namely equation (19), is carried out on the staggered grid arrangement shown in Figure 2. Staggered grids have been extensively adopted for computation of incompressible fluid flows owing to their many advantages.[24,25] The variables and their respective control points are located as indicated in Figure 2. The interface is thus required to be tracked on such a grid arrangement. The definition of the control volumes, which is carried out according to an intersection procedure,[11] is now applicable to each type of control volume shown, namely the u, v and (p, T) control volumes. For a given control volume then, the interface tracking procedure provides the information shown in Figure 3. Thus, at each iteration, explicit definition of the interface location/control volumes for each variable and its grid is available. This facilitates the application of boundary conditions on the faces of the cells containing the interface, as explained later.

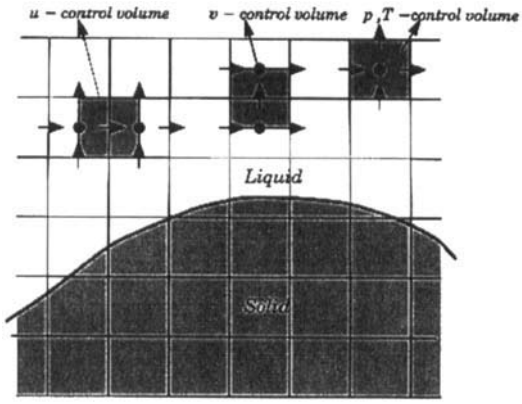


Figure 2. Control volumes for u, v, p and T variables showing staggered arrangement

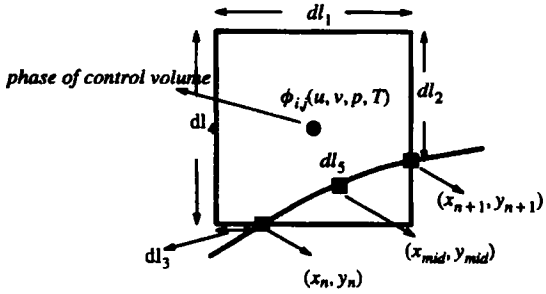


Figure 3. Illustration of information provided by interface tracking module during assembly of coefficients for control volumes

2.3. *Computation of the pressure field*

For incompressible flows, since no explicit equation exists for the pressure field, some means needs to be devised to compute the pressure field. To extract the pressure, one makes use of the continuity equation and obtains a correction equation for the pressure and velocities that enforces mass conservation. The correction procedure is continued until convergence is achieved for each time step. Such a pressure correction method has been widely used and is described in detail elsewhere.[24,25] The essential feature is that the following equation results for the pressure correction at each point, where the right-hand side represents the mass deficit in the control cell and is required to be nullified at convergence:

$$a_P p'_P - a_N p'_N - a_S p'_S - a_E p'_E - a_W p'_W = \frac{(\rho_P^0 - \rho_P)A_{cv}}{\delta t} + \rho_w u_w \, dl_w - \rho_e u_e \, dl_e + \rho_s v_s \, dl_s - \rho_n v_n \, dl_n + (\rho_I u_{nI}) dl_I. \quad (20)$$

The individual terms on the right-hand side are mass fluxes through the faces of the control volume, including the interfacial segments.

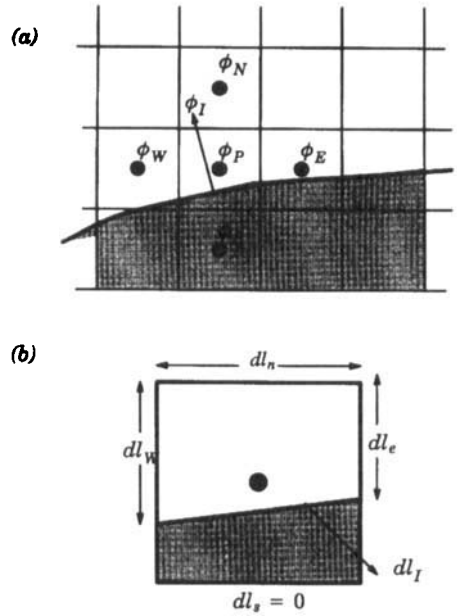


Figure 4. Illustration of nomenclature: (a) control points; (b) control volume sides

2.4. Computation of convective and diffusive fluxes and source terms

Returning now to equation (19) for the situation shown in Figure 4, for the variable ϕ the convective fluxes are evaluated as

$$\sum_{k=1}^{5} (\rho u_n)_k(\phi_k - \rho_P)\,\mathrm{d}l_k = (\rho_e u_e)(\phi_e - \rho_P)\,\mathrm{d}l_e - (\rho_w u_w)(\phi_w - \phi_P)\,\mathrm{d}l_w + (\rho_n u_n)(\phi_n - \phi_P)\,\mathrm{d}l_n - (\rho_s u_s)(\phi_s - \phi_P)\,\mathrm{d}l_s \pm (\rho_I u_{nI})(\phi_I - \phi_P)\,\mathrm{d}l_I, \tag{21}$$

which may be written as

$$\sum_{k=1}^{5}(\rho u_n)_k(\phi_k - \phi_P)\,\mathrm{d}l_k = F_e(\phi_e - \phi_P) - F_w(\phi_w - \phi_P) + F_n(\phi_n - \phi_P) - F_s(\phi_s - \phi_P) \pm \rho_I u_{nI}(\phi_I - \phi_P)\,\mathrm{d}l_I. \tag{22}$$

Here the Fs stand for the mass fluxes through the faces of the control volume. The lowercase subscripts indicate the values at the cell faces. The manner in which these fluxes are evaluated, i.e. the specific shape function assumed for the variable ϕ for evaluation of these fluxes, determines the order of accuracy of the scheme employed. In equation (22) the sign of the last term is to be decided. What is required is the outward normal flux from the interfacial segment. We shall return to this point momentarily.

Now the diffusion fluxes can be written as

$$\sum_{k=1}^{5} \Gamma_k\left(\frac{\partial\phi}{\partial n}\right)_k \mathrm{d}l_k = \Gamma_e\left(\frac{\partial\phi}{\partial x}\right)_e \mathrm{d}l_e - \Gamma_w\left(\frac{\partial\phi}{\partial x}\right)_w \mathrm{d}l_w + \Gamma_n\left(\frac{\partial\phi}{\partial y}\right)_n \mathrm{d}l_n - \Gamma_s\left(\frac{\partial\phi}{\partial y}\right)_s \mathrm{d}l_s \pm \Gamma_I\left(\frac{\partial\phi}{\partial y}\right)_I \mathrm{d}l_I. \tag{23}$$

