INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
*Int. J. Numer. Meth. Fluids* 2001; **36**: 1–33

# An adaptive Lagrangian boundary element approach for three-dimensional transient free-surface Stokes flow as applied to extrusion, thermoforming, and rheometry

Roger E. Khayat[a,*,1] and Delphine Genouvrier[b]

[a] *Department of Mechanical and Materials Engineering, The University of Western Ontario, London, Ontario, Canada*
[b] *Laboratoire de Mecanique et Materiaux, Division Structures, Ecole Centrale de Nantes, Nantes Cedex, France*

## SUMMARY

An adaptive (Lagrangian) boundary element approach is proposed for the general three-dimensional simulation of confined free-surface Stokes flow. The method is stable as it includes remeshing capabilities of the deforming free surface and thus can handle large deformations. A simple algorithm is developed for mesh refinement of the deforming free-surface mesh. Smooth transition between large and small elements is achieved without significant degradation of the aspect ratio of the elements in the mesh. Several flow problems are presented to illustrate the utility of the approach, particularly as encountered in polymer processing and rheology. These problems illustrate the transient nature of the flow during the processes of extrusion and thermoforming, the elongation of a fluid sample in an extensional rheometer, and the coating of a sphere. Surface tension effects are also explored. Copyright © 2001 John Wiley & Sons, Ltd.

KEY WORDS: free-surface Stokes flow; Lagrangian method; polymer processing; remeshing

## 1. INTRODUCTION

The numerical simulation of transient free surface flow problems remains challenging despite the advent of powerful techniques. Typically, a boundary value problem of the moving type, which involves geometrical non-linearities, must be addressed. In contrast to conventional problems in fluid dynamics, the domain of computation, which is bounded in part by the free surface, is not known *a priori* since the shape of the free surface itself must be determined as part of the solution. A number of iterations are usually needed in order to reach the precise form of the free surface. The problem becomes even more challenging when, in addition, the shape of the free surface evolves with time, generating large distortions in the discretized

* Correspondence to: Department of Mechanical and Materials Engineering, The University of Western Ontario, London, Ontario, Canada N6A 5B9.
[1] E-mail: rkhayat@eng.uwo.ca

*Received 9 November 1999*
*Revised 15 June 2000*

domain of the fluid. It is needless to say that while large distortions have been reasonably well handled for two-dimensional free surface flows, major issues remain regarding complex three-dimensional problems.

Several numerical techniques have been developed for the solution of moving boundary/initial value problems. These techniques may be classified as Eulerian, Lagrangian, and mixed Eulerian–Lagrangian [1]. In the Eulerian description of the flow the grid points remain stationary or move in a predetermined manner [2–5]. Typically, the fluid moves in and out of the computational cells. The method can handle arbitrarily large free surface deformations without loss of accuracy. Its main disadvantage, however, is the lack of sharp definition of the free surface, and the consequent difficulty of imposing the kinematic and dynamic boundary conditions on the free surface. In the Lagrangian approach, the grid points move with local fluid particles [6,7]. The free surface is sharply defined and it is easy to impose the necessary boundary conditions. However, Lagrangian methods require mesh refinement or remeshing for large deformations of the free surface. Hybrid methods have also been developed that combine the advantages of the Eulerian and Lagrangian methods [8]. Generally, an adaptive Lagrangian approach becomes difficult to implement when a volume method, such as the finite element method (FEM), is used. On the other hand, the boundary element method (BEM) is much easier to use along with adaptive remeshing as the dimension of the problem is reduced by one.

The BEM relates velocities at points within the fluid to the velocity and stress on the bounding surfaces. It is thus an ideal method for studying moving-boundary problems, where the velocity on the free surface is the quantity of prime interest. The advantages of the BEM include a reduction of problem dimensionality, a direct calculation of the interfacial velocity, the ability to track large surface deformations, and the potential for easy incorporation of interfacial tension as well as other surface effects.

The present paper is part of a series of studies on the applicability of the BEM to problems of the moving-boundary type. Such problems include the planar deformation of a drop in a confined medium [9–11], gas-assisted injection molding [12], air venting during blow molding and thermoforming [13], and the transient mixing of Newtonian and viscoelastic fluids [14]. The present work addresses the numerical solution of a class of moving-boundary problems in a confined medium of the free-surface type. An adaptive Lagrangian boundary element approach is adopted to determine the evolution of the free surface. The formulation and numerical implementation are illustrated for a flow advancing inside and exiting a confining cavity.

There is a number of simplifying assumptions to be adopted for the BEM to become applicable. The inherent transient nature of the flow process and the presence of a moving free surface make the simulation of the problem challenging because of the non-linearities involved [1]. The challenge becomes even greater if both inertia and viscoelastic (or, more generally, non-Newtonian) effects are accounted for. Such non-linear phenomena have been addressed in moving-boundary problems with relevance to polymer processing. For instance, viscoelastic effects were examined on the growth of spherical and cylindrical shells of a fluid obeying a highly non-linear viscoelastic constitutive model [15]. It was found that even under constant driving pressure oscillatory growth results from elastic normal stress effects. In order to assess the mathematical intricacies in the case of pressure-driven flows, Khayat [16] examined the

 

small planar deformation of a viscoelastic column of fluid obeying the upper-convected Maxwell fluid by applying a regular perturbation approach. It was found that the governing equations are indeed hyperbolic and therefore, unlike the Navier–Stokes equations, they can entertain an oscillatory solution for a statically stressed fluid. The same problem was later examined for large deformations using the FEM [17]. Non-linear effects, such as those stemming from fluid elasticity, fluid inertia, and shear thinning, are difficult to account for in a boundary element approach despite the advent of recent techniques to handle non-linear and transient problems [18–21].

A simple algorithm for adaptive refinement of the (two-dimensional) triangular mesh of the free surface is implemented. A number of algorithms for the adaptive generation of triangular meshes have been proposed before. Some methods generate entirely new meshes [22,23], others subdivide elements of an initial mesh [24,25], and still others can both generate new meshes and subdivide the meshes adaptively [26]. A comprehensive survey of automatic mesh generation algorithms is given by Sheppard [27]. In the present study, the initial mesh is assumed to be regular so that no initial mesh refinement is needed. As the free surface evolves with time, the triangular elements grow and become distorted. Beyond a prescribed level of distortion, an element is subdivided into two elements. Additional nodes are added only at the mid-side of the longest edge of the element. This method ensures the preservation of a good aspect ratio for the elements in the mesh. Rivara [25] has implemented this method but his procedure requires a search process to repair any incompatibilities that may be generated. Dow and Byrd [28] also employed this approach in the refinement of regions with simple rectangular boundaries.

The paper is organized as follows. The problem formulation is presented in Section 2, where basic equations and boundary conditions are covered. The solution procedure, including the time marching scheme, adaptive meshing of the free surface, determination of local curvature, and the contact problem are discussed in Section 3. Numerical assessment and results are covered in Section 4. Finally, some concluding remarks are given in Section 5.

## 2. PROBLEM FORMULATION

In this section, the governing equations, domain description, and boundary conditions are reviewed, together with some of the assumptions taken for the moving-boundary flow of viscous incompressible Newtonian fluids. Only low-Reynolds number flows, typically characterized by small velocities, small length scales, and/or high viscosity, will be considered. In this limit, the inertia terms in the momentum equation are negligible, so the flow is in a state of creeping motion. The formulation is thus limited to Stokes flow.

### *2.1. Governing equations*

At any instant, $t$, the fluid is assumed to occupy a three-dimensional region, $\Omega(t)$, which is bounded by $\Gamma(t)$. It is convenient to take $\Omega(t)$ as the inner domain, excluding $\Gamma(t)$. Thus, $\Omega(t) \cup \Gamma(t)$ constitutes the domain occupied by the fluid. The fluid is taken to be neutrally buoyant so the effects of gravity and any external body forces are negligible. The conservation of mass and linear momentum equations are given by

 

$$\nabla\cdot\mathbf{u}(\mathbf{x}, t) = 0, \qquad \nabla\cdot\sigma(\mathbf{x}, t) = \mathbf{0}, \quad x\in\Omega(t)\cup\Gamma(t) \tag{1}$$

where $\nabla$ is the gradient operator, $\mathbf{x}$ is the position vector, $\mathbf{u}(\mathbf{x}, t)$ is the velocity vector, and $\sigma(\mathbf{x}, t)$ is the total stress tensor given in terms of the hydrostatic pressure $p(\mathbf{x}, t)$ and excess stress tensor $\tau(\mathbf{x}, t)$. Here $\sigma(\mathbf{x}, t) = -p(\mathbf{x}, t)\mathbf{I} + \tau(\mathbf{x}, t)$, where $\mathbf{I}$ is the identity tensor. In the present study, the fluid is assumed to be Newtonian, so that

$$\tau(\mathbf{x}, t) = \mu \lfloor \nabla\mathbf{u}(\mathbf{x}, t) + \nabla\mathbf{u}^T(\mathbf{x}, t) \rfloor, \quad \mathbf{x}\in\Omega(t)\cup\Gamma(t) \tag{2}$$

where $\mu$ is the viscosity of the fluid. The superscript $T$ denotes the transpose of the matrix. It is important to note that the acceleration term $\partial\mathbf{u}/\partial t$ in the momentum conservation equation has been neglected, so that for a Newtonian fluid the formulation in question is not strictly unsteady but quasi-steady. This quasi-steady state assumption is valid whenever $L^2/\nu \ll T$, where $L$ and $T$ are typical characteristic length and time of the flow, and $\nu = \mu/\rho$ is the kinematic viscosity ($\rho$ being the density). In the case of cavity flow, $T \sim L/U$, where $U$ is a typical value of the driving velocity. Thus, for the quasi-steady state assumption to be applicable, one must have $UL/\nu \ll 1$. This is indeed typically the case for fluids of interest to materials processing. Note also that this inequality is implied by the smallness of the Reynolds number. Physically, the quasi-steady state approximation means that a Newtonian fluid immediately adjusts to changes in the movement of the boundary or boundary conditions.

### *2.2. Domain of computation of the free-surface flow problem*

There are various classes of free-surface flow problems that can be considered by the proposed formulation, with direct relevance to polymer processing. Most notable examples are conventional and gas-assisted injection molding, coating, extrusion, thermoforming, and blow molding. These problems may be conveniently divided into two categories: continuous and discrete flows. A continuous flow involves the growing of a domain as a result of an extended influx of fluid over a period of time, such as during injection molding and extrusion. In contrast, a discrete flow is induced by the action of an external force, usually pressure, which acts on a given and constant amount of fluid, such as during gas-assisted injection molding, thermoforming, and blow molding. The continuous flow is illustrated in Figure 1 and the discrete flow is illustrated in Figure 2.

The flow in Figure 1 is reminiscent of the filling stage during injection molding or the developing free-surface flow inside a duct. The boundary $\Gamma(t)$ is composed at any time of a part spanning the source region, $\Gamma_s$, the wetted part of the cavity, $\Gamma_w(t)$, and the moving front, $\Gamma_f(t)$. The wetted part of the cavity is taken to be time-dependent since it grows as more fluid comes in contact with the wall. The overall boundary may thus be expressed as $\Gamma(t) = \Gamma_s\cup\Gamma_w(t)\cup\Gamma_f(t)$. The liquid occupying the domain $\Omega(t)\cup\Gamma(t)$ is called the melt, in reference to polymeric liquids usually involved in injection molding.

