INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng. **46**, 1559–1573 (1999)

FINITE ELEMENT SIMULATION OF THE FILLING OF THIN MOULDS

RAMON CODINA* AND ORLANDO SOTO

International Center for Numerical Methods in Engineering, Universitat Politècnica de Catalunya, Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain

SUMMARY

In this work we present a finite element formulation to simulate the filling of thin moulds. The model has as starting point a 3-D approach using either div-stable elements, such as the Q2/P1 element (tri-quadratic velocities and piecewise linear, discontinuous pressures) or stabilized finite element formulations. The tracking of the free surface is based on the Volume-Of-Fluid (VOF) method. The velocity profile is assumed to be parabolic in the direction normal to the mid-plane, so that one element along the width of the mould is enough to reproduce this profile if this element is quadratic. The velocity is prescribed to zero on the upper and lower surfaces and the normal to the mid-plane is also prescribed to zero. In the case of div-stable elements, the pressure profile is prescribed to be constant along the width of the mould. This can be achieved by using as interpolation degrees of freedom the pressure values at the element centroid and its derivatives in the directions tangent and normal to the mid-plane, and prescribing the latter to zero. No modifications are needed when stabilized formulations are employed. To advance in time the function used in the VOF technique, we use a constant velocity across the width of the mould, which is taken as the projection on the tangent plane of each element of the nodal velocities. This is needed in order to have mass conservation. Copyright © 1999 John Wiley & Sons, Ltd.

KEY WORDS: finite elements; incompressibility constraint; surface flow; free surface

1. INTRODUCTION

Mould filling is the first stage of the casting process, which starts with the pouring of a molten material into a mould until it is filled and concludes when the solid nature of the material is restored. The numerical approximation of this process is extremely complex, since in general it involves the approximation of the incompressible Navier–Stokes equations in a transient and turbulent regime, together with the tracking of the free surface of the fluid filling the mould. However, in many applications the mould to be filled has one dimension much smaller than the other two, in the sense that it can be considered as a 'shell'. This simplifies the problem considerably, since under appropriate simplifications the problem can be made two-dimensional [1].

The purpose of this paper is to present a finite element formulation for the numerical simulation of the filling of thin moulds. For that, we start from a 3-D finite element method for mould filling

* Correspondence to: Ramon Codina, c/o Professor R. Lohner, Institute of Computational Sciences and Information, George Mason University, Science and Technology Building 1 MS 5C3, 4400 University Drive, Fairfax VA 22030-4444, U.S.A. E-mail: codina@ce.berkeley.edu

CCC 0029-5981/99/331559–15$17.50

Received 1 September 1998
Revised 1 February 1999

based on the Volume Of Fluid (VOF) technique [2] as the method to track the free surface. The basic flow model is based either on the use of stable velocity–pressure spaces or in stabilized finite formulations allowing the use of equal velocity–pressure interpolations, such as that presented in [3]. These basic formulations are then applied to the case of the flow in thin moulds, for which several assumptions are made in order to transform the problem into a 2-D one. This reduction of the problem has important consequences related to the stability of the pressure. We discuss them using both stable and stabilized velocity–pressure spaces. The tracking of a free surface for a flow in thin regions is considered next. A numerical simulation consisting of the filling of a spoon and another showing the filling of a cylindrical shell demonstrate the effectiveness of the numerical model developed. Finally, in the last section of the paper we summarize the basic numerical ingredients of the model and draw some conclusions.

2. PROBLEM STATEMENT

2.1. Three-dimensional problem

In this section we shall consider the general problem for an incompressible fluid in laminar regime and taking into account the existence of a free surface within the domain Ω to be occupied by the fluid. We also consider the possibility of having a thermo-coupled flow.

2.1.1. Flow equations. The equations describing the problem are

$$\rho\left[\frac{\partial u}{\partial t}+(\mathbf{u}\cdot\nabla)\mathbf{u}\right]-2\nabla\cdot[\mu\boldsymbol{\varepsilon}(\mathbf{u})]+\nabla p=\rho\mathbf{f} \tag{1}$$

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

$$\rho C_p\left[\frac{\partial T}{\partial t}+(\mathbf{u}\cdot\nabla)T\right]-\nabla\cdot[k\nabla T]=Q \tag{3}$$

to be solved in $\Omega\times(0,t_{\rm fin})$, where Ω is the computational domain and $[0,t_{\rm fin}]$ is the time interval to be considered. In (1)–(3) $\mathbf{u}$ denotes the velocity field, p is the pressure, T is the temperature, ρ is the density, μ is the dynamical viscosity, which may depend on the invariants of the symmetrical part of the velocity gradient $\boldsymbol{\varepsilon}(\mathbf{u})$ and the temperature, C_p is the specific heat at constant pressure, k is the thermal conductivity, $\mathbf{f}$ is the vector of body force and Q is the heat source, which may include the term $2\mu\boldsymbol{\varepsilon}(\mathbf{u})\colon\boldsymbol{\varepsilon}(\mathbf{u})$ if the mechanical dissipation into heat needs to be taken into account.

Let $\boldsymbol{\sigma}$ be the stress tensor and $\mathbf{n}$ the unit outward normal to the boundary $\partial\Omega$. Denoting by an overbar prescribed values, the boundary conditions for the velocity to be considered here are

$$\begin{aligned}\mathbf{u}=\bar{\mathbf{u}}\quad&\text{on }\Gamma_{\rm du}\\ \mathbf{n}\cdot\boldsymbol{\sigma}=\bar{\mathbf{t}}\quad&\text{on }\Gamma_{\rm nu}\\ \mathbf{u}\cdot\mathbf{n}=\bar{u}_n,\quad \mathbf{n}\cdot\boldsymbol{\sigma}\cdot\mathbf{g}_1=\bar{t}_1,\quad \mathbf{n}\cdot\boldsymbol{\sigma}\cdot\mathbf{g}_2=\bar{t}_2\quad&\text{on }\Gamma_{\rm mu}\end{aligned} \tag{4}$$

for $t\in(0,t_{\rm fin})$. The boundary $\partial\Omega$ has been considered split into three sets of disjoint components $\Gamma_{\rm du},\Gamma_{\rm nu}$ and $\Gamma_{\rm mu}$, the latter being the part of the boundary where mixed conditions are prescribed: the normal velocity and the tangent stresses. Vectors $\mathbf{g}_1$ and $\mathbf{g}_2$ (for the three-dimensional case) span the space tangent to $\Gamma_{\rm mu}$.