Substituting the expressions provided by (22) and (23) into equation (19) and simplifying, we now obtain

$$\frac{\rho_P^o(\phi_P - \phi_P^o)A_{cv}}{\delta t} + (J_e - F_e\phi_P) - (J_w - F_w\phi_P) + (J_n - F_n\phi_P) - (J_s - F_s\phi_P) = \pm \Gamma_I\left(\frac{\partial\phi}{\partial n}\right)_I dl_I \mp \rho_I u_{nI} dl_I(\phi_I - \phi_P) + \bar{S}, \quad (24)$$

where

$$J_e = \rho_e u_e \, dl_e \phi_e - \Gamma_e\left(\frac{\partial\phi}{\partial x}\right)_e dl_e, \quad (25)$$

$$J_n = \rho_n v_n dl_n \phi_n - \Gamma_n\left(\frac{\partial\phi}{\partial y}\right)_n dl_n, \quad (26)$$

with similar expressions for J_w and J_s. The following notation is standard for the SIMPLE algorithm:[24]

$$J_e - F_e\phi_P = a_E(\phi_P - \phi_E), \quad (27)$$

$$J_w - F_w\phi_P = a_W(\phi_W - \phi_P), \quad (28)$$

with similar expressions for a_N and a_S. The forms assumed by $a_{E,W,N,S}$ determine the order of accuracy of the differencing scheme employed. For example,

$$a_E = D_E A(|P_e|) + [-F_e, 0], \quad (29)$$

where $P_e = F_e/D_e$ is the cell Peclet number and the square brackets imply the maximum of the two quantities. In this work the second-order central difference scheme is employed to discretize spatial derivatives, for which[24]

$$A(|P|) = 1 - 0{\cdot}5|P|. \quad (30)$$

Equation (24) can now be written as

$$\frac{\rho_P^o(\phi_P - \phi_P^o)A_{cv}}{\delta t} + a_E(\phi_P - \phi_E) - a_W(\phi_W - \phi_P) + a_N(\phi_P - \phi_N) - a_S(\phi_S - \phi_P) = \pm \Gamma_I\left(\frac{\partial\phi}{\partial n}\right)_I dl_I \mp \rho_I u_{nI} \, dl_I(\phi_I - \phi_P) + \bar{S}. \quad (31)$$

Letting

$$\frac{\rho_P^o A_{cv}}{\delta t} = a_P^o \quad (32)$$

and

$$a_P = a_P^o + a_E + a_W + a_N + a_S, \quad (33)$$

we have the final discretized form

$$a_P\phi_P = a_N\phi_N + a_S\phi_S + a_E\phi_E + a_W\phi_W + b, \quad (34)$$

where

$$b = \bar{S} \pm \Gamma_I\left(\frac{\partial\phi}{\partial n}\right)_I dl_I \mp \rho_I u_{nI} \, dl_I(\phi_I - \phi_P). \quad (35)$$

In the above discretization the value of ϕ at any point P is dependent on its four immediate neighbours. When cast in matrix form, the equation reads

$$[C][\Phi] = [B], \tag{36}$$

where $[C]$ is a pentadiagonal coefficient matrix, Φ is the solution vector and B is a source vector. The procedure usually adopted is to solve the system of equations in iterative fashion employing a line SOR method, which calls for a tridiagonal matrix solution procedure.

2.5. *Computation of interfacial fluxes*

We now proceed to detail the method for obtaining the interfacial flux terms. Considering the fluxes

$$\pm \Gamma_I \left(\frac{\partial \phi}{\partial n}\right)_I \mp \rho_I u_{nI}\, dl_I(\phi_I - \phi_P) \tag{37}$$

from the interface, we need to estimate the quantities

$$\left(\frac{\partial \phi}{\partial n}\right)_I, \quad \rho_I u_{nI}, \quad \phi_I \tag{38}$$

at the interface and determine the missing signs to evaluate the flux terms. As for the above values at the interface, ρ_I is easily obtained for incompressible flow as the value of the density at the liquid control point closest to the interface, while u_{nI}, the fluid velocity normal to the interface, is evaluated from the boundary condition at the interface, equation (9), once the interfacial velocity at that point is known. The interfacial velocity is of course determined from the Stefan condition during the course of the calculation and is coupled to the temperature gradients in the vicinity of the interface. The normal gradient of any variable ϕ, i.e. $(\partial\phi/\partial n)_I$, is evaluated in the same fashion as for the pure conduction problem. A probe is inserted in each phase and a biquadratic shape function is described in the vicinity of the interface. The derivative can then be estimated. The value of the variable at the interface, ϕ_I, is again obtained from the boundary conditions. In particular:

(i) the u-momentum equation, $\phi = u$ and $u_I = u_{nI} n_x$ as shown in Figure 5;
(ii) the v-momentum equation, $\phi = v$ and $v_I = u_{nI} n_y$ as shown in Figure 5;
(iii) for the energy equation, $\phi = T$ and T_I is obtained from the Gibbs–Thomson condition at the interface. Next we need to decide upon the signs to be assigned to each of the interfacial fluxes. The diffusion flux with the as yet undetermined sign is

$$\pm \Gamma_I \left(\frac{\partial \phi}{\partial n}\right)_I dl_I. \tag{39}$$

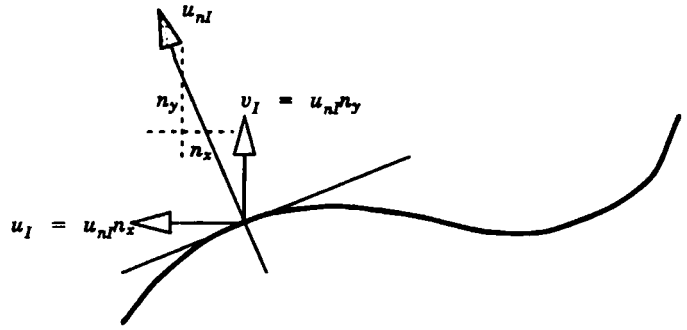


Figure 5. Illustration of velocity components of fluid at interface

The positive sign is applicable when an outflux is evaluated. By the procedure outlined above, the gradient evaluated is

$$\frac{\partial \phi}{\partial n} = \frac{\phi_{I+dn} - \phi_I}{dn} \tag{40}$$

for the situation shown in Figure 6. The gradient as evaluated above represents the influx into the control volume in each phase. A negative sign is therefore required to provide the efflux from the control volume. As for the sign of the convective interfacial flux, we have adopted the convention that the normal to the interface points into the liquid phase. Hence u_{nI} is the normal velocity of the fluid at the interface and is directed into the liquid control volume. Therefore the convective flux into the control volume is given by

$$\rho_I u_{nI}\, dl_I(\phi_I - \phi_P). \tag{41}$$

To obtain an efflux from the control volume, a negative sign should appear in front of this flux term. Thus the final form of the source term b is given by

$$b = -\Gamma_I\, dl_I\left(\frac{\partial \phi}{\partial n}\right)_I + \rho_I u_{nI}\, dl_I(\phi_I - \phi_P) + \bar{S}. \tag{42}$$

2.6. *Evaluation of the source term*

For each of the equations, namely the continuity and momentum equations, the source term $\bar{S}$ assumes a different form. The pressure terms are included in this source term and its evaluation involves certain considerations which we now detail. For example, for the u-momentum equation we have

$$\bar{S}_u = \int_A \left(-\frac{\partial p}{\partial x}\right) dA \tag{43}$$

and for the v-momentum equation

$$\bar{S}_v = \int_A \left(-\frac{\partial p}{\partial y}\right) dA. \tag{44}$$

Consider the term corresponding to the u-momentum equation as in (43) in relation to the u-control volume shown in Figure 7(a). Applying the divergence theorem, equation (43) can be written as

$$\bar{S}_u = -\sum_{k=1}^{5} p_i (dl_y)_k, \tag{45}$$

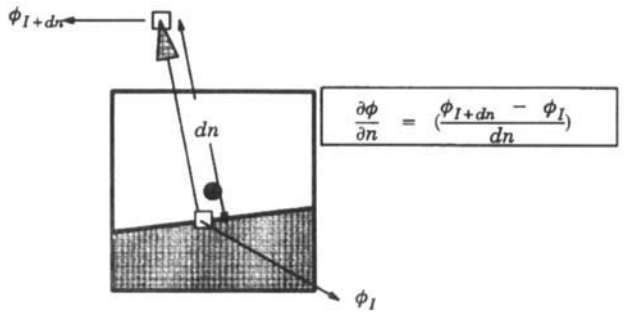


Figure 6. Interfacial gradient of a field variable obtained by normal projection procedure