The discrete flow process is shown in Figure 2, typically as encountered during thermofoming, blow molding, and gas-assisted injection molding, and is also closely related to Taylor–Safman viscous fingering. In practice, the discrete process (such as gas-assisted injection molding) consists of an initial stage of a continuous process (such as conventional injection

 

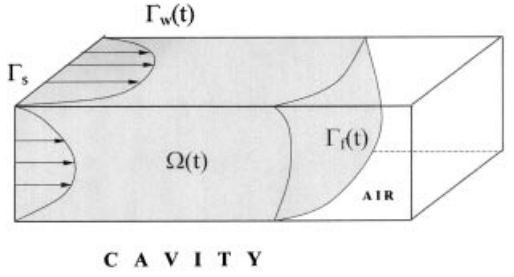


Figure 1. Continuous transient free-surface flow illustrated schematically with notations used. The flow is reminiscent of the cavity-filling stage during the injection molding process or the developing free-surface flow inside a duct.

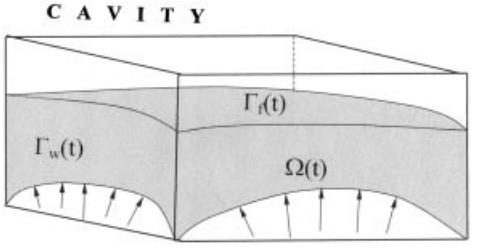


Figure 2. Discrete transient free-surface flow illustrated schematically with notations used. The flow is reminiscent of the gas-assisted injection molding process, thermoforming or blow molding.

molding), during which the melt is driven inside the cavity similarly to Figure 1 [35,36]. At some stage of the process, the melt stops being injected. Instead, the existing amount of liquid is driven by a gas (usually nitrogen), as depicted from Figure 2. There is a wetted part of the cavity, as well as a free-surface (front), but there is no source of fluid. Instead, there is a moving (gas/melt) interface, $\Gamma_f(t)$. In this case, $\Gamma(t) = \Gamma_f(t) \cup \Gamma_w(t) \cup \Gamma_f(t)$. The driving pressure is often assumed constant throughout the gas region. In principle, the motion of the gas behind the melt, and that of the air ahead, must both be accounted for, but, as a first approximation, only the flow inside the melt region is examined.

### *2.3. Boundary and initial conditions*

While the boundary conditions on the source and wetted cavity are straightforward to implement, those on the melt front and gas/melt interface must be examined more closely. The velocity is assumed to be fully prescribed along the source boundary, $\Gamma_s$. A general flow, $\mathbf{u}_s(\mathbf{x})$, usually Poiseuille flow, is imposed

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_s(\mathbf{x}), \quad \mathbf{x} \in \Gamma_s \tag{3a}$$

 

If a gas/melt interface exists instead of a source, a driving (gas) pressure, $\Delta p$, is assumed to apply such that the traction at the interface is given by

$$\mathbf{t}(\mathbf{x}, t) = -\Delta p \mathbf{n}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma_i(t) \tag{3b}$$

Here $\mathbf{t}(\mathbf{x}, t) = \sigma(\mathbf{x}, t) \cdot \mathbf{n}(\mathbf{x}, t)$ is the traction, $\mathbf{n}$ is the normal unit vector at $\Gamma_i(t)$. Equation (3b) represents the dynamic condition at the interface. The fluid is assumed to adhere to the cavity boundary, so that stick and no-penetration boundary conditions apply on the wetted part of the cavity

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{0}, \quad \mathbf{x} \in \Gamma_w(t) \tag{4}$$

The proper choice and implementation of a kinematic condition is generally not obvious [1].

The dynamic conditions on the free surface (melt front) are based on the continuity of the tangential stress and discontinuity of normal stress caused by the interfacial tension

$$\mathbf{t}(\mathbf{x}, t) = \gamma \mathbf{n}(\mathbf{x}, t) \nabla \cdot \mathbf{n}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma_f(t) \tag{5}$$

where $\mathbf{n}$ is the normal vector outward to the front and $\gamma$ is the surface tension coefficient. Note that boundary condition (5) is derived under conditions of equilibrium and uniform interfacial tension. Its validity under dynamic conditions is simply assumed [29,30]. The condition also assumes implicitly that the flow activity of the fluid outside the moving boundary (air) is negligible with the (atmospheric) pressure taken as zero.

In addition, the kinematic condition is needed, which relates the evolution of the moving boundary to the local velocity field. The moving boundary deforms in accordance with the instantaneous velocity field, thus determining new moving boundary positions with time. In a Lagrangian representation, the moving boundary may be assumed to deform with the fluid velocity, such that the evolution of $\Gamma_f(t)$ and $\Gamma_i(t)$ is governed by Equation (6a)

$$\frac{d\mathbf{x}}{dt} = \mathbf{u}(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma_f(t) \cup \Gamma_i(t) \tag{6a}$$

Although easy to implement, the resulting scheme based on Equation (6a) tends to sweep points on the moving boundary along the tangent to the moving boundary, even if only small shape changes take place. Consequently, frequent redistribution of the moving boundary points or remeshing becomes necessary.

Alternatively, the moving boundary can be assumed to deform pointwise along the normal with the normal projection of the fluid velocity at the moving boundary [1]. This method keeps the points evenly distributed on the moving boundary. Thus, the following kinematic boundary condition results:

$$\frac{d\mathbf{x}}{dt} = \mathbf{n}(\mathbf{x}, t)[\mathbf{n}(\mathbf{x}, t) \cdot \mathbf{u}(\mathbf{x}, t)], \quad \mathbf{x} \in \Gamma_f(t) \cup \Gamma_i(t) \tag{6b}$$

 

Unlike Equation (6a), the above equation prevents the nodes to be swept along the tangent to the moving boundary. However, it leads to numerical instability of the saw-tooth type. This difficulty is usually circumvented by applying a smoothing technique of the moving boundary. Given the remeshing capabilities of the present approach, the use of Equation (6a) turned out to be the good choice for the class of problems covered in this study.

As to the initial conditions, when the fluid is assumed to be at rest initially, the following condition is used:

$$\mathbf{u}(\mathbf{x}, t=0) \equiv \mathbf{0}, \quad \mathbf{x} \in \Omega(t=0) \cup \Gamma(t=0) \tag{7}$$

Thus, the flow field is determined through the solution of Equations (1) and (2) subject to initial condition (7) and the boundary conditions above using the boundary integral method.

### *2.4. Boundary integral equation*

The general time-dependent integral equation for a moving domain is given by [31]

$$\int_{\Gamma(t)} \mathbf{t}(\mathbf{y}, t) \cdot \mathbf{J}(\mathbf{x}|\mathbf{y}) \, \mathrm{d}\Gamma_{\mathbf{y}} - \int_{\Gamma(t)} \mathbf{n}(\mathbf{y}, t) \cdot \mathbf{u}(\mathbf{y}, t) \cdot \mathbf{K}(\mathbf{x}|\mathbf{y}) \, \mathrm{d}\Gamma_{\mathbf{y}} = c(\mathbf{x}, t) \cdot \mathbf{u}(\mathbf{x}, t), \quad \mathbf{x} \in \Omega(t) \cup \Gamma(t) \tag{8}$$

Here $\mathbf{J}$ and $\mathbf{K}$ are the usual symmetric and anti-symmetric tensors with respect to $\mathbf{r} = \mathbf{x} - \mathbf{y}$, and are given as [31]

$$\mathbf{J}(\mathbf{x}|\mathbf{y}) = \frac{1}{8\pi\mu}\left(\frac{\mathbf{I}}{r} - \frac{\mathbf{rr}}{r^3}\right), \qquad \mathbf{K}(\mathbf{x}|\mathbf{y}) = \frac{3}{4\pi}\frac{\mathbf{rrr}}{r^5} \tag{9}$$

where $r = |\mathbf{r}|$. The function $c(\mathbf{x},t)$, for $\mathbf{x} \in \Gamma(t)$, depends on the geometrical form of the boundary; its value arises from the jump in the value of the velocity integrals as the boundary is crossed. When the boundary is Lyapunov-smooth, which requires that a local tangent to the free surface exists everywhere, the function $c(\mathbf{x}, t) = 1/2$. This is the case if constant boundary elements are used. Thus, the assumption of boundary smoothness is generally not valid in the vicinity of sharp corners, cusps, or edges. In general, since $c(\mathbf{x}, t)$ depends solely on geometry, it may be evaluated assuming that a uniform velocity field such as $\mathbf{u}(\mathbf{x}, t) = u\mathbf{e}$ is applied over the boundary, $\mathbf{e}$ being the direction of the velocity and $u$ is its magnitude. Under these conditions, all derivatives (including tractions and stresses) must vanish. Hence, at any time $t$, Equation (8) reduces to

$$c(\mathbf{x}, t) = \int_{\Gamma(t)} \mathbf{n}(\mathbf{y}, t) \cdot [\mathbf{e} \cdot \mathbf{K}(\mathbf{x}|\mathbf{y}) \cdot \mathbf{e}] \, \mathrm{d}\Gamma_{\mathbf{y}}, \quad \mathbf{x} \in \Gamma(t) \tag{10}$$

Thus, at any time $t$, the form of the boundary $\Gamma(t)$ is determined and the function $c(\mathbf{x}, t)$ is evaluated using the above equation.

 

## 3. SOLUTION PROCEDURE

In this section, a time marching scheme is proposed to discretize Equation (6). Once the flow field is determined at a given time step from Equation (8), the location of the moving boundary can be determined by solving Equation (6). As the boundary elements are distorted, the mesh is refined through element subdivision. The problem of contact between evolving moving boundary and surrounding cavity walls is also discussed.

### *3.1. Time marching scheme and moving boundary evolution*

Consider now the application of the integral equation (8) for a point on the boundary, i.e., for $\mathbf{x}\in\Gamma(t)$. The discussion is limited to the cavity-filling problem in Figure 1, with obvious extensions to other problems. The flow field at any interior point $\mathbf{x}\in\Omega(t)$ is obtained once the variables at the boundary are known. Since the velocity is fully prescribed on $\Gamma_s\cup\Gamma_w(t)$, only the traction (or stress) will be determined there. The traction is imposed on the moving boundary, $\Gamma_f(t)$, where the value of the velocity will be found. More explicitly, Equation (8) may be rewritten as

$$\begin{aligned}&\int_{\Gamma_s\cup\Gamma_w(t)}\mathbf{t}(\mathbf{y},t)\cdot\mathbf{J}(\mathbf{x}|\mathbf{y})\,d\Gamma_{\mathbf{y}}-\int_{\Gamma_f(t)}\mathbf{u}(\mathbf{y},t)\cdot[\mathbf{n}(\mathbf{y},t)\cdot\mathbf{K}(\mathbf{x}|\mathbf{y})]\,d\Gamma_{\mathbf{y}}\\&\quad+\gamma\int_{\Gamma_f(t)}[\mathbf{n}(\mathbf{y},t)\nabla\cdot\mathbf{n}(\mathbf{y},t)]\cdot\mathbf{J}(\mathbf{x}|\mathbf{y})\,d\Gamma_{\mathbf{y}}+\int_{\Gamma_s}\mathbf{u}_s(\mathbf{y})\cdot[\mathbf{n}(\mathbf{y})\cdot\mathbf{K}(\mathbf{x}|\mathbf{y})]\,d\Gamma_{\mathbf{y}}\\&=\begin{cases}c(\mathbf{x},t)\mathbf{u}_s(x), & \mathbf{x}\in\Gamma_s\\ \mathbf{0} & \mathbf{x}\in\Gamma_w(t)\\ c(\mathbf{x},t)\mathbf{u}(\mathbf{x},t), & \mathbf{x}\in\Gamma_f(t)\end{cases}\end{aligned}\tag{11}$$

where conditions (3) and (4) are used. The unknowns in Equation (11) are thus $\mathbf{t}(\mathbf{x}\in\Gamma_s\cup\Gamma_w,t)$ and $\mathbf{u}(\mathbf{x}\in\Gamma_f,t)$, so that the values of the third and fourth integrals on the left-hand side are known.