The boundary conditions to be considered for the temperature are

$$\begin{aligned} T &= \bar{T} \quad \text{on } \Gamma_{\mathrm{dT}} \\ \mathbf{n}\cdot(k\nabla T) &= \bar{h} \quad \text{on } \Gamma_{\mathrm{nT}} \end{aligned} \tag{5}$$

for $t \in (0, t_{\mathrm{fin}})$, where again Γ_{dT} and Γ_{nT} are disjoint components of $\partial\Omega$, now where Dirichlet and Neumann boundary conditions for the temperature are prescribed.

Initial conditions have to be appended to problem (1)–(3). They are of the form

$$\mathbf{u}(\mathbf{x},0) = \mathbf{u}^0(\mathbf{x}), \quad \mathbf{x}\in\Omega$$

$$T(\mathbf{x},0) = T^0(\mathbf{x}), \quad \mathbf{x}\in\Omega$$

where $\mathbf{u}^0(\mathbf{x})$ and $T^0(\mathbf{x})$ are given initial data.

2.1.2. Free surface tracking. Concerning the tracking of the free surface, we use the (VOF) technique [2]. The basic idea of this method is to define a scalar function, say $\psi(\mathbf{x},t)$, over the computational domain Ω in such a manner that its value at a certain point $\mathbf{x}\in\Omega$ indicates the presence or absence of fluid. This function may be considered as a fictitious fluid property. For instance, we may assign the value 1 to regions where the fluid has already entered and the value 0 to air-filled regions. The position of the fluid front will be defined by the isovalue contour $\psi(\mathbf{x},t) = \psi_{\mathrm{c}}$, where $\psi_{\mathrm{c}} \in [0,1]$ is a critical value defined *a priori*. We usually take $\psi_{\mathrm{c}} = 0{\cdot}5$. This value is immaterial if ψ is a true step function, but is needed in the finite element discretization.

The conservation of ψ (assumed to be sufficiently smooth) in any control volume $V_t \subset \Omega$ which is moving with the divergence-free velocity field $\mathbf{u}$ leads to

$$\frac{\partial\psi}{\partial t} + (\mathbf{u}\cdot\nabla)\psi = 0 \quad \text{in } \Omega,\ t\in(0,t_{\mathrm{fin}}) \tag{6}$$

This equation is hyperbolic and therefore boundary conditions for ψ have to be specified at the inflow boundary, defined as

$$\Gamma_{\mathrm{inf}} := \{\mathbf{x}\in\partial\Omega \mid \mathbf{u}\cdot\mathbf{n} < 0\}$$

The definition of the position of the fluid front will be given by the physical properties. Let π be any of these, i.e., density (ρ), viscosity (μ), specific heat (C_p) or thermal conductivity (k). We will have that

$$\pi(\mathbf{x},t) = \begin{cases} \pi_{\mathrm{fluid}}(\mathbf{x},t) & \text{if } \mathbf{x}\in\Omega_t \\ \pi_{\mathrm{air}} & \text{if } \mathbf{x}\in\Omega\backslash\Omega_t \end{cases}$$

where

$$\Omega_t := \{\mathbf{x}\in\Omega \mid \psi(\mathbf{x},t) \geqslant \psi_{\mathrm{c}}\}$$

and the function ψ is the solution of the following problem:

$$\begin{aligned} \frac{\partial\psi}{\partial t} + (\mathbf{u}\cdot\nabla)\psi &= 0 \quad \text{in } \Omega,\ t\in(0,t_{\mathrm{fin}}) \\ \psi &= \bar{\psi} \quad \text{on } \Gamma_{\mathrm{inf}},\ t\in(0,t_{\mathrm{fin}}) \\ \psi(\mathbf{x},0) &= \psi^0(\mathbf{x}) \quad \text{in } \Omega \end{aligned}$$

The initial condition ψ^0 is chosen in order to define the initial position of the fluid to be analysed. The boundary condition $\bar{\psi}$ determines whether fluid enters or not through a certain point of the inflow boundary. If it does, a value $\bar{\psi} \geqslant \psi_\mathrm{c}$ is to be prescribed (for example, $\bar{\psi} = 1$); else, $\bar{\psi} < \psi_\mathrm{c}$.

Observe that since the physical properties will be discontinuous across the fluid front, the differential equations (1)–(3) will not exactly describe the conservation of momentum, mass, and internal energy, since the jump of these properties has been simply ignored. Observe also that since the fluid under consideration and the air are treated at once, no boundary conditions at the interface between them are needed.

2.1.3. Air release. As explained in [4], one of the problems of the VOF formulation as described above is the evacuation of air bubbles. Since we deal with incompressible flows, air cannot shrink and air bubbles near the corners will remain if a method to evacuate them is not devised. In practice, moulds are made of porous materials (usually sand in casting applications). Therefore, air can leave the mould without resistence. Numerically, a possible way to evacuate air is to introduce 'holes' on the boundary and to block them when the fluid touches the wall. Instead of using discrete holes, it is also possible to consider all the walls open for the air.

When the computational domain is discretized using a finite element mesh (see Section 3), the above idea leads to a change in the type of boundary conditions for the velocity at the nodes of the mesh corresponding to the holes, changing from a Neumann-type boundary condition to a Dirichlet one. In order to avoid the need for changing the size of the problem associated to this (or the definition of the boundary condition), we leave the nodes located at the holes always 'free'. When the fluid reaches them, the velocity (or perhaps only the component normal to the wall) is prescribed to zero using the following standard strategy. Let us consider a generic linear system of the form

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{7}$$

where $\mathbf{x}$ is a vector of n unknowns. Suppose that the ith component of $\mathbf{x}$ is to be prescribed to a value $\bar{x}$, i.e. $x_i = \bar{x}$. From (7) we will have that

$$a_{ii} x_i = b_i - \sum_{j=1, j \neq i}^{n} a_{ij} x_j$$

Assume that the component a_{ii} of matrix $\mathbf{A}$ is not zero and replace

$$a_{ii} \leftarrow 1, \quad a_{ij} \leftarrow 0, \quad i \neq j, \quad b_i \leftarrow \bar{x} \tag{8}$$

This can be done at the element level, since the assembly operator is linear. The unknown will be exactly prescribed at the required value.

