INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 21, 1153–1170 (1995)

EXTENSION OF THE VOLUME-OF-FLUID METHOD FOR ANALYSIS OF FREE SURFACE VISCOUS FLOW IN AN IDEAL GAS

JOHN F. McKIBBEN

The Procter & Gamble Company, 7800 Palace Drive, Cincinnati, OH 45249, U.S.A.

AND

CYRUS K. AIDUN*

Institute of Paper Science and Technology, 500 10th Street NW, Atlanta, GA 30318-5794, U.S.A.

SUMMARY

The volume-of-fluid (VOF) method is a simple and robust technique for simulating free surface flows with large deformations and intersecting free surfaces. Earlier implementations used Laplace's formula for the normal stress boundary condition at the interface between the liquid and vapour phases. We have expanded the interfacial boundary conditions to include the viscous component of the normal stress in the liquid phase and, in a limited manner, to allow the pressure in the vapour phase to vary. Included are sample computations that show the accuracy of added third-order-accurate differencing schemes for the convective terms in the Navier–Stokes equation (NSE), the viscous terms in the normal stress at the interface and the solution of potential flow in the vapour phase coupled with the solution of the NSE in the liquid phase. With these modifications we show that the VOF method can accurately predict the instability of a thin viscous sheet flowing through a stagnant vapour phase.

KEY WORDS: thin film stability; volume-of-fluid method; free surface flows

1. INTRODUCTION

Current techniques for computational analysis of free surface flows include primarily Lagrangian and Eulerian approaches. Reviews of free surface techniques can be found elsewhere, particularly in References 1–4. In this paper we present a Eulerian approach with advantages in a broad class of free surface flow problems with large surface deformations. Finally we present results of several test problems which show the accuracy possible with these refinements.

Regardless of the method used to track the location of the interface, the equations to be solved for isothermal, incompressible flow are the continuity equation

$$\nabla \cdot \mathbf{v} = 0 \tag{1}$$

and the Navier–Stokes equation (NSE)

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = \mathbf{g} - \frac{1}{\rho} \nabla P + \nu \nabla^2 \mathbf{v}, \tag{2}$$

where $\mathbf{v}$ is the velocity vector, $\mathbf{g}$ is a body forced vector, P is the pressure, ρ is the density and ν is the kinematic viscosity. These equations are solved subject to boundary conditions at the edges of the

* Author to whom correspondence should be addressed.

CCC 0271–2091/95/121153–18

Received 11 July 1994
Revised 17 March 1995

computational domain, along interior obstacles and at the interface between the liquid and vapour phases.

There is no doubt that the extension of a free surface computational method to a two-phase problem would be trivial if the same equations were applied to both phases and solved along with the full interface conditions.[5] However, when the viscous effects in the gas phase are not important, then it is considerably more cost-effective to solve the inviscid equations for the gas phase. In this study we present the formulation and results for a viscous liquid phase adjacent to an inviscid gas phase. Also, because of the large difference in dynamic viscosity between the phases, the viscous region of the gas phase is very often limited to a thin boundary at the interface. To correctly impose the interfacial shear stress, the boundary layer needs to be numerically resolved at great computational expense. There is a large class of two-phase flow problems where one is only concerned with the pressure variations along the interface and not the viscous boundary layer. For these problems one can either consider the full equations in both phases and neglect the shear stress at the interface (which may be difficult to implement numerically) to avoid the computational expense of having to resolve the gas boundary layer or one can assume that the gas phase flow is irrotational and only consider the inviscid equations for the gas phase along with the full equations for the liquid phase, as we have done in this study. We outline the method, followed by an example problem involving the Kelvin–Helmholtz-type instability of a liquid sheet emanating from a circular jet impinging on a plate.

The interfacial boundary conditions are derived from the velocity and stress balances at the interface and the continuity of velocity. We begin with the definition of the stress balances by defining a localized auxiliary function for the location of the interface,

$$H(x, y) \equiv y - \eta(x) = 0. \tag{3}$$

The unit surface normal vector $\mathbf{n} = (n_x, n_y)$ is computed from the gradient of $H(x, y)$ as

$$n_x = -\eta'(\eta'^2 + 1)^{-1/2}, \qquad n_y = (\eta'^2 + 1)^{-1/2}, \tag{4}$$

where $\eta' = \partial\eta/\partial x$. The unit tangential vector $\mathbf{t} = (t_x, t_y)$ may then be computed from the orthogonality condition as

$$t_x = (\eta'^2 + 1)^{-1/2}, \qquad t_y = \eta'(\eta'^2 + 1)^{-1/2}. \tag{5}$$

Finally, the surface curvature κ is given by

$$\kappa = \eta''(\eta'^2 + 1)^{-3/2}. \tag{6}$$

With these definitions in mind and the assumption of an inviscid vapour phase the boundary conditions at the interface arising from the normal and tangential stress balances

$$P_l - \mathbf{n} \cdot \tau_l \cdot \mathbf{n} = P_v - \sigma\kappa \tag{7}$$

and

$$\mathbf{t} \cdot \tau_l \cdot \mathbf{n} = 0 \tag{8}$$

respectively, where subscripts 'ℓ' and 'v' refer to the liquid and vapour phases respectively, σ is the surface tension and

$$\tau = \mu(\nabla\mathbf{v} + \nabla\mathbf{v}^T) \tag{9}$$

is the stress tensor. The remaining boundary condition at the interface is due to the continuity of normal velocity given by

$$\mathbf{v}_l \cdot \mathbf{n} = \mathbf{v}_v \cdot \mathbf{n}. \tag{10}$$

Since the vapour phase is assumed to be inviscid, continuity of the tangential component of velocity cannot be imposed.

In our work we have chosen to use one of the volume-tracking techniques to retain the advantages of simplicity in treating flows with large deformations and folding free surfaces. In order to increase the range of problems which can be accurately studied with this technique, we have extended the SOLA-VOF method to include the viscous terms in the interfacial boundary condition and, in a limited manner, to allow flow of the vapour phase. As we will show below, the consideration of vapour phase flow and variations in pressure is critically important for stability analysis.

The VOF method is derived from the first generally successful volume-tracking free surface programme, the marker-and-cell (MAC) method.[6] The MAC method tracks the location of the fluid within a fixed Eulerian mesh through the use of massless marker particles. These particles are convected through the computational domain at the end of each time step using the interpolated local fluid velocity. The free surface is constructed from the cells partially filled with marker particles and having neighbouring empty cells. In the MAC method the normal stress boundary condition at the interface is simplified to

$$P_l = P_v. \tag{11}$$

This simplified boundary condition, applied at the cell centre rather than at the actual interface location, greatly reduces the accuracy of the computational technique. The first use of donor–acceptor differencing in a MAC-type free surface method which considers both phases appears to be by Ramshaw and Trapp.[5] However, they neglect the effects of surface tension, which is often the critical parameter in free surface and multiphase flow problems

The MAC method has evolved into the VOF technique, which can be looked upon as allowing the number of marker particles to become infinite and integrating to determine the fraction of the cell containing fluid. Thus the liquid is tracked by a step function F representing the fraction of each computational cell occupied by liquid. Transport of F through the computational mesh is governed by the F-convection equation

$$\frac{\partial F}{\partial t} = -\mathbf{v} \cdot \nabla F, \tag{12}$$

which ensures that the amount of each phase is conserved.