The time derivative in Equation (6) is approximated by an explicit Eulerian finite difference scheme, with higher-order terms in the time increment, $\Delta t$, being neglected. The integral equation (10) relates the velocity and traction at the current time. Once the flow field is determined at each time step, $t$, the position of the moving boundary is updated. The evolution of $\Gamma_f(t)$ is dictated by Equation (6). The updated position of the nodes that belong to the moving boundary is thus determined once the velocity at the moving boundary is obtained from the solution of Equation (11).

The integrals in Equation (11) are discretized into a finite sum of contributing terms over the boundaries. In this work, the boundary elements are assumed to be geometrically linear so that the velocity and traction are constant over each element. This makes the proposed adaptive remeshing method and estimation of curvature less difficult to implement since no interpolation of the flow variables is needed at each time step. The use of higher-order elements is possible, but may not be crucial given the mesh refinement and remeshing capabilities involved

 

in the current procedure. The traction is constant over flat linear element, and is multiply valued at a corner node if higher-order elements are used. In two dimensions, the traction may be assumed to be double valued at every node of a curved boundary. Another advantage of the constant boundary element is that the value of $c(\mathbf{x}, t)$ is always and everywhere equal to $\frac{1}{2}$. In addition, the normal vector to each element is determined exactly.

### *3.2. Adaptive meshing*

A simple algorithm is proposed for adaptive refinement of the triangular mesh of the evolving free surface. The method is similar to that proposed by Nambiar *et al.* [32] for adaptive and *h* refinement of two-dimensional triangular finite element meshes. Initially ($t < 0$), the fluid is assumed to occupy a three-dimensional region, $\Omega_0 = \Omega(t = 0)$, bounded by part of the cavity (source) and free surface. Typically, at $t \geq 0$, additional fluid penetrates $\Omega_0$ (see Figure 1) and the volume starts to grow together with the boundary elements. At some point, some elements become too distorted, and mesh refinement or remeshing is needed.

The refinement is carried out by subdividing the elements that are too distorted. Generally, the criteria for subdivision are based on the element area and the length of the edges. However, it turns out that the length of the longest edge is a reliable criterion by itself. In this case, at each time step of the flow a list of elements is established, with the length of the longest edge greater than an imposed tolerance, $D_{\max}$. Thus, an element is assumed to satisfy the criterion for subdivision if its longest edge is longer than the imposed value of $D_{\max}$. The list is sorted in the order of increasing length of the longest edges of each element. In order to avoid generating mesh incompatibilities or elements with poor aspect ratio during refinement, larger elements are subdivided first. The subdivision starts from the last (i.e., the longest edge) element in the list, and is continued recursively until the list is empty. The subdivision is carried out by bisecting the largest of the edges of the element in the list.

Two distinct cases of element subdivision arise, which are illustrated in Figure 3. The figure displays the subdivision process for a three-dimensional surface that has expanded. In the first case (Figure 3(a)), an element inside the domain (e.g. elements 9 and 21) is subdivided, and in the second case (Figure 3(b)), the element has an edge on the boundary of the domain (element 14). In the latter case, the subdivision process is straightforward. Once the subdivision criteria are reached, the element is simply subdivided into two elements. This is typically illustrated in Figure 3(b) for element 14, which is subdivided into elements 14 and 28. In the former case, the edge of the element is in the interior and additional options must be considered. To prevent creation of any non-conforming interior elements, the element that shares the common longest edge is also bisected along with the first element. Creation of four elements in such a manner is carried out only when the bisected edge is the longest edge of both elements. If this condition is not met, the second element is added to the end of the element list, which contains the distorted elements so that the second element is now the current element for subdivision and the process is repeated. Again, the element selected for subdivision will have an edge that is the longest among the edges of the elements in the list.

In order to facilitate the searching process for the second element and the longest edge of elements, the input data are initially processed to create three data structures, one structure each for nodes, triangles, and edges. A node is defined by its $x$-, $y$-, and $z$-co-ordinate. The

 

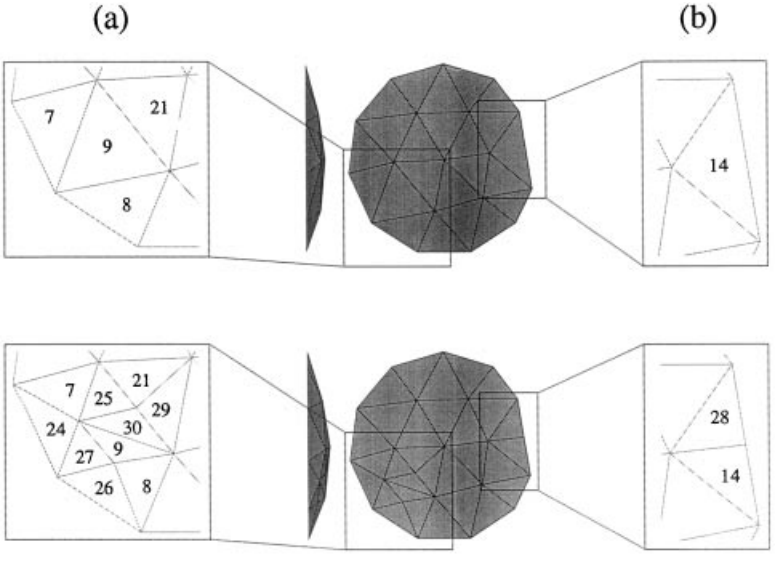


Figure 3. Adaptive meshing and element subdivision of free surface elements, for an element with the longest edge for an internal element (a), and for an element at the boundary (b).

properties selected for a triangle are its three nodes ($A$, $B$, and $C$), its three edges ($AB$, $BC$, and $CA$), and its area. The above information is structured such that the node numbers are ordered in a counterclockwise direction for each element, and edge $AB$ is the longest edge of each element. The properties of an edge are its two end nodes, its two neighboring triangles, and the length of the edge.

The major advantage of dividing only the largest edge in a triangle is that the smallest angle in the original mesh is not further subdivided. If the largest angle $C$ is greater than 90° none of the newly created angles can be smaller than the original smallest angle $A$. If $C$ is equal to 90° the smallest angle $A$ is duplicated as $D$. If, however, $C$ is smaller than 90° the newly created angle $D$ is slightly smaller than $A$. Further, if an angle $A$ is smaller that 60°, it will never be bisected using this algorithm. Rosenberg and Stenger [33] have shown that the smallest angle that can be created in any subsequent mesh produced by the above method is bounded by half the minimum angle present in the initial mesh. Thus, the aspect ratio of the triangles in the mesh remains in an acceptable and known range. The process of subdivision shown in Figure 3(a) illustrates the general sequence of mesh refinement. Elements 7 and 9 are first subdivided each into two triangles; element 7 is now divided into the current elements 7 and 24, while element 9 is divided into one triangle currently composed of 25 and 30, and another currently composed of elements 9 and 27. Subsequently, element 21 is subdivided into elements 21 and 29, along with an element currently composed of 25 and 30. Similarly, element 8 is divided into elements 8 and 26, along with the element currently composed of 9 and 27.

 

The first step in the solution procedure consists of creating the data file containing the description of the problem domain, boundary conditions, loading, and initial mesh. The initial mesh is first examined to check for initially distorted elements. This mesh comes from a computer aided design (CAD) system, such as I-DEAS, PATRAN, or PROENGINEER. The initial mesh is then refined by the adaptive remeshing scheme described above. An example is given in Figure 4, where the discretization of a cylindrical wedge is shown for three values of $D_{\max}$. The mesh ($D_{\max} \to \infty$) corresponds to the coarse mesh provided by I-DEAS with no mesh refinement. The wedge height-to-radius ratio is 0.5. The figure displays two levels of refinement. For $D_{\max} = 1.2$, the triangular (right) elements in the initial mesh are divided into two elements each. When $D_{\max}$ is decreased further to $D_{\max} = 0.7$, each previous element is further subdivided into two.

### *3.3. Determination of local curvature*

The value of the curvature at a particular location (node or element) on the free surface is needed if surface tension is accounted for. The curvature is obviously related to the divergence of the normal vector $\mathbf{n}(x, y, z, t)$ at the location in question. Thus, the determination of the curvature is based on the estimation of the derivative of the normal vector components in the three directions. For this it is convenient to define local co-ordinates $(\xi, \eta, n)$ spanned by the plane tangent to the surface at the local point and the normal to the plane.

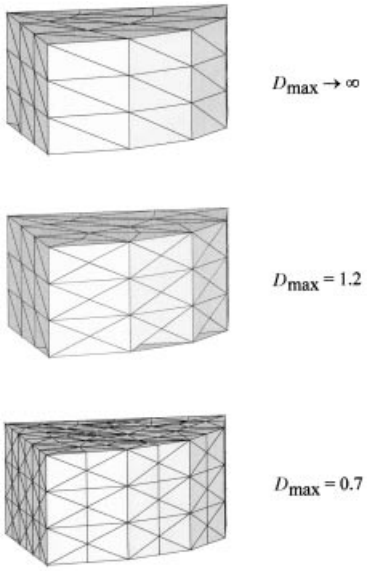


Figure 4. Influence of $D_{\max}$ on element subdivision for a cylindrical wedge of unit radius and height equal to 0.5. The figure shows the initial mesh in the absence of remeshing ($D_{\max} \to \infty$) and two refined meshes for $D_{\max} = 1.2$ and 0.7.

 

Consider now the curvature at an element (centroid) of the discretized surface. The curvature is estimated directly at the centroid of the element rather than on a smooth interpolated surface going through the element vertices. Surface interpolation and fitting can be very costly. Generally, each node of the triangle belongs to an arbitrary number of elements, and the normal at the node is not uniquely defined. The normal vector is then taken as the area average of the normal vectors to the elements to which the node belongs. The normal vector anywhere to the element, with vertices 1, 2, and 3, may then be generally written as

$$\mathbf{n}(\mathbf{x}, t) = \sum_{i=1}^{3} \mathbf{n}_i(t)\phi_i[\xi(\mathbf{x}, t),\ \eta(\mathbf{x}, t)] \tag{12}$$

where $\mathbf{n}_i(t)$ are the normal vectors at node $i$ and $\phi_i(\mathbf{x}, t)$ are suitably introduced interpolation functions. In this work, given the fact that flat triangular elements are used, $\phi_i(\mathbf{x}, t)$ are taken to be linear. The partial derivatives of the normal vector are obtained by differentiating Equation (12). If the $(\xi, \eta)$ co-ordinate axes are taken to lie in the plane of the element, then

$$\nabla\cdot\mathbf{n} = \frac{\partial n_\xi}{\partial \xi} + \frac{\partial n_\eta}{\partial \eta} \tag{13}$$

and the curvature is just given by $-\nabla\cdot\mathbf{n}/2$. This procedure is now validated by computing the curvature for simple surfaces, discretized into triangular flat elements. Two examples are treated; namely, the case of a spherical shell of radius equal to one, and the case of a parabolic surface.