The way to block the holes is now clear. For a certain time step (see Section 3), the value of ψ at the point of interest is computed. If this value ψ is lower than ψ_c, nothing is done for the system analogous to (7) arising from the fully discrete and linearized Navier–Stokes equations and the redefinition (8) is not performed. Otherwise, the previous strategy to prescribe values is followed, taking $\bar{x} = 0$ in (8).

Consider now the transport equation for the free-surface function. If for a certain time step the velocity at a node lying on the hole is left free, it may point into the mould due to suction effects. In this situation, the hole must be considered as a part of the inflow boundary Γ_{inf} and therefore the function ψ must be prescribed there. Otherwise, it may happen that values of ψ higher than ψ_c be transported into the mould, thus introducing spurious fluid. The way to see this is the following. Let V_t be any control volume surrounding this node. Multiplying (6) by ψ, integrating over V_t and using the fact that $\mathbf{u}$ is divergence-free yields

$$\frac{\mathrm{d}}{\mathrm{dt}}\int_{V_t}\psi^2 = -\frac{1}{2}\int_{\partial V_t}(\mathbf{n}\cdot\mathbf{u})\psi^2$$

If ψ is not prescribed where $\mathbf{n}\cdot\mathbf{u}<0$, the integral of ψ^2 over V_t may increase as time goes on, and this happens for any control volume V_t, that is, a spurious fluid-filled region may appear around the hole.

Having these considerations in mind, it is clear that the free-surface function must be prescribed at the temporary free wall nodes where $\mathbf{n}\cdot\mathbf{u}<0$. For a certain time step, the value of the prescription will be the value obtained in the previous step. The way to implement this is the same as for the velocities in the Navier–Stokes equations. Let ψ_{b}^n the value of the free surface function at the node under consideration for time step n. Considering that the system to be solved to find ψ for time step $n+1$ is (7), the redefinition (8) will be employed, with $\bar{x}=\psi_{\mathrm{b}}^n$.

This completes the formulation of the VOF method. This method, also known as pseudo-concentration technique, has been used by several authors to follow free surfaces of creeping flows and viscoplastic flows in the context of metal forming processes, such as extrusion, forging or rolling [5, 6], as well as to mould filling [4, 7, 8].

2.2. Flow in a surface

In the previous section we have presented the most general problem of tracking the free surface using the VOF method in a 3-D flow. The idea is to consider now the particular case in which the computational domain is 'thin' in a certain direction. More precisely, let $\mathscr{S}$ be a surface in the three-dimensional space described by a mapping of the form

$$(s_1,s_2)\mapsto \mathbf{x}(s_1,\,s_2)=(x_1(s_1,\,s_2),x_2(s_1,\,s_2),\,x_3(s_1,\,s_2))$$

We assume that Ω is of the form $\mathscr{S}\times\mathscr{I}$, where for each $\mathbf{x}\in\mathscr{S}$, $\mathscr{I}$ is a segment centered at $\mathbf{x}$ of length $2a$ (which may vary from point to point in $\mathscr{S}$), small compared to the dimensions of $\mathscr{S}$. This segment is assumed to be parameterized by the arc parameter s_3, and therefore $-a\leqslant s_3\leqslant a$.

For this particular type of computational domain, several assumptions may be done in order to reduce the 3-D problem for the velocity and the pressure to another equivalent (from the computational point of view) to a 2-D one. These assumptions can be written in terms of the parameterization $(s_1,\,s_2,\,s_3)$ of Ω by expressing $\mathbf{u}$ in terms of these parameters:

H1 The velocity field vanishes at the boundaries defined by the end-points of $\mathscr{I}$, that is to say, $\mathbf{u}(s_1,\,s_2,\,\pm a)=\mathbf{0}$. This is the classical no-slip condition for viscous flow.

H2 The velocity field is tangent to $\mathscr{S}$ for $s_3=0$.

H3 For fixed values of s_1 and s_2, the velocity $\mathbf{u}$ is parabolic as a function of s_3. This assumption, combined with H1 and H2, implies that there is no flow in the direction normal to $\mathscr{S}$.

H4 For fixed values of s_1 and s_2, the pressure is constant as a function of s_3. This condition is physically consistent with H3.

The physical situation implied by all these assumptions is schematically depicted in Figure 1.

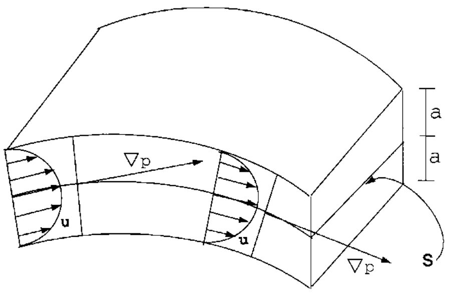

Figure 1. Velocity and pressure for the flow in a surface

With all these assumptions the physical problem can be transformed into a 2-D one, the unknowns being the two velocity components tangent to $\mathscr{S}$ and the pressure, all these variables depending only on the surface co-ordinates. The modified problem can then be approximated by any numerical technique [1]. However, the flow equations are different from the standard Navier–Stokes equations (1) written in Cartesian co-ordinates and therefore standard codes available for 3-D flows have to be substantially modified. In the following section we shall describe how to adapt a 3-D code to account for assumptions H1–H4 with minor modifications.

The basic assumption on the function ψ that defines the position of the free surface is that it is constant in s_3, so that it needs to be interpolated only in $\mathscr{S}$. In order to be able to satisfy this assumption, the advection velocity **u** to be used in (6) must be constant in s_3, since if the true parabolic profile is used then the position of the fluid front would not advance at the boundaries defined by the end points of $\mathscr{I}$. The velocity we take for equation (6) is $2\mathbf{u}(s_1, s_2, 0)/3$, that is, the average velocity along the width of the domain.

Concerning the thermal problem, the same assumption on the temperature T as on the function ψ may be used. However, there is also the possibility of solving for the temperature without taking into account the fact that the domain is thin.