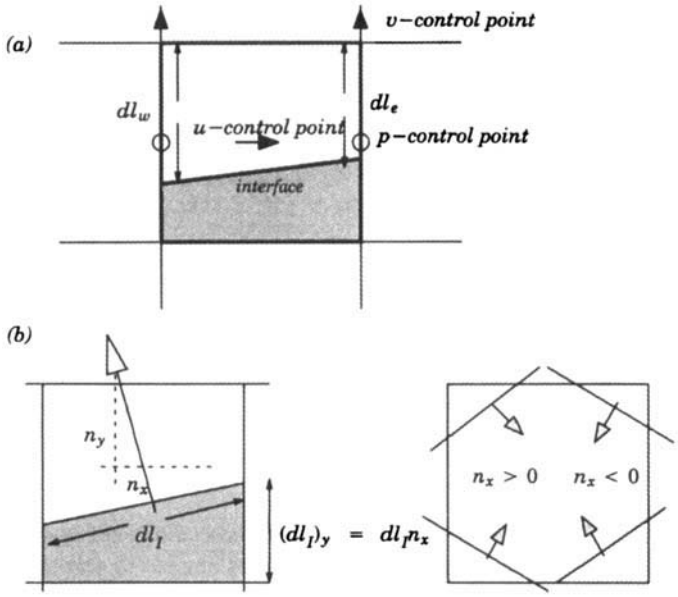


Figure 7. Schematic and geometric representation of cut cell: (a) typical control volume with interfacial segment; (b) directions of normals and projected lengths for typical control volume (n_x and n_y are components of unit normal in directions x and y respectively)

where the length dl_y is the projected length of a side along the y-direction. Thus, for the control volume shown, the pressure contributions take the form

$$\bar{S}_u = -p_e\, dl_e + p_w\, dl_w \pm p_l (dl_l)_y. \tag{46}$$

Now, as shown in Figure 7(b), $(dl_l)_y = dl_l n_x$, where n_x is the x-component of the unit normal vector to the interface in that cell and is already available from the interface tracking information. From observing Figure 7(b), it is evident that for $n_x < 0$ a negative sign is required in front of the pressure term for the interfacial segment and for $n_x > 0$ a positive sign is required. Thus the appropriate form for the source term is now

$$\bar{S}_u = -p_e\, dl_e + p_w\, dl_w + p_l\, dl_l n_x. \tag{47}$$

To estimate the value of the pressure at the interface, p_l, a bilinear extrapolation is performed from the neighbouring liquid phase pressure control points onto the interface location.

2.7. The continuity equation: pressure correction equation

The pressure correction equation is of the form shown in (20). The right-hand side contains mass fluxes through each side of the control volume. Such terms are easily evaluated, since the velocity values required in computing the mass fluxes are available at the cell faces, a feature of the staggered grid. The coefficients for the pressure correction equation are assembled in the standard way.

2.8. Dealing with cut cells

When the interface passes through the staggered grid arrangement as shown in Figure 2, the control volumes in the vicinity of the interface become fragmented. Consider the control volume for the u-velocity, $u(i, j)$, as shown in Figure 8(a). The u-control point lies in the liquid phase here, while the

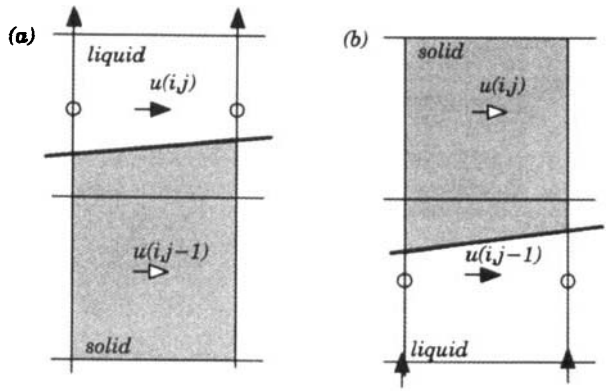


Figure 8. Definition of cut cells and their partners

portion of the control volume shown shaded lies in the solid phase. However, in Figure 8(b) the shaded region belongs to cell $(i, j-1)$ but lies in the solid phase. Thus one has to account for the existence of such fragments of control volumes and assign them to the proper phase. This is done by defining partner cells for each type of cell, a procedure discussed in more detail in Reference 11. In Figure 8(a), for example, cell (i, j) is now redefined to be the unshaded irregular cell. Similarly, in Figure 8(b) the new u-cell (i, j) is defined by the shaded region. This procedure of redefining the cut cells and their partners is accomplished by running through the array of interfacial cells. By assigning partner cell fluxes and redefining dimensions, it is possible to maintain consistency at cell faces and conservation of fluxes.

2.9. *Conservation and consistency at cell faces*

In solving the set of conservation laws, it is important to set up the control volume formulation and differencing schemes so that strict conservation of fluxes is maintained. Consider the situation shown in Figure 9. Of concern to us are the flux evaluations through the faces of cells (i, j) and $(i \pm 1, j)$ and their relation to the partner cell definitions. The newly redefined cells after partner cell assignment are shown hatched. The fluxes through the west face of (i, j) are now magnified by the factor $\{\mathrm{d}l_w(i, j) + [\mathrm{d}y - \mathrm{d}l_w(i, j-1)]\}/\mathrm{d}l_w(i, j)$ to correct for the augmented length of the west face. A similar modification is made for cell $(i-1, j)$ when its partner cell $(i-1, j-1)$ is being dealt with. Similarly, in cell $(i, j-1)$ the pressure force on the west side is magnified by the factor above. Fluxes

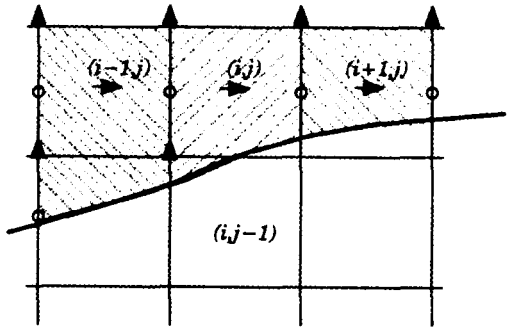


Figure 9. Illustration of partner cell definitions

from the interface computed in cell $(i, j-1)$ are assigned to cell (i, j) and from cell $(i-1, j-1)$ to cell $(i-1, j)$. Thus the partner assignment procedure and the cut cell coefficient assembly algorithms achieve explicit flux conservation in the cells affected by the interface.