*3.3.1. Case of a spherical surface*. Two mesh sizes are considered for the sphere of radius one. The first mesh consists of 72 elements and 38 nodes, and the second of 272 elements and 138 nodes. The curvature in this case, with the normal vector pointing outward, is equal to $-1$. The computed curvature for the sphere composed of 72 elements lies within the interval $[-0.912, 0.885]$. The average value of the curvature is equal to $-0.903$, which corresponds to a global error of 9.7 per cent. For this mesh, the small dispersion in the numerical values of the curvature indicates that the uniformity in the mesh is preserved. The curvature is computed at the centroid of the elements. Given the construction of the mesh, which places the nodes on a sphere of radius one, the centroids do not coincide exactly with the sphere. The deviation in position between the centroids of the elements and the corresponding position on the sphere is typically equal to 8 per cent. This value is of the same order of magnitude as the error in the estimation of the curvature. The results of the numerical computation of the curvature for this mesh are thus conclusive: the uniformity of the entire mesh is preserved, and the resulting error corresponds to that of the mesh size. This consistency is not always achieved for any mesh as the next case shows.

The influence of mesh size on the computed curvature is examined by considering the mesh of 272 elements. In this case, the computed average curvature is found to be closer to $-1$, namely, $-0.95$. However, the difference between the maximum and the minimum values is larger. The former is equal (in magnitude) to $-1.25$, and the latter is equal to $-0.68$. These

 

extreme values correspond to regions where the elements are not regular. A curvature can have a magnitude that is too small, for instance, two or more elements happen to lie almost in the same plane. The opposite is true when, for instance, the planes of two elements make an angle of less than 90°. These mesh irregularities do indeed happen and there is little that can be done to control them. Localized errors can be large, exceeding 30 per cent for a couple of nodes in the present example. However, if the couple of irregular points are excluded, the average error in the computed curvature is only 3 per cent, which also corresponds to the error in the location of the centroids of the elements relative to a sphere of radius one.

This second mesh confirms that a good estimate of the curvature depends on two major factors: mesh size and regularity. Obviously, a finer mesh leads globally to an accurate estimate of the curvature. However, strong irregularity results in large errors locally. The accuracy of the current method hinges on the accuracy of the normal vectors at the nodes and (their average) over each element. It is thus essential to start with an optimized original mesh.

Indeed, four mesh sizes are examined that help establish the convergence rate with mesh refinement. The meshes are generated with reasonable regularity. The mesh sizes considered correspond to 192, 312, 406, and 600 elements. The resulting average relative error and standard deviation for each mesh size are reported in Table I. The error and standard deviation decrease with mesh refinement, confIrming the convergence of the current procedure for estimating the curvature.

*3.3.2. Case of a parabolic surface*. Consider the case of the surface defined over the octant $x, y, z \geq 0$ by the relation $x = 16y(y-1)z(z-1)$. Comparison between the analytical and numerical curvatures indicates a large disparity in error. The error is found to be much larger along the edges, especially around the corners, compared with the core region. The mesh is not optimized along the edges. Although the mesh chosen is relatively fine, the discretized surface does not adhere well along the edges. As in the case of the sphere, all nodes coincide with the analytically prescribed surface, but the deviation between the element centroids and the surface can exceed 40 per cent in the corner regions. The normal vectors to these elements are thus inaccurately estimated. Consequently, the estimation of the curvature, which is based on that of the normal vectors, is expected to be inaccurate in the corner region.

From a practical point of view, the inaccuracy in the curvature value along the edges and corners for the present case does not pose any problem. This is confirmed upon comparison of the analytical curvature with the computed ones for the four different mesh sizes as shown in Figure 5. The figure indicates clearly that the curvature increases sharply as one moves from the edges to the core region. There is a flattening of the surface along the edges. Consequently,

Table I. Influence of mesh size on the accuracy in estimating the curvature for a sphere.

| Number of elements | Relative error | Standard deviation |
|---|---|---|
| 192 | 0.0302 | 0.0291 |
| 312 | 0.0175 | 0.0170 |
| 406 | 0.0148 | 0.0145 |
| 600 | 0.0125 | 0.0123 |

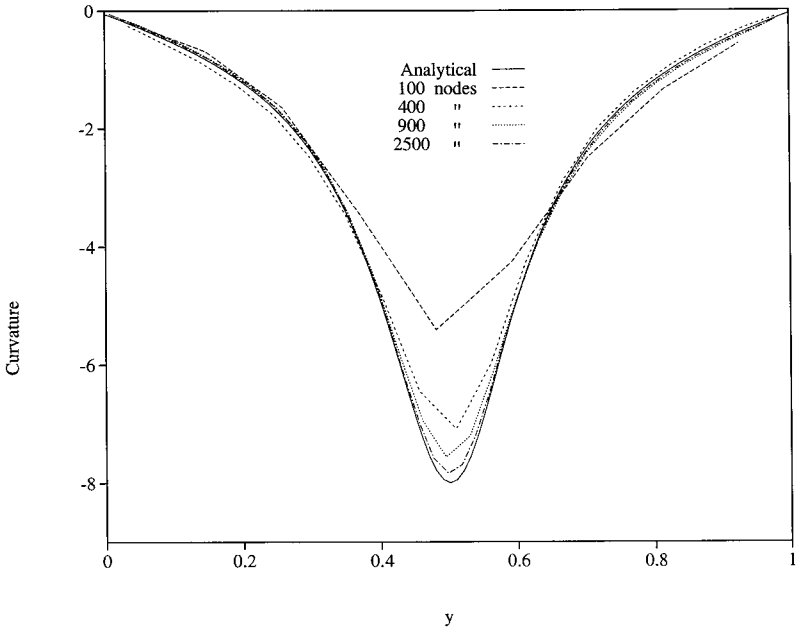


Figure 5. Influence of mesh size on the curvature of a parabolic surface, $z = 16x(x-1)y(y-1)$. The figure shows the curvature vs $y$ along the mid-section at $x = 0.5$.

the curvature becomes small enough to be considered essentially zero, leading to no significant loss in the accuracy in the estimation of surface tension effects.

Generally, the rate of convergence with mesh refinement is good, as Table II indicates. Four meshes are used which consist of 100, 400, 900, and 2500 elements. Each mesh is constructed on the basis of the imposed number of nodes along the $y$- and $z$-axes. The relative error varies from 0.524 for the coarsest mesh to 0.046 for the finest, and the standard deviation varies from 1.910 to 0.044. The curves in Figure 5 also reflect the fast rate of convergence of the proposed algorithm. The figure displays the behavior of the curvature along the $y$-axis at $z = 0$.

In conclusion, the proposed scheme for the estimation of the curvature depends strongly on the mesh of the discretized surface. The major influencing factors that lead to a good estimate

Table II. Influence of mesh size on the accuracy in estimating the curvature for a parabolic surface.

| Number of elements | Relative error | Standard deviation |
|---|---|---|
| 100 | 0.524 | 1.910 |
| 400 | 0.180 | 0.213 |
| 900 | 0.097 | 0.090 |
| 2500 | 0.046 | 0.044 |

 

are: the mesh size, the mesh regularity, and a good estimate of the normal vectors to the elements.

### *3.4. Contact between free surface and confining wall(s)*

Initially, the fluid is assumed to occupy a prescribed volume, $\Omega_0$, which is taken as the starting step for the computation. Typically, $\Omega_0$ is bounded by a source surface and the free surface. The source surface is assumed to be part of the cavity at all times. The initial volume of fluid is thus in contact with the cavity only at the intersection of the free surface and the source surface. The free surface mesh is independent of the source surface mesh. The mesh of the source surface does not change with the subsequent motion. In fact the source area is part of the cavity and its mesh coincides with that of the cavity.

The cavity walls are discretized into an appropriate number of elements that serve only to confine the fluid but do not come into the flow calculation. The number of cavity elements can thus be arbitrarily large, leading to an accurate representation of the cavity shape. This is particularly advantageous for practical situations where the geometry is typically complex. Obviously, the mesh density of the evolving free surface need not match that of the surrounding cavity. In other words, the accuracy in cavity representation is usually far superior to that of the surface bounding the moving fluid. A node or element that comes in contact with the wall is assumed to subsequently adhere to it. Contact is assumed to be established once the fluid has come to within a certain distance to the cavity wall, which is usually taken to be of the order of an element size. In fact, contact usually occurs once the node has intersected and penetrated the cavity surface. Obviously, the error generated in this case is controlled by decreasing the time increment. The influence of the time increment will be assessed below.

The boundary conditions imposed at a given element depend on whether the element belongs to the free surface or the cavity. As mentioned above, initially the domain boundary consists of a free surface and a source surface. At the free surface, conditions (6) and (7) apply. The flow is entirely specified at the source surface. It is either of the Poiseulle or plug flow type. These boundary conditions remain applicable as the flow evolves during the early stages, until the free surface comes in contact with the surrounding cavity wall. At the contact region, which may be simply or multiply connected, the boundary conditions change from those on a free surface to stick conditions at the wall.

## 4. DISCUSSION AND RESULTS

The potential of the formulation and solution procedure is demonstrated for three-dimensional complex flow configurations of the moving boundary type, with emphasis on transient problems from polymer processing. In particular, illustrations are borrowed from the processes of extrusion, thermoforming, extensional flow, as well as surface coating, including the effects of surface tension. The numerical accuracy is first assessed by examining the influence of the mesh size for a developing free-surface flow inside and out of a circular and a square tube. Given the similarity characteristic of the Stokes equations, all results are reported in dimensionless form.

 

### 4.1. Transient extrusion through circular and square dies

The proposed formulation is first illustrated for the flow through and out of a circular and a square die. The flow out of a duct is essentially the transient stage of the extrusion of a solid circular or square cylinder. That is, the flow is examined as the fluid emerges at the exit of the tube. The diameter (side)-to-length ratio of the tube is equal to one. Poiseuille flow is imposed at the entrance to the tube (away from the front). Surface tension effects are neglected at first but will be examined shortly.

Consider first the flow inside and out of a circular die. Note that although the flow is axisymmetric, the calculations are carried out for a three-dimensional problem. The resultant sequence of flows is displayed in Figure 6 at three time stages. The fluid is assumed to flow inside the tube before it actually reaches the tube exit, and then emerges out of it. The tube acts only as a confining wall to the developing flow, and is not shown in the figure. In fact, the tube is generated by the freezing (adherence) of the fluid surface as the fluid reaches the unit radius. Initially, the fluid volume is limited to an almost flat surface. In fact, the free surface is initially taken to be parabolic with a height equal to one-tenth the radius of the base. As the flow develops, the fluid moves freely inside the tube (without touching it) until it comes in contact with the surrounding wall (Figure 6; $t = 0.5$). As the fluid emerges out of the tube (Figure 6; $t = 1.2$) it experiences a degree of swelling at the exit which increases with time. A

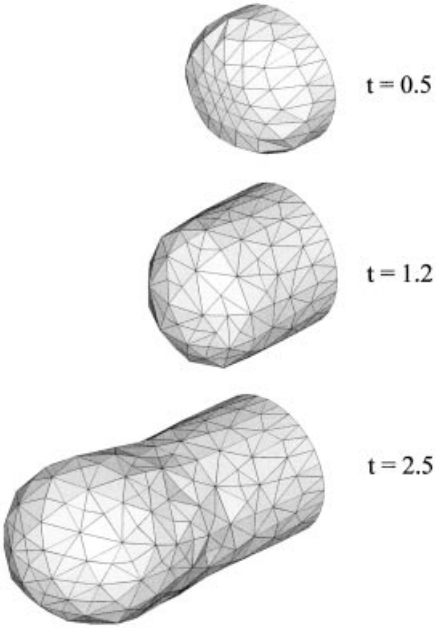


Figure 6. Transient extrusion of a solid circular cylinder with no surface tension. Early stage of flow at $t = 0.5$, intermediate stage at $t = 1.2$ when the fluid begins to exit the tube, and long-term behavior at $t = 2.5$ with almost steady state conditions downstream from the exit.