3. FINITE ELEMENT MODEL FOR SURFACE FLOW

3.1. *Space and time discretization*

We consider in this section the numerical approximation of the *original equations* (1)–(3) and (6). In the following section we shall see how this discretization can be applied to surface flows.

Let us consider first the temporal discretization, for which we use the generalized trapezoidal rule. Let $0 = t^0 < t^1 < \cdots < t^N = t_{\text{fin}}$ be a partition of the time interval and $\theta \in [0, 1]$, and let us also introduce the notation

$$\delta f^n := f^{n+1} - f^n, \quad f^{n+\theta} := \theta f^{n+1} + (1-\theta) f^n$$

where f is a generic function of time and f^n denotes the value of f at time t^n or an approximation to it. For simplicity of notation, we shall take the time step size $\delta t := t^{n+1} - t^n$ constant for all n.

The generalized trapezoidal rule applied to equations (1)–(3) and (6) leads to the following time-discrete problem: from known $\mathbf{u}^n, p^n, T^n$ and ψ^n, find $\mathbf{u}^{n+1}, p^{n+1}, T^{n+1}$ and ψ^{n+1} such that

$$\rho\left[\frac{\delta\mathbf{u}^n}{\delta t}+(\mathbf{u}^{n+\theta}\cdot\nabla)\mathbf{u}^{n+\theta}\right]-2\nabla\cdot[\mu\boldsymbol{\varepsilon}(\mathbf{u}^{n+\theta})]+\nabla p^{n+1}=\rho\mathbf{f}^{n+\theta} \tag{9}$$

$$\nabla\cdot\mathbf{u}^{n+1}=0 \tag{10}$$

$$\rho C_{\mathrm{p}}\left[\frac{\delta T^n}{\delta t}+(\mathbf{u}^{n+\theta}\cdot\nabla)T^{n+\theta}\right]-\nabla\cdot[k\nabla T^{n+\theta}]=Q^{n+\theta} \tag{11}$$

$$\frac{\delta\psi^n}{\delta t}+(\mathbf{u}^{n+\theta}\cdot\nabla)\psi^{n+\theta}=0 \tag{12}$$

and satisfying the boundary conditions. For simplicity, we have considered the force vector $\mathbf{f}$ and the heat source Q continuous in time. The values of interest of the parameter θ are $\theta=\frac{1}{2}$ and $\theta=1$, corresponding to the Crank–Nicolson and the backward Euler schemes, respectively.

In order to discretize in space the above equations, let $\{\Omega^e\}$ be a finite element partition of the domain Ω, with index e ranging from 1 to the number of elements n_{el}. We denote with a subscript h the finite element approximation to the unknown functions, and by $\mathbf{v}_h$, q_h, S_h and ϕ_h the velocity, pressure, temperature and free-surface test functions associated to $\{\Omega^e\}$.

The standard Galerkin formulation applied to equations (9)–(12) has several well-known difficulties. Let us discuss first the choice of the finite element spaces for the velocity and the pressure. It is well known that due to the zero-divergence restriction (10) these spaces have to satisfy the Babuŝka–Brezzi (BB) stability condition (often referred to as div-stability in the present situation, see for example 9). Some of the possible velocity pressure pairs available for the problem we consider are discussed in the following section.

Concerning the finite element space for the temperature and the free-surface function, we have used for them the same interpolation as for the velocity components.

The second well-known numerical problem arises when the convective term in a transport equation becomes important. In this case, the standard Galerkin formulation fails and numerical oscillations occur. Upwind techniques must be devised to solve this problem.

In order to overcome the problems described, we have applied a stabilized finite element formulation to solve (9)–(12). This formulation is presented in [10] for the general case of systems of convection–diffusion–reaction equations, and applied also to the incompressible Navier–Stokes equations. The bottom line of the method is to test the continuous equations by the standard Galerkin test functions (belonging to the finite element space where the solution is sought) plus perturbations that depend on the operator representing the differential equation being solved. When applied to the time-discrete Navier–Stokes equations (9)–(10), the method consists of finding $\mathbf{u}_h^{n+1}$ and p_h^{n+1} belonging to the velocity and pressure finite element spaces such that

$$\begin{aligned}&\int_\Omega(\mathbf{v}_h+\boldsymbol{\zeta}_{h,u})\cdot\rho\left[\frac{\delta\mathbf{u}_h^n}{\delta t}+(\mathbf{u}_h^{n+\theta}\cdot\nabla)\mathbf{u}_h^{n+\theta}\right]\\&\quad+\int_\Omega 2\mu\boldsymbol{\varepsilon}(\mathbf{v}_h):\boldsymbol{\varepsilon}(\mathbf{u}_h^{n+\theta})-\int_{\Omega'}2\boldsymbol{\zeta}_{h,u}\cdot(\nabla\cdot[\mu\boldsymbol{\varepsilon}(\mathbf{u}_h^{n+\theta})])\\&\quad-\int_\Omega p_h^{n+1}\nabla\cdot\mathbf{v}_h+\int_{\Omega'}\boldsymbol{\zeta}_{h,u}\cdot\nabla p_h^{n+1}+\int_\Omega q_h(\nabla\cdot\mathbf{u}_h^{n+1}+\boldsymbol{\varepsilon}p_h^{n+1})\\&=\int_{\Omega'}(\mathbf{v}_h+\boldsymbol{\zeta}_{h,u})\cdot\rho\mathbf{f}^{n+\theta}+\int_{\Gamma_{\mathrm{nu}}}\mathbf{v}_h\cdot\bar{\mathbf{t}}+\int_{\Gamma_{\mathrm{mu}}}(v_{h,1}\bar{t}_1+v_{h,2}\bar{t}_2)\end{aligned} \tag{13}$$

for all test functions $\mathbf{v}_h$ and q_h, where the perturbation function $\boldsymbol{\zeta}_{h,u}$ is given by

$$\boldsymbol{\zeta}_{h,u} = \tau_u \left[(\mathbf{u}^n \cdot \nabla)\mathbf{v}_h + \sigma \left(2\nabla \cdot [\mu \boldsymbol{\varepsilon}(\mathbf{v}_h)] + \nabla q_h \right) \right] \tag{14}$$

and parameter τ_u is computed for each element as [11, 12]

$$\tau_u = \beta_0 \left[\frac{2|\mathbf{u}^n|}{h} + \frac{4\mu}{\rho h^2} \right]^{-1} \tag{15}$$

with $\beta_0 = 1$ for linear elements and $\frac{1}{2}$ for quadratic elements. Both in (14) and in (15) the velocity can be taken at the current time step, that is, $\mathbf{u}_h^{n+1}$ can be used instead of $\mathbf{u}_h^n$, although this introduces a further non-linearity in the problem. In (15) and below, h is the element size. Observe that the boundary conditions (4) have been incorporated in (13).

