INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2003; **42**:1105–1134 (DOI: 10.1002/fld.582)

Numerical simulation of mold filling in foam reaction injection molding

Dongjin Seo, Jae Ryoun Youn and Charles L. Tucker III*,†

School of Materials Science and Engineering, Seoul National University, Seoul 151-744, South Korea

SUMMARY

A numerical simulation of reaction injection molding (RIM) of polymeric foam is developed, using a finite volume method (FVM). In this study we predict mold filling with a variable-density fluid that fills a mold by self-expansion. We deal with two-dimensional, isothermal cases. With the assumptions of ideal mixing and rapid bubble nucleation, the foam is modelled as a continuum with a time-dependent density. The continuum is assumed to be a Newtonian fluid. We develop a pressure-based FVM for unstructured meshes that includes the SIMPLE algorithm with treatment of fluid compressibility. Cell-based, co-located storage is used for all physical variables. To treat the moving interface, an explicit high-resolution interface capturing method is used. Foam flow in a slit is investigated, and the numerical calculations are in good agreement with an approximate analytic solution. For fountain flow in a rectangular cavity, the shape of the flow front is flatter and the traces of the particles are more complicated for an expanding foam than for a constant-density fluid. An example of mold filling by an expanding foam demonstrates the geometric flexibility of the method. Copyright © 2003 John Wiley & Sons, Ltd.

KEY WORDS: foam molding; injection molding; self-expanding fluid; finite volume method; unstructured meshes; fountain flow

INTRODUCTION

Reaction injection molding (RIM) is a widely used process for producing various kinds of complex parts including automobiles, furniture, appliances, and housings. In RIM, products are made from two or more chemical components through mixing, chemical reaction, and molding [1, 2]. Liquid reactants from two supply tanks flow at high pressure into a mix head, where they impinge at high velocity. After impingement mixing, the mixture flows out into the mold cavity and the polymerization is initiated. During the process, foaming can occur.

*Correspondence to: C. L. Tucker, Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, 140 Mechanical Engineering Building, MC-244, 1206 West Green Street, Urbana, IL 61801, U.S.A.
†E-mail: ctucker@uiuc.edu

Contract/grant sponsor: Applied Rheology Center (ARC)

Received 10 February 2002
Revised 3 April 2003

Chemical and/or physical blowing agents are generally used in RIM for foaming [3–8]. There are many advantages of foamed materials, including low cost, light weight, enhanced thermal and electrical insulation, and high impact strength [9].

In order to model the foam reaction injection molding (FRIM) process completely, one would have to understand the phenomena of mixing, bubble nucleation, reaction kinetics, bubble growth, foam rheology, and flow behaviour during mold filling. Each of these is a complex subject that has been studied intensively, and it is difficult to consider all of them simultaneously. For this reason, there have been few studies reported on FRIM. There have been some numerical simulations of the RIM process [10–14]. In these studies, numerical methods similar to those used for injection molding were used to simulate the process. Mold filling was driven by an external injection pressure. Mixing was generally assumed to be ideal. Reaction kinetics and heat transfer during mold filling were taken into account, but bubble growth and foam rheology were not considered.

Arai *et al.* [15] did experiments in which pre-mixed foam reactants were poured into an L-shaped mold, and allowed to fill the mold as the mixture foamed and expanded. The distribution of the foam density was then evaluated in the solid part. In spite of some variations due to the amine catalyst used, the density varied from about 30 to 40 kg/m^3 depending on the location in the mold, while the density of the free-rising foam was about 22 kg/m^3. Their research showed that foam density is affected by the pressure of the foam fluid, even though this is not the main factor that controls foam density.

Lefebvre and Keunings [16, 17] simulated numerically the continuous flow of chemically reactive polymeric liquids in two-dimensional geometries using a finite element method. The gelling reaction which leads to the formation of polyurethane, and a blowing reaction from a chemical blowing agent, were taken into account. A spine method was used to treat the free surface, however, this method is not appropriate for filling molds of complex geometry. The density of the foam was a function only of temperature, which varied because the reactions are exothermic.

In this study, we propose a model and predict mold filling with a variable-density fluid that fills a mold by self-expansion. We deal with two-dimensional cases as a basic investigation. With the assumptions of ideal mixing and rapid bubble nucleation, the foam is modelled as a continuum with a time-dependent density decrease that is induced by bubble growth. The continuum is assumed to be a Newtonian fluid. Reaction and temperature are not taken into consideration, as the focus in this study is on self-expansion of the foam.

For numerical calculations, we develop a pressure-based finite volume scheme for unstructured meshes [18] that includes the SIMPLE algorithm [19] with treatment of fluid compressibility. Cell-based, co-located storage is used for all physical variables. For treating the moving interface, an explicit high-resolution scheme that is similar to the CICSAM (compressive interface capturing scheme for arbitrary meshes) method [20] is used.

MODELLING

The mold that forms the computational domain for FRIM is partially filled with foam and has a moving interface. In our simulation the empty regions in the mold are considered to be completely filled with a fictitious fluid that has different physical properties from the foam [21]. If only the isothermal case is considered and surface tension at the interface is

negligible, the general governing equations include the continuity equation,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{1}$$

and the momentum equation,

$$\frac{\partial}{\partial t}(\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v}\mathbf{v} - \mathbf{T}) = \rho \mathbf{g} \tag{2}$$

Here ρ is density, t is time, $\mathbf{v}$ is the velocity vector, and $\mathbf{g}$ is the acceleration due to gravity. $\mathbf{T}$ is the total stress, which is defined as

$$\mathbf{T} = -p\mathbf{I} + \boldsymbol{\tau} \tag{3}$$

where p is pressure, $\mathbf{I}$ is the unit tensor, and $\boldsymbol{\tau}$ is the extra stress tensor.

In order to solve these equations, several assumptions are needed. The foam is considered as a continuum which can be modelled as a Newtonian fluid. Although the pressure in the foam fluid affects the density through the bubble growth in real FRIM systems [6, 22, 23], in this study density change from self-expansion is assumed to be a known function of time only. This approximates self-expansion due to a spatially homogeneous chemical reaction. The equations that follow assume that density can vary with position as well as time, but our numerical examples assume that every point in the foam has the same density at any given time. Our examples assume that viscosity is constant, but viscosity could be a function of time in order to represent curing phenomena by a chemical reaction [2, 24] or rheological changes as the foam expands [25]. Laminar flow is assumed, but transient and inertia effects are considered because the viscosity is very low at the start of molding.

For a compressible Newtonian fluid, the extra stress tensor is defined as

$$\boldsymbol{\tau} = \mu[(\nabla \mathbf{v}) + (\nabla \mathbf{v})^{\mathrm{T}}] + \lambda(\nabla \cdot \mathbf{v})\mathbf{I} \tag{4}$$

where a superscript T indicates the transpose of tensor, μ is the shear viscosity and λ is the dilatational viscosity. We use $\lambda = -2\mu/3$, following Stokes' hypothesis [26]. Now the governing equations for the foam fluid are

$$\nabla \cdot \mathbf{v} = -\frac{1}{\rho}\left[\frac{\partial \rho}{\partial t} + \mathbf{v} \cdot (\nabla \rho)\right] \tag{5}$$

$$\frac{\partial}{\partial t}(\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = -\nabla p + \mu \nabla^2 \mathbf{v} + \frac{1}{3}\mu \nabla(\nabla \cdot \mathbf{v}) + \rho \mathbf{g} \tag{6}$$

The third term on the right-hand side of Equation (6) represents a viscous stress from density variation, and does not appear for the incompressible fluid.

For solving the empty domain which is actually filled with air, some additional assumptions are needed. The air is assumed to be an incompressible Newtonian fluid. Laminar flow is assumed. The density of air is about $1\,\mathrm{kg/m^3}$ and its viscosity is about $10^{-5}\,\mathrm{Pa\,s}$. If the air is replaced with a fictitious fluid whose viscosity is approximately $1\,\mathrm{Pa\,s}$, then transient, inertia, and gravity effects can be neglected in the fictitious fluid, but the solution in the foam fluid

is essentially unchanged [27]. Then the governing equations for the empty part reduce to

$$(\nabla \cdot \mathbf{v}) = 0 \tag{7}$$

$$-\nabla p + \mu \nabla^2 \mathbf{v} = 0 \tag{8}$$

In order to track the interface between foam and air, the fractional volume function $f(\mathbf{x}, t)$ is defined such that [28]

$$f(\mathbf{x}, t) = \begin{cases} 1 & \text{for the point } (\mathbf{x}, t) \text{ filled with foam} \\ 0 & \text{for the point } (\mathbf{x}, t) \text{ filled with air} \end{cases} \tag{9}$$

This function is governed by a scalar advection equation,

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f = 0 \tag{10}$$

The interface is located in the cells whose average value of f lies between 0 and 1. When necessary, the approximate location of the interface is reconstructed by drawing the contour of $f = 0.5$. This is strictly a post-processing operation, and has no effect on the calculation itself.

For partially filled cells, material properties such as viscosity and density are interpolated using

$$\mu = \mu_{\text{foam}} f + \mu_{\text{air}}(1 - f) \tag{11}$$

$$\rho = \rho_{\text{foam}} f + \rho_{\text{air}}(1 - f) \tag{12}$$

In our examples the mold does not have inlets, because some portion of the mold is assumed to be filled with un-expanded foams at the beginning of molding. The initial velocity is set to zero at every point within the mold and the initial pressure has a hydrostatic distribution. A no-slip condition is applied between the mold wall and the foam. On the other hand, the air is assumed to be free to leave or enter the mold, so the boundary between the mold wall and air is traction-free. The boundary condition at the mold wall must change as the mold fills. The change is made according to the fractional volume function, as follows [29, 30]:

$$\mathbf{v} = 0 \quad \text{where } f \geqslant f_c \tag{13}$$

$$\mathbf{n} \cdot \mathbf{T} = 0 \quad \text{where } f < f_c \tag{14}$$

where f_c is a critical fractional volume. Typically we use $f_c = 0.90$.

When solving mold filling with constant-density fluid, we need a boundary condition at the inlet to the mold. This boundary condition is typically either a constant flow rate with a parabolic velocity profile, or a constant pressure with a fully developed flow condition.

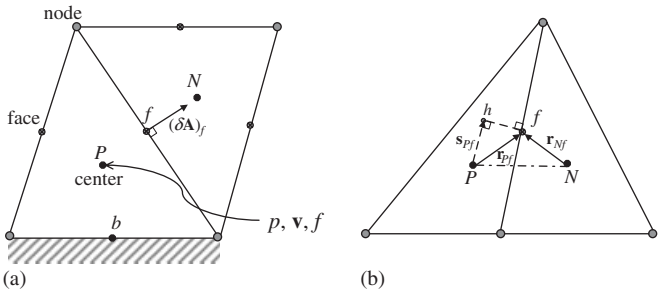

Figure 1. Cell-based control volumes.

NUMERICAL FORMULATION

Our numerical approach uses a pressure-based FVM scheme for unstructured meshes, and includes the SIMPLE algorithm with fluid compressibility. All variables are stored at the cell centres, and the momentum interpolation method is used to prevent checkerboard pressure modes. In order to capture the sharp fluid interface on an arbitrary mesh, an explicit high-resolution interface capturing method is used. The following formulations are developed for two-dimensional problems, but can easily be extended to three-dimensional problems. We will discuss only the numerical formulation of the governing equations for the foam, because the fictitious fluid can be solved using the same discretized equations as the foam but with different viscosity and density.