 

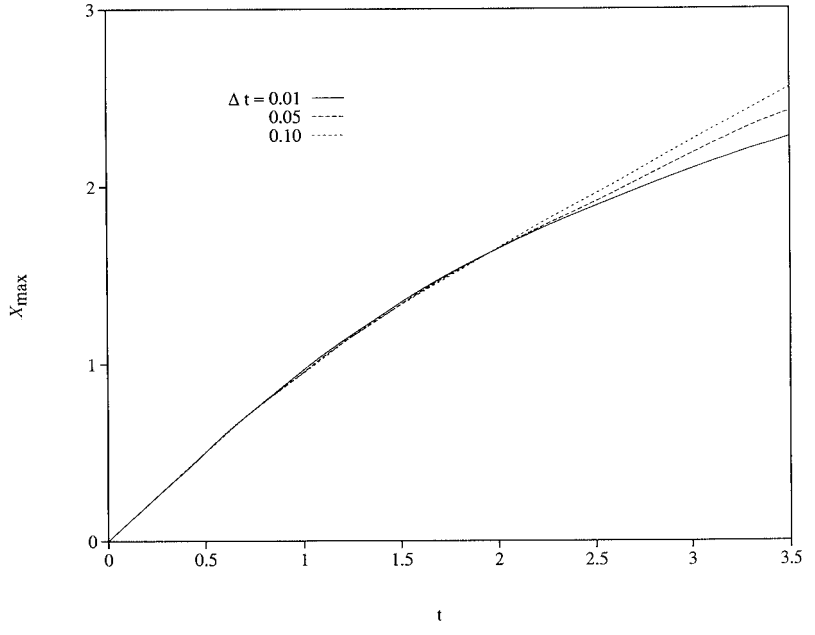


Figure 7. Influence of time increment on the evolution of front tip for the flow in Figure 6. Here $D_{max} = 0.5$ and $0.01 \leq \Delta t \leq 0.1$.

swell ratio of approximately 10 per cent is eventually established as steady state conditions near the exit are established (Figure 6; $t = 2.5$). This is in agreement with existing steady state calculations [34].

The influence of the time increment and mesh size is an important issue for any type of numerical method. These two parameters are now examined as they affect the accuracy of calculations in the present problem. Figure 7 shows the influence of the time increment, $\Delta t$, on the evolution of the front tip position, $X_{max}$, with time equal to over 3.5 time units. In this case, $D_{max}$ is kept equal to 0.5 and $0.01 \leq \Delta t \leq 0.1$. The figure shows, as expected, that the evolution of the front tip (and flow) is affected by the time increment. However, the influence of $\Delta t$ is relatively minimal as Figure 7 suggests; the difference is only a few per cent at most when $\Delta t$ is raised by one order of magnitude, from 0.01 to 0.1. The rather insignificant influence of the time increment is of course encouraging, as considerable CPU can be saved when dealing with complex three-dimensional free-surface flows. At least for the array of problems covered so far, it is found that the influence of time increment depicted in Figure 7 is typical. This relatively little influence of the time increment is contrasted by the strong effect that the mesh size can have on the numerical accuracy.

The influence of mesh size is found generally to be more important than that of the time increment. This influence is depicted from Figure 8 for $D_{max} \in [0.6, 0.75]$. The figure shows the evolution of the relative error on the volume of fluid for each value of $D_{max}$. The error is equal

 

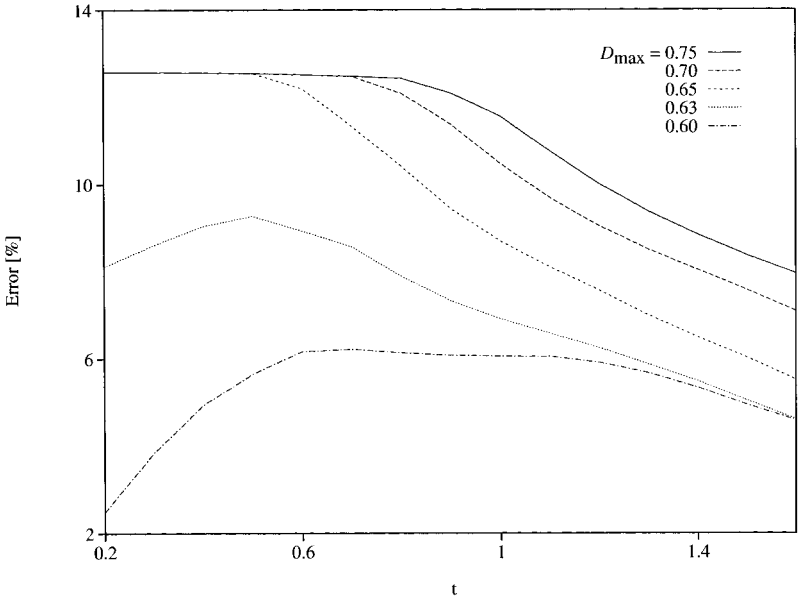


Figure 8. Influence of mesh refinement on the relative error for the extrusion of a solid cylinder. The error is based on the difference between the computed and theoretical volumes of fluid with time. Here $D_{max} \in [0.6, 0.75]$ and $\Delta t = 0.1$.

to the relative difference between the volume of fluid at time $t$ and the theoretical amount of fluid that is supposed to have emerged at the exit of the tube during the time $t$. The figure clearly shows that the overall error decreases as $D_{max}$ decreases. For any $D_{max}$, the error is also found to generally decrease with time, except for the lower $D_{max}$ value considered. There is an initial error that is obviously larger when $D_{max}$ is large, since the volume of the initial fluid is poorly approximated as a result of the crude mesh used in the surface discretization. In this case, the initial error at $t = 0$ is maximum. As $D_{max}$ decreases the maximum in error is shifted from $t = 0$, but a maximum seems to always exist. It is interesting to observe that the influence of $D_{max}$ is greatest in the initial stages of flow. The error appears to continuously decrease at the later stages of the flow. There is an initial error that is obviously larger when $D_{max}$ increases. The initial error (for $t < 0.5$) is the same for $D_{max} = 0.75$, 0.70, and 0.65. This stems from the fact that there is no element subdivision initially for this range of $D_{max}$ values.

Consider next the flow through and out of a square die. The length-to-side ratio is equal to one. Poiseuille flow is imposed upstream of the exit. The resultant sequence of flows is displayed in Figure 9 at three time stages. The fluid is assumed to flow inside the tube before it actually reaches the tube exit, and then emerges out of it. Initially, the fluid volume is limited to an almost flat surface. As the flow develops, the fluid moves freely inside the tube until it comes in contact with the surrounding wall ($t = 0.4$). As the fluid emerges out of the tube ($t > 1$), it experiences a degree of swelling at the exit that increases with time. A swell ratio of

 

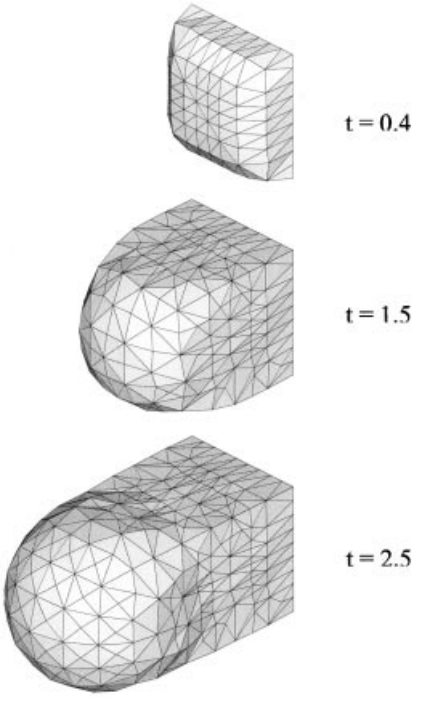


Figure 9. Transient extrusion of a solid square cylinder with no surface tension. Early stage of flow at $t = 0.4$, intermediate stage at $t = 1.5$ when the fluid begins to exit the tube, and long-term behavior at $t = 2.5$ with almost steady state conditions downstream from the exit.

approximately 10 per cent is eventually established as steady state conditions near the exit are established ($t = 2.5$).

Further calculations were also carried out including surface tension effects. Since one of the major objectives of the simulation is to examine the evolution of the shape of the front during a given process, the influence of surface tension becomes important. It is interesting to observe that surface tension plays a similar role to fluid elasticity, as it opposes flow. Here the influence of surface tension is explored for the transient extrusion through a circular and a square die. Figure 10 shows the evolution of the front tip position, $X_{\max}$, during the extrusion of the circular cylinder, over a period of 3.5 time units, as function of time for $Ca^{-1} \in [0, 7]$. The figure indicates clearly the prohibiting effect of surface tension on the motion of the melt front. The figure shows a considerable decrease in the rate of advancement of the fluid tip as surface tension increases.

A similar effect of the surface tension is observed during the extrusion of the square cylinder. Figure 11 shows the flows at $t = 2.5$ for $Ca^{-1} = 0$, 2, and 5, which are labeled (a), (b), and (c) respectively. The shape of the front is drastically influenced by surface tension effect. In fact, in the absence of surface tension, the flow has a tendency to emerge like a jet, thrusting

 

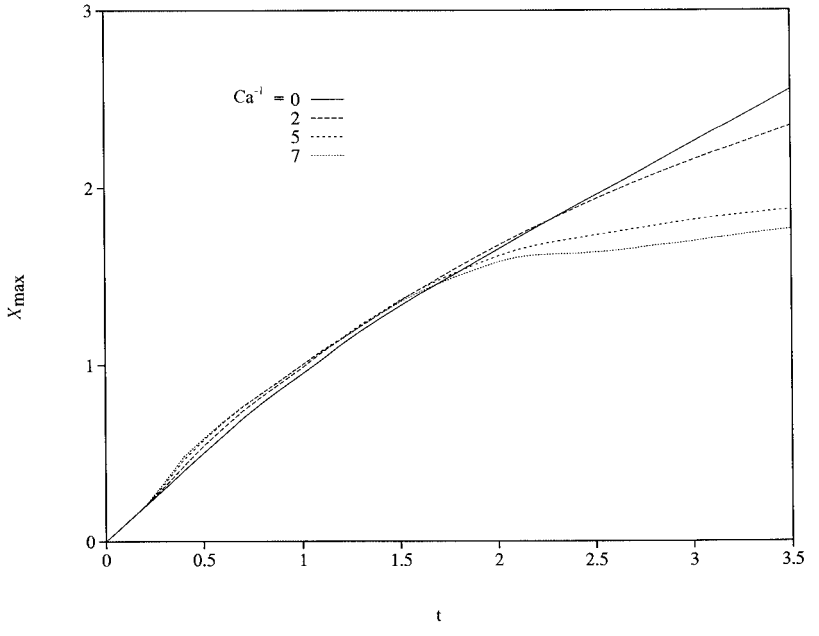


Figure 10. Influence of surface tension on the evolution of the front tip during the extrusion of a solid circular cylinder as in Figure 6. Here $Ca^{-1} \in [0, 7]$.

forward with little swell (roughly 10 per cent). As surface tension is added, the level of swell increases, making the front more rounded, with a swell of more than 12 per cent (Figure 11(b)). Finally, in the presence of higher surface tension ($Ca^{-1} = 5$), the flow tends to move laterally, almost with adverse flow, leading to the mushroom-like shape depicted in Figure 11(c), with a swell of more than 15 per cent. It is not difficult to appreciate the significant influence that surface tension can have in materials processing. This is of course obvious in the present case of the extrusion of a square cylinder. Although surface tension is usually negligible for most liquid melts used in polymer processing, the illustrations in Figures 10 and 11 are still of relevance to interfacial flow, such as during co-extrusion. It is the level of swell that is of interest in practice.