The interface between the phase is determined on a cellwise basis from local F-values. Cells with $F = 1$ are liquid cells, cells with $F = 0$ are vapour cells and cells with intermediate values of F are free surface cells. Once the free surface cells have been identified, the location and shape of the interface within the free surface cells may be reconstructed from gradients of the F-function.

The VOF implementation, SOLA-VOF, included the effects of surface tension, yielding Laplace's formula

$$P_l = P_v - \sigma\kappa \tag{13}$$

as the free surface boundary condition. In addition, the NASA-VOF[7] technique incorporates an interpolation scheme for applying the boundary condition at the free surface location rather than at the centre of the computational cell. Improvements in algorithms for computing surface curvature and methods for treating obstacles within the computational domain and extension to three dimensions for cylindrical co-ordinates led to the NASA-VOF3D programme.[8] However, all three programmes neglect the viscous components of the liquid normal stress in the liquid phase at the interface and assume that the pressure in the vapour phase remains constant. These assumptions impose severe limitations on the applicability of this method to free surface problems where viscous stresses are important.

In the study of free surface flows such as the die swell problem, inclusion of the viscous terms in the interfacial boundary condition is vital.[9,10] Therefore implementations which neglect these components are unable to accurately solve this problem. In addition, in the study of the stability of thin liquid films the viscous terms at the interface and variations in the vapour phase pressure along the interface are the primary factors inducing instability and wave formation.[11]

For these reasons we extend the VOF technique to (1) allow inclusion of the liquid phase viscous stress at the interface, (2) allow variation of the pressure in the vapour phase and (3) greatly extend the applicability of the VOF method to free surface problems. We will outline the various solution algorithms, followed by the numerical treatment of the static contact line and the implementation of viscous components at the interface. Next the numerical solution of the vapour phase flow coupled with the liquid phase solution will be presented, followed by a comprehensive examination of the technique's accuracy.

We solve the lid-driven cavity problem to examine the accuracy of the solution of the Navier–Stokes equations. The die swell problem is solved to test the implementation of the viscous stresses at the interface. Finally we solve for the stability of a liquid sheet and compare the results with those of linear stability analysis. We accurately compute the growth rate of waves in a thin liquid sheet in agreement with predictions from linear stability analysis. The above test problems are solved to demonstrate that, despite popular perceptions, this method can be very accurate and reliable when the complete interfacial condition is considered.

2. NUMERICAL TECHNIQUE

We begin the description of the numerical technique with a brief outline of the solution algorithm employed in the SOLA family of programmes. This is followed by a description of the additions we have made to improve the accuracy and extend the capabilities of our programme, IPST-VOF3D. Specifically, we highlight more accurate methods for differencing the convective terms in the NSE, modifications needed to treat a static contact point on an interior obstacle, inclusion of the viscous terms in the liquid at the interface and solution of the potential flow equations in the vapour phase to yield the pressure in the vapour phase.

2.1. SOLA solution algorithm

In the SOLA family of programmes the velocity and pressure fields are solved on a staggered grid (Figure 1). In this representation vector quantities are stored on cell faces and scalar quantities at the cell centres.

Here we briefly describe the numerical method used to solve the NSE in the SOLA family of

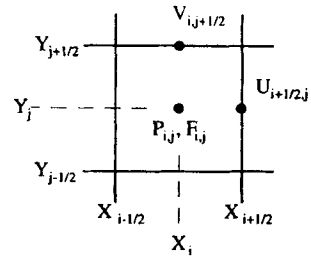

Figure 1. Schematic diagram of computational grid geometry

programmes. Generally this can be described by defining an explicit guess

$$\tilde{\mathbf{v}}^n = \mathbf{v}^n + \delta t\left(\mathbf{g} - \frac{1}{\rho}\nabla P^n + \nu\nabla^2\mathbf{v}^n - \mathbf{v}^n\cdot\nabla\mathbf{v}^n\right) \tag{14}$$

for the new velocity field, where the superscript n refers to the time step. Except as described below for the convective terms, the specific of the finite difference representations used can be found in References 7, 8 and 12. The velocity field after the time step can then be written as the explicit guess plus a correction term due to the pressure change across the time step, i.e.

$$\mathbf{v}^{n+1} = \tilde{\mathbf{v}}^n - \frac{\delta t}{\rho}\nabla(\delta P^{n+1}). \tag{15}$$

Since mass must be conserved at all times, we may substitute (15) into (1) to yield

$$\frac{\delta t}{\rho}V\nabla\cdot[\nabla(\delta P^{n+1})] = V\nabla\cdot\tilde{\mathbf{v}}^n, \tag{16}$$

where V is the volume of the computational cell, needed to ensure a symmetric system of equations.[7] The Poisson equation for pressure, (16), yields a sparse, symmetric linear system of equations that can be solved using a variety of numerical methods such as the successive overrelaxation (SOR) or conjugate residual (CR) methods.[8] With the new pressure field available, the updated velocity field is then computed from (15).

The F-convection equation (12) is solved using donor–acceptor differencing[8] to assist in maintaining a sharp interface between the liquid and vapour phases. Once the new fluid configuration has been obtained, it is possible to reconstruct the localized interface configuration needed for computation of the surface tension force.[8] Again the details of this process are presented elsewhere.[7,8,12]

2.2. *Differencing of convective term*

As we will show below, as the Reynolds number increases, the accuracy of the finite difference representation of the convective terms in the NSE limits the accuracy of the entire solution. Therefore, in addition to the standard differencing for the convective terms present in the SOLA programme, we apply and evaluate three third-order-accurate differencing options: quadratic upstream interpolation for convective kinematics (QUICK),[13,14] third-order-accurate upwind differencing (THIRD)[12,15] and the method of Kawamura and Kuwahara (KANDK).[12,16]

As an example, the constant grid formula for the convective term involving the x-component of velocity in the x-direction is given by

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2}. \tag{17}$$

Analogous formulae have been developed for the remaining terms for grids with variable cell spacing, except as noted. Presentation of these formulae and their derivations can be found in Reference 12.