The computational domain, which covers the entire mold, is divided into unstructured control volumes, also called cells. Cells are either interior cells or boundary cells. Each interior cell consists of cell faces, which have outward normal area vectors; cell nodes, which are located at the vertices of the cells; and a cell centre, which lies at the mass centre of the cell, as shown in Figure 1. Each boundary cell is a line element located on the boundary of the domain, and has zero volume. The neighbour cells (N) of a specific cell (P) are the cells that share a common face (f) with cell P. For two-dimensional triangular meshes, each interior cell has three neighbour cells and three faces. All physical variables and their derivatives are stored at the cell centre, but face properties such as mass flux are calculated at the face centres. Derivatives of variables in the cells of unstructured meshes cannot be obtained easily. To evaluate derivatives, a linear reconstruction method [18] based on the divergence theorem is used, as shown in Appendix A.

In order to discretize the governing equations, we need to evaluate the diffusive flux of some variables at the cell faces. The method of calculating the gradient on an unstructured mesh is also used to calculate diffusive flux terms at the cell faces, as shown in Appendix B.

Continuity equation

Integrating the continuity equation (1) over the control volume and applying the divergence theorem and a fully implicit time discretization, we obtain the discretized continuity

equation as

$$a_P^{\rho} - a_P^{\rho^0} + \sum_f \rho_f F_f = 0 \tag{15}$$

$$a_P^{\rho} = \frac{\rho_P}{\Delta t}(\Delta V)_P \tag{16}$$

$$a_P^{\rho^0} = \frac{\rho_P^0}{\Delta t}(\Delta V)_P \tag{17}$$

$$F_f = (\delta \mathbf{A})_f \cdot \mathbf{v}_f \tag{18}$$

Here a superscript 0 indicates a value at the previous time step, Δt is the time interval, $(\Delta V)_P$ is the volume of cell P, $(\delta \mathbf{A})_f$ is the outward normal area vector of the face f, $\mathbf{v}_f$ is the velocity at the face f, and F_f represents the volume flux at the cell face f. In order to calculate face volume flux, the face velocity must be obtained using an appropriate interpolation scheme, because velocity is defined only at cell centre. This will be discussed later. From another version of the continuity equation (5), we can also obtain the divergence of the velocity in the cell P, which represents the density variation.

$$\sum_f F_f = (\nabla \cdot \mathbf{v})_P (\Delta V)_P = -\frac{1}{\rho_P}\left[a_P^{\rho} - a_P^{\rho^0} + \mathbf{v}_P \cdot (\nabla \rho)_P (\Delta V)_P\right] \tag{19}$$

If the density change is known, the divergence of velocity can be calculated using this equation.

Momentum equation

Integrating the momentum equation (6) over the control volume and applying the divergence theorem gives

$$\int_P \frac{\partial \rho \mathbf{v}}{\partial t}\,\mathrm{d}V + \sum_f (\delta \mathbf{A}) \cdot (\rho \mathbf{v}\mathbf{v} - \mu \nabla \mathbf{v}) = \int_P \nabla p\,\mathrm{d}V + \frac{1}{3}\int_P \mu \nabla (\nabla \cdot \mathbf{v})\,\mathrm{d}V + \int_P \rho \mathbf{g}\,\mathrm{d}V \tag{20}$$

We now focus on the x-direction component of the momentum equation, which is

$$\int_P \frac{\partial \rho u}{\partial t}\,\mathrm{d}V + \sum_f (\delta \mathbf{A}) \cdot (\rho \mathbf{v} u - \mu \nabla u) = \int_P \frac{\partial p}{\partial x}\,\mathrm{d}V + \frac{1}{3}\int_P \mu \frac{\partial}{\partial x}(\nabla \cdot \mathbf{v})\,\mathrm{d}V + \int_P \rho g_x\,\mathrm{d}V \tag{21}$$

Here u is the x-direction velocity and g_x is the x component of the gravitational acceleration. Using a fully implicit time derivative and discretizing the diffusion term, we obtain a discrete momentum equation as follows:

$$\begin{aligned} &a_P^{\rho} u_P - a_P^{\rho^0} u_P^0 + \sum_f \rho_f F_f u_f + \sum_f (D_f (u_P - u_N) - S_{f,u}^{\mathrm{non}}) \\ &\quad = -\left(\frac{\partial p}{\partial x}\right)_P (\Delta V)_P + \frac{1}{3}\mu_P \left(\frac{\partial}{\partial x}(\nabla \cdot \mathbf{v})\right)_P (\Delta V)_P + \rho_P g_x (\Delta V)_P \end{aligned} \tag{22}$$

$$D_f = \mu_f \frac{(\delta \mathbf{A})_f \cdot (\delta \mathbf{A})_f}{(\delta V)_f} \tag{23}$$

$$S_{f,u}^{\text{non}} = \mu_f \left[(\delta \mathbf{A})_f \cdot (\overline{\nabla u})_f - \frac{(\delta \mathbf{A})_f \cdot (\delta \mathbf{A})_f}{(\delta V)_f} (\overline{\nabla u})_f \cdot \mathbf{r}_{PN} \right] \tag{24}$$

where P represents the present cell, N is a neighbour cell, and f is the face between P and N. D_f is the diffusion coefficient for viscous stress that includes geometric information, and $S_{f,u}^{\text{non}}$ is a secondary diffusion term for x-direction viscous force at face f. The bar over $(\nabla u)_f$ represents a length-inverse weighted average between cells P and N, as defined in Equation (A2).

Using the discretized continuity equation (15), the transient terms in Equation (22) can be changed into the following relation:

$$a_P^{\rho} u_P - a_P^{\rho^0} u_P^0 = a_P^{\rho^0} (u_P - u_P^0) - \sum_f \rho_f F_f u_P \tag{25}$$

The x-direction total momentum flux at face f, which consists of convective and diffusive fluxes, must be modified using a differencing scheme in order to obtain a converged solution.

$$\rho_f F_f u_f + D_f (u_P - u_N) - S_{f,u}^{\text{non}} = \rho_f F_f u_P + a_N (u_P - u_N) - S_{f,u}^{\text{non}} \tag{26}$$

$$a_N = D_f A(|P_f|) + \max\{0, -\rho_f F_f\} \tag{27}$$

$$P_f = \rho_f \frac{F_f}{D_f} \tag{28}$$

where P_f is the Peclet number at face f. $A(|P_f|)$ would be unity for the upwind differencing scheme. We use a power-law differencing scheme, for which the function is

$$A(|P_f|) = \max\{0, (1 - 0.1|P_f|)^5\} \tag{29}$$

If Equations (25) and (26) are substituted into Equation (22), the final discretized equation for x momentum is found to be

$$a_P u_P = \sum_f a_N u_N - \left(\frac{\partial p}{\partial x} \right)_P (\Delta V)_P + S_{P,u}^{\text{non}} \tag{30}$$

$$a_P = a_P^{\rho^0} + \sum_f a_N \tag{31}$$

$$S_{P,u}^{\text{non}} = a_P^{\rho^0} u_P^0 + \sum_f S_{f,u}^{\text{non}} + \frac{1}{3} \mu_P \left(\frac{\partial}{\partial x} (\nabla \cdot \mathbf{v}) \right)_P (\Delta V)_P + \rho_P g_x (\Delta V)_P \tag{32}$$

The y- and z-direction momentum equations are discretized in the same manner. The final discretized momentum equation can be written in vector form as follows:

$$a_P \mathbf{v}_P = \sum_f a_N \mathbf{v}_N - (\nabla p)_P (\Delta V)_P + \mathbf{S}_P^{\text{non}} \tag{33}$$

$$\mathbf{S}_P^{\text{non}} = a_P^{\rho^0} \mathbf{v}_P^0 + \sum_f \mathbf{S}_f^{\text{non}} + \frac{1}{3} \mu_P (\nabla(\nabla \cdot \mathbf{v}))_P (\Delta V)_P + \rho_P \mathbf{g} (\Delta V)_P \tag{34}$$

The source term $\mathbf{S}_P^{\text{non}}$ contains the momentum of the previous time step, the secondary diffusion of momentum, the viscous force from density change, and gravity force.

These discretized momentum equations can be under-relaxed to obtain converged velocity solutions, using

$$\mathbf{v}^{\text{new}} = \alpha_{\text{v}} \mathbf{v} + (1 - \alpha_{\text{v}}) \mathbf{v}^{(n-1)} \tag{35}$$

where α_{v} is a velocity under-relaxation factor with a value between 0 and 1, $\mathbf{v}$ is the corrected velocity vector without relaxation calculated using Equation (33), and $\mathbf{v}^{(n-1)}$ represents the velocity from the previous iteration. After some algebra, Equations (33) and (35) can be combined to give

$$\frac{a_P}{\alpha_{\text{v}}} \mathbf{v}_P = \sum_f a_N \mathbf{v}_N - (\nabla p)_P (\Delta V)_P + \mathbf{S}_P^{\text{non}} + \left[(1 - \alpha_{\text{v}}) \frac{a_P}{\alpha_{\text{v}}} \right] \mathbf{v}_P^{(n-1)} \tag{36}$$

This is the equation used for velocity iterations in our code.

Boundary conditions

Boundary conditions for the momentum equations include a no-slip condition where the foam contacts the wall and a no-traction condition where the mold edge is in contact with air. The diffusive momentum flux between an interior cell and a boundary cell can be written as (see Appendix B):

$$\mu_b (\delta \mathbf{A})_b \cdot (\nabla \mathbf{v})_b = D_b (\mathbf{v}_b - \mathbf{v}_P) + \mathbf{S}_b^{\text{non}} \tag{37}$$

For a no-slip condition, velocity and volume flux are set to zero. Thus, the diffusive flux of momentum can be easily calculated and applied in the overall momentum equation. For a traction-free condition note that, in our algorithm, pressure and the divergence of velocity $(\nabla \cdot \mathbf{v})$ are treated as volume properties that are integrated over each control volume. So, at a traction-free boundary the diffusive flux of momentum is set to zero and the boundary velocity is updated as follows:

$$\mathbf{v}_b = \mathbf{v}_h = \mathbf{v}_P + (\nabla \mathbf{v})_P^{\text{T}} \cdot \mathbf{s}_{Pf} \tag{38}$$

The point h lies on the line normal to the face f and passing through the centre of the face, whose normal distance to the face is the same as the cell centre P (see Figure 1(b)). $\mathbf{s}_{Pf}$ is the vector from P to h point.

A fully developed flow condition at the inlet boundary is also treated by setting the diffusive flux of momentum to zero. For a constant flow rate condition at the inlet boundary, the velocity is already known, so the diffusive flux of momentum can be calculated using Equation (37).

Pressure-correction equation

If the pressure gradient is known, Equation (36) can be easily solved. However, the pressure must also be calculated as a part of the solution, so another equation for pressure is needed. Pressure is a primary variable, but does not have its own equation. Here the SIMPLE algorithm is employed to obtain a pressure-correction equation from the continuity equation [19]. For this purpose, the volume flux at each cell face must be known. But the face flux cannot be obtained easily, because velocity is stored only at the cell centre. So we need a method of evaluating flux at the cell face. Checkerboard pressure solutions will result if one uses simple linear interpolation [19]. In this study we use the momentum interpolation method (MIM), as first proposed by Rhie and Chow [31].