Finally, the effect of surface tension is further analyzed by comparing the swell for circular extrusion to that of square extrusion. Figure 12 shows the flow in both cases at $t = 2.5$ and $Ca^{-1} = 5$. It is interesting to observe that, although the die geometry is different, the shape of the front is very similar. Note that this similarity is not present in the absence of surface tension, as is easily inferred by comparing the flow at $t = 2.5$ in Figures 6 and 9. This comparison shows that the fluid keeps the square shape as it emerges out of the die. In contrast, Figure 12 indicates that surface tension tends to round the edges at the exit of the square die. This is, of course, a result of the relatively high normal force acting in the corner regions, which prevents the fluid from expanding, and letting the fluid far from the corners

 

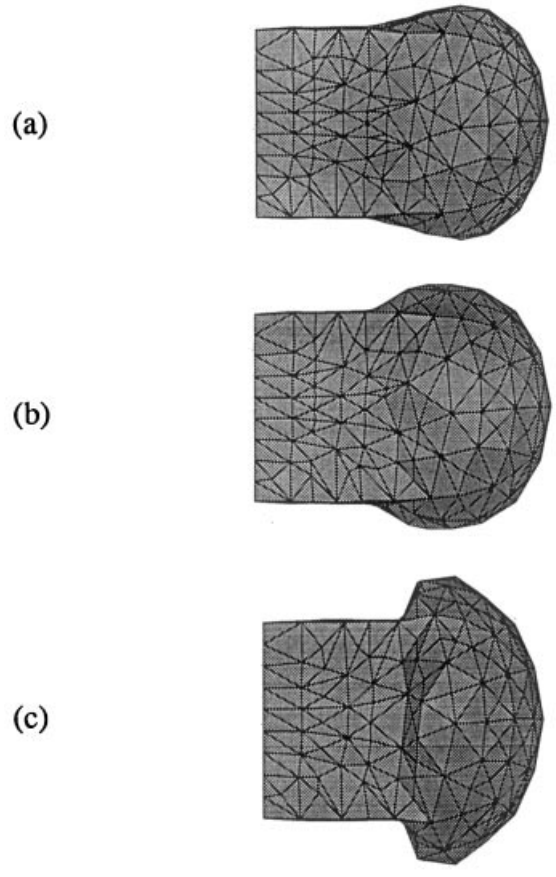


Figure 11. Influence of surface tension for the transient extrusion of a solid square cylinder as in Figure 9 at $t = 2.5$, for $Ca^{-1} = 0$ (a), 0.2 (b), and 5 (c).

swell at a relatively faster rate. Although surface tension effects may not be significant in material processing, especially for polymers, the similarity in Figure 12 can have a drastic implication in practice in processes involving interfacial flow, such as during the co-extrusion or co-injection molding of two materials.

### *4.2. Flow during the thermoforming process*

This is the second category of problems that is addressed in the present study. In this case, the flow is induced by the deformation of a free surface as a result of a pressure force acting on it. In the thermoforming (or blow molding) process, the pressure acts on one of the two free surfaces. The second surface is traction-free. The process consists of forcing the fluid sheet inside a mold until the (traction-free) free surface embraces completely the inner walls of the

 

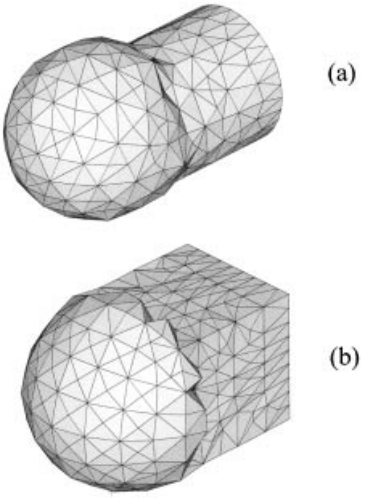


Figure 12. Effect of surface tension on the swell ratio at $t = 2.5$ and $Ca^{-1} = 5$ for the flow out of a circular die (a) and out of a square die (b).

mold cavity. In practice, the thermoformed part undergoes additional deformation, in the form of warpage and shrinkage, which is caused by the build-up of residual stresses resulting from elastic and non-isothermal effects. The present simulation is only an approximation to the real process since the flow is assumed to be inelastic and isothermal.

Consider the thermoforming of a square thick sheet of length-to-thickness ratio equal to 20. The situation is schematically illustrated in Figure 2. The fluid is assumed to deform inside a square cavity of depth equal to twice the sheet thickness. The objective of the simulation is to find the thickness distribution of the deformed fluid medium. The thickness distribution of the thermoformed part is the most crucial characteristic targeted during the process. Obviously, if the initial thickness is uniform, as in the present example, then the final thickness distribution will not be uniform. In practice, the initial thickness is pre-programmed iteratively until the desired final thickness distribution is reached. This results in a large number of costly trial-and-error operations in order to reach optimal processing conditions. The present simulation is an illustration of how CAD can be used to reduce the number of the iterative operations.

The fluid is assumed to be clamped at all times along the four edges. Since the problem is symmetric, only the quarter of the domain is considered. Thus, the initial domain of computation is taken as $[0, 0.1] \times [0, 1] \times [0, 1]$ along $x$, $y$, and $z$ respectively. The pressure is applied on the plane $x = 0$ initially. The faces $y = 0$ and $z = 0$ are fixed. The adherence and no-penetration conditions are applied at the planes $y = 0$ and $z = 0$, whereas symmetry conditions are applied at the planes $y = 1$ and $z = 1$. There is a pressure force, $\mathbf{n}$, of unit strength, which acts continuously on the lower surface, $S_1$, which is located initially at $x = 0$, where $\mathbf{n}$ is the normal unit vector to $S_1$. The second free surface, $S_2$, which is located initially at $x = 0.1$ is assumed to be traction free. Thus

 

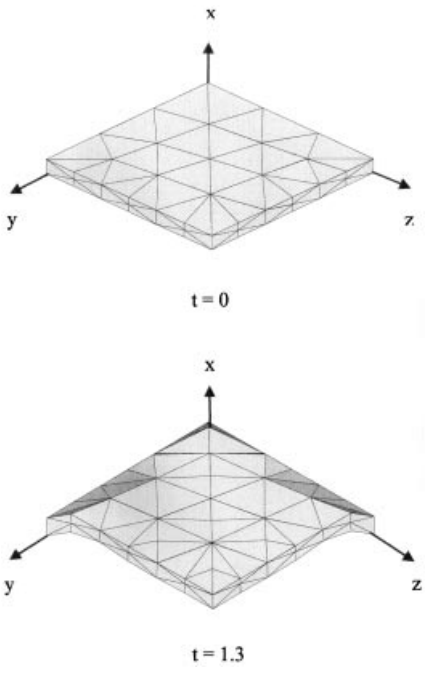


Figure 13. Transient thermoforming of a square sheet. Only one quarter of the domain is shown of unit half side and initial thickness equal to 0.1. Initial shape and mesh ($t = 0$). Final stage, after forming is completed at $t = 1.3$. Here $Ca^{-1} = 0$.

$$\mathbf{t}(\mathbf{x} \in S_1, t) = -\mathbf{n}(\mathbf{x} \in S_1, t), \qquad \mathbf{t}(\mathbf{x} \in S_2, t) = \mathbf{0} \tag{14}$$

These dynamic conditions hold in the early stages of flow. When surface $S_2$ touches the mold, the fluid is assumed to subsequently stick to the mold wall.

Figure 13 shows the initial stage ($t = 0$) and the final stage ($t = 1.3$). The initial number of elements is equal to 116 with 90 nodes. The initial mesh size is unimportant since remeshing is done at any time once a $D_{\max}$ value is imposed. For the present problem, $D_{\max} = 0.3$ and $\Delta t = 0.0015$. This step size turned out to be appropriately small for the free surface nodes not to exceed too much the mold upon contact. Note that the mold here is simple enough not to require any meshing. In this case, the mold geometry is specified plane by plane. However, the algorithm can handle any mold shape, which, when complex, must be geometrically specified through a proper mesh.

From a practical point view, the thickness is probably the most important characteristic of a thermoformed part. The evolution of the thickness in the mid-section of the part is shown in Figure 14. The figure shows the section of the part in the $(x, y)$ plane at $z = 1$ between the initial and final time, after 1.1. The early stage, corresponding to unconfined free surface flow is typically shown at $t = 0.0025$. This stage lasts until (roughly) 0.004, when the upper free

 

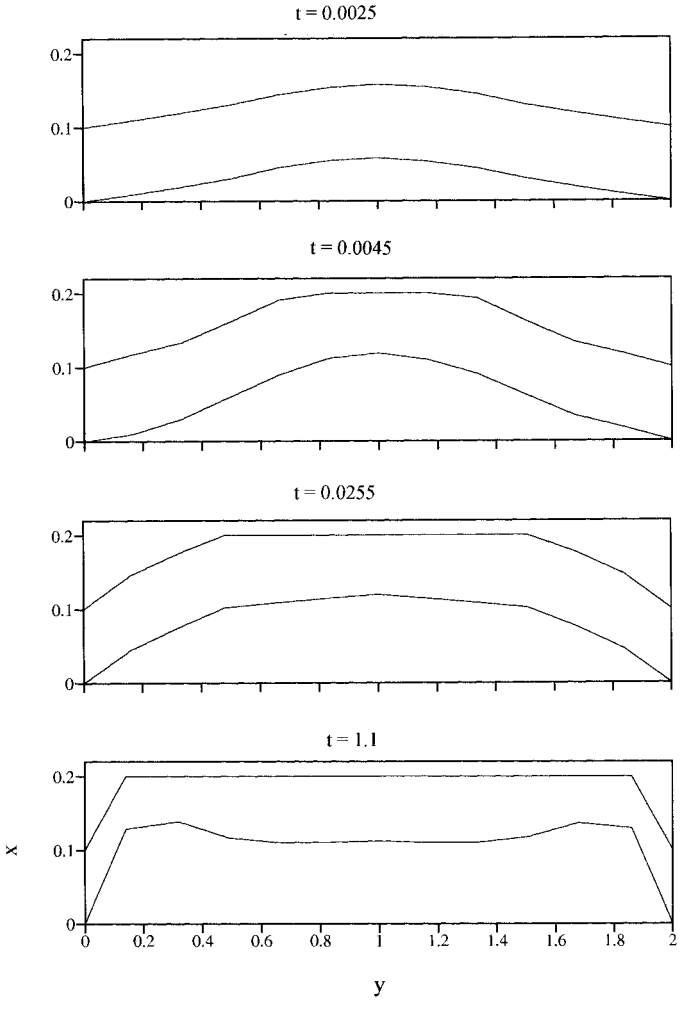


Figure 14. Evolution of the thickness distribution during the thermoforming process. The figure shows the unconfined flow in the early stages ($t < 0.004$), early contact with upper wall of the mold at $x = 0.2$ at 0.004.

surface comes in contact with the mold wall $x = 0.2$, typified by the stages $t = 0.0045$ and 0.025. It is important to note the small time scale over which most of the filling stage takes (less than 0.1 time unit). Although the rate of filling depends on the magnitude of the driving pressure, it is typically short. The filling stage is essentially completed after $t = 1.1$ as shown in Figure 14. At this stage, the thickness has lost its uniformity. Pronounced thinning occurs near the corners, accompanied by a bulge in the middle.