The perturbation given by (14) is discontinuous across inter-element boundaries. To give sense to the integrals of the second derivatives of finite element functions appearing in the above equations we have introduced the notation

$$\int_{\Omega'} := \sum_{e=1}^{n_{\rm el}} \int_{\Omega^e}$$

There are several remarks to be made to equation (13):

(1) The single equation (13) can be split into two equations, one corresponding to the momentum equation and the other to the incompressibility constraint, as is usually done. This can be done by setting $q_h = 0$ in the first case and $\mathbf{v}_h = \mathbf{0}$ in the second.

(2) When $\sigma = 0$ the method considered reduces to the classical SUPG formulation for the Navier–Stokes equations [13]. In this case, the velocity and pressure finite element spaces have to satisfy the BB condition. This point is further commented in the following subsection.

(3) When $\sigma = 1$ the method corresponds to the algebraic version of the subgrid scale approach [14, 10]. In this case it is possible to use equal velocity pressure interpolations (see Section 3.3).

(4) In (13) we have introduced a parameter $\boldsymbol{\varepsilon}$ that corresponds to a penalty parameter for the incompressibility constraint. The use of penalty methods is very useful when the pressure interpolation is discontinuous, since in this case pressure degrees of freedom are easily eliminated at the element level (see Section 3.2).

The discrete variational statement for the heat equation (incorporating the boundary condition (5)) and the free-surface equation using the same type of stabilized finite element method (now to deal only with convection) is the following: find T^{n+1} and ψ^{n+1} such that

$$\begin{aligned} \int_\Omega (R_h + \zeta_{h,T})\rho C_{\rm p} \left[\frac{\delta T_h^n}{\delta t} + (\mathbf{u}_h^{n+\theta} \cdot \nabla) T_h^{n+\theta} \right] \\ + \int_\Omega \nabla R_h \cdot (k \nabla T_h^{n+\theta}) - \int_\Omega \zeta_{h,T} \nabla \cdot (k \nabla T_h^{n+\theta}) \\ = \int_{\Omega'} (R_h + \zeta_{h,T}) Q^{n+\theta} + \int_{\Gamma_{\rm nT}} R_h \cdot \bar{h} \end{aligned}$$

$$\int_\Omega (\phi_h + \zeta_{h,\psi}) \left[\frac{\delta \psi_h^n}{\delta t} + (\mathbf{u}_h^{n+\theta} \cdot \nabla)\psi_h^{n+\theta} \right] = 0$$

for all test functions R_h and ϕ_h, where the perturbation functions $\zeta_{h,T}$ and $\zeta_{h,\psi}$ are given by

$$\zeta_{h,T} = \tau_T \left[(\mathbf{u}^n \cdot \nabla)R_h + \sigma\nabla \cdot (k\nabla R_h)\right]$$

$$\zeta_{h,\psi} = \tau_\psi(\mathbf{u}^n \cdot \nabla)\phi_h$$

and the parameters τ_T and τ_ψ are

$$\tau_T = \beta_0 \left[\frac{2|\mathbf{u}^n|}{h} + \frac{4k}{\rho C_{\mathrm{p}} h^2}\right]^{-1}$$

$$\tau_\psi = \beta_0 \frac{h}{2|\mathbf{u}^n|}$$

3.2. *Mixed velocity–pressure interpolations*

Let us discuss now how to adapt the previous general approximation for the 3-D problem to the flow in a surface as explained earlier. We consider in this section that mixed div-stable velocity–pressure interpolations are needed, that is, $\sigma = 0$ in (13). In particular, we shall use discontinuous pressure interpolations. The degrees of freedom for such interpolation are described next.

In order to fulfill assumptions H1–H4 indicated in Section 2.2, we discretize the 3-D domain as follows:

S1 Only *one quadratic element* is used along the width of Ω. This condition implies that H3 will hold.

S2 The velocity is prescribed to zero at the end-points of $\mathscr{I}$, that is, at the extreme nodes of the quadratic element defined along the width of the domain. This is equivalent to assumption H1.

S3 The velocity is prescribed to be tangent to $\mathscr{S}$. This is done by *rotating the nodal velocities* and expressing them in the basis tangent to each node, which has to be computed for all the nodes in $\mathscr{S}$. This is precisely assumption H2. The resulting velocity field within an element is shown in Figure 2.

S4 In order to be able to satisfy H4, the pressure is interpolated using as degrees of freedom its values at the nodes on the surface $\mathscr{S}$ together with its derivatives in the direction normal to it. These derivatives are then prescribed to zero.

Using this approach, conditions H1–H4 are satisfied. However, the effects of the curvature of the shell are not exactly taken into account as it would happen if the continuous equations for the flow in a surface were discretized [1].

The interpolation used for the pressure is common for some elements, such as the classical Q_2/P_1. Let us discuss what happens in this case. The velocity is quadratic in each direction, so that steps S1–S3 above can be easily implemented.

Referring to the pressure interpolation, for the Q_2/P_1 element it is common to interpolate the pressures in the parent domain as

$$p(\xi, \eta, \gamma) = p_0 + p_1\xi + p_2\eta + p_3\gamma$$

where ξ, η, γ are the isoparametric co-ordinates. If the numbering of the nodes preserves the orientation from the numbering in the parent domain, then step S4 can be done simply by setting $p_3 = 0$, that is to say, by taking

$$p(\xi, \eta, \gamma) = p_0 + p_1\xi + p_2\eta$$

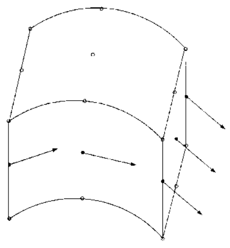

Figure 2. Velocity vectors in an element

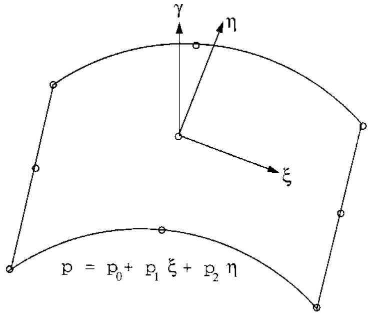

Figure 3. Pressure co-ordinates

The co-ordinates for the pressure interpolation are shown in Figure 3 . From the implementation point of view, this modification (from a standard 3-D code) can be easily done by setting to zero the third pressure interpolation function and its derivatives. The rest of the code does not need any further modification.