The first technique used in the SOLA family of programmes consists of a linear combination of first-order-accurate upwind differencing and second-order-accurate central differencing. This leads to the constant grid formulae

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{2\Delta x}[(1+\alpha)u_{i+3/2} - 2\alpha u_{i+1/2} - (1-\alpha)u_{i-1/2}], \quad u_{i+1/2} < 0, \tag{18a}$$

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{2\Delta x}[(1-\alpha)u_{i+3/2} + 2\alpha u_{i+1/2} - (1+\alpha)u_{i-1/2}], \quad u_{i+1/2} > 0, \tag{18b}$$

where Δx is the cell spacing and α controls the fraction of central differencing. Setting $\alpha = 1$ yields the first-order-accurate upwind differencing and setting $\alpha = 0$ yields second-order-accurate central differencing. Numerical stability considerations limit the fraction of central differencing.[8]

The first of the three third-order-accurate differencing schemes is based on the QUICK differencing technique which uses quadratic upstream interpolation to compute the value of the convected variable at each face of a control volume.[13] These interpolated values are then used to form a centred finite difference formula. We have combined simplified forms of the QUICK interpolation formuale[14] to yield the finite difference formulae

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{8\Delta x}(3u_{i+3/2} + 3u_{i+1/2} - 7u_{i-1/2} + u_{i-3/2}), \quad u_{i+1/2} > 0, \tag{19a}$$

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{8\Delta x}(-u_{i+5/2} + 7u_{i+3/2} - 3u_{i+1/2} - 3u_{i-1/2}), \quad u_{i+1/2} < 0, \tag{19b}$$

for our implementation of QUICK differencing.

The second of the third-order-accurate differencing techniques, THIRD, was derived for constant grids by Agarwal.[15] This technique again uses upstream differencing for stability but is derived in a different manner.[12] The finite difference formulae for constant grid spacing are

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{6\Delta x}(2u_{i+3/2} + 3u_{i+1/2} - 6u_{i-1/2} + u_{i-3/2}), \quad u_{i+1/2} > 0, \tag{20a}$$

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{6\Delta x}(-u_{i+5/2} + 6u_{i+3/2} - 3u_{i+1/2} - 2u_{i-1/2}), \quad u_{i+1/2} < 0, \tag{20b}$$

The final third-order-accurate technique, termed KANDK, is a differencing scheme developed by Kawamura and Kuwahara.[16] They used an alternative approach to derive a third-order-accurate scheme, beginning with a second-order-accurate upwind scheme and eliminating the term leading to the third-order error. The constant grid formulae for KANDK are

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{6\Delta x}(u_{i+5/2} - 2u_{i+3/2} + 9u_{i+1/2} - 10u_{i-1/2} + 2u_{i-3/2}), \quad u_{i+1/2} > 0, \tag{21a}$$

$$\left(u\frac{\partial u}{\partial x}\right)_{i+1/2} = \frac{u_{i+1/2}}{6\Delta x}(-2u_{i+5/2} + 10u_{i+3/2} - 9u_{i+1/2} + 2u_{i-1/2} - 2u_{i-3/2}), \quad u_{i+1/2} < 0, \tag{21b}$$

Attempts to derive variable grid finite difference formulae in the manner used by Kawamura and Kuwahara were unsuccessful. Our derivation of a variable grid version of Kawamura and Kuwahara's method is presented in the Appendix, with other details reported elsewhere.[12]

2.3. Treatment of a static contact line

Many free surface problems have a contact point or line which joins the liquid, vapour and solid phases. A static contact is an intersection between vapour, liquid and solid phases where the point of contact is fixed but the contact angle can vary. The variation in the contact angle is part of the solution and often has a significant effect on the free surface shape. An example is the die swell problem described below.

In previous studies[7,8,17] the dynamic contact lines are treated by modifying the surface tension component of the interfacial boundary condition in the cell adjacent to the wall. The contact angle is

specified in the programme input and the contact line determined from the local fluid configuration. The surface force is then computed from the contact angle and the surface tension. In the case of a static contact line we compute the contact angle from the local fluid configuration and then apply the surface force in the same manner as in the previous studies.[7,8,17]

2.4. *Viscous component of interfacial boundary condition*

As mentioned above, the previous VOF technique use a simplified boundary condition, equation (13), for the normal stress balance. To eliminae the assumption that the viscous terms in the interfacial boundary condition are negligible, we have included an option for computing the viscous forces. The local unit vector normal to the interface is computed in the manner used in Reference 8 during surface tension computations. Once the co-ordinate axis most nearly normal to the interface has been determined, a local height function analogous to (3) is computed and the unit surface normal is obtained from (4).

Next the components of the viscous stress tensor (9) are computed using the provisional velocities $\tilde{\mathbf{v}}_t^n$, where only velocities within the liquid phase are included in the finite difference formulae. For example, with reference to Figure 2, the components of the viscous stress tensor, assuming constant grid spacing, are computed as

$$\tau_{xx} = 2\mu \frac{U_{i+1/2,j} - U_{i-1/2,j}}{\Delta x}, \tag{22a}$$

$$\tau_{yy} = 2\mu \frac{(1+2F_{i,j})V_{i,j+1/2} - 4F_{i,j}V_{i,j-1/2} - (1-2F_{i,j})V_{i,j-3,2}}{2\Delta y}, \tag{22b}$$

$$\tau_{xy} = \tau_{yx} = \mu\left(\frac{U_{i+1/2,j} + U_{i-1/2,j} - U_{i+1/2,j-1} - U_{i-1/2,j-1}}{2\Delta y} + \frac{V_{i+1,j+1/2} - V_{i-1,j+1/2}}{2\Delta x}\right), \tag{22c}$$

With the viscous stress tensor and the unit normal vector available, the viscous component of the interfacial boundary condition is computed from $\mathbf{n} \cdot \tau_l \cdot \mathbf{n}$.

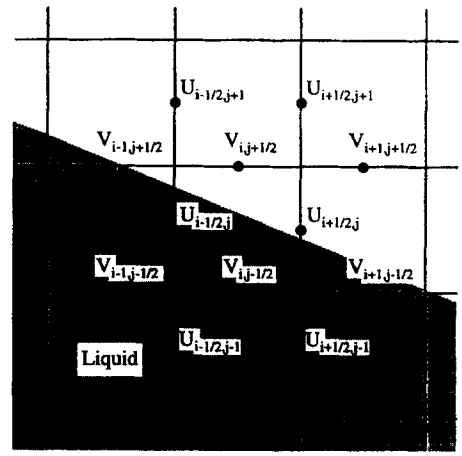

Figure 2. Example fluid configuration for viscous term computation

2.5. *Potential flow in vapour phase*

As stated above, for stability problems such as flow of a thin liquid sheet, allowing the pressure in the vapour phase to vary is vital. We have implemented a method for solving the potential flow equation in the vapour phase which is coupled to the full NSE in the liquid phase through the interfacial conditions. This allows computation of the pressure in the vapour phase as a function of time and position.

With the assumptions that the vapour phase is inviscid and the flow in the vapour phase is irrotational, the vapour phase may be modelled using the potential flow

$$\nabla^2 \phi_v = 0, \tag{23}$$

where ϕ_v is the vapour phase potential. The pressure and velocities in the vapour phase are defined as

$$P_v = -\rho_v \frac{\partial \phi_v}{\partial t} - \tfrac{1}{2}\rho_v |\mathbf{v}_v|^2, \tag{24}$$

$$\mathbf{v}_v = \nabla \phi_v. \tag{25}$$

Therefore in the vapour phase we must solve Laplace's equation in a region with curved boundaries having Neuman boundary conditions. This is accomplished using standard second-order-accurate finite difference formulae for (23) in the bulk of the vapour phase and adjacent to straight boundaries. At the interface between the two fluids a more complex treatment is required.