 

### 4.3. Extension of a cylindrical fluid

Consider the extension of a fluid sample between two plates that are displaced at a certain rate. This is the classical flow configuration encountered in rheometry. Equivalently, the traction may be imposed at the plates rather than the displacement. Thus, knowing the traction (stress) and the rate of displacement, the fluid properties can be inferred from the kinematic measurements. In this section, the extension flow of a cylindrical sample will be examined. The initial cylindrical fluid sample is drawn by the two plates at the same constant but opposite velocities in the axial direction. Since the configuration is axisymmetric, only one eighth of the problem will be analyzed. The influence of surface tension will be examined as well. Numerical accuracy is assessed by examining the influence of the mesh size.

The initial and final configurations are shown in Figure 15. Consider the initial configuration ($t = 0$), which corresponds to a fluid cylinder of initial radius-to-height ratio equal to one. The cylinder ends are assumed to each adhere to the moving plates. The co-ordinate frame of reference is taken as shown in the figure. The origin and the $(y, z)$ plane are located halfway between the two plates. This is an unconfined flow, i.e., the boundary conditions do not change with time since no additional contact between fluid and rigid wall occurs. The domain of computation is given by the set

$$\{(x, y, z)|\sqrt{y^2 + z^2} \leq r(x, t),\ x \in [0, h(t)]\ \}$$

where $r(x, t)$ is the local radius of the fluid and $h(t)$ is the half height. Note that $r(x, 0) = 1$ and $h(0) = 0.5$.

The dynamic condition at the (only) free surface, $S(t)$, is given by

$$\mathbf{t}(\mathbf{x} \in S, t) = -Ca^{-1}\mathbf{n}(\mathbf{x} \in S, t)\nabla \cdot \mathbf{n}(\mathbf{x} \in S, t) \tag{15}$$

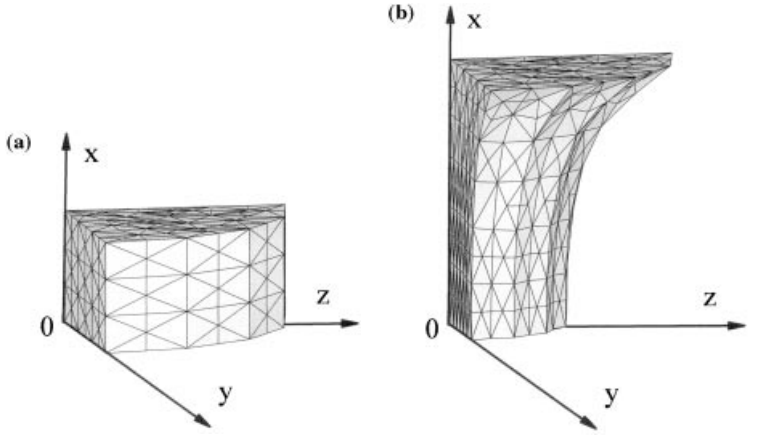


Figure 15. Extension of a cylindrical fluid sample. The figure shows (a) one eighth of the initial sample ($t = 0$), and (b) one eighth of the stretched sample ($t = 2.4$).

 

It is clear from the radial and vertical scales in the figure that the triangular elements in the initial mesh are too stretched. This initial ill-conditioning is, however, only temporary as the elements are subsequently subdivided. The mesh regularity is also evident from the flow at $t = 2.4$ in Figure 15 The details of the flow field is shown in Figure 16 during the extensional flow process at $t = 2.4$. There is a strong axial flow as a result of the stretching in the $x$-direction. The radial flow is obviously most pronounced in the $(y, z)$ plane, and is felt throughout the fluid, especially in the lower mid region between the $(y, z)$ plane of symmetry and the plate that is pulling the fluid, as shown in the $(x, z)$ plane. There is a strong axial flow as a result of the stretching in the $x$-direction. Note that the absence of arrows on the free surface is not an indication of an absence of flow activity on the surface.

The influence of surface tension is explored for the cylindrical sample of Figure 15. Figure 17 shows the evolution of the sample diameter $D_0$ at $z = 0$ as function of time for $Ca^{-1} \in [0, 6]$. The figure indicates clearly that $D_0$ decreases with time and surface tension tends to accelerate the drop in the diameter. While the change in the rate of decrease is always postive in the absence of surface tension, it changes sign at some time for $Ca^{-1} > 0$, reflecting an enhanced drop in the diameter with time.

The influence of the mesh size is examined to assess its effect on the numerical accuracy. Figure 18 shows the evolution of the relative error with time, which, as in thermoforming, is equal to relative change in volume. A relatively large time increment is used, $\Delta t = 0.2$. The figure shows the evolution of the error for $D_{\max} \in [0.8, \infty)$. Note that an infinite value for $D_{\max}$ corresponds to no remeshing. The figure indicates the growth is monotonic for any mesh size. In fact, the error grows linearly in the absence of remeshing. As

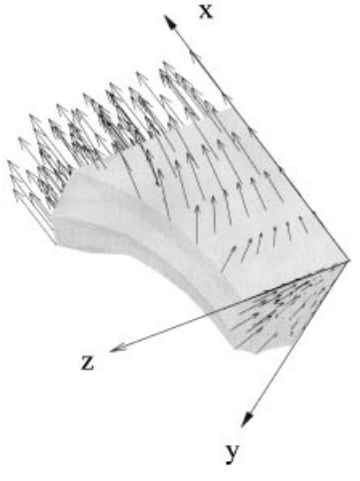


Figure 16. Flow field during the extension of the fluid sample in Figure 14 at $t = 2.4$.

 

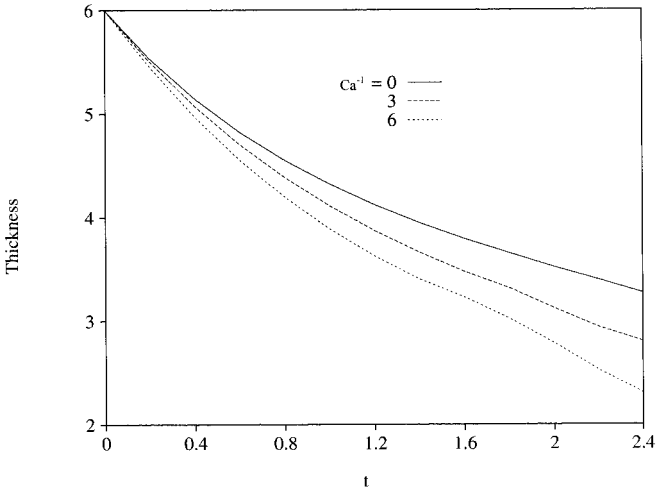


Figure 17. Influence of surface tension for the transient extrusion of a solid square cylinder as in Figure 14. Here $Ca^{-1} \in [0, 7]$.

remeshing is used, the error tends to taper off after an initial linear growth that is present for the higher $D_{\max}$ values. The error growth is effectively controlled by lowering $D_{\max}$, which is limited to less than 2 per cent for $D_{\max}$, which corresponds to a relatively large mesh size.

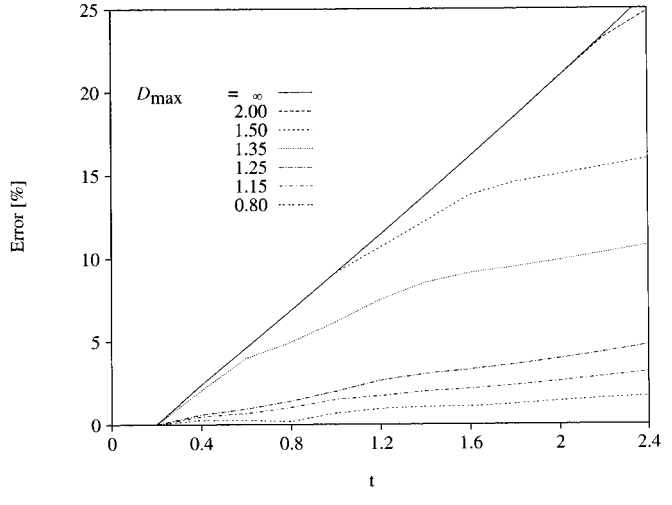


Figure 18. Influence of mesh size for the extension of a fluid sample for $D_{\max} \in [0.8, \infty)$.

 

### 4.4. Coating of a spherical object

Coating is another important problem in material processing. Several variants of this process exist, including blade coating and wire coating, Most coating processes are typically analyzed under steady state conditions, as the long-term behavior is of practical interest since the process is usually continuous as it involves the coating of long sheets or wires. When discrete parts are coated, however, transient effects become important. In fact, steady state conditions never really set in because the part is dipped in or sprayed by the coating material. The problem considered in the present section is the coating of a spherical object by a jet of fluid. The first problem examined is that of a sphere centered on the axis of the jet, $x$, and the second deals with a sphere located off axis. All dimensions are given relative to the jet radius at the exit. In both problems, the sphere has a radius equal to 0.75 and is centered at a unit distance from the plane of the jet exit. The influence of surface tension is also examined.

Although the first problem is axisymmetric, it will be analyzed using the three-dimensional formulation and algorithm developed earlier. An important reason for choosing an axisymmetric flow configuration is to demonstrate the robustness of the proposed methodology. As will be seen, there is essentially no deviation of the three-dimensional flow from axisymmetry despite the large number of elements involved and the long time over which the simulation is performed. Figure 19 shows two stages of the coating process over a time period equal to 7 in the absence of surface tension ($Ca \to \infty$). The early stage of flow is shown at $t = 0.8$ after the jet has just touched the sphere at $x = 1$. A certain degree of swelling is visible. Note that the sphere is invisible as the fluid enrobes it. At an advanced stage, $t = 7$, the fluid remains separated in the wake of the sphere.

In the presence of surface tension, the coating process is expected to become more difficult. Indeed, Figure 20 displays the flow configuration at an early stage ($t = 0.8$) and at an advanced stage ($t = 8$) for $Ca = 0.2$. In this case, the fluid tends to move radially as opposed to the axial movement observed in Figure 19. Even at this advanced stage, the fluid is incapable of covering the sphere. Surface tension effects are obviously present in regions of high curvature, particularly at the most advanced part of the fluid. Comparisons between Figures 19 and 20 indicate that the coating fluid is likely going to be more viscous for a larger surface.

Finally, the coating of a sphere located off axis is examined in the absence of surface tension. The center of the sphere is located above the jet axis at (1.75, 0, 0.5). In this case, one expects the coating to be practically impossible, at least with a jet of small radius at the exit. This is inferred from the simulation shown in Figure 21, which shows the flow at two stages, $t = 0.3$ and 6. It is clear from the early stage that the flow essentially moves only under the sphere. Even at $t = 6$, the flow continues to flow under the sphere, with very little flow above it.