2.10. Anomalous cases

However, some anomalous situations may be encountered in the coefficient assembly process. For instance, in Figure 10 the u-control point (i, j) is adjoined by a solid phase control point. Therefore the flux through the east face is not straightforward to compute. The way this is done to maintain consistency at that face is by noting that cell $(i+1, j+1)$ has been redefined by the partner cell assignment procedure to be of the shape shown. Thus a part of the west face of cell $(i+1, j+1)$ is now adjacent to the east face of cell (i, j). Therefore the flux through the west face of cell $(i+1, j+1)$ has to be consistent with the fluxes through the east faces of (i, j) and $(i, j+1)$. The flux through the east face of cell (i, j) is now obtained by redistributing the west face flux of cell $(i+1, j+1)$, weighted according to the dimensions $dl_e(i, j)$ and $dl_e(i, j+1)$. In addition, if the east face pressure control point of cell (i, j) is also in the solid phase, the pressure contribution from cell $(i+1, j+1)$ is redistributed consistently between cells $(i, j+1)$ and (i, j). Such anomalous cases exist for all eight types of cells and for all control volumes. Treating such cases adds to the tedium of assembly of the coefficients but is critical in obtaining solutions to flow problems. If consistency at cell faces is violated, convergence cannot be achieved owing to the existence of spurious sources/sinks of mass, momentum and energy at such locations.

2.11. Distinction between liquid and solid cells

In the cell assembly procedure one has to be careful not to step across the interface into the opposite phase in performing the flux computations. The advantage in combining the Eulerian (field solver) and Lagrangian (interface tracking) methods here is that the two phases can be treated separately. Thus, unlike in the Eulerian methods, the interface separating the two phases can be explicitly defined and treated as a discontinuity. Also, in contrast with the Lagrangian methods, grid redistribution for conformity with the moving, distorting boundary is avoided. Thus, in each phase, the operations required to obtain the flux estimates should involve points in the same phase and the interfacial values only. The treatment of the liquid and solid phase control points follows the same procedure as far as the coefficient assembly is concerned. However, the solid is passive as far as fluid flow is concerned, unlike in the purely Eulerian methods, where it is assumed to have some porosity in the proximity of the interface to ease the sudden property jump across it. In our case, once the coefficients are assembled, a

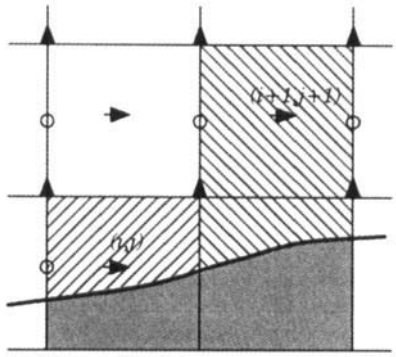


Figure 10. Illustration of an anomalous case in defining fluxes

flag denoting solid is employed to turn off the fluid flow computations in the solid phase, such that $a_P = 1$ and $a_E, a_W, a_N, a_S, b = 0$ in the solid, which results in $u = v = p' = 0$ in the solid phase. This of course is not applicable for the energy equation, where heat conduction in the solid is to be accounted for.

Once the solid and liquid cell coefficients are assembled, the solution procedure does not distinguish between the two phases. The ADI method along with the tridiagonal matrix solution is employed for the entire domain of calculation. The difference scheme used in this work is the $O(\delta x^2)$ central difference scheme for all the cell face fluxes except the interface flux terms. The interface flux estimation is performed based on the two-point difference described in Section 2.5. By using a three-point scheme, the accuracy of the interface fluxes can be enhanced.

2.12. Moving boundary problems—treatment of cells that change phase

When the interface moves, e.g. owing to a change in phase, the shape of the interface at two time instants (or iterations) may develop as shown in Figure 11. Thus cells (i, j), $(i+1, j)$, etc. have moved from the solid phase into the liquid. In the implicit solution procedure adopted here, information regarding the previous state of a control point is required to evaluate the time derivative, i.e. the quantity $\phi^n_{i,j}$ corresponding to the new phase is non-existent for that control point. To overcome this difficulty, flags are employed indicating the current and previous phases of the control point. When a control point changes phase, the value at that location corresponding to the phase in which it finds itself is estimated by performing a linear interpolation as shown in Figure 12. The new value in a cell which undergoes a phase change is thus obtained by interpolation from neighbouring cells in the same phase as

$$\phi^n_{i,j} = (\mathrm{d}y_I \phi^n_{i,j+1} + \mathrm{d}y_N \phi^n_I)/(\mathrm{d}y_I + \mathrm{d}y_N). \tag{48}$$

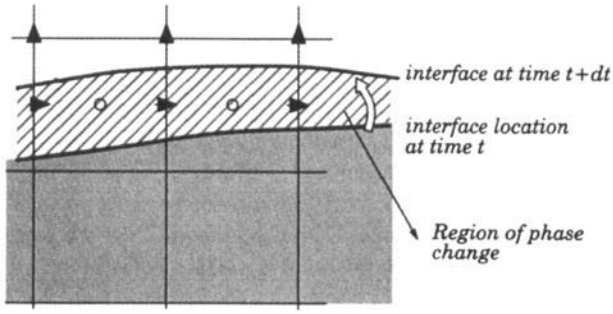


Figure 11. Change in phase of cells as interface traverses domain

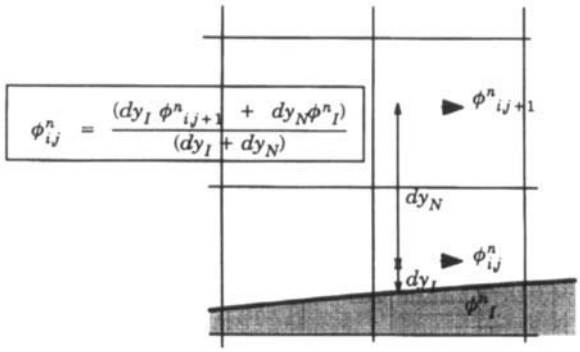
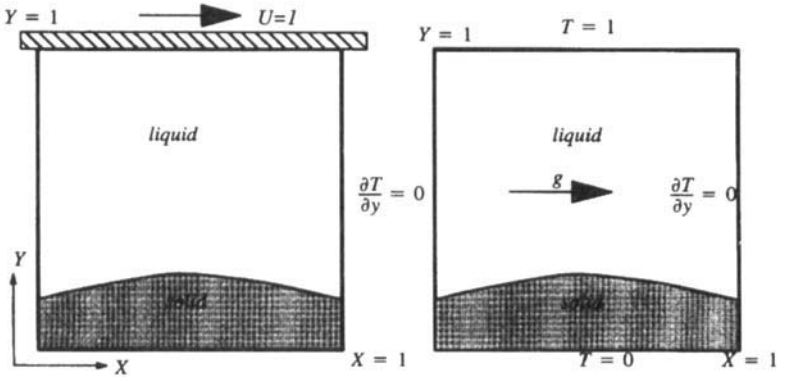


Figure 12. Illustration of interpolation procedure for estimation of field quantities in cells that have changed phase



(a) Driven cavity flow *(b) Natural convection*

Figure 13. Illustration of computational domain and boundary conditions for test cases presented. For both configurations, no-slip conditions are applied on all solid surfaces. A Cartesian grid is used, with the interface passing through the grid. The interface amplitude was equal to 0·1, which is 10% of domain size

In particular, the value $\phi^n_{i,j}$ is then obtained via the implicit solution procedure by iteration. In our computations, flipping of a control point between two phases was practically non-existent. This problem exists in purely Eulerian methods. In our case, since the interface is a continuous entity, each point on the interface is influenced by the motion of all its neighbouring along the interfacial curve and this global influence helps in avoiding phase flipping. In contrast, interface reconstruction procedures rely on local fluid fraction information in a cell to define the interface and thus it is possible for a cell to flip between phases in the course of the iterations.