We have implemented a modified form of a second-order-accurate method for solving Poisson's equation in a region with curved boundaries having mixed boundary conditions. Bramble and Hubbard[18] define a second-order-accurate operator

$$\delta_n \phi_0 = \phi_0 \sum_{i=1}^{3} a_i - \sum_{i=1}^{3} a_i \phi_i, \tag{26}$$

for the normal derivative using three points within the region of interest, where $\delta_n \phi_0$ is the normal derivative of ϕ_0 at the surface point of interest, ϕ_i are three points within the vapour phase and the coefficients a_i are determined from the solution of the system of equations

$$\begin{bmatrix} \bar{y}_1 & \bar{y}_2 & \bar{y}_3 \\ \bar{x}_1 & \bar{x}_2 & \bar{x}_3 \\ \bar{x}_1^2 - \bar{y}_1^2 & \bar{x}_2^2 - \bar{y}_2^2 & \bar{x}_3^2 - \bar{y}_3^2 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \tag{27}$$

Here $\bar{y}_i$ and $\bar{x}_i$ are the distances from the surface point of interest to the points ϕ_i in the normal and tangential directions respectively. In addition, Bramble and Hubbard[18] present criteria which ensure that the operator yields a diagonally dominant system of equations.

The boundary condition for the vapour phase potential at the interface is the conservation of normal velocity, (10). Thus the boundary operator is equal to the normal velocity in the liquid phase plus a small correction arising from the derivation, i.e.

$$\delta_n \phi_0 = v_{\bar{y}} + \frac{\partial v_{\bar{y}}}{\partial \bar{x}} \sum_{i=1}^{3} a_i \bar{x}_i \bar{y}_i, \tag{28}$$

where $v_{\bar{y}}$ is the normal velocity at the interface and $\partial v_{\bar{y}}/\partial \bar{x}$ is the tangential derivative of the normal velocity at the interface.[18] The value of the vapour phase potential at the interface, needed to solve Laplace's equation, can be obtained by combining (27) and (28) and rearranging to yield

$$\phi_0 = \frac{v_{\bar{y}} + (\partial v_{\bar{y}}/\partial \bar{x}) \sum_{i=1}^{3} a_i \bar{x}_i \bar{y}_i + \sum_{i=1}^{3} a_i \phi_i}{\sum_{i=1}^{3} a_i}. \tag{29}$$

As stated above, since we have assumed the vapour phase to be inviscid, no restrictions are placed on the tangential velocity at the interface.

Incorporation of the liquid phase viscous terms and variations in the vapour phase pressure into the interfacial boundary condition yield the following solution procedure. First, compute the surface curvature from the local liquid configuration. Second, compute the surface normal velocity from the change in surface position and solve (23) for the vapour phase potential. Third, compute the vapour phase pressure from (24). Fourth, compute the explicit guess for the liquid phase velocity field from (14). Fifth, compute the interfacial liquid phase stress from (9). Sixth, compute the pressure on the liquid side of the interface from (13). Seventh, solve the Poisson pressure equation (16) to yield the new liquid phase pressure field. Eighth, update the liquid phase velocity field using (15). Finally, solve (12) to yield the new fluid configuration.

This procedure may be repeated until the desired time is reached. The second and third steps have been added to allow for variations in the vapour phase pressure, while the fifth step is required for inclusion of the vapour phase viscous terms in the interfacial conditions.

3. NUMERICAL RESULTS

In this section we present results from three sample problems chosen to demonstrate the accuracy of each of the major extensions to the VOF family of programmes. First we present results for flow in a lid-driven cavity which demonstrate the accuracy of the convective term differencing schemes. Then we give examples of the die swell phenomenon where the liquid phase viscous component of the interfacial boundary condition is important. Finally we present results from a study of the stability of a thin liquid sheet flowing through an inviscid vapour phase.

3.1. Flow in a lid-driven cavity

The lid-driven cavity (LDC) problem is commonly used for testing numerical solutions of the NSE. We have chosen to use a square cavity (Figure 3) at $Re = HV/\nu = 1000$ and with an aspect ratio $H/W = 1$

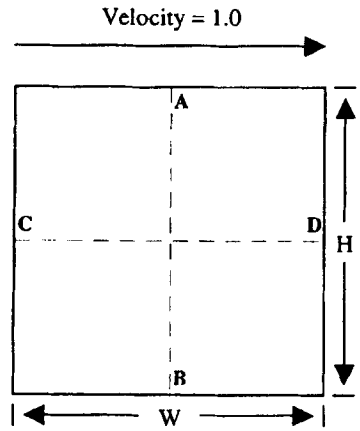

Figure 3. Schematic diagram of lid-driven cavity problem

Table I. Error for lid-driven cavity problem at $Re = 1000$

Convective term differencing scheme	Grid	
	Constant	Variable
SOLA ($\alpha = 0{\cdot}5$)	54·3%	38·8%
QUICK	15·4%	4·8%
Third-order-accurate upwind	13·0%	4·2%
Kawamura and Kuwahara	12·4%	6·7%

to test the accuracy of SOLA differencing with $\alpha = 0{\cdot}5$ and the three third-order-accurate techniques described above.

Two computational grids were used for each differencing scheme, the first having 40 equally spaced cells in each direction and the second having cell spacings one-half the mean spacing adjacent to the walls and twice the mean spacing in the centre of the cavity. Results of these eight simulations, with the results of Ghia *et al.*[19] included for comparison, are presented in Figure 4 for the horizontal component of velocity along the vertical centreline AB and in Figure 5 for the vertical component of velocity along the horizontal centreline CD.

The accuracies in predicting the local extrema in Figures 4 and 5 for each case as compared with the results of Ghia *et al.*[19] are presented in Table I. Using variable grid spacing, we were able to get within 5% of the result of Ghia *et al.* while employing only 10% as many computational cells.