Using the discretized momentum equation, the volume flux F_f at each cell face can be calculated according to the following equation (see Appendix C):

$$F_f = (\delta\mathbf{A})_f \cdot \bar{\mathbf{v}}_f - \frac{(\delta\mathbf{A})_f \cdot (\delta\mathbf{A})_f}{\bar{a}_f}((p_N - p_P) - (\overline{\nabla p})_f \cdot \mathbf{r}_{PN}) \tag{39}$$

This formulation is the same as that of Mathur and Murthy [18]. When variations of velocity are large, which occur often near the wall, and the cell face centre does not lie on the line connecting the centres of neighbour cells as shown in Figure 1(b), a linear average of velocity at the cell face is not appropriate. So we modify the above equation as follows:

$$F_f = (\delta\mathbf{A})_f \cdot \tilde{\mathbf{v}}_f - \frac{(\delta\mathbf{A})_f \cdot (\delta\mathbf{A})_f}{\bar{a}_f}((p_N - p_P) - (\overline{\nabla p})_f \cdot \mathbf{r}_{PN}) \tag{40}$$

$$\tilde{\mathbf{v}}_f = \frac{\mathbf{v}_{f,P} + \mathbf{v}_{f,N}}{2} \tag{41}$$

$$\mathbf{v}_{f,P} = \mathbf{v}_P + (\nabla\mathbf{v})_P^{\mathrm{T}} \cdot \mathbf{r}_{Pf} \tag{42}$$

$$\mathbf{v}_{f,N} = \mathbf{v}_N + (\nabla\mathbf{v})_N^{\mathrm{T}} \cdot \mathbf{r}_{Nf} \tag{43}$$

where $\mathbf{r}_{Pf}$ and $\mathbf{r}_{Nf}$ are described in Figure 1(b).

If the volume flux correction term is defined by the following equation;

$$F_f' \equiv -\frac{(\delta\mathbf{A})_f \cdot (\delta\mathbf{A})_f}{\bar{a}_f}(p_N' - p_P') = \beta_f(p_P' - p_N') \tag{44}$$

a pressure-correction equation can be obtained after substituting the volume flux into the discretized continuity equation, Equation (19).

$$\beta_P p_P' = \sum_f \beta_f p_N' + b_P \tag{45}$$

$$\beta_P = \sum_f \beta_f \tag{46}$$

$$b_P = -\frac{1}{\rho_P}[a_P^{\rho} - a_P^{\rho^0} + \mathbf{v}_P \cdot (\nabla\rho)_P(\Delta V)_P] - \sum_f F_f^* \tag{47}$$

Here F_f^* is a volume flux which is calculated from the velocity that satisfies the momentum equation, and b_p is called the residue of continuity. After the pressure correction term is obtained, some variables are corrected using the result. Volume flux at the cell face, as well as pressure and velocity at the cell centre, should be corrected by using the following equations sequentially:

$$F_f = F_f^* + F_f' \tag{48}$$

$$p_P = p_P^* + \alpha_p p_P' \tag{49}$$

$$\mathbf{v}_P = \mathbf{v}_P^* - \frac{(\Delta V)_P}{a_P} (\nabla p')_P \tag{50}$$

where p^* and $\mathbf{v}^*$ are values after solving the momentum equation and α_p is an under-relaxation coefficient for the pressure.

At the wall boundary there is no flux, so the volume flux correction is equal to zero,

$$F_b' = \beta_b (p_P' - p_b') = 0 \tag{51}$$

and the wall boundary pressure is updated as follows:

$$p_b = p_h = p_P + (\nabla p)_P \cdot \mathbf{s}_{Pf} \tag{52}$$

At a free boundary, pressure is assumed to be zero, so the pressure correction is zero. Then the volume flux correction term can be represented as

$$F_b' = \beta_b p_P' \tag{53}$$

For a constant pressure condition at an inlet boundary, the pressure correction term is set to zero. A constant flow rate condition is treated by setting the volume flux correction to zero.

Solving fractional volume function

After solving for the distribution of velocity, the new position of the interface must be updated by solving the fractional volume equation. Ubbink and Issa [20] developed the CICSAM method for the accurate capturing of fluid interfaces on meshes of arbitrary topology. Their scheme is implicit, and theoretically has second-order temporal accuracy because the Crank–Nicolson scheme is used for temporal discretization. A similar method is used in our explicit scheme.

The conservative form of the volume fraction advection equation is

$$\frac{\partial f}{\partial t} + \nabla \cdot (f \mathbf{v}) = f(\nabla \cdot \mathbf{v}) \tag{54}$$

where the term on the right-hand side will be zero for the fictitious fluid. Integrating this equation over a control volume and applying the divergence theorem leads to

$$\int_P \frac{\partial f}{\partial t} \, \mathrm{d}V + \sum_f f_f (\delta \mathbf{A})_f \cdot \mathbf{v}_f = \int_P f(\nabla \cdot \mathbf{v}) \, \mathrm{d}V \tag{55}$$

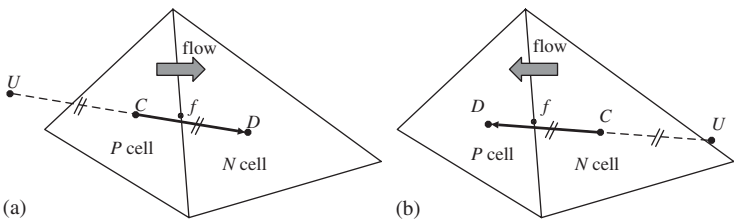

Figure 2. Designation of upwind, downwind, and centre cell according to the flow direction for arbitrary cell topology.

Using an explicit Euler scheme, and assuming that the variation of velocity is much smaller than that of volume fraction during the small time step so that the newest velocities can be used, we can write

$$\frac{f_P - f_P^0}{\Delta t}(\Delta V)_P = -\sum_f f_f^0(\delta \mathbf{A})_f \cdot \mathbf{v}_f + f_P^0(\nabla \cdot \mathbf{v})_P(\Delta V)_P \tag{56}$$

or, rearranging,

$$f_P = f_P^0(1 + (\nabla \cdot \mathbf{v})_P \Delta t) - \sum_f f_f^0 c_{f,P} \tag{57}$$

$$c_{f,P} = \frac{\Delta t}{(\Delta V)_P}(\delta \mathbf{A})_f \cdot \mathbf{v}_f = \frac{\Delta t}{(\Delta V)_P} F_f \tag{58}$$

where $c_{f,P}$ is called the face Courant number for face f in cell P.

It is very important to use appropriate face volume fraction values (f_f^0) at the previous time step in order to obtain a bounded and non-diffusive solution. The central differencing scheme has second-order accuracy, but gives an unbounded solution. The upwind scheme gives bounded results, but the solution is very diffusive. The downwind scheme gives a compressive, non-diffusive but unbounded solution. The CICSAM method chooses between two schemes, according to the flow direction and the interface orientation. These are the Hyper-C (HC) scheme and the ULTIMATE-QUICKEST (UQ) scheme.

The HC scheme [32] is known to be very compressive, because it turns every finite gradient into a step profile. This is very desirable property for calculating sharp interfaces when the interface is normal to the flow direction. This differencing scheme is best explained using normalized variables. The normalized value of f is defined as

$$\hat{f} \equiv \frac{f^0 - f_{\mathrm{U}}^0}{f_{\mathrm{D}}^0 - f_{\mathrm{U}}^0} \tag{59}$$

where subscripts U and D indicate the upwind and downwind values, defined according to the flow direction as shown in Figure 2. The downwind value is the value at the centre of the downwind cell, while the upwind value is taken at the same distance from the centre cells,

but in the opposite direction. The value of f at the upwind cell is approximated using

$$f_{\mathrm{U}}^{0} = f_{\mathrm{C}}^{0} - (\overline{\nabla f})_{f}^{0} \cdot \mathbf{r}_{\mathrm{CD}} \tag{60}$$

where the subscript C refers to the centre cell, f is the cell face between the centre and downwind cells, and $\mathbf{r}_{\mathrm{CD}}$ is the vector connecting the centre and downwind cells. This approximation does not guarantee a bounded value, so it is necessary to bound the result with the known value of f.

$$f_{\mathrm{U}}^{0} = \min\{\max\{f_{\mathrm{U}}^{0}, f_{\min}\}, f_{\max}\} \tag{61}$$

In our study, $f_{\min}$ and $f_{\max}$ are zero and one, respectively. Now the HC scheme can be represented as follows:

$$\hat{f}_{\mathrm{HC}} = \begin{cases} \min\left\{1, \dfrac{\hat{f}_{\mathrm{C}}}{c}\right\} & \text{when } 0 \leqslant \hat{f}_{\mathrm{C}} \leqslant 1 \\ \hat{f}_{\mathrm{C}} & \text{when } \hat{f}_{\mathrm{C}} < 0 \text{ or } \hat{f}_{\mathrm{C}} > 1 \end{cases} \tag{62}$$

$$c = \sum_{f} |c_{f,\mathrm{C}}| \tag{63}$$

where c is the Courant number of the centre cell.

The UQ method [32] is a third-order differencing scheme for the convection equation that has a good performance, but is diffusive for sharp interfaces. So it is used when the interface is tangential to the flow direction. The UQ scheme can be represented as follows [20]:

$$\hat{f}_{\mathrm{UQ}} = \begin{cases} \min\left\{\dfrac{8c\hat{f}_{\mathrm{C}} + (1-c)(6\hat{f}_{\mathrm{C}} + 3)}{8}, \hat{f}_{\mathrm{HC}}\right\} & \text{when } 0 \leqslant \hat{f}_{\mathrm{C}} \leqslant 1 \\ \hat{f}_{\mathrm{C}} & \text{when } \hat{f}_{\mathrm{C}} < 0 \text{ or } \hat{f}_{\mathrm{C}} > 1 \end{cases} \tag{64}$$

CICSAM switches smoothly these two schemes using the weighting factor $0 \leqslant \gamma_f \leqslant 1$,

$$\hat{f}_{f} = \gamma_f \hat{f}_{\mathrm{HC}} + (1 - \gamma_f)\hat{f}_{\mathrm{UQ}} \tag{65}$$

where γ_f is calculated based on the angle θ_f between the interface normal and the flow direction,

$$\gamma_f = \min\left\{k_\gamma \frac{\cos(2\theta_f) + 1}{2}, 1\right\} \tag{66}$$

$$\theta_f = \arccos\left|\frac{(\nabla f)_{\mathrm{C}}^{0} \cdot \mathbf{v}_{\mathrm{C}}}{|(\nabla f)_{\mathrm{C}}^{0}||\mathbf{v}_{\mathrm{C}}|}\right| \tag{67}$$

Here $k_\gamma \geqslant 0$ is a constant introduced to control the dominance of the different schemes. In our case this value is unity, which is recommended by Ubbink and Issa [20], and $\mathbf{v}_{\mathrm{C}}$ is the velocity vector at the centre cell. Ubbink and Issa [20] used $\mathbf{r}_{\mathrm{CD}}$ in place of $\mathbf{v}_{\mathrm{C}}$, but we obtained less

mesh dependence using Equation (67). After the normalized face volume fraction is obtained, f_f^0 is reconstructed as follows:

$$f_f^0 = \hat{f}_f(f_D^0 - f_U^0) + f_U^0 \tag{68}$$

After finding all face values in this way, the cell values of fractional volume function are updated explicitly according to Equation (57).