3. RESULTS AND DISCUSSION

We first compare solutions for a stationary interface with those from well-tested solution procedures using body-fitted co-ordinates.[25] This exercise is designed to validate the conservation and consistency characteristics of the cut cell method. The interface is sufficiently deformed that all types of cut cells are encountered. In the absence of a consistent discretization in the vicinity of the interface it is found that convergence cannot be achieved and that care needs to be exercised in performing flux calculations in the grid cells affected by the interface. The computational domain is as shown in Figure 13(a). The square cavity is a frequently adopted test bed for numerical experiments on incompressible flows and benchmarks exist. We deform the base of the cavity, the amplitude being 10% of the base. A 121×121 Cartesian grid is employed. We first present the results for a driven cavity flow where the top wall of the cavity is pulled at velocity $U = 1$ corresponding to a Reynolds number of 1000. The results from the present method are compared with a previously benchmarked body-fitted formulation with the same grid size. The results are shown in Figure 14. The streamline patterns in Figures 14(a) and 14(b) show good agreement. A quantitative comparison can be seen in Figures 14(c) and 14(d), where the centreline velocities are plotted.

In Figure 15 we compare the results for a stationary interface with natural convection in the cavity shown in Figure 13(b). The same grid size and interface shape as above are used. The Rayleigh number computed is 10^5 and $Pr = 1$. Again the streamline patterns shown in Figures 15(a) and 15(b) are in good agreement. In Figures 15(e)–15(g) we compare the values of velocity components u and v and temperature respectively along the centreline of the cavity. The results agree closely with the body-fitted code. Thus it has been shown that the current scheme yields accurate results for the case of a stationary interface in the presence of complex flow fields. It is noted that the formulation developed here would be

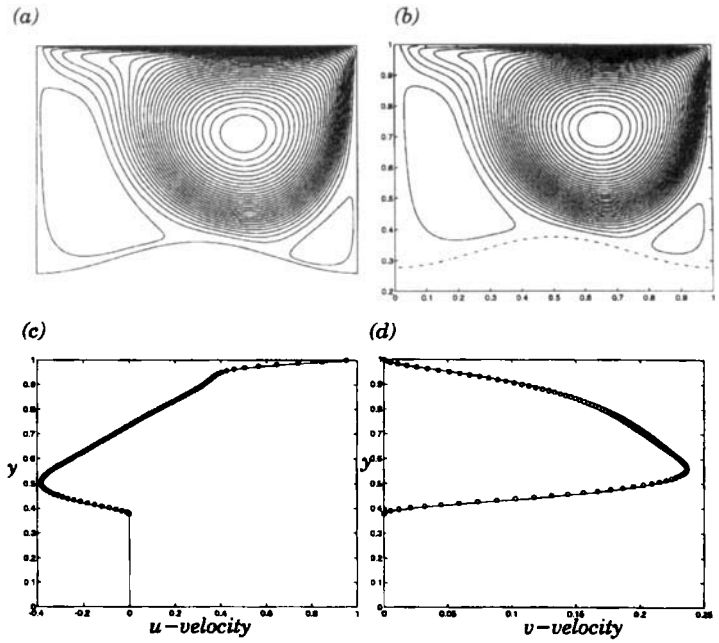


Figure 14. Streamlines for a driven cavity with a deformed base. $Re = 1000$, 121×121 grid. (a) Contours from boundary-fitted grid computation. (b) On a fixed Cartesian grid using current method. Comparison of solutions from boundary-fitted and current Cartesian grid method. (c) Centreline u-velocity. Circles represent solution from boundary-fitted grid; full line is from current calculation. (d) Centreline v-velocity

useful in computing incompressible flows around complex shapes employing fixed Cartesian grids, circumventing the need for generation of boundary-fitted grids.

We now proceed to test the numerical procedure developed here for a situation involving phase change. Hitherto much effort has been devoted to numerically duplicating the results of Gau and Viskanta[26] (hereinafter referred to as G&V) on the melting of gallium from a vertical wall in a rectangular enclosure. Gallium is adopted as the experimental material since it is a metal with a low melting point and thus is easy to handle. Unfortunately, numerical values of interface positions were not presented by G&V and the initial conditions are ambiguous. The authors also present interface shapes viewed from the front and rear of their experimental set-up and poor correspondence is observed. Furthermore, recent experiments of Campbell *et al.*[27] appear to differ from G&V in regard to interface shapes and positions. Underlying these facts is the difficulty in performing experiments in relation to flow fields and interface positions in opaque melts. Despite these limitations, the experiments of G&V have been extensively employed for comparison. Needless to say, the agreement between numerics and experiment is at best modest.[28] A more effective comparison may be between numerical techniques of essentially disparate nature, e.g. purely Lagrangian and Eulerian methods. Lacroix and Voller[29] have performed such a comparison. The grid sizes used by these authors, however, may not be sufficiently fine to resolve all the flow features. In our work we found the presence of multiple convection cells in the initial stages of development of the interface in some cases. It is not conceivable that such cells can be resolved by coarse grids. The presence of such cells is important to capture, in particular because the interface shape reflects the presence of these flow features. Thus two different numerical schemes can yield the same results for the same grid spacing, but neither may actually be an accurate calculation. In fact, the level of numerical dissipation, i.e. the order of accuracy of the numerical technique, was found to determine the types of flow features resolvable, especially at higher Rayleigh numbers. Thus the