Any of the third-order-accurate methods produced markedly superior results to SOLA differencing for this problem; however, all the third-order cases required longer computation times. While KANDK yielded the most accurate results for the constant grid, the accuracy of this method was the worst of the

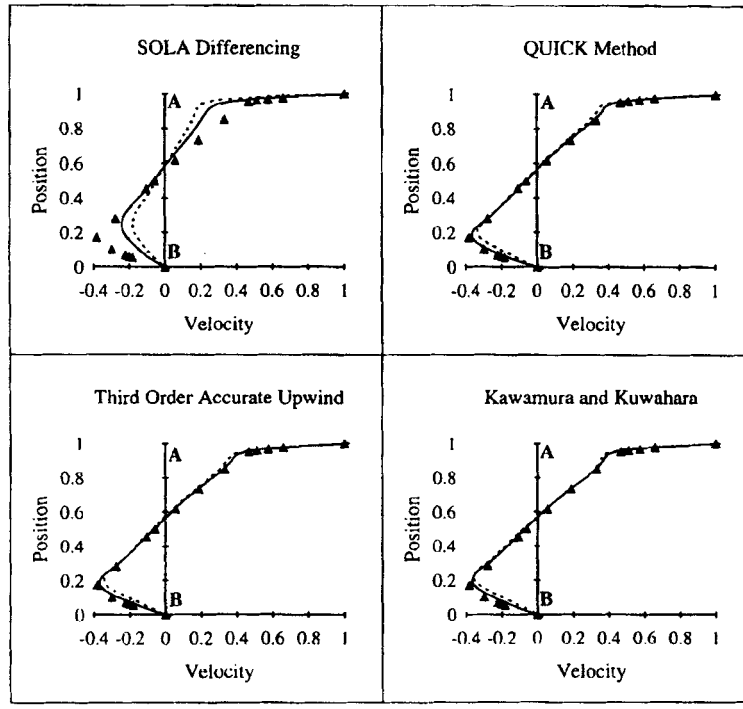

Figure 4. Plots of horizontal component of velocity along vertical centreline: ▲, Ghia *et al.*;[19] ·····, variable grid; ————, constant grid

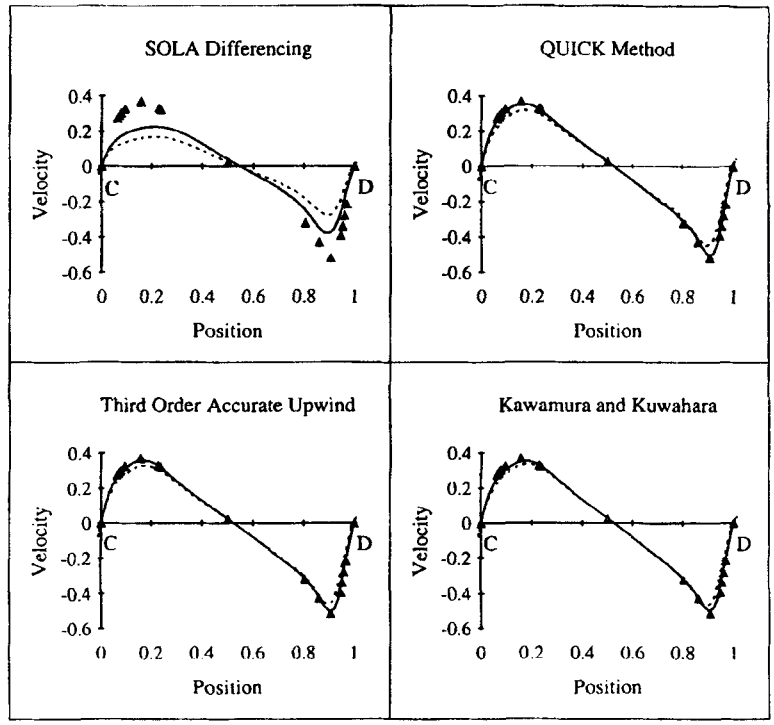

Figure 5. Plots of vertical component of velocity along horizontal centreline: ▲, Ghia *et al.*;[19] ·····, variable grid; ————, constant grid

third-order-accurate techniques for the variable grid. This difference in ranking is attributed to the inaccuracies present in the variable grid formulation of Kawamura and Kuwahara's[16] method as outlined in the Appendix.

3.2. Cartesian die swell problem

We have studied the Cartesian die sell problem (Figure 6) both with and without surface tension to test the accuracy of the computation of the liquid phase viscous stress at the interface. Relatively minor additions were necessary to modify the surface tension algorithm in Reference 18 to allow computation of the surface curvature for problems in Cartesian co-ordinates. Additional modifications to the velocity boundary conditions at the corner of the die were implemented in a manner analogous to that used by Hill.[9,10] Details of these changes can be found in Reference 12.

Results from a die swell case without surface tension at $Re = HV/\nu = 300$ and $Ca^{-1} = \sigma/V\mu = 0$, where V is the average inlet velocity, are presented. As shown in Figure 7(a), the initial condition consisted of the free surface level with the edge of the die. The initial velocity profile (Figure 7(b)) was

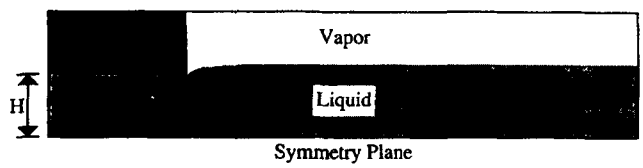

Figure 6. Schematic diagram of die swell problem

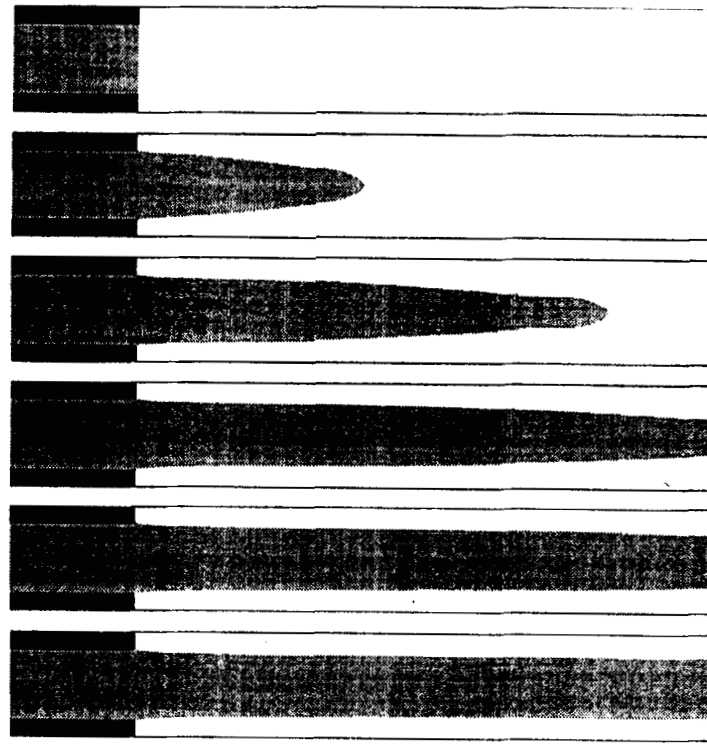

Figure 7. Results for die swell problem at $Re = 300$ and $Ca^{-1} = 0$

parabolic throughout the liquid phase and the liquid phase viscous component of the interfacial boundary condition was included. Figures 7(c)–7(f) show the evolution of the solution as a function of time until a steady state solution was obtained.

The predicted die swell for the case with $Re = 300$ and $Ca^{-1} = 0$ is $-15{\cdot}66\%$ This is in good agreement with results from the literature of $-15{\cdot}24\%$[20] and $-15{\cdot}52\%$.[21] Results from several solutions using different computational grids are presented in Table II. All simulations used an entrance zone within the die of $3{\cdot}5H$ and had varying domain lengths as indicated, while the computational grid was graded with the minimum cell spacing in each direction adjacent to the static contact line.