The Courant number must be less than unity in order to obtain converged solutions, because an explicit scheme is used. In our algorithm, the time step is chosen to keep the maximum Courant number less than 0.5. Courant numbers at every cell are different because velocity and cell volume vary from cell to cell. For this reason, the newly calculated volume fractions of some cells can occasionally be greater than unity. This excess volume fraction is distributed into downwind neighbour cells according to the ratio of face Courant numbers. However, if a downwind neighbour cell is boundary cell, it does not receive any of this excess volume fraction.

At the boundary, the fractional volume function f is calculated as

$$f_b = f_P + (\nabla f)_P \cdot \mathbf{r}_{Pb} \tag{69}$$

where $\mathbf{r}_{Pb}$ is the vector from P to point b (see Figure 1(a)). At both walls and free boundaries, the face flux of fractional volume is set to zero to prevent mass loss. At the inlet boundary f is always set to unity.

Overall solution procedure

The solution procedure for predicting mold filling of the foam is as follows:

1. Initialize fractional volume function f at all cells in the mold and set the boundary conditions.
2. Initialize pressure and the velocity vector at all interior and boundary cells. Compute the face volume flux F_f.
3. Calculate density and viscosity of the foam using the given functions of time, and then obtain the overall density and viscosity in the mold according to the fractional volume function.
4. Solve the momentum equation, Equation (36), with current values of the pressure p^* and the face volume flux.
5. Update face volume flux F_f^* using the newly calculated velocity $\mathbf{v}^*$.
6. Obtain the pressure correction p' by solving the pressure correction equation, Equation (45).
7. Correct the volume flux, pressure, and velocity using Equations (48)–(50).
8. Repeat steps 4–7 until flow field solution is converged.
9. Solve the fractional volume equation, Equation (57).
10. Update the boundary condition according the newly calculated fraction volume function.
11. Advance the time step and go step 3.

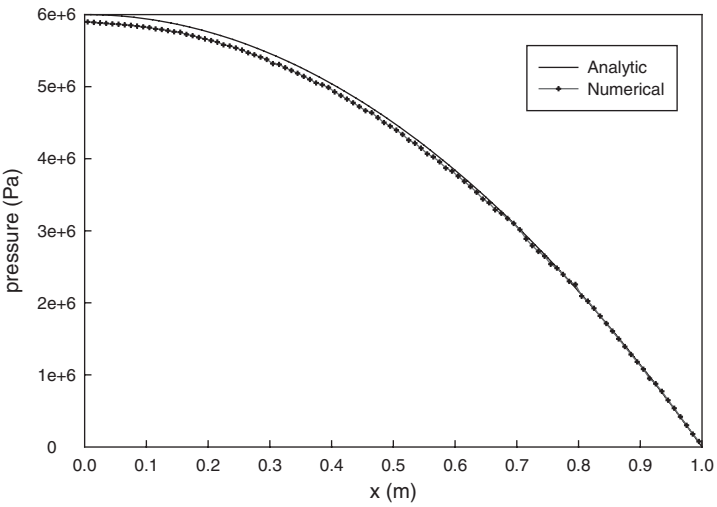

Figure 3. Pressure profiles for expanding foam in the thin slit. $\mu = 10\,000$ Pa s, $\rho_0 = 100\,\mathrm{kg/m^3}$, and $\kappa = 1\,\mathrm{s^{-1}}$.

RESULTS AND DISCUSSION

Foam flow in a slit

As a test problem, foam flow in a rectangular slit was solved. This problem has an approximate analytic solution for the case of creeping flow, which is developed in Appendix D. The length L of the slit used for numerical calculations is 1 m and the height $2H$ is 0.1 m. The mesh of this slit is unstructured and has 2170 elements and 1196 nodes. The slit is initially filled completely with expanding foam whose density decays exponentially with time, as given in Equation (D1). The right-hand side of the slit has a free surface boundary condition while the other sides are walls. In this case, the viscosity is 10 000 Pa s, the initial density is $100\,\mathrm{kg/m^3}$, and the rate of exponential decay κ, which is defined in Equation (D1), is $1\,\mathrm{s^{-1}}$. This makes the Reynolds number 2.5×10^{-5}, so inertia is negligible. We also neglect gravity.

Figure 3 shows the pressure profiles of the analytical and numerical solutions for expanding foam. The numerical data is taken along the lower wall, $y = -H$. The maximum pressure of the analytical solution at $x = 0$ is 6.00×10^6 Pa, while the numerical solution gives $p = 5.90 \times 10^6$ Pa. The reason is that, near the left wall, the x-direction velocity is so small that it is not the dominant velocity component. Therefore, the assumptions that are used for deriving the analytical solution are invalid. The numerical velocities are in excellent agreement with the analytical solution except at $x < H$ and $x > L - H$, where the analytical solution is not accurate.

Fountain flow in a rectangular cavity

In order to verify the method of interface capturing and compare the flow behaviours of an expanding foam fluid and a fluid with constant density, we solved the advance of a moving flow front in a two-dimensional rectangular cavity. It is important to investigate the evolution

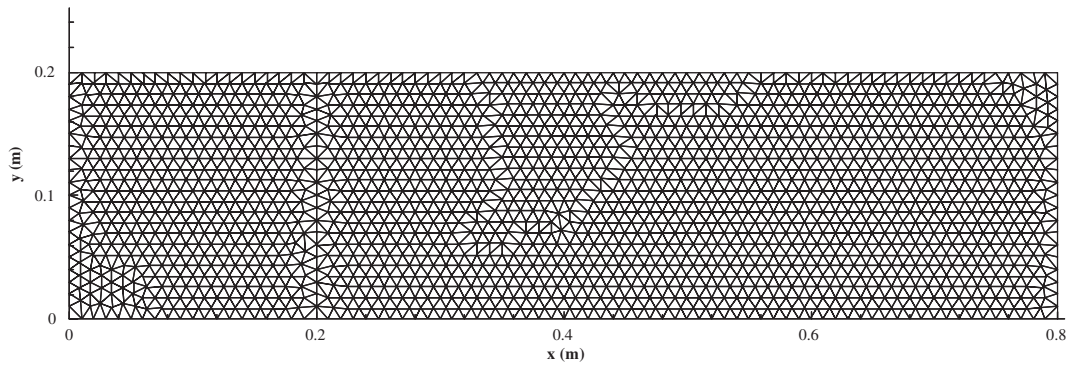

Figure 4. The unstructured triangular mesh of the rectangular cavity for numerical calculation of the fountain flow. 3584 elements and 1839 nodes.

of the flow front shape because of fountain flow effects. Fountain flow refers to movement of the fluid particles from the central region toward the wall, close to the flow front [33, 34]. We consider two cases: a Newtonian fluid with constant density, and a Newtonian foam with exponentially decreasing density.

Figure 4 shows the rectangular cavity, with an unstructured triangular mesh of 3584 elements used for numerical calculations. The length of this cavity is 0.8 m and the height is 0.2 m. The part of the cavity with $0 \leqslant x \leqslant 0.2$ is initially filled with fluid. The initial pressure and velocity are set to zero, and we neglect gravity. For the constant-density fluid, the left-hand side of the cavity is an inlet gate where a constant flow rate condition is applied. There the x-direction velocity is

$$u = 25[0.01 - (0.1 - y)^2] \ (\mathrm{m/s}) \tag{70}$$

and the other velocity component is set to zero. The viscosity of the fluid is 1000 Pa s and its density is $1000\,\mathrm{kg/m^3}$.

For the expanding foam, whose density decays exponentially with time, there is no inlet gate. This is the same problem as free-rise foaming without gravity. The viscosity is 1000 Pa s, the initial density is $\rho_0 = 1000\,\mathrm{kg/m^3}$, and the rate of exponential decay is $\kappa = 1\,\mathrm{s^{-1}}$ (see Equation (D1)). For both cases, the viscosity and density of the fictitious fluid are 1 Pa s and $1.23\,\mathrm{kg/m^3}$.

Figure 5 shows the profiles of the flow front at different times, defined as the contour where $f = 0.5$. Because the mesh is irregular, the profiles near the upper and lower wall exhibit some wiggles. But the overall shape of the flow front for the constant-density fluid agrees well with the results of Behrens *et al.* [35], even though our study is not an exact creeping flow because $Re = 0.025$. Behrens *et al.* studied the transient creeping fountain flow of an isothermal Newtonian fluid both in a circular tube and in a channel with parallel walls. For the experiments in a circular tube, a steady flow front shape was reached after the front had traveled about 1.5 radii, and at the steady state the axial distance between the fluid at the centreline and at the wall was 0.83 ± 0.04 radii. Using a numerical method that gave good results for the circular tube, they obtained a difference value of about 0.94 half-heights in a

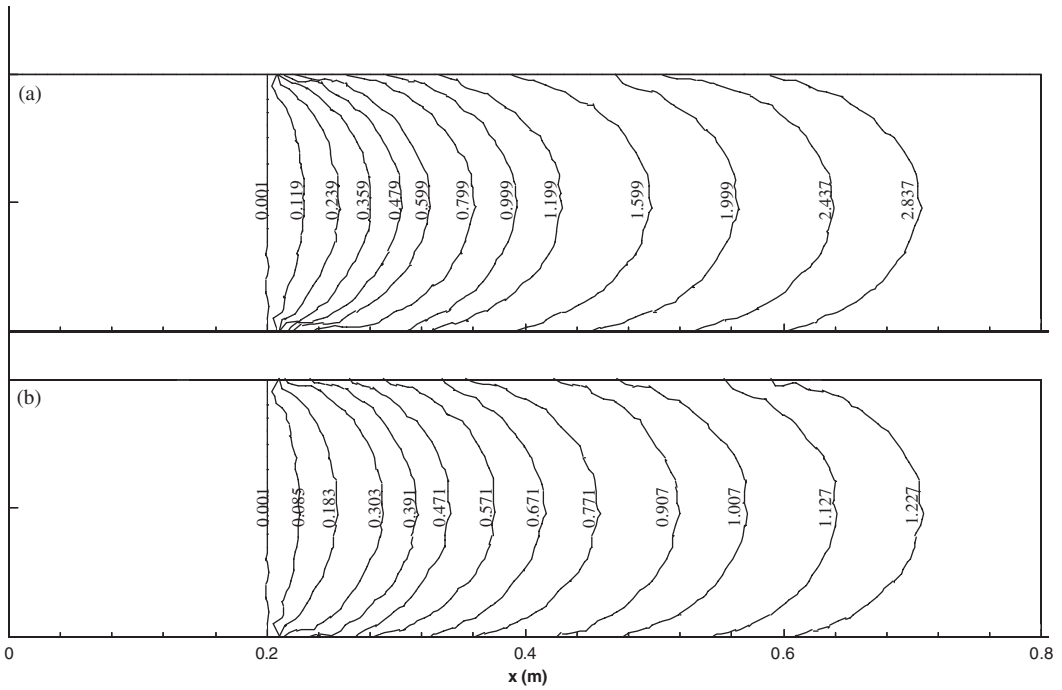

Figure 5. Profiles of the flow front ($f = 0.5$) at different times (s) for (a) constant-density fluid and (b) expanding foam in fountain flow.

parallel planar plate. In our case, a steady state is reached after the front travels about 2.0 half-heights and the difference is about 1.0 half-heights. We obtained similar results on coarse and/or structured meshes.