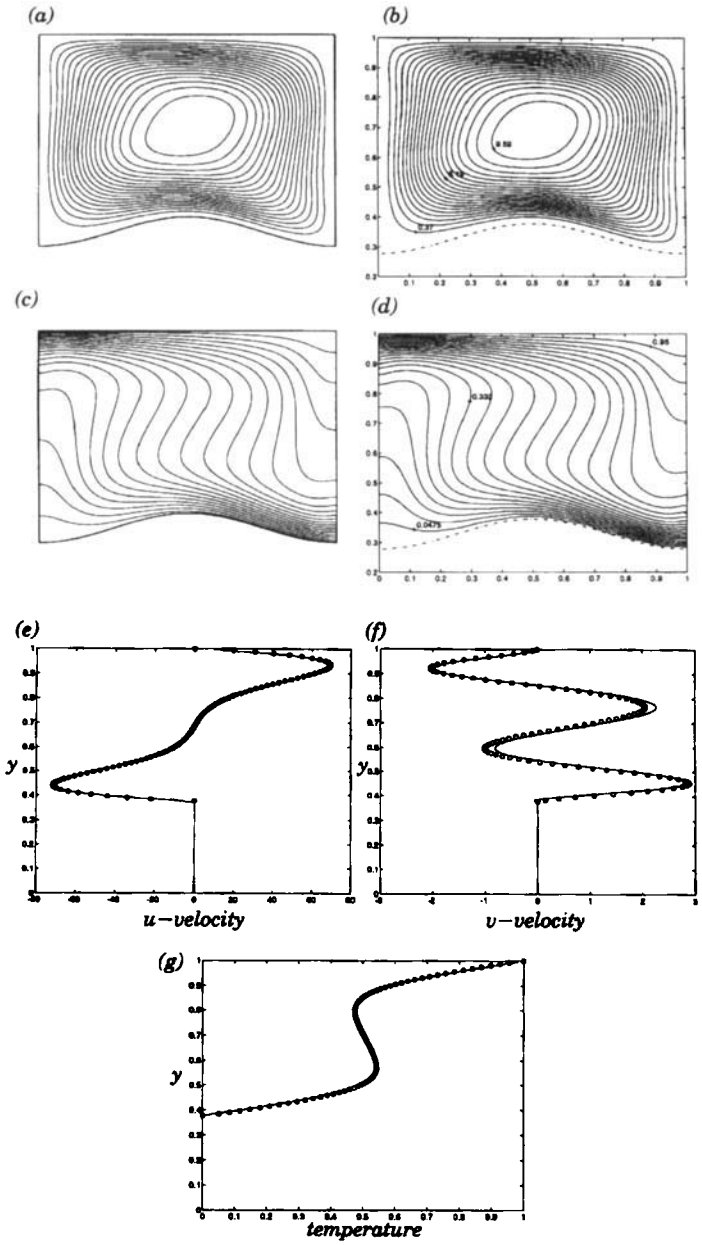


Figure 15. Streamlines for natural convection in a deformed cavity. $Re = 10^5$, $Pr = 1$, 121 × 121 grid. (a) Result from boundary-fitted grid computation. (b) Results from Cartesian grid computation. (c) Isotherm contours from boundary-fitted formulation. (d) Isotherm contours from Cartesian grid computation. (e) Centreline u- velocity. Circles are from boundary-fitted grid computation; full line is from current method. (f) Centreline v-velocity. (g) Centreline temperature

entire situation, especially for the higher Reynolds numbers, is found to be highly sensitive to the numerical conditions employed. Since the interface shape is strongly linked to the flow features and vice versa, great care must be employed in performing the computations.

Here we present calculations for the melting of gallium from a vertical wall for the configuration shown in Figure 16. The current method is compared with an enthalpy-based Eulerian method[30] employing an 81×81 grid. Unless otherwise mentioned, the second-order central difference scheme is used for both methods. In a method dealing with the temperature as a variable and using an interface tracking procedure, it is not possible to initiate melting in a domain that is entirely solid. Thus the computation using the present method is started from an initial condition generated by the enthalpy method, so that a thin initial melt layer exists at the start. The flow field and temperature field are obtained from the purely Eulerian method. Melting is initiated at the left wall. In Figure 17 we show the results for a Rayleigh number of 10^4. The Prandtl number of gallium is 0·021. The Stefan number for this case is 0·042. The interfacial shape and position are compared in this case with those of the enthalpy-based method in Figure 17(a). As can be seen, close agreement is maintained between the predictions of the two methods. The present interface tracking method is marginally slower than the purely Eulerian method for large times. However, it is likely that with further refinement of the grid better agreement could be obtained. It may be noted that the location of the elbow in the phase front is correctly predicted and thus the size of the recirculation zone is obtained accurately. This can be seen from the plots of the streamlines at two different time instants shown in Figures 17(c)–17(f). Furthermore, the centreline velocities and temperature profiles shown in Figure 17(b) are in good agreement. Thus this experiment has proved that the methods employed in treating the interface, in particular the flux computations for the cut cells, the treatment of pressure terms and the procedures involved with the newly formed cells, are borne out by these calculations.

We next present in Figure 18 the results for a higher Stefan number. In this case, as above, $Ra = 10^4$ and $Pr = 0{\cdot}021$, but the Stefan number $St = 0{\cdot}42$, a 10-fold increase. In Figure 18(a) the interface positions are compared, plotted at equal intervals of time $\delta t^* = 10$. As can be seen, the results are very close except for the lagging of the current method for larger times. However, even for the rapidly moving interface the front shapes are well predicted, which implies that the bulk flow features, i.e. recirculation zones, are obtained accurately. In the figures containing streamfunction and isotherm contours, Figures 18(c)–18(h), the interface has been represented in each case by plotting the temperature contours $T = -0{\cdot}005$, 0·0 and 0·005. In the case of the enthalpy method this is the best approximation to the interface shape that one can obtain. In fact, in the comparisons of interface shapes the values corresponding to the enthalpy method are obtained as those corresponding to the $f_s = 0{\cdot}5$ contour by interpolation. Thus there is an uncertainty in the order of the grid spacing in identifying the interface position in the enthalpy method. In contrast, the interface tracking procedure explicitly yields

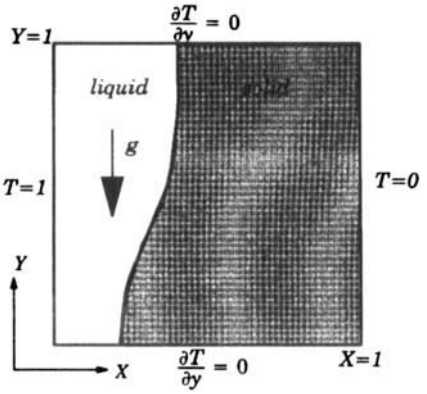


Figure 16. Illustration of computational domain and boundary conditions for melting of gallium. The melting is initiated at the left wall. No-slip velocity conditions are imposed on all solid surfaces. Computations are performed on a Cartesian grid