Table II. Results of solutions of die swell problem at $Re = 300$ and $Ca^{-1} = 0$

	Minimum cell spacing			
Domain length	$0{\cdot}04H$	$0{\cdot}03H$	$0{\cdot}02H$	$0{\cdot}01H$
$20H$	$-14{\cdot}48\%$	$-14{\cdot}92\%$	$-15{\cdot}20\%$	$-15{\cdot}12\%$
$25H$	$-14{\cdot}78\%$	$-15{\cdot}08\%$	$-15{\cdot}52\%$	$-15{\cdot}37\%$
$30H$	$-15{\cdot}05\%$	$-15{\cdot}14\%$	$-15{\cdot}62\%$	$-15{\cdot}53\%$
$35H$	$-15{\cdot}19\%$	$-15{\cdot}17\%$	$-15{\cdot}66\%$	$-15{\cdot}63\%$
$40H$	$-15{\cdot}31\%$	$-15{\cdot}21\%$	$-15{\cdot}66\%$	$-15{\cdot}67\%$

Table III. Results of solutions of die swell problem at $Re = 75$ and $Ca^{-1} = 2$

Number of y-direction computational cells	Minimum cell spacing			
	0·0333	0·0278	0·0222	0·0167
30	−10·99%	−11·61%	−11·38%	−11·26%
36	*	−11·62%	−11·43%	−11·53%
45	*	*	−10·91%	−11·37%
60	*	*	*	−11·05%

*Combination not possible.

Results of a similar series of solutions obtained for a case with surface tension at $Re = 75$ and $Ca^{-1} = 2$ are presented in Table III. In this case the accuracy of the solution was improved by increasing the number of computational cells in the direction perpendicular to the flow direction and refining the grid in the region adjacent to the corner of the die. The predicted die swell shows more scatter than in the previous case but remains in reasonable agreement with the literature results of −11·16%,[20] −10·92%[22] and −10·48%.[21]

3.3. *Stability of a two-dimensional viscous liquid sheet*

The final test problem presented results from a study of the stability of a thin liquid sheet of fluid flowing through an inviscid vapour phase. Figure 8 presents a diagram of the problem. The variation in the surface position ε shown in Figure 8 is assumed to be of the form

$$\varepsilon = \varepsilon_0 e^{\omega t + ikx} \tag{30}$$

where ε_0 is the initial amplitude, $\omega = \omega_r + i\omega_i$ is the complex growth rate, $i = (-1)^{1/2}$ and k is the wave number of the disturbance. A linear stability analysis[11] yields dispersion relations for antisymmetric,

$$0 = (\tilde{\omega}_1 + 4m^2 Z)\tilde{\omega}_1 \tanh(m) + 4m^3 Z^2 \{m \tanh(m) + (m^2 + \tilde{\omega}_1/Z)^{1/2} \tanh[(m^2 + \tilde{\omega}_1/Z)^{1/2}]\} + \tilde{\rho}\tilde{\omega}^2 + m^3, \tag{31}$$

and axisymmetric,

$$0 = (\tilde{\omega}_1 + 4m^2 Z)\tilde{\omega}_1 \coth(m) + 4m^3 Z^2 \{m \coth(m) + (m^2 + \tilde{\omega}_1/Z)^{1/2} \coth[(m^2 + \tilde{\omega}_1/Z)^{1/2}]\} + \tilde{\rho}\tilde{\omega}^2 + m^3, \tag{32}$$

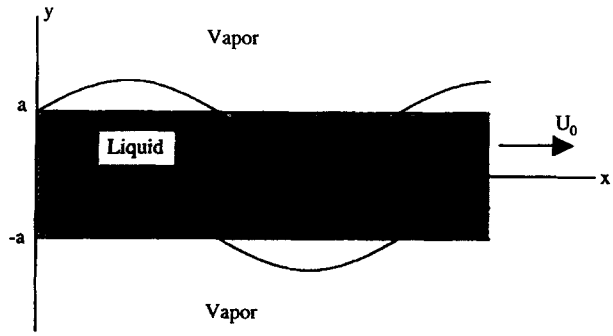

Figure 8. Schematic diagram of sheet instability problem

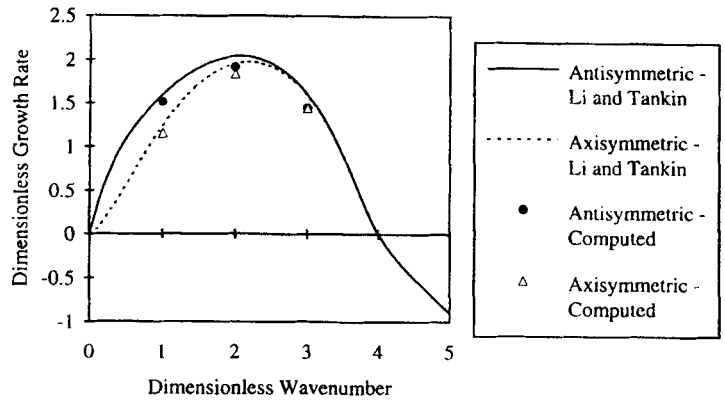

Figure 9. Non-dimensional growth rate for $We_l = 40$, $Z = 0{\cdot}1$ and $\tilde{\rho} = 0{\cdot}1$ obtained from numerical solution of Li and Tankin's[11] dispersion relations. Open and full symbols represent results of our computational analysis

disturbances. Here $\tilde{\omega} = \tilde{\omega}_r + i We_l^{1/2} \tilde{\omega}_i$, $\omega_1 = \tilde{\omega} + i We_l^{1/2} m$, $\tilde{\omega}_r = \omega_r (\sigma/\rho_l a^3)^{-1/2}$, $\tilde{\omega}_i = \omega_i (a/U_0) m$, a is the initial sheet half-thickness, $m = ka$ is the dimensionless wave number, and U_0 is the initial sheet velocity. The remaining parameters are defined as the Weber number $We_l = \rho_l U_0^2 a/\sigma$, the Ohnesorge number $Z = \mu_l (\rho_l a \sigma)^{-1/2}$ and the density ratio $\tilde{\rho} = \rho_v/\rho_l$.

It is possible to solve the dispersion relations (31) and (32) for given We_l, Z and $\tilde{\rho}$ to yield the complex growth rate $\tilde{\omega}$ as a function of the wave number. The real part of $\tilde{\omega}$ is the dimensionless growth rate of a disturbance with wave number m. Results for a case with $We_l = 40$, $Z = 0{\cdot}1$ and $\tilde{\rho} = 0{\cdot}1$ are shown in Figure 9 for both antisymmetric and axisymmetric disturbances, represented by full and broken curves respectively. The data points plotted in Figure 9 represent our computed results for antisymmetric and axisymmetric disturbances, represented by full and open symbols respectively.