The flow fronts for the expanding foam are slightly flatter than those of the Newtonian fluid. A steady state is reached after the front travels about 1.6 half-heights, and the steady state difference is about 0.8 half-heights. This value increases slightly as flow front travels down the cavity. These results differ from those of Lefebvre and Keunings [16], who calculated a flow front that was almost flat except near the contact point. However, Lefebvre and Keunings were simulating free rising foam, with only a single wall constraining the motion, and they used a partial slip condition at the wall.

To understand the differences between the expanding foam and the constant-density fluid, we need to investigate the pressure and velocity fields. The pressure contours for the constant-density fluid and the expanding foam are shown in Figure 6. For the constant-density fluid the isobars are parallel to the inlet gate, except near the flow front where the shape of the pressure contours approaches the shape of flow front. For the expanding foam the pressure contours are more complicated. Near the left wall, the pressure is higher at the top and bottom than at the centre. However, near the flow front the pressure is higher at the centre than at the wall.

In both cases, there are small regions with negative pressure near the contact point, where the boundary condition changes from no-slip to traction-free. These negative pressures appear

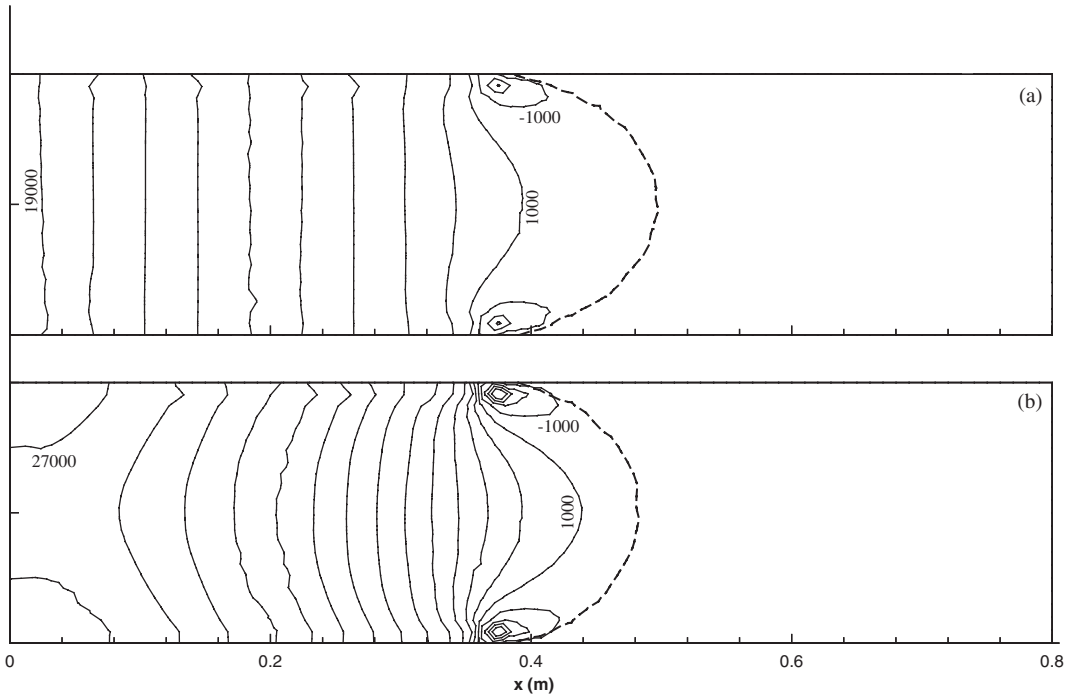

Figure 6. Pressure (Pa) contours in fountain flow for (a) constant-density fluid at $t = 1.60$ and (b) expanding foam at $t = 0.83$. Dotted line is the line where fractional volume $f = 0.5$.

in the interior domain because our algorithm forces the free boundary pressure to zero. The size of the area with negative pressure decreases, and its magnitude increases, as mesh size decreases. These kinds of phenomena do not appear when the flow front shape is a straight line. When the flow front shape is like a half circle, the fountain flow closely resembles a stick–slip flow. The stick–slip problem has a singularity at the wall where boundary condition changes [36], so it is difficult to solve that problem numerically. Georgiou *et al.* [37] solved the stick–slip problem using singular finite elements, whose shape functions embody the form of the singularity. But it is difficult to adopt their method in our algorithm, because the singular point moves with time, and FVM does not use shape functions. Instead, we allow the negative pressures in our solution, recognizing that these negative pressures rarely affect the velocity profile. If the pressure value is used for other physical calculations, such a pressure-dependent foam density, then the negative pressure values must be corrected.

Figure 7 shows the velocity vectors and streamlines near the flow front, for the constant-density fluid at $t = 1.60$ and the expanding foam at $t = 0.83$. The expanding foam has a more diverging velocity distribution than the constant-density fluid. Foam near the wall flows a little bit toward the centre because of volumetric expansion, and flows into the wall again beyond the flow front. The flow front shape of an expanding foam is less sharp, because the foam expands at every point, including the flow front tip. The expansion compels foam to flow away from the wall, altering the streamline and front shape.

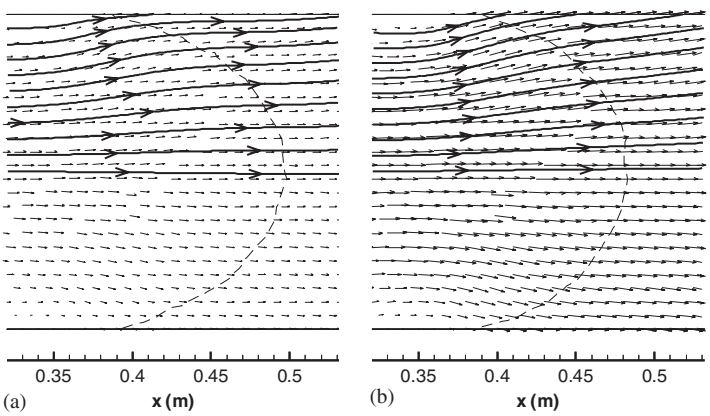

Figure 7. Velocity vectors and streamlines in fountain flow for (a) constant-density fluid at $t = 1.60$ and (b) expanding foam at $t = 0.83$. Dotted line is the line where fractional volume $f = 0.5$.

More differences between the constant-density fluid and expanding foam are found in the traces of fluid particles. In order to trace particles in our numerical method, a longer cavity than the above test is needed, so we simulated filling in the thin rectangular cavity with a length of $L = 2.5$ m and a height of $2H = 0.2$ m. Because of the symmetry of the problem, only half of the cavity is analyzed. The mesh uses structured triangles with 1250 elements. The other conditions are the same as the above test. Figure 8 shows the traces of some particles in the fountain flow. The normalized length ξ is defined as x/l, where l is the average distance of flow front from $x = 0$. The normalized height η is also defined as y/H, where H is the half height of the cavity and $y = 0$ is the midplane.

For the constant-density fluid, the average position l of the flow front increases linearly with time, because the inlet condition is a constant flow rate. In the internal flow region, where there is no effect of the fountain flow, the particles move forward parallel to the walls. But when the particles approach the flow front, they experience fountain flow. These particles go outside to the wall, and subsequently flow slowly. As shown in Figure 8(a), some traces move straight forward and then curve upward and finally go backward, when plotted in the ξ–η domain. Others traces move straight forward to a certain limiting curve, and do not advance further in the ξ–η coordinates.

Figure 8(c) shows analytical results for the movement of particles in the internal flow region. Because, for a constant-density fluid, the x-direction velocity profile is parabolic and the y-direction velocity is zero, material points with $\eta = \eta^* \equiv 1/\sqrt{3}$ have a local velocity that equals with an average velocity. When $\eta < \eta^*$, the velocity of the particles is greater than the velocity of the flow front. As time advances, these particles catch up with the flow front, near $\xi = 1$. But when $\eta > \eta^*$, the velocity of the particles is less than the velocity of the flow front. As time advances, the distance from these particles to the flow front increases, and the normalized length ξ approaches a constant value of $1.5(1 - \eta^2)$, which is the ratio between local velocity and flow front velocity. Thus, a particle with $\xi < 1.5(1 - \eta^2)$ will never reach the flow front, and its ξ will increase with time. However, particles with $\xi > 1.5(1 - \eta^2)$ have experienced the fountain flow, and their values of ξ decrease with time [38].

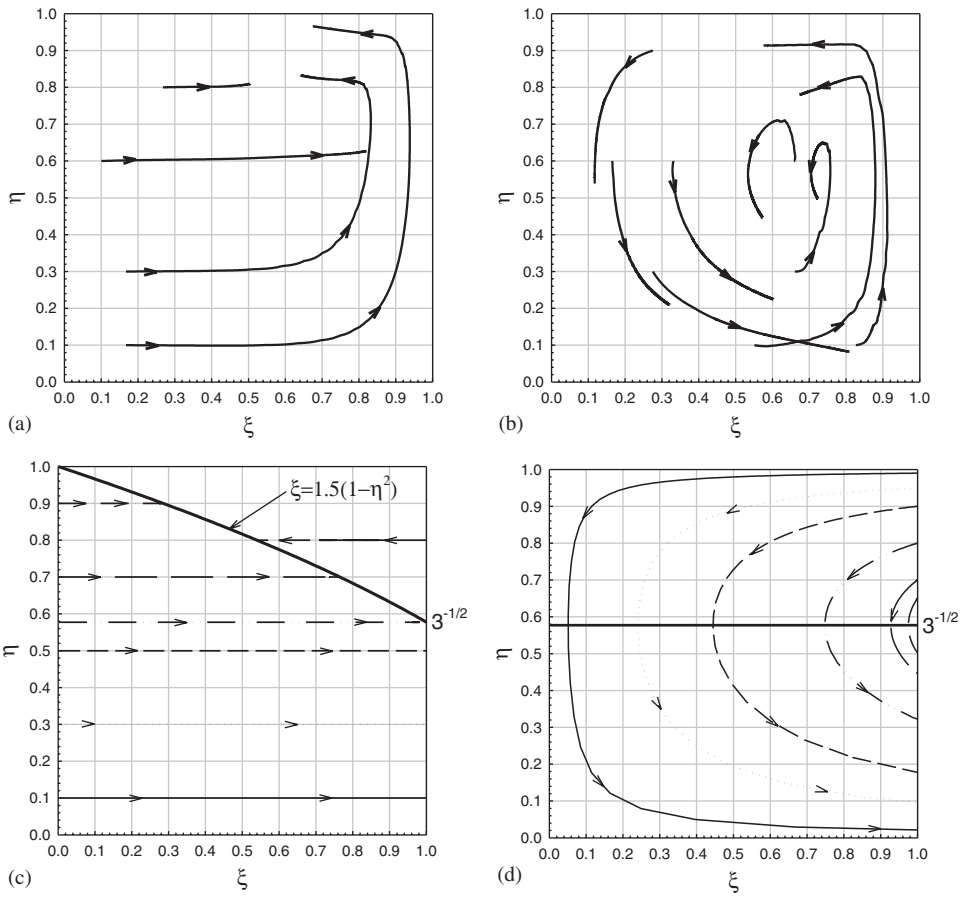

Figure 8. Traces of some particles in the fountain flow. Note that $\xi = x/l$ and $\eta = y/H$. (a) Numerical results of constant-density fluid, (b) Numerical results of expanding foam, (c) Analytical results of constant-density fluid in the internal flow region, (d) Analytical results of expanding foam in the internal flow region.