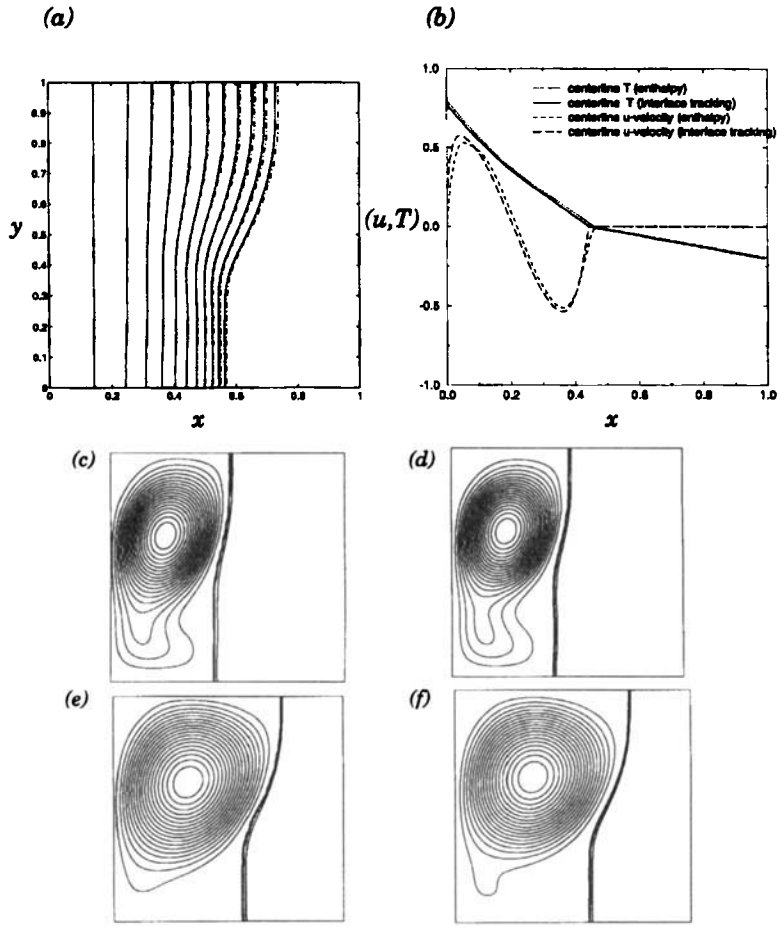


Figure 17. Melting of gallium. $Ra = 10^4$, $Pr = 0{\cdot}021$, $St = 0{\cdot}042$, 81×81 grid. Melting is initiated at the left. (a) Comparison of interface positions at equal intervals of time. Chain lines represent enthalpy-based method; full lines correspond to present method. (b) Comparison of centreline profiles. Streamline contours at $t = 50$ and 100. Comparison of enthalpy and present methods. (c) Current interface tracking method, $t = 50$. (d) Enthalpy method, $t = 50$. (e) Current method, $t = 100$. (f) Enthalpy method, $t = 100$. The interface has been represented by plotting the temperature contours between $T = -0{\cdot}005$ and $0{\cdot}005$. In the case of the current method the interface position is actually available explicitly and exactly. In the enthalpy method the only information regarding the interface is the contours shown

information regarding interface shape. Also, it is noted that in the interface tracking method no modelling of the physics is necessary in the vicinity of the interface, while in the enthalpy method there is an unavoidable smearing of information. In addition, the D'Arcy law treatment and the mushy zone model[31] obscure the transport processes in this region.

The results presented above were applicable to phase change at the macroscopic scale. The interface at such scales is only mildly distorted, as seen in Figures 17(a) and 18(a). However, at the microscopic level the interface can be highly deformed and can assume highly branched forms called dendrites and deep cellular morphologies. The transport mechanisms at that level are dominated by thermal diffusion. However, significant effects of convection have been reported.[32–34] Results for the evolution of diffusion-controlled morphological instabilities are presented in Reference 11. The results for such diffusion-controlled growth demonstrate the expected effects of surface tension at the microscopic level.

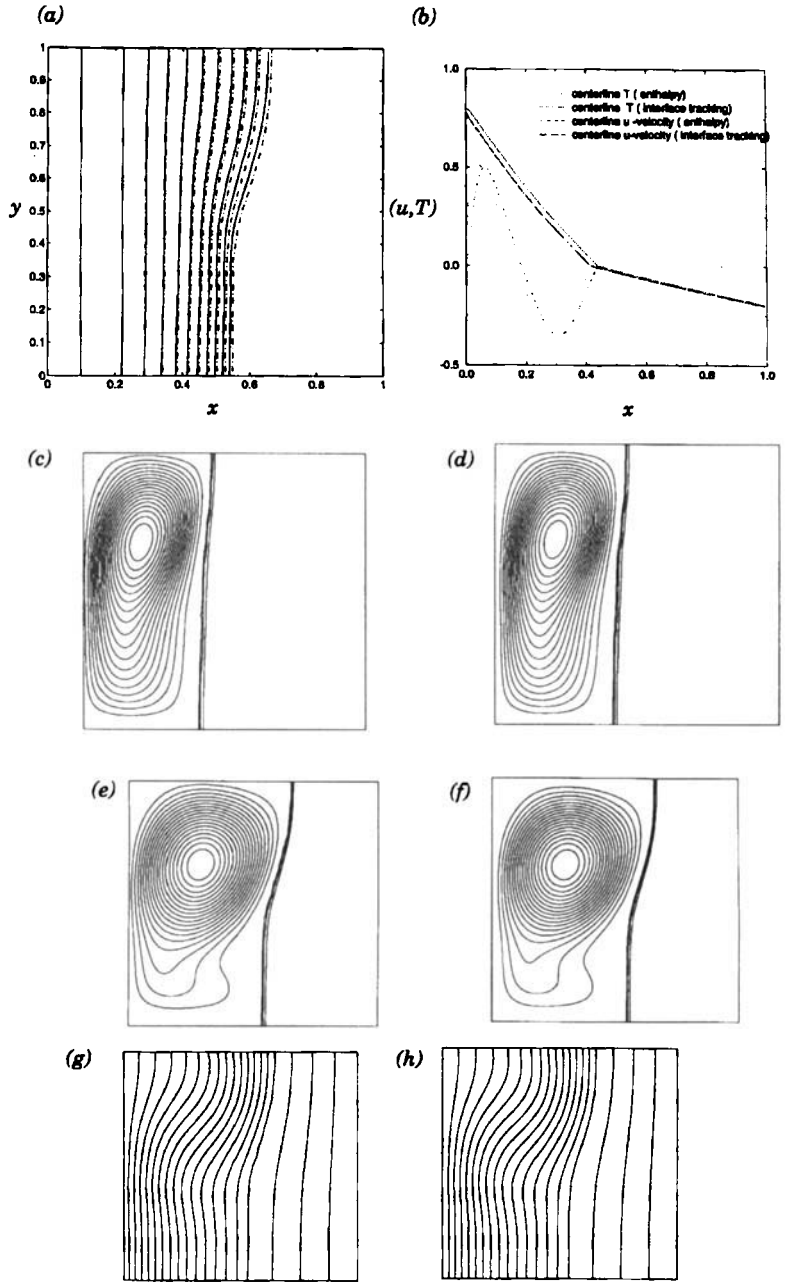


Figure 18. Melting of gallium. Comparison of solutions from enthalpy method and present interface tracking method. Higher-Stefan-number case. $Re = 10^4$, $Pr = 0{\cdot}021$, $St = 0{\cdot}042$, 81×81 grid. (a) Comparison of interface shapes at equal intervals of time. Full lines correspond to present method; chain lines represent enthalpy method. (b) Comparison of centreline values. The legend is self-explanatory. (c) Streamline contours for current method at $t = 50$. (d) Streamline contours for enthalpy method at $t = 100$. In (c) and (d) the interface is represented by plotting the temperature contours $T = -0{\cdot}005$, $0{\cdot}0$ and $0{\cdot}005$. (e) Streamline contours at $t = 100$, interface tracking method. (f) Streamline contours at $t = 100$, enthalpy method. (g) Isotherms at $t = 100$, interface tracking method. (h) Isotherms at $t = 100$, enthalpy method

The present method has been shown to be capable of tracking evolving interfaces for large times and distortions. Computations of microstructure evolution with effects of convection included are in progress.

4. CONCLUSIONS

ELAFINT has been developed to combine the strengths of the purely Eulerian and Lagrangian methods. This has resulted in a method that has the following features.