Solutions at $m = 1$ were obtained on a computational domain of size $2\pi a$ in the primary direction of flow and $8a$ perpendicular to the flow. This problem was discretized on a computational grid with 360 cells constantly spaced cells in the direction of flow and 100 graded cells perpendicular to the primary direction of flow. The grading was done so that a region of constant cell spacing was maintained adjacent to the interface. Problems with larger wave numbers used the same computational grid with a shorter computational domain, keeping the number of computational cells per wave constant.

The computed growth rates were obtained from an initial surface purturbation of $0{\cdot}0001a$, i.e. 0·5% of the computational grid spacing adjacent to the interface. Displacements of adjacent peaks and troughs on both sides of the fluid were tracked through over four e-foldings and the growth rate determined from the slope of the average displacement on a semilogarithmic plot.

4. CONCLUSIONS

We have presented several extensions to the VOF method for tracking the location of the interface between a liquid and a vapour phase included in the SOLA method for solving the NSE. These have included more accurate methods for treating the convective terms in the NSE, a method for treating a static contact line, inclusion of the liquid phase viscous terms in the interfacial conditions and the ability to solve for flow in the vapour phase coupled with the flow in the liquid phase.

These modifications allow the VOF technique to be applied to a wider variety of problems, including the die swell problem and study of the stability of a thin viscous sheet flowing through an inviscid vapour phase. We have shown that when the complete boundary conditions at the interface

between a viscous liquid and an inviscid liquid are imposed, the VOF technique coupled with the SOLA algorithm can in fact yield accurate solutions for complex problems.

ACKNOWLEDGEMENTS

This study is supported by industrial contributions and the National Science Foundation under grant CTS-9258667. The computations are conducted in part using the Cornell National Supercomputer Center, a resource of the Center for Theory and Simulation in Science and Engineering at Cornell University, which is funded in part by the National Science Foundation, New York State and the IBM Corporation. Portions of this work were used by J.F.M. as partial fulfilment of the requirements for the Ph.D. degree at the Institute of Paper Science and Technology.

APPENDIX: DERIVATION OF VARIABLE GRID KAWAMURA AND KUWAHARA METHOD

We begin by reviewing the derivation of Kawamura and Kuwahara. This is followed by an attempt to directly reproduce their derivation scheme for a variable grid. Finally we will present our approximation of the variable grid KANDK method. For simplicity these derivations will be carried out using the points u_{i+2}, u_{i+1}, u_i, u_{i+1}, and u_{i+2} rather than the points at the cell faces used earlier and the constant grid spacing will be denoted h.

Kawamura and Kuwahara's derivation

We begin with a second-order upwind differencing scheme

$$\left(u\frac{\partial u}{\partial x}\right)_i = \frac{u_i}{2h}(3u_i - 4u_{i-1} + u_{i-2}), \quad u_i > 0, \tag{33a}$$

$$\left(u\frac{\partial u}{\partial x}\right)_i = \frac{u_i}{2h}(-u_{i+2} + 4u_{i+1} - 3u_i), \quad u_i < 0, \tag{33b}$$

These formulae can be combined to yield a single formula independent of the flow direction:

$$\left(u\frac{\partial u}{\partial x}\right)_i = \frac{u_i}{4h}(-u_{i+2} + 4u_{i+1} - 4u_{i-1} + u_{i-2}) + \frac{|u_i|}{4h}(u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}). \tag{34}$$

From Taylor series expansions this formula can be rewritten as

$$\left(u\frac{\partial u}{\partial x}\right)_i = u_i\left(\frac{\partial u}{\partial x} - \frac{h^2}{2}\frac{\partial^3 u}{\partial x^3} + O(h^4)\right) + |u_i|\left(h^3\frac{\partial^4 u}{\partial x^4} + O(h^5)\right). \tag{35}$$

Thus the leading error in (33) or (34) can be reduced by eliminating the term

$$\frac{h^2}{2}\frac{\partial^3 u}{\partial x^3}. \tag{36}$$

Improved accuracy is obtained by replacing the first term in (35),

$$\frac{\partial u}{\partial x} - \frac{h^2}{2}\frac{\partial^3 u}{\partial x^3} + O(h^4), \tag{37}$$

with

$$\frac{\partial u}{\partial x} + O(h^4) = \frac{-u_{i+2} + 8u_{i+1} - 8u_{i-1} + u_{i-2}}{12h}, \tag{38}$$

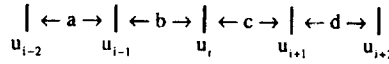

Figure 10. Diagram of variable grid dimensions

yielding

$$\left(u\frac{\partial u}{\partial x}\right)_i = u_i\left(\frac{\partial u}{\partial x} + O(h^4)\right) + |u_i|\left(h^3\frac{\partial^4 u}{\partial x^4} + O(h^5)\right). \tag{39}$$

The resulting analogue of (34) with error of $O(h^4)$ is

$$\left(u\frac{\partial u}{\partial x}\right)_i = \frac{u_i}{12h}(-u_{i+2} + 8u_{i+1} - 8u_{i-1} + u_{i-2}) + \frac{|u_i|}{4h}(u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}). \tag{40}$$

Attempt at Kawamura and Kuwahara's derivation with variable grid

In this subsection we follow the steps of Kawamura and Kuwahara's derivation as far as possible for a variable grid. All derivations will be with respect to the dimensions shown in Figure 10.

For a variable grid (33) becomes

$$\left(u\frac{\partial u}{\partial x}\right)_i = u_i\left(\frac{a+2b}{b(a+b)}u_i - \frac{a+b}{ab}u_{i-1} + \frac{b}{a(a+b)}u_{i-2}\right), \quad u_i > 0, \tag{41a}$$

$$\left(u\frac{\partial u}{\partial x}\right)_i = u_i\left(-\frac{c}{d(c+d)}u_{i+2} + \frac{c+d}{cd}u_{i+1} - \frac{2c+d}{c(c+d)}u_i\right), \quad u_i < 0. \tag{41b}$$

Thus (34) can be rewritten as

$$\left(u\frac{\partial u}{\partial x}\right)_i = \frac{u_i}{2}\left[-\frac{c}{d(c+d)}u_{i+2} + \frac{c+d}{cd}u_{i+1} + \left(-\frac{2c+d}{c(c+d)} + \frac{a+2b}{b(a+b)}\right)u_i - \frac{a+b}{ab}u_{i-1} + \frac{b}{a(a+b)}u_{i-2}\right]$$

$$+\frac{|u_i|}{2}\left[\frac{c}{d(c+d)}u_{i+2} - \frac{c+d}{cd}u_{i+1} + \left(\frac{2c+d}{c(c_d)} + \frac{a+2b}{b(a+b)}\right)u_i - \frac{a+b}{ab}u_{i-1} + \frac{b}{a(a+b)}u_{i-2}\right]. \tag{42}$$