The movement of particles in the expanding foam is quite different. The average position l of the flow front increases exponentially with time,

$$l = l_0 \exp(\kappa t) \tag{71}$$

where l_0 is the initial position of the flow front and κ is the rate of exponential decay of density. In the internal region, every particle moves toward the midplane of the cavity. At the same time, each particle moves toward the flow front and, if the flow continues for enough time, wil1 catch up to the flow front. As the particle reaches the flow front, it experiences the fountain flow and moves out toward the wall, where the particles move more slowly in the x direction. As shown in Figure 8(b), the trace of the every particle circulates counter-clockwise in the ξ–η domain. If the flow continues for a large enough time, each particle will circulate

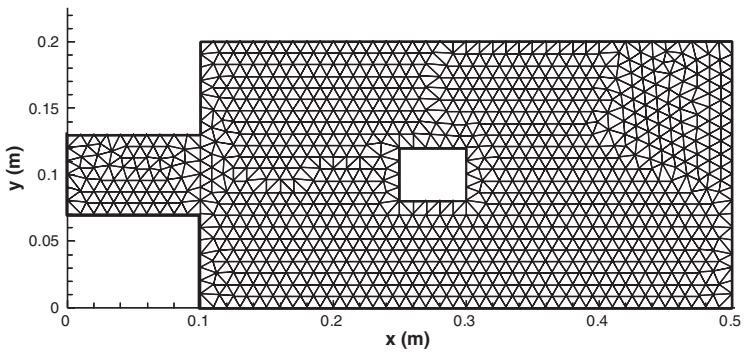

Figure 9. The unstructured triangular mesh of the mold with an inserted block. 1860 elements and 1009 nodes.

multiple times. Thus, every particle can experience the fountain flow many times, provided the cavity is enough long.

Figure 8(d) shows the particle traces in the internal flow region, based on the approximate analytic solution of expanding foam in the thin slit for the case of creeping flow (see Appendix D). If we integrate Eqs. (D13) and (D14), we can find the particle position as a function of time in this region. The normalized positions are

$$\xi = \xi_0 \eta_0^3 e^{-\kappa t} \left[1 + \left(\frac{1}{\eta_0^2} - 1 \right) e^{\kappa t} \right]^{3/2} \tag{72}$$

$$\eta = \left[1 + \left(\frac{1}{\eta_0^2} - 1 \right) e^{\kappa t} \right]^{-1/2} \tag{73}$$

where ξ_0 and η_0 are the normalized co-ordinates of the initial position of the particle. The normalized length ξ of every particle will become greater than 1 as time advances. That means every particle in the internal flow region will catch up to the flow front and experience the fountain flow. When $\eta < \eta^*$, the x-direction velocity of the particle is greater than flow front velocity, and ξ increases with time. But when $\eta > \eta^*$, the x-direction velocity of the particle is less than the flow front velocity, and ξ decreases with time.

Mold filling

As a final demonstration problem, filling of a complex mold geometry by an expanding foam is simulated. A two-dimensional mold with an insert was selected. Figure 9 shows the unstructured mesh of the mold, with 1860 elements and 1009 nodes. Foam is poured into the mold and expands to fill the mold. The left side of the mold, $0 \leqslant x \leqslant 0.1$, is initially filled with the foam. The initial pressure and velocity are set to zero, and we neglect gravity. The density of the foam decays exponentially with time as given in Equation (D1). The initial density is $\rho_0 = 1000\,\mathrm{kg/m^3}$ and the rate of exponential decay is $\kappa = 1\,\mathrm{s^{-1}}$. The viscosity of the foam is 10 000 Pa s. When we formulate the numerical algorithm, inertia and gravity effects

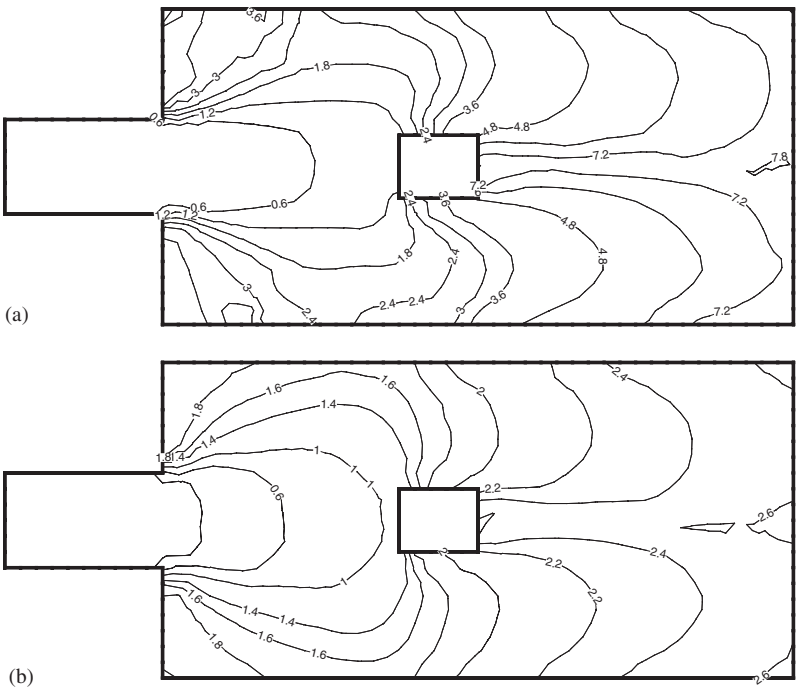

Figure 10. Fill time (s) contours of mold filling in (a) constant-density fluid and (b) expanding foam.

are taken into account because the viscosity of a polyurethane foam may vary from 0.1 Pa s to an infinite value. However, we consider only a high viscosity case in this example.

For the comparison, we also simulated the mold filling of the Newtonian fluid with constant density. The constant-density fluid initially fills the same part as the foam does initially. A constant flow rate condition is applied at the inlet gate, $x = 0$. This flow rate is $9 \times 10^{-3}\,\mathrm{m}^2/\mathrm{s}$, and the velocity is assumed to have a parabolic profile. The density is $1000\,\mathrm{kg/m}^3$ and the viscosity is 10 000 Pa s. For the both cases, the viscosity and density of the fictitious fluid are 1 Pa s and $1.23\,\mathrm{kg/m}^3$.

Figure 10 shows the fill time contours. The filling time of the mold is 7.824 s for the constant-density fluid and is 2.612 s for the expanding foam. These values are less than analytical filling times of 8.667 and 2.639 s, respectively, because some air is trapped behind the insert. Also, the cells near the boundary are not filled completely because the critical fractional volume value f_c, which is used for switching the boundary condition from traction-free to no-slip, is 0.9 in our case (see Equations (13) and (14)).

For the constant-density fluid, there are some wiggles in the left upper and lower corners. The mesh is too coarse to trace exactly the flow front when the flow front is almost parallel to the velocity. But for the expanding foam, the orientation of the flow front is almost normal to the flow direction in the left corner, so there are no wiggles. In both cases, there are weldlines and some air entrapment behind the insert. Note that our simulation allows air to pass freely through the mold walls. Thus, any residual air in the simulation at the end of mold filling is

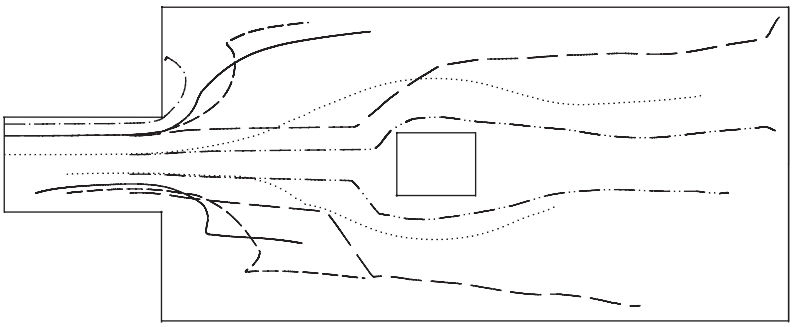

Figure 11. Traces of some particles during the mold filling. The upper half of the mold is for the constant-density fluid and the lower half is for the expanding foam.

due to the finite size of the mesh. The amount of this residual air will decrease as the mesh is refined.

Figure 11 shows the paths of some fluid particles during the mold filling. Paths in the upper half of the mold are for the constant-density fluid and the paths in the lower half are for the expanding foam. For the constant-density fluid, there is a vortex in the left corner of the mold after the corner fills. Once the particles go into this vortex region, they remain there during the remainder of mold filling. For expanding foam there is no vortex in the mold, so particles which go into the left corner of the mold eventually flow to the centre of the mold, due to foam expansion after the corner fills.

CONCLUSIONS

Mold filling with a variable-density fluid that fills a mold by self-expansion was predicted by numerical simulation. A pressure-based finite volume method and an explicit high-resolution interface capturing method for unstructured meshes with treatment of fluid compressibility was successfully applied to this moving boundary problem. Expanding foam, which is modelled as a Newtonian fluid with temporal density change, showed fundamentally different flow behaviours compared to a constant-density fluid. The shape of the flow front in fountain flow is flatter for expanding foam than for a constant-density fluid. Also, fluid particles of expanding foam flow follow more complicated paths in the mold. This study will be a basis for a non-isothermal and three-dimensional numerical simulation of foam reaction injection molding which will consider bubble growth, reaction kinetics, and foam rheology.

APPENDIX A: EVALUATION OF SCALAR QUANTITIES AT CELL FACES AND GRADIENTS AT CELL CENTRES

When using an unstructured cell-based control volume as shown in Figure A1, it is necessary to evaluate primary quantities at the cell faces and their gradients at the cell centres. These two evaluations are coupled with each other.

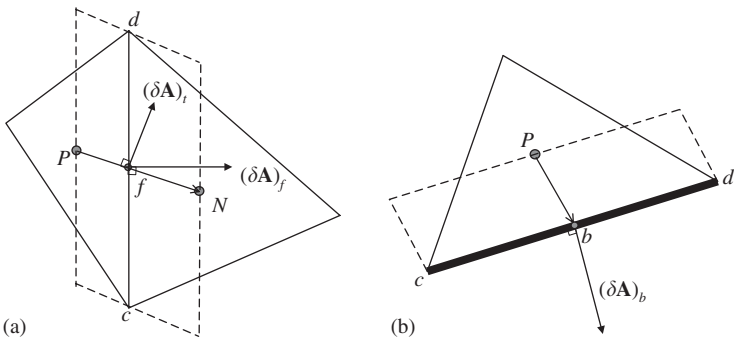

Figure A1. Calculation of gradient using finite concept method. Neighbour cell is (a) interior cell, (b) boundary cell.

To evaluate the gradient of a scalar quantity at the cell centre, a linear reconstruction method [18] based on the divergence theorem can be used. The gradient of any field variable ϕ at the cell centre can be approximated as

$$(\nabla\phi)_P = \frac{1}{(\Delta V)_P}\sum_f (\delta\mathbf{A})_f \phi_f \tag{A1}$$

where P represents the given cell, f denotes the faces of cell P, $(\Delta V)_P$ is the volume of cell P, and $(\delta\mathbf{A})_f$ is the outward area vector normal to face f as shown in Figure A1(a). Because physical variables are only stored at cell centres, the cell face values ϕ_f must be acquired from cell centre values using some appropriate method.