With the modifications indicated above, the velocity and pressure degrees of freedom for the Q_2/P_1 are the same as for the 2-D version of this element and thus the element is stable. In general, the velocity and pressure spaces for the 3-D 'surface element' must be isomorphic to those of the 2-D element. This is what is achieved using S1–S4.

Although the modifications for the Q_2/P_1 element are very small, this is not the case for other element interpolations. For example, in order to reproduce the 2-D P_2/P_1 element, with continuous quadratic velocities and continuous linear pressures, it is necessary to implement an element prismatic and quadratic in the direction normal to $\mathscr{S}$. Another 2-D element satisfying the BB condition that has to be modified is for example the P_2/P_0 element (quadratic velocities, piecewise constant pressures). The modification for the Q_2/Q_1 element (biquadratic velocities, continuous bilinear pressures) is simple and the Q_2^-/P_0 (serendipid velocities, piecewise constant pressures) needs no modification.

3.3. Stabilized methods

Instead of using velocity–pressure finite element interpolations satisfying the BB stability condition it is also possible to use stabilized formulations that allow to use *equal* interpolations. One of such methods is the one described above, and given by (13) *with* $\sigma = 1$.

The consequence of the use of stabilized methods for the case of flow in a surface is that step S4 indicated above *is not necessary*. The pressure interpolation is stable regardless of the velocity space chosen, and thus it is not mandatory to assume that the pressure is constant along the width of the domain.

4. MOULD FILLING OF THIN MOULDS

Once the basic numerical method for the approximation of the velocity and the pressure has been described, it remains to define the way in which the free surface of the fluid filling the mould is tracked. As for the general 3-D case, this is done by solving (6) for ψ. We take this function constant along the width of Ω. This can be done in two ways:

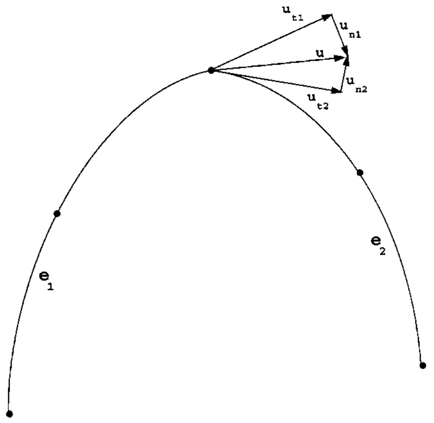

Figure 4. Elements sharing a node

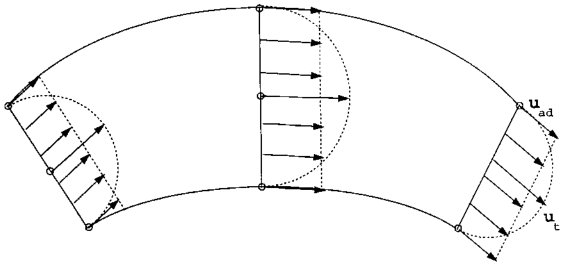

Figure 5. Advection velocity for ψ

(a) By using the same strategy as for the pressure. This would lead to non-standard finite element interpolations for ψ. Instead of this, the approach we follow is

(b) to use a constant velocity along the width of Ω.

For strategy (b) it is necessary to define the advection velocity for ψ. In order to have conservation of mass, we take this constant velocity as $\frac{2}{3}$ of $\mathbf{u}$ at the surface $\mathcal{S}$.

There is also another effect that needs to be taken into account. Consider the situation depicted in Figure 4, in which the mid-planes of elements $e1$ and $e2$ share a node P. Suppose that $\mathbf{u}$ at P is tangent to the analytical surface $\mathcal{S}$ and let $\mathbf{u}_{ti}$ and $\mathbf{u}_{ni}$ be the tangent and normal projections of $\mathbf{u}$ in the local basis at P tangent to ei, $i=1, 2$. Clearly, $\mathbf{u}_{n1}$ produces a flow normal to $e1$ that is not necessarily compensated by the flow that $\mathbf{u}_{n2}$ produces in $e2$. This will lead to mass conservation problems for the free surface equation. The way to avoid this problem is very simple: we simply take $\mathbf{u}_{t1}$ instead of $\mathbf{u}$ when evaluating the element matrix for element $e1$ in the transport of ψ. The final advection velocity $\mathbf{u}_{\text{ad}}$ within an element for this equation is shown in Figure 5.

As it has already been mentioned, the same approach as for the free-surface function can be used for the temperature, although there is no need to do this. In the first example of the following section, the original 3-D problem for the temperature has been solved. Moreover, it has to be remarked that if the adiabatic boundary condition $\bar{h}=0$ is prescribed on the top and bottom surfaces, the temperature will be (weakly) constant along the width of the domain.

5. NUMERICAL EXAMPLES

In this section we present two simple numerical examples to demonstrate the effectiveness of the numerical model described in this paper. The first example is the simulation of the filling of a spoon. The domain and the finite element mesh used for this case are shown in Figure 6. This mesh consists of 152 Q_2/P_1 elements (split into Q_1 elements in the plot) and 2040 nodal points, only 680 of which are on the mid-surface of the spoon.