In order to continue with Kawamura and Kuwahara's derivation, we begin by defining the terms in (35):

$$\frac{\partial u}{\partial x} + O(h^2) = \frac{b}{c(b+c)}u_{i+1} + \frac{c-b}{bc}u_i - \frac{c}{b(b+c)}u_{i-1}, \tag{43}$$

$$\frac{\partial^3 u}{\partial x^3} + O(h^4) = \frac{6(a+2b-c)}{d(c+d)(b+c+d)(a+b+c+d)}(u_{i+2} - u_i) - \frac{6(a+2b-c-d)}{cd(a+b+c)(b+c)}(u_{i+1} - u_i)$$

$$+\frac{6(d+2c-b-a)}{ab(b+c+d)(b+c)}(u_{i-1} - u_i) - \frac{6(d+2c-b)}{a(a+b)(a+b+c)(a+b+c+d)}(u_{i-2} - u_i), \tag{44}$$

$$\frac{\partial^4 u}{\partial x^4} + O(h^5) = \frac{24}{d(c+d)(b+c+d)(a+b+c+d)}(u_{i+2} - u_i) - \frac{24}{cd(a+b+c)(b+c)}(u_{i+1} - u_i)$$

$$-\frac{24}{ab(b+c+d)(b+c)}(u_{i-1} - u_i) + \frac{24}{a(a+b)(a+b+c)(a+b+c+d)}(u_{i-2} - u_i). \tag{45}$$

When (43)–(45) are substituted into (35), it does not yield (34), implying that the derivation of Kawamura and Kuwahara's method is inaccurate for variable grids.

Approximate Kawamura and Kuwahara derivation with variable grid

We can obtain an approximate form of Kawamura and Kuwahara's technique by beginning the derivation with (39). After substituting the variable grid analogue of (38),

$$\frac{\partial u}{\partial x} + O(h^4) = -\frac{bc(a+b)}{d(c+d)(b+c+d)(a+b+c+d)}(u_{i+2} - u_i) + \frac{b(a+b)(c+d)}{cd(a+b+c)(b+c)}(u_{i+1} - u_i)$$

$$-\frac{c(a+b)(c+d)}{ab(b+c+d)(b+c)}(u_{i-1} - u_i) + \frac{bc(c+d)}{a(a+b)(a+b+c)(a+b+c+d)}(u_{i-2} - u_i), \tag{46}$$

and (45) into (39), the resulting approximate formula for Kawamura and Kuwahara's method on a variable grid becomes

$$\left(u\frac{\partial u}{\partial x}\right)_i = \frac{-bc(a+b)u_i + 24|u_i|}{d(c+d)(b+c+d)(a+b+c+d)}(u_{i+2} - u_i) + \frac{b(a+b)(c+d)u_i - 24|u_i|}{cd(a+b+c)(b+c)}(u_{i+1} - u_i)$$

$$-\frac{c(a+b)(c+d)u_i - 24|u_i|}{ab(b+c+d)(b+c)}(u_{i-1} - u_i) + \frac{bc(c+d)u_i + 24|u_i|}{a(a+b)(a+b+c)(a+b+c+d)}(u_{i-2} - u_i). \tag{47}$$

REFERENCES

1. R. W. Yeung, 'Numerical methods in free-surface flows', *Ann. Rev. Fluid Mech.*, **14**, 395–442 (1982).
2. J. M. Hyman, 'Numerical methods for tracking interfaces', *Physica D*, **12**, 396–407 (1984).
3. K. J. Laskey, E. S. Oran and J. P. Boris, 'Approaches to resolving and tracking interfaces and discontinuities', *Naval Research Laboratory Rep. 5999*, Arlington, VA, 1987.
4. J. M. Floryan and H. Rasmussen, 'Numerical methods for viscous flows with moving boundaries', *Appl. Mech. Rev.*, **42**, 323 (1989).
5. J. D. Ramshaw and J. A. Trapp, *J. Comput. Phys.*, **21**, 438 (1976).
6. F. H. Harlow and J. E. Welch, 'Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface', *Phys. Fluids*, **8**, 2182 (1965).
7. M. D. Torrey, L. D. Cloutman, R. C. Mjolsness and C. W. Hirt, 'NASA-VOF2D: a computer program for incompressible flows with free surfaces', *LA-10612-MS*, Los Alamos National Laboratory, 1985.
8. M. D. Torrey, R. C. Mjolsness and L. R. Stein, 'NASA-VOF3D: a three-dimensional computer program for incompressible flows with free surfaces', *LA-11009-MS*, Los Alamos National Laboratory, 1987.
9. G. A. Hill, 'Numerical simulation of free surface flows', *Doctoral Dissertation*, University of Saskatchewan, Saskatoon, 1979.
10. G. A. Hill, C. A. Shook and M. N. Esmail, 'Finite difference simulation of die swell for a Newtonian fluid', *Can. J. Chem. Eng.*, **59**, 100 (1981).
11. X. Li and R. S. Tankin, 'On the temporal instability of a two-dimensional viscous liquid sheet', *J. Fluid Mech.*, **226**, 425 (1991).
12. J. F. McKibben, 'A computational fluid dynamics model for transient three-dimensional free surface flows', *Doctoral Dissertation*, Institute of Paper Science and Technology, Atlanta, GA, 1993.
13. B. P. Leonard, 'A stable and accurate convective modeling procedure based on quadratic upstream interpolation', *Comput. Methods Appl. Mech. Eng.*, **19**, 59 (1979).
14. C. J. Freitas, R. L. Street, A. N. Findikakis an dJ. R. Koseff, 'Numerical simulation of three-dimensional flow in a cavity', *Int. j. numer. methods fluids*, **5**, 561 (1985).
15. R. K. Agarwal, 'A third-order-accurate upwind scheme for Navier–Stokes solutions at high Reynolds numbers', *AIAA Paper AIAA-81-0112*, 1981.
16. T. Kawamura and K. Kuwahara, 'Computation of high Reynolds number flow around a circular cylinder with surface roughness', *AIAA Paper AIAA-84-0340*, 1984.
17. C. W. Hirt and B. D. Nichols, 'Volume of fluid (VOF) method for dynamics of free boundaries', *J. Comput. Phys.*, **39**, 201 (1981).
18. J. H. Bramble and B. E. Hubbard, 'Approximation of solutions of mixed boundary value problems for Poisson's equation by finite differences', *J. Assoc. Comput. Mach.*, **12**, 114 (1965).

19. U. Ghia, K. N. Ghia and C. T. Shin, 'High-*Re* solutions for incompressible flow using the Navier–Stokes equations and a multigrid method', *J. Comput. Phys.*, **48**, 387 (1982).
20. *FIDAP User's Manual*, Rev. 5.0, 1st edn, 6-1, Fluid Dynamics International, Evanston, IL, 1990.
21. J. Omedei, 'Computer solutions of a plane Newtonian jet with surface tension', *Comput. Fluids*, **7**, 79 (1979).
22. F. Dupret, 'A method for the computation of viscous flow by finite elements with free boundaries and surface tension', in T. Kawai (ed.), *Finite Element Flow Analysis*, University of Tokyo Press, Tokyo, 1982.