The simplest way to obtain values at the cell face is to use linear interpolation from the neighboring cells. This gives the face value of ϕ as

$$\phi_f = \bar{\phi}_f \equiv \frac{l_P\phi_N + l_N\phi_P}{l_P + l_N} \tag{A2}$$

where the bar indicates a length-inverse weighted linear interpolation, N is the neighbour cell sharing face f with cell P, l_P is the distance between P and f, and l_N is the distance between N and f. This method has second-order accuracy as long as the centre of the cell face lies at the mid-point between the centres of cells P and N. If not, and the gradient varies between the cells, then the accuracy of this method drops rapidly. This case occasionally occurs when finding the velocity profile on an unstructured mesh near the wall.

In order to increase the order of accuracy, values at the cell face can be obtained from a combination of the value and the gradient stored at the cell centre as follows:

$$\phi_f = \phi_P + (\nabla\phi)_P \cdot \mathbf{r}_{Pf} \tag{A3}$$

where $(\nabla\phi)_P$ is calculated from a combination of Equations (A1) and (A2) and $\mathbf{r}_{Pf}$ is the vector from the cell centre P to the cell face f. The face value ϕ_f must lie between the maximum and minimum value of neighbour cells of cell P.

$$\phi_f = \min\{\max\{\phi_{\min}, \phi_f\}, \phi_{\max}\} \tag{A4}$$

Because each cell face is shared by two neighbour cells, Mathur and Murthy [18] averaged the two values that are calculated from Equation (A3) from the two cells sharing the face.

This approach can be generalized to evaluate a scalar quantity at the cell face and its gradient at the cell centre iteratively as follows:

$$(\nabla\phi)_P^n = \frac{1}{(\Delta V)_P}\sum_f (\delta\mathbf{A})_f \hat{\phi}_f^n$$

$$\hat{\phi}_f^n = \frac{\phi_{f,P}^n + \phi_{f,N}^n}{2} \tag{A5}$$

$$\phi_{f,P}^n = \phi_P + (\nabla\phi)_P^{n-1}\cdot\mathbf{r}_{Pf} \quad \phi_{f,N}^n = \phi_N + (\nabla\phi)_N^{n-1}\cdot\mathbf{r}_{Nf}$$

where the superscript indicates an iteration step, and $n \geqslant 2$. The first iteration values are found using

$$(\nabla\phi)_P^1 = \frac{1}{(\Delta V)_P}\sum_f (\delta\mathbf{A})_f \hat{\phi}_f^1$$

$$\hat{\phi}_f^1 = \bar{\phi}_f \tag{A6}$$

If n is two, this method is the same as Mathur and Murthy [18].

For regular meshes and a smooth profile of the variable, the gradient calculated using $n=1$ is very close to that using $n=2$. But for other cases, especially when the cell face is not coincident with the centre position of the two neighbour cells and gradient is large, many iterations are required to obtain accurate solutions. So we suggest a method that is a kind of over-estimation.

$$(\nabla\phi)_P = \frac{1}{(\Delta V)_P}\sum_f (\delta\mathbf{A})_f \phi_f$$

$$\phi_f = \frac{\phi_{f,P} + \phi_{f,N}}{2}$$

$$\phi_{f,P} = \phi_P + (\nabla\phi)_P^1\cdot(\mathbf{r}_{Pf} + \alpha\mathbf{s}_{Pf}) \tag{A7}$$

$$\phi_{f,N} = \phi_N + (\nabla\phi)_N^1\cdot(\mathbf{r}_{Nf} + \alpha\mathbf{s}_{Nf})$$

Here $\mathbf{s}_{Pf}$ (see Figure 1(b)) is vector from P to h, a point that lies on the line normal to the face and passing through the centre of the face, and is as close as possible to point P. α is an over-estimating factor. When α is zero, the above method is same as Mathur and Murthy's $n=2$ method. We use Equation (A7) with $\alpha = 0.75$, a value that was found by trial and error to give good solutions.

APPENDIX B: EVALUATION OF THE DIFFUSION TERM AT A CELL FACE

In order to evaluate the gradient at the cell face, we use the finite concept method (FCM) that was proposed by Kim [39]. The gradient of a variable ϕ at the cell face for the

two-dimensional case can be estimated as

$$(\nabla\phi)_f = \frac{1}{(\delta V)_f}[(\delta\mathbf{A})_f(\phi_N - \phi_P) + (\delta\mathbf{A})_t(\phi_d - \phi_c)] \tag{B1}$$

where

$$\begin{aligned} \mathbf{r}_{PN} &= \mathbf{r}_N - \mathbf{r}_P, \quad \mathbf{r}_{cd} = \mathbf{r}_d - \mathbf{r}_c \\ (\delta\mathbf{A})_f &= \mathbf{r}_{cd} \times \mathbf{k}, \quad (\delta\mathbf{A})_t = \mathbf{k} \times \mathbf{r}_{PN} \\ (\delta V)_f &= (\mathbf{r}_{PN} \times \mathbf{r}_{cd}) \cdot \mathbf{k} \end{aligned} \tag{B2}$$

Here $\mathbf{r}$ is the position vector, $\mathbf{k}$ is the third-direction unit vector, and the subscripts indicate the position as defined in Figure A1(a). $(\delta\mathbf{A})_f$ is the outward area vector normal to face f in cell P, and $(\delta\mathbf{A})_t$ is the tangential area vector that is normal to vector $\mathbf{r}_{PN}$. $(\delta V)_f$ is a virtual control volume at the cell face f that is described by the dotted line in Figure A1. If the centre of the cell face has the same position as the mid-point between the centres of its neighbour cells, this method has a second-order accuracy. This representation does not contain vectors constrained by any particular co-ordinate system.

Using the gradient calculated this way, the diffusion term at the cell face can be calculated as

$$\Gamma_f(\delta\mathbf{A})_f \cdot (\nabla\phi)_f = \Gamma_f \frac{(\delta\mathbf{A})_f \cdot (\delta\mathbf{A})_f}{(\delta V)_f}(\phi_N - \phi_P) + \Gamma_f \frac{(\delta\mathbf{A})_f \cdot (\delta\mathbf{A})_t}{(\delta V)_f}(\phi_d - \phi_c) \tag{B3}$$

Here Γ_f is the diffusion coefficient at the cell face that is calculated by appropriate interpolation of values between neighbour cells. This formulation is exactly same as that of Mathur and Murthy [18] in spite of the different description. The first term on the right-hand side of this equation represents primary diffusion, which occurs by the difference of values in the outward normal direction to the face. The second term, which is called cross-diffusion, is zero in orthogonal grid, that is when $(\delta\mathbf{A})_t$ is normal to $(\delta\mathbf{A})_f$. For an unstructured grid this term is generally not zero and can be significant, so its correct evaluation becomes very important. Mathur and Murthy [18] proposed a new formulation for evaluating the secondary diffusion term. To avoid the use of the face tangent area vectors $(\delta\mathbf{A})_t$ and face node values (ϕ_c, ϕ_d), Equation (B3) can be represented as follows:

$$\begin{aligned} \Gamma_f(\delta\mathbf{A})_f \cdot (\nabla\phi)_f = {} & \Gamma_f \frac{(\delta\mathbf{A})_f \cdot (\delta\mathbf{A})_f}{(\delta V)_f}(\phi_N - \phi_P) \\ & + \Gamma_f \left[(\delta\mathbf{A})_f \cdot (\overline{\nabla\phi})_f - \frac{(\delta\mathbf{A})_f \cdot (\delta\mathbf{A})_f}{(\delta V)_f}(\overline{\nabla\phi})_f \cdot \mathbf{r}_{PN}\right] \end{aligned} \tag{B4}$$

where the bar again means a length-inverse weighted average value between adjacent cells, and $\mathbf{r}_{PN}$ is defined in Equation (B2). The above equation can be written as

$$\Gamma_f(\delta\mathbf{A})_f \cdot (\nabla\phi)_f = D_f(\phi_N - \phi_P) + S_f^{\text{non}} \tag{B5}$$

Here D_f contains the diffusion coefficient and geometric information. S_f^{non} is a secondary diffusion term that includes averaged gradients of the scalar quantity, which must be calculated using another procedure as described in Appendix A.

The diffusion term at the face between an interior cell and a boundary cell is calculated as follows:

$$\Gamma_b(\delta\mathbf{A})_b \cdot (\nabla\phi)_b = \Gamma_b \frac{(\delta\mathbf{A})_b \cdot (\delta\mathbf{A})_b}{(\delta V)_b}(\phi_b - \phi_P) + \Gamma_b \left[(\delta\mathbf{A})_b \cdot (\nabla\phi)_P - \frac{(\delta\mathbf{A})_b \cdot (\delta\mathbf{A})_b}{(\delta V)_b}(\nabla\phi)_P \cdot \mathbf{r}_{Pb}\right] \tag{B6}$$

where

$$\begin{aligned} \mathbf{r}_{Pb} &= \mathbf{r}_b - \mathbf{r}_p, \quad \mathbf{r}_{cd} = \mathbf{r}_d - \mathbf{r}_c \\ (\delta\mathbf{A})_b &= \mathbf{r}_{cd} \times \mathbf{k} \\ (\delta V)_b &= (\mathbf{r}_{Pb} \times \mathbf{r}_{cd}) \cdot \mathbf{k} \end{aligned} \tag{B7}$$

or

$$\Gamma_b(\delta\mathbf{A})_b \cdot (\nabla\phi)_b = D_b(\phi_b - \phi_P) + S_b^{\text{non}} \tag{B8}$$

where the subscripts indicate points defined in Figure A1(b).

APPENDIX C: DERIVATION OF THE FACE FLUX USING MIM

Because a finite volume method with cell centred storage is used, it is very important to obtain a good description of face flux in order to avoid checkerboard pressure modes. In this study, we use a method similar to the momentum interpolation method (MIM), which was first proposed by Rhie and Chow [31].

Using the discretized momentum equation, Equation (33), the velocity at each cell centre can be represented as

$$\mathbf{v}_P = \frac{\sum_f a_N \mathbf{v}_N + \mathbf{S}_P^{\text{non}}}{a_P} - \frac{(\Delta V)_P}{a_P}(\nabla p)_P = \mathbf{H}_P - \frac{(\Delta V)_P}{a_P}(\nabla p)_P \tag{C1}$$

If $\mathbf{H}_P$ and $1/a_P$ are assumed to vary linearly between cells, the average face velocity can be approximated as

$$\bar{\mathbf{v}}_f = \bar{\mathbf{H}}_f - \frac{(\delta V)_f}{\bar{a}_f}(\overline{\nabla p})_f \tag{C2}$$

where the bar means a length-inverse weighted average, and $(\delta V)_f$ is a virtual control volume at the cell face as defined at Appendix 2 and Figure A1(a). If the pressure gradient at the cell face is known, the face velocity can be written as

$$\mathbf{v}_f = \bar{\mathbf{H}}_f - \frac{(\delta V)_f}{\bar{a}_f}(\nabla p)_f \tag{C3}$$

The above two equations can be combined to eliminate $\bar{\mathbf{H}}_f$,

$$\mathbf{v}_f = \bar{\mathbf{v}}_f - \frac{(\delta V)_f}{\bar{a}_f}[(\nabla p)_f - (\overline{\nabla p})_f] \tag{C4}$$

then the face volume flux can be easily obtained as follows:

$$F_f \equiv (\delta \mathbf{A})_f \cdot \mathbf{v}_f = (\delta \mathbf{A})_f \cdot \bar{\mathbf{v}}_f - \frac{(\delta V)_f}{\bar{a}_f}(\delta \mathbf{A})_f \cdot [(\nabla p)_f - (\overline{\nabla p})_f] \tag{C5}$$

Applying the expression for the gradient at the cell face (see Appendix B) to Equation (C5) gives

$$F_f = (\delta \mathbf{A})_f \cdot \bar{\mathbf{v}}_f - \frac{(\delta \mathbf{A})_f \cdot (\delta \mathbf{A})_f}{\bar{a}_f}[(p_N - p_P) - (\overline{\nabla p})_f \cdot \mathbf{r}_{PN}] \tag{C6}$$

This expression is used for all faces that separate two interior cells.