The geometry and physical properties employed for this example correspond to the simulation of a real case of the injection of a plastic in a mould to make a spoon. Using kg for the mass

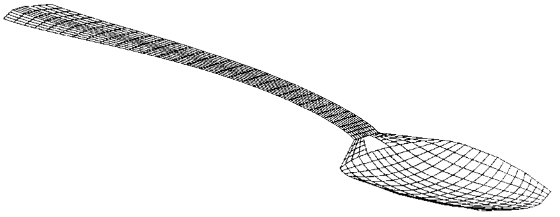

Figure 6. Mesh for the spoon

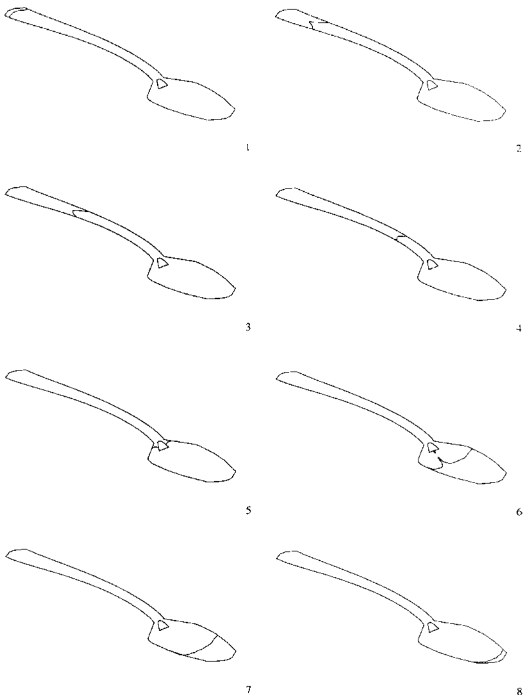

Figure 7. Position of the fluid front at time $t = 0{\cdot}1$, $0{\cdot}2$, $0{\cdot}3$, $0{\cdot}4$, $0{\cdot}5$, $0{\cdot}7$, $0{\cdot}9$ and $1{\cdot}1$ (from the left to the right and from the top to the bottom)

and mm for the distance units, these properties are $\mu = 1{\cdot}5$, $\rho = 7{\cdot}7 \times 10^{-4}$, $C_p = 2{\cdot}931 \times 10^7$ and $k = 11$. The properties taken for the 'air' (fictitious fluid) are $\mu_{\rm a} = 1{\cdot}5 \times 10^{-2}$, $\rho_{\rm a} = 7{\cdot}7 \times 10^{-7}$, $C_{p,a} = 2{\cdot}931 \times 10^{-10}$ and $k_{\rm a} = 1{\cdot}1 \times 10^{-2}$. These are not the physical properties of air, but have been chosen such that the motion of the fictious fluid does not affect that of the fluid filling the mould.

The maximum length of the spoon is $L = 480$ and its width $2a = 2$. The plastic is injected with a velocity $V = 2000$ at a temperature of 100°C. The initial temperature is 20°C in the air.

A time step size of $\delta t = 0{\cdot}1$ has been used for the time integration of the equations, for which the backward Euler scheme has been employed ($\theta = 1$ in (9)–(12)).

The position of the fluid front at times 0·1, 0·2, 0·3, 0·4, 0·5, 0·7, 0·9 and 1·1 is shown in Figure 7, whereas Figure 8 shows the velocity vectors and the pressure contours at $t = 0{\cdot}5$ and the temperature contours at $t = 0{\cdot}1$ and 0·3. Finally, Figure 9 shows the evolution of the temperature at $t = 0{\cdot}5$, 0·7, 0·9 and 1·1.

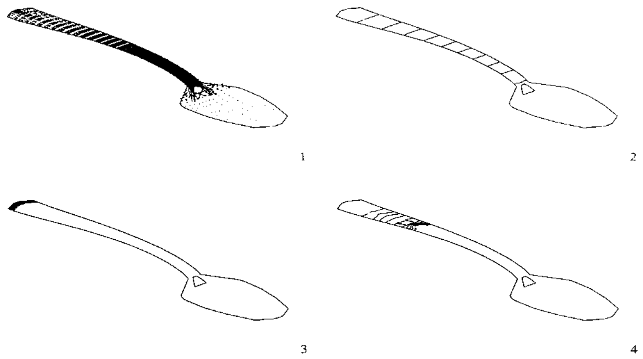

Figure 8. Velocity vectors and pressure contours at $t = 0{\cdot}5$ (top) and temperature contours at $t = 0{\cdot}1$ and $0{\cdot}3$ (bottom)

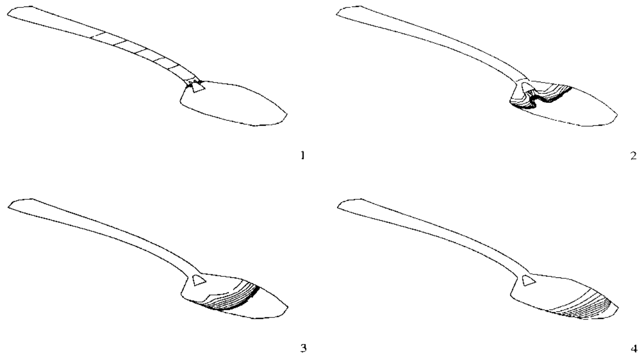

Figure 9. Temperature contours at $t = 0{\cdot}5$, $0{\cdot}7$, $0{\cdot}9$ and $1{\cdot}1$

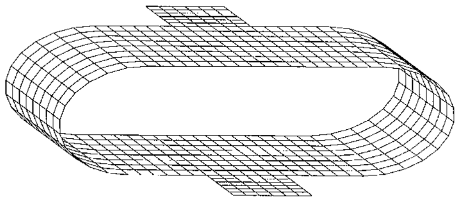

Figure 10. Mesh for the cylinder

The second numerical example is the filling of the shell shown in Figure 10. The finite element mesh consists in this case of 138 Q_2/P_1 elements and 1872 nodal points (624 in the mid-surface). The physical properties are the same as in the previous simulation. In this case, no thermal calculation has been performed. The maximum diameter of the cylinder is 100 and the width $2a = 2$. The injection velocity is 1000. The time-step size employed is $0{\cdot}01$, again using the backward Euler time integration scheme.