1. Highly distorted interfaces can be handled by the Lagrangian component, namely the interface tracking scheme. In previous work[22] we have demonstrated the ability of the scheme to capture the interactions, i.e. merger and break-up of different shapes and their subsequent evolution. The methods that were developed to handle diffusion-controlled instability phenomena[11] have here been extended to incorporate convection effects.
2. The computations are performed on a fixed Cartesian grid. Thus the problems of grid redistribution associated with body-fitted adaptive grids is avoided. Also, the procedures employed in the discretization of the equations on such a grid lend themselves to the use of well-tested implicit pressure-based flow solvers. The solutions are achieved by discretizing the equations via a control volume formulation, with particular attention to conservation and consistency in assembling the fluxes of conserved variables. The solution is achieved by using the line SOR method, leading to the iterative solution of tridiagonal matrices.
3. The interface definition is strictly maintained and the boundary conditions applied on the control volumes. The release of latent heat during phase change has been incorporated via the Stefan condition. Several details involved in dealing with the presence of the interface have been presented. Some procedures have been devised to deal with the flux of quantities at the interface.

The results presented here demonstrate the performance of the computational procedure by comparing with previously tested methods. It can also be incorporated to treat many other moving boundary problems, such as those reviewed in Reference 35.

ACKNOWLEDGEMENTS

The work presented here was partially supported by the AFOSR/URI Program and GE Aircraft Engines. Computations were performed on the CRAY-YMP at Eglin Air Force Base, Pensacola, FL.

REFERENCES

1. J. Crank, *Free and Moving Boundary Problems*, Oxford University Press, Oxford, 1984.
2. J. M. Floryan and H. Rasmussen, 'Numerical methods for viscous flows with moving boundaries', *Appl. Mech. Rev.*, **42**, 323–341 (1989).
3. J. F. Thompson, Z. U. A. Warsi and C. W. Mastin, *Numerical Grid Generation*, Elsevier, New York, 1985.
4. W. Shyy, H. S. Udaykumar and S.-J. Liang, 'An interface tracking method applied to morphological evolution during phase change', *Int. J. Heat Mass Transfer*, **36**, 1833–1834 (1993).
5. W. Shyy, 'An adaptive grid method for Navier–Stokes flow computation', *Appl. Math. Comput.*, **21**, 201–219 (1987).
6. C. W. Hirt and B. D. Nichols, 'Volume of fluid (VOF) method for the dynamics of free boundaries', *J. Comput. Phys.*, **39**, 201–225 (1981).
7. J. A. Sethian, 'Numerical algorithms for propagating interfaces: Hamilton–Jacobi equations and conservation laws', *J. Diff. Geom.*, **31**, 131–161 (1990).
8. R. Kobayashi, 'Modelling and numerical simulations of dendritic crystal growth', *Physica D*, **63**, 410–423 (1993).
9. A. A. Wheeler, B. T. Murray and R. J. Schaefer, 'Computations of dendrites using a phase field model', *Physica D*, **66**, 243–262 (1993).
10. P. Y. Liang, 'Numerical method for calculation of surface tension flows in arbitrary grids', *AIAA J.*, **29**, 161–167 (1991).

11. H. S. Udaykumar and W. Shyy, 'Simulation of interfacial instabilities during solidification; Part I: Conduction and capillarity effects', *Numer. Heat Transfer B*, **27**, 127–155 (1994).
12. H. Miyata, 'Finite difference simulation of breaking waves', *J. Comput. Phys.*, **65**, 179–214 (1986).
13. K. Morinishi, 'A finite difference solution of the Euler equations on body-fitted Cartesian grids', *Comput. Fluids*, **21**, 331–344 (1992).
14. J. J. Quirk, 'An alternative to unstructured grids for computing gas dynamic flows around arbitrarily complex two-dimensional bodies', *ICASE Rep. 92-7*, NASA Langley Research Center, Hampton, VA, 1992.
15. D. D. Zeeuw and K. G. Powell, 'An adaptively-refined Cartesian mesh solver for the Euler equations', *AIAA Paper 90-0000*, 1990.
16. S. A. Bayyuk, K. G. Powell and B. van Leer, 'A simulation technique for 2-D unsteady inviscid flows around arbitrarily moving and deforming bodies of arbitrary geometry', *AIAA Paper 93-3391*, 1993.
17. D. B. Kothe and R. C. Mjolsness, 'RIPPLE: a new method for incompressible flows with free surfaces', *AIAA J.*, **30**, 2694–2700 (1992).
18. D. A. Kessler, J. Koplik and H. Levine, 'Pattern selection in fingered growth phenomena', *Adv. Phys.*, **37**, 255–339 (1988).
19. J. S. Langer, 'Instabilities and pattern formation in crystal growth', *Rev. Mod. Phys.*, **52**, 1–56 (1980).
20. M. E. Glicksmann, S. R. Coriell and G. B. McFadden, 'Interaction of flows with the crystal–melt interface', *Ann. Rev. Fluid Mech.*, **18**, 307–336 (1986).
21. S. Ostrach, 'Fluid mechanics in crystal growth', *J. Fluids Eng.*, **105**, 5–20 (1983).
22. H. S. Udaykumar and W. Shyy, 'Development of a grid-supported marker particle scheme for interface tracking', *AIAA Paper 93-3384*, 1993.
23. J. C. T. Wang and G. F. Widhopf, 'A high-resolution TVD finite volume scheme for the Euler equations in conservation form', *J. Comput. Phys.*, **84**, 145–173 (1989).
24. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, Washington, DC, 1980.
25. W. Shyy, *Computational Modelling for Fluid Flow and Interfacial Transport*, Elsevier, Amsterdam, 1994.
26. G. Gau and R. Viskanta, 'Melting and solidification of a pure metal on a vertical wall', *Trans. ASME*, **108**, 174–181 (1986).
27. T. A. Campbell, R. E. Pool and J. N. Koster, 'Melting and solidification of a liquid metal at a vertical wall', *AIAA Paper 94-0792*, 1994.
28. M. Lacroix, 'Computation of heat transfer during melting of a pure substance from an isothermal wall', *Numer. Heat Transfer B*, **15**, 191–210 (1989).
29. M. Lacroix and V. R. Voller, 'Finite difference solutions of solidification phase change problems: transformed versus fixed grids', *Numer. Heat Transfer B*, **17**, 25–41 (1990).
30. W. Shyy and M. M. Rao, 'Enthalpy based formulations for phase change problems with application to g-jitter', *AIAA Paper 93-2831*, 1993; also *Micrograv. Sci. Technol.*, **7**, 41–49 (1994).
31. V. R. Voller and C. Prakash, 'A fixed grid numerical modelling methodology for convection–diffusion mushy region phase change problem', *Int. J. Heat Mass Transfer*, **30**, 1709–1719 (1987).
32. Ph. Bouissou, B. Perrin and P. Tabeling, 'Influence of an external periodic flow on dendritic crystal growth', in F. H. Busse and L. Kramer (eds), *Nonlinear Evolution of Spatio-temporal Structures in Dissipative Dynamical Systems*, Plenum, New York, 1990.
33. T. Sato, W. Kurz and K. Ikawa, 'Experiments on dendrite branch detachment in the succinonitrile–camphor alloy', *Trans. Jpn. Inst. Met.*, **28**, 1012–1021 (1987).
34. S. H. Tirmizi and W. N. Gill, 'Effect of natural convection on growth velocity and morphology of dendritic ice crystals', *J. Cryst. Growth*, **85**, 488–502 (1987).
35. W. Shyy, H. S. Udaykumar, M. M. Rao and R. W. Smith, *Computational Fluid Dynamics with Moving Boundaries*, Hemisphere, Washington, DC, 1995.