At a boundary face, the volume flux can be written as

$$F_b = (\delta \mathbf{A})_b \cdot \mathbf{v}_P - \frac{(\delta \mathbf{A})_b \cdot (\delta \mathbf{A})_b}{a_P}[(p_b - p_P) - (\nabla p)_P \cdot \mathbf{r}_{Pb}] \tag{C7}$$

APPENDIX D: ANALYTICAL SOLUTION FOR A FOAM FLOW WITH TEMPORAL DENSITY CHANGE IN A THIN SLIT

An approximate analytic solution can be obtained for the case of creeping flow in the thin slit. The basic assumptions are as follows: isothermal flow, no slip at the wall, no body force, laminar flow, and Newtonian fluid that follows Stokes' hypothesis. Figure D1 shows the geometry, which is initially filled completely with a foam whose density changes with time. The geometry is a two-dimensional rectangular shape with three walls and a free surface. The free surface is maintained as flat, with the assumption that foam is removed as soon as it flows beyond the free surface line.

Density is assumed to be a function only of time, using

$$\rho(t) = \rho_0 \exp(-\kappa t) \tag{D1}$$

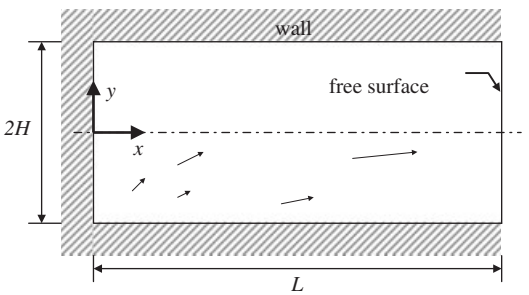

Figure D1. The geometry used for the approximate analytical solution for a foam flow with temporal density change.

where ρ_0 is the initial density, κ is the rate of exponential decay of density, and t is time. Every point in the foam has the same density at any given time. The Reynolds number for this problem can be defined as

$$Re = \frac{\rho_0 \kappa H^2}{\mu} \tag{D2}$$

where μ is the viscosity and H the half-height defined in Figure D1. If $Re \ll 1$ then inertia can be neglected, and the continuity and momentum equations can be written as follows:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = \kappa \tag{D3}$$

$$\frac{\partial p}{\partial x} = \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \tag{D4}$$

$$\frac{\partial p}{\partial y} = \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \tag{D5}$$

where u and v are x- and y-direction velocities, respectively, and p is pressure. All equations are independent of time, even though the problem is originally transient. If $L \gg H$, then the y-direction velocity can be neglected and the x-direction derivatives of velocity are also negligible, reducing the x-direction momentum equation to

$$\frac{\partial p}{\partial x} = \mu \frac{\partial^2 u}{\partial y^2} \tag{D6}$$

If pressure is assumed to be a function of x only and boundary conditions are applied, the x-direction velocity is

$$u = \frac{1}{2\mu} \frac{\partial p}{\partial x} (y^2 - H^2) \tag{D7}$$

Integrating the velocity in the y direction from zero to H gives

$$\int_0^H u \, \mathrm{d}y = \frac{1}{2\mu} \frac{\partial p}{\partial x} \int_0^H (y^2 - H^2) \, \mathrm{d}y \tag{D8}$$

or

$$\bar{u} = \frac{H^2}{3\mu} \left(-\frac{\partial p}{\partial x} \right) \tag{D9}$$

where the bar now means the average over the y direction. Integrating the continuity equation along the y direction and applying boundary conditions gives

$$\int_0^H \frac{\partial u}{\partial x} \, \mathrm{d}y + \int_0^H \frac{\partial v}{\partial y} \, \mathrm{d}y = \int_0^H \kappa \, \mathrm{d}y \tag{D10}$$

or

$$\bar{u} = \kappa x \tag{D11}$$

Substituting Equation (D11) into Equation (D9) and applying the boundary condition $p = 0$ at $x = L$, we obtain the pressure solution as follows:

$$p = -\frac{3\mu\kappa}{2H^2}(x^2 - L^2) \tag{D12}$$

The x-direction velocity is then obtained as

$$u = -\frac{3\kappa}{2H^2}x(y^2 - H^2) \tag{D13}$$

When this equation is substituted into the continuity equation, the y-direction velocity is also obtained

$$v = \frac{\kappa}{2H^2}(y^3 - H^2 y) \tag{D14}$$

ACKNOWLEDGEMENTS

This study was supported by the Korea Science and Engineering Foundation through the Applied Rheology Center (ARC), an officially KOSEF-created engineering research centre at Korea University, Seoul, South Korea. The authors thank Prof. S. P. Vanka of the University of Illinois for his helpful suggestions.

REFERENCES

1. Sweeney FM. *Reaction Injection Molding Machinery and Processes*. Marcel Dekker: New York, 1987.
2. Macosko CW. *RIM, Fundamentals of Reaction Injection Molding*. Hanser Publishers: New York, 1989.
3. Park H, Youn JR. Processing of cellular polyurethane by ultrasonic excitation. *Journal of Engineering for Industry* 1992; **114**:323–328.
4. Cho WJ, Park H, Youn JR. Ultrasonic bubble nucleation in reaction injection moulding of polyurethane. *Journal of Engineering and Manufacturing* 1994; **208**:121–128.
5. Park H, Youn JR. Study on reaction injection molding of polyurethane microcellular foam. *Polymer Engineering and Science* 1995; **35**:1899–1906.
6. Youn JR, Park H. Bubble growth in reaction injection molded parts foamed by ultrasonic excitation. *Polymer Engineering and Science* 1999; **39**:457–468.
7. Kim C, Youn JR. Environmentally friendly processing of polyurethane foam for thermal insulation. *Polymer-Plastics Technology and Engineering* 2000; **39**:163–185.
8. Koo MS, Chung K, Youn JR. Reaction injection molding of polyurethane foam for improved thermal insulation. *Polymer Engineering and Science* 2001; **41**:1177–1186.
9. Klempner D, Frisch KC. *Handbook of Polymeric Foams and Foam Technology*. Hanser Publishers: New York, 1991.
10. Hayes RE, Dannelongue HH, Tanguy PA. Numerical simulation of mold filling in reaction injection molding. *Polymer Engineering and Science* 1991; **31**:842–848.
11. Perrett FJ, Reible DD, McIlhenny RC. Modeling fountain flow and filling front shape in reaction injection molding. *Polymer Engineering and Science* 1993; **33**:716–720.
12. Anturkar NR. A model of advancing flow front in RIM. *Polymer Engineering and Science* 1994; **34**: 1450–1454.
13. Lo Y, Reible DD, Collier JR, Chen C. Three-dimensional modeling of reaction injection molding. I. *Polymer Engineering and Science* 1994; **34**:1393–1400.
14. Lo Y, Reible DD, Collier JR, Chen C. Three-dimensional modeling of reaction injection molding. II: application. *Polymer Engineering and Science* 1994; **34**:1401–1405.
15. Arai S, Tsutsumi Y, Lowe DW. Amine catalyst for refrigerator rigid foam application. *Proceedings of the SPI-29th Annual Technicall/Marketing Conference*, 1985; 272–278.
16. Lefebvre L, Keunings R. Finite element modelling of the flow of chemically reactive polymeric liquids. *International Journal for Numerical Methods in Fluids* 1995; **20**:319–334.

17. Lefebvre L, Keunings R. Mathematical Modeling and Computer Simulation of the Flow of Chemically-Reacting Polymeric Foams. In *Mathematical Modelling for Materials Processing*, Cross M, Pittman JFT, Wood RD (eds). Clarendon: Oxford, 1993; 399–417.
18. Mathur SR, Murthy JY. A pressure-based method for unstructured meshes. *Numerical Heat Transfer*, *Part B* 1997; **31**:195–215.
19. Patankar SV. *Numerical Heat Transfer and Fluid Flow*. McGraw-Hill: New York, 1980.
20. Ubbink O, Issa RI. A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics* 1999; **153**:26–50.
21. Thompson E. Use of pseudo-concentrations to follow creeping viscous flows during transient analysis. *International Journal for Numerical Methods in Fluids* 1986; **6**:749–761.
22. Kostrzewski W. Foam rheometer: its principle and applications. *Proceedings of the SPI-29*th *Annual Technical/Marketing Conference*, 1985; 209–213.
23. Niyogi D, Kumar R, Gandhi KS. Water blown free rise polyurethane foams. *Polymer Engineering and Science* 1999; **39**:199–209.
24. Sun X, Toth J, Lee LJ. Chemorheology of poly(urethane/isocyanurate) formation. *Polymer Engineering and Science* 1997; **37**:143–152.
25. Kraynik AM. Foam flows. *Annual Review of Fluid Mechanics* 1988; **20**:325–357.
26. Stokes GG. On the theories of internal friction of fluids in motion. *Transaction of the Cambridge Philosophical Society* 1845; **8**:287–305.
27. Haagh GAAV, van de Vosse FN. Simulation of three-dimensional polymer mould filling processing using a pseudo-concentration method. *International Journal for Numerical Methods in Fluids* 1998; **28**:1355–1369.
28. Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
29. Hetu JF, Gao DM, Garcia-Rejon A, Salloum G. 3D finite element method for the simulation of the filling stage in injection molding. *Polymer Engineering and Science* 1998; **38**:223–236.
30. Chang RY, Yang WH. Numerical simulation of mold filling in injection molding using a three-dimensional finite volume approach. *International Journal for Numerical Methods in Fluids* 2001; **37**:125–148.
31. Rhie CM, Chow WL. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal* 1983; **21**:1525–1532.
32. Leonard BP. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering* 1991; **88**:17–74.
33. Rose W. Fluid-fluid interfaces in steady motion. *Nature* 1961; **191**:242–243.
34. Coyle DJ, Blake JW, Macosko CW. The kinematics of fountain flow in mold-filling. *AIChE Journal* 1987; **33**:1168–1177.
35. Behrens RA, Crochet MJ, Denson CD, Metzner AB. Transient free-surface flows: motion of a fluid advancing in a tube. *AIChE Journal* 1987; **33**:1178–1986.
36. Richardson S. A 'stick-slip' problem related to the motion of a free jet at low Reynolds numbers. *Proceedings of the Cambridge Philosophical Society* 1970; **67**:477–489.
37. Georgiou GC, Olson LG, Schultz WW, Sagan S. A singular finite element for Stokes flow: the stick-slip problem. *International Journal for Numerical Methods in Fluids* 1989; **9**:1353–1367.
38. Manas-Zloczower I, Blake JW, Macosko CW. Space-time distribution in filling a mold. *Polymer Engineering and Science* 1987; **27**:1229–1235.
39. Kim CJ. *Introduction to Computational Fluid Dynamics*. Munundang: Seoul, 1998.