The position of the fluid front at times $t = 0{\cdot}01$, $0{\cdot}1$, $0{\cdot}2$, $0{\cdot}3$, $0{\cdot}4$, $0{\cdot}6$, $0{\cdot}7$ and $0{\cdot}8$ is shown in Figure 11. The pressure contours and velocity field at $t = 0{\cdot}4$ and $0{\cdot}7$ are shown in Figure 12. When the air is in contact with the boundary, we let it leave the domain freely, whereas when

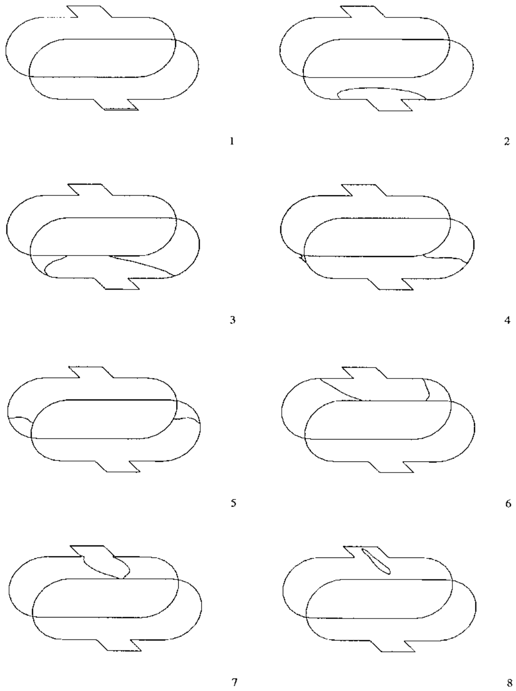

Figure 11. Position of the fluid front at $t = 0{\cdot}01$, $0{\cdot}1$, $0{\cdot}2$, $0{\cdot}3$, $0{\cdot}4$, $0{\cdot}6$, $0{\cdot}7$ and $0{\cdot}8$

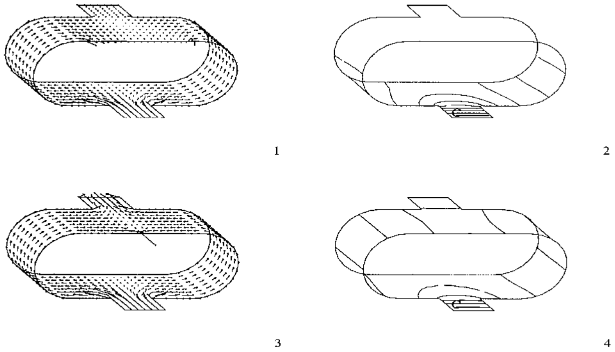

Figure 12. Velocity vectors (right) and pressure contours (left) at $t = 0{\cdot}4$ (top) and $t = 0{\cdot}7$ (bottom)

the fluid filling the mould touches the boundary we change the boundary conditions and use the no-slip condition instead of the free outflow. This explains the difference in the flow-field pattern from time step $t = 0{\cdot}4$–$0{\cdot}7$.

6. SUMMARY AND CONCLUSION

In this paper we have discussed some aspects related to the implementation of a finite element model to simulate the filling of thin moulds. Apart from the general approximation of the

Navier–Stokes equations for incompressible fluids, the specific items related to the fact that the mould to be filled is *thin* can be summarized as follows:

(1) Only *one quadratic element* is used along the width of Ω.
(2) The velocity is prescribed to zero at the end-points of $\mathscr{I}$.
(3) The velocity is prescribed to be tangent to $\mathscr{S}$.
(4) The pressure is taken as constant along the width of Ω when div-stable velocity–pressure pairs are used. Nothing needs to be done when stabilized formulations are employed.
(5) A constant velocity along the width of Ω is used for the advection of the function ψ. This velocity is equal to $\frac{2}{3}$ of the velocity at $\mathscr{S}$.
(6) The velocity used in the element matrices for the advection of ψ is the projection of the velocity on the mid-plane of the elements.

These are the only modifications that have to be introduced in a general 3-D code for mould filling. Using them the problem has the same number of degrees of freedom as a 2-D one, incorporating automatically assumptions H1–H4 described in Section 2 for the flow over a surface. Numerical experiments have shown the effectiveness of the proposed methodology, even though the effects of the curvature of the shell are not exactly taken into account as it would happen if the continuous equations for the flow in a surface were discretized.

REFERENCES

1. Nakayama PI, Romero NC. Numerical method for almost three-dimensional incompressible fluid flow and a simple internal obstacle treatment. *Journal of Computational Physics* 1971; **8**:230–240.
2. Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
3. Hughes TJR, Franca LP, Balestra M. A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška–Brezzi condition: a stable Petrov–Galerkin formulation for the Stokes problem accommodating equal-order interpolations. *Computer Methods in Applied Mechanics and Engineering* 1986; **59**:85–99.
4. Codina R, Schäfer U, Oñate E. Mould filling simulation using finite elements. *International Journal of Numerical Methods for Heat and Fluid Flow* 1994; **4**:291–310.
5. Dvorkin E, Petöcz EG. On the modelling of 2d metal forming processes using the flow formulation and the pseudo-concentration technique. In *COMPLAS III, Proceedings of the Third International Conference on Computational Plasticity*, Pineridge Press/CIMNE: Barcelona, Spain, 1992.
6. Thompson E, Smelser RE. Transient analysis of forging operations by the pseudo-concentration method. *International Journal for Numerical Methods in Engineering* 1988; **25**:177–189.
7. Dhatt G, Gao DM, Ben Cheikh A. A finite element simulation of metal flow in moulds. *International Journal for Numerical Methods in Engineering* 1990; **30**:821–831.
8. Lewis RW, Usmani AS, Cross JT. Efficient mould filling simulation in metal castings by an explicit finite element method. *International Journal for Numerical Methods in Engineering* 1995; **20**:493–506.
9. Brezzi F, Fortin M. *Mixed and Hybrid Finite Element Methods*. Springer: New York, 1991.
10. Codina R, On stabilized finite element methods for linear systems of convection–diffusion–reaction equations. *Computer Methods in Applied Mechanics and Engineering*, submitted.
11. Codina R, Oñate E, Cervera M. The intrinsic time for the streamline upwind/Petrov–Galerkin formulation using quadratic elements. *Computer Methods in Applied Mechanics and Engineering* 1992; **94**:239–262.
12. Codina R. Comparison of some finite element methods for solving the diffusion–convection–reaction equation. *Computer Methods in Applied Mechanics and Engineering* 1998; **156**:185–210.
13. Brooks AN, Hughes TJR. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equation. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**:199–259.
14. Hughes TJR. Multiscale phenomena: Green's function, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized formulations. *Computer Methods in Applied Mechanics and Engineering* 1995; **127**:387–401.