## 5. CONCLUSION

The simulation of three-dimensional transient free surface flow is carried out for case studies from material processing. An adaptive Lagrangian three-dimensional boundary element approach is proposed for the flow of Newtonian viscous fluids. The mesh refinement algorithm

 

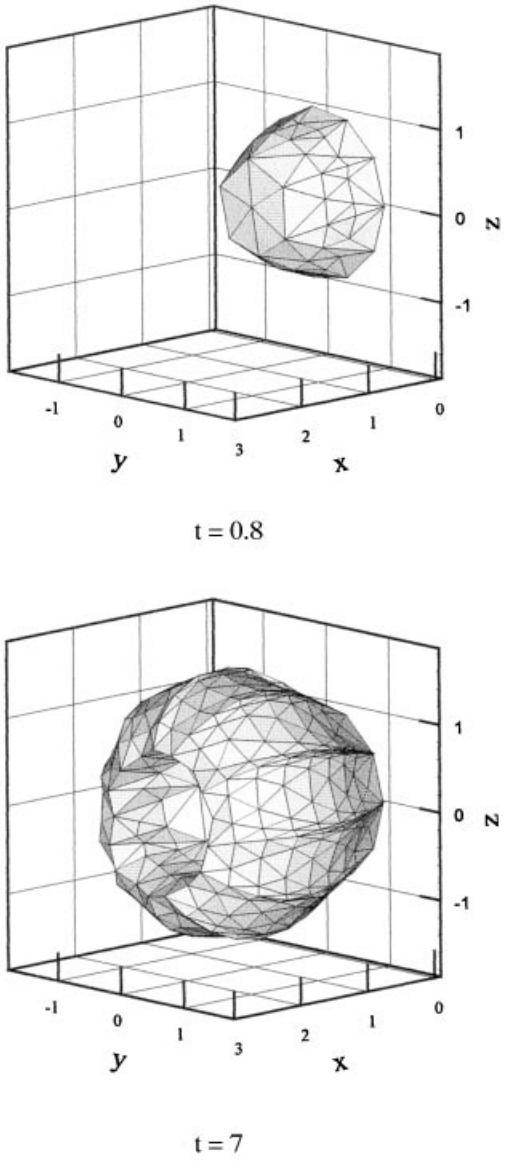


Figure 19. Coating of a sphere centered on the jet axis in the absence of surface tension. The figure shows the flow just as it comes in contact with the sphere ($t = 0.8$), and after the flow has surrounded the sphere ($t = 7$).

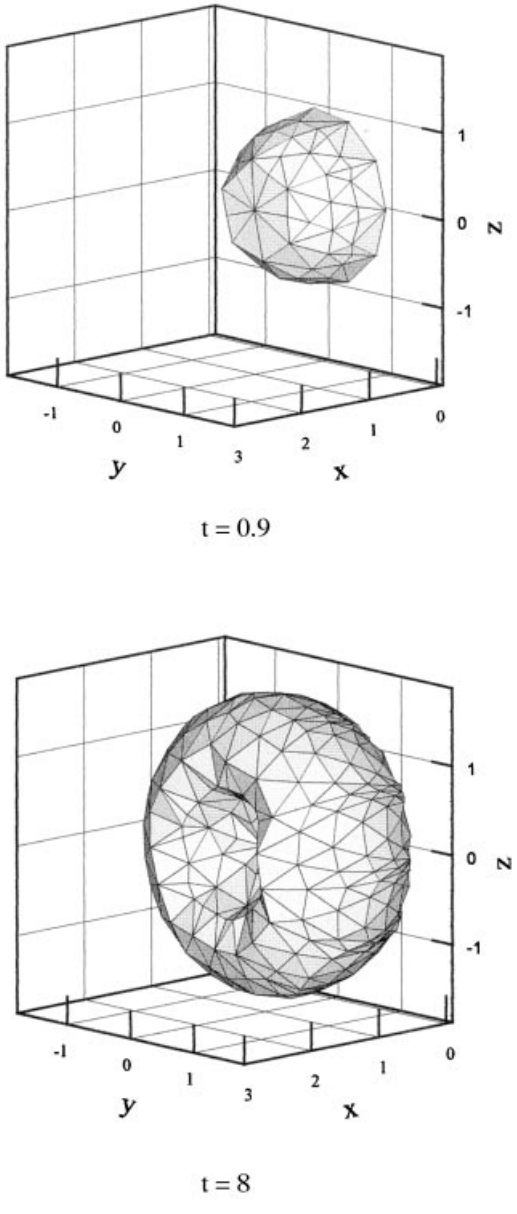


Figure 20. Coating of a sphere centered on the jet axis in the presence of surface tension ($Ca^{-1} = 5$). The figure shows the flow just as it comes in contact with the sphere ($t = 0.9$), and after the flow has begun to surround the sphere ($t = 8$).

 

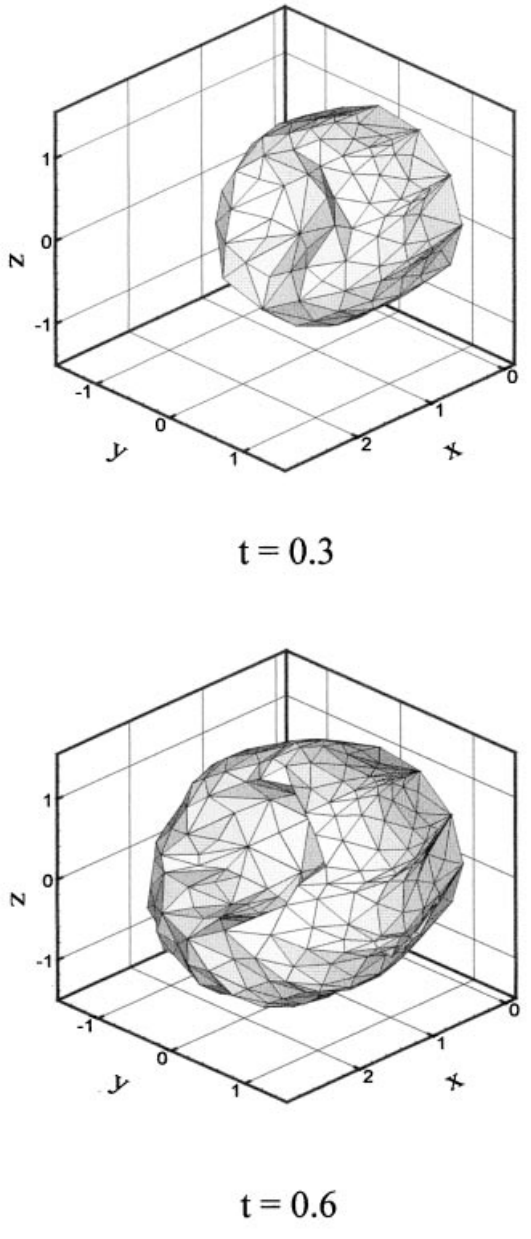


Figure 21. Coating of a sphere centered below the jet axis in the absence of surface tension. The figure shows the flow just as it comes in contact with the sphere ($t = 0.3$), and after the flow has begun to surround the sphere ($t = 0.6$).

 

is simple and yet is found to be robust and suitable for moving free surface flow. The method is illustrated through transient extrusion of a solid circular and square cylinders showing no limitation as the degree of deformation or flow in the absence or presence of surface tension. Other flow configurations were also considered, such as the thermoforming of a fluid sheet, the extension of a cylindrical fluid sample, and the coating of a sphere.

Future work is currently undertaken to extend the present formulation to more complex problems in polymer processing and multiphase flow. Additional convergence and accuracy assessment is needed to confirm the validity of the method in more complex flow configurations. Direct comparative studies with existing numerical results would have been helpful. However, to the author's knowledge, there has not been a numerical implementation for the three-dimensional viscous flow against which comparison can be made.

## ACKNOWLEDGMENTS

This work is supported by the Natural Sciences and Engineering Council of Canada.

## REFERENCES

1. Floryan JM, Rasmussen H. Numerical methods for viscous flows with moving boundaries. *Applied Mechanics Reviews* 1989; **42**: 323.
2. Nickell RE, Tanner RI, Caswell B. The solution of viscous incompressible jet and free-surface flows using finite element method. *Journal of Fluid Mechanics* 1974; **65**: 189.
3. Siliman WJ, Scriven LE. Separating flow near a static contact line: slip at a wall and shape of a free surface. *Journal of Computational Physics* 1980; **34**: 287.
4. Ruschak KJ. A method of incorporating free boundaries with surface tension in finite element fluid flow simulation. *International Journal for Numerical Methods in Engineering* 1980; **15**: 639.
5. Kawahara M, Miwa T. Finite element analysis of wave motion. *International Journal for Numerical Methods in Engineering* 1984; **20**: 1193.
6. Bach P, Hassager O. An algorithm for the use of the Lagrangian specification in Newtonian fluid mechanics and applications to free surface flows. *Journal of Fluid Mechanics* 1985; **152**: 173.
7. Ramaswany B, Kawahara M. Lagrangian finite element analysis applied to viscous free surface fluid flow. *International Journal for Numerical Methods in Fluids* 1987; **7**: 953.
8. Chipada S, Jue TC, Joo SW, Wheeler MF, Ramaswamy R. Numerical simulation of free-boundary problems. *Computational Fluid Dynamics* 1996; **7**: 91.
9. Khayat RE, Luciani A, Utracki LA. Boundary-element analysis of planar drop deformation in confined flow. Part I. Newtonian fluids. *Engineering Analysis and Boundary Elements* 1997; **19**: 279.
10. Khayat RE, Huneault M, Utracki LA, Duquette R. A boundary element analysis of planar drop deformation in the screw channel of a mixing extruder. *Engineering Analysis and Boundary Elements* 1998; **21**: 155.
11. Khayat RE. Boundary-element analysis of planar drop deformation in confined flow. Part II. Viscelastic fluids. *Engineering Analysis and Boundary Elements* 1998; **22**: 291.
12. Khayat RE, Derdouri A, Hebert LP. A boundary-element approach to three-dimensional gas-assisted injection molding. *Journal of Non-Newtonian Fluid Mechanics* 1995; **57**: 253.
13. Khayat RE, Raducanu P. A coupled finite element/boundary element approach for the three-dimensional simulation of air venting in blow molding and thermoforming. *International Journal for Numerical Methods in Engineering* 1998; **43**: 151.
14. Khayat RE, Derdouri A, Frayce D. Boundary-element analysis of three-dimensional transient mixing processes of Newtonian and viscoelastic fluids. *International Journal for Numerical Methods in Fluids* 1998; **28**: 815.
15. Khayat RE, Garcia-Rejon A. Uniaxial and biaxial unsteady inflation of a viscoelastic material. *Journal of Non-Newtonian Fluid Mechanics* 1992; **43**: 31.
16. Khayat RE. A perturbation approach to planar flow of a viscoelastic fluid with two moving free boundaries. *Quarterly Journal of Mechanics and Applied Mathematics* 1994; **47**(3): 342.

 

17. Mao W, Khayat RE. Numerical simulation of transient planar flow of a viscoelastic materials with two moving free surfaces. *International Journal for Numerical Methods in Fluids* 1995; **21**: 1137.
18. Wrobel LC. The dual reciprocity boundary element formulation for nonlinear problems. *Computing Methods in Applied Mechanics and Engineering* 1987; **65**: 147.
19. Nowak AJ. Application of the multiple reciprocity method for solving nonlinear problems. In *Advanced Computational Methods in Heat Transfer II*, *Vol I: Conduction*, *Radiation and Phase Change*, Wrobel LC, Brebbia CA, Nowak AJ (eds). Computational Mechanics Publications: Southampton, 1995.
20. Neves AC, Brebbia CA. The multiple reciprocity boundary element method for transforming domain integrals to the boundary. *International Journal for Numerical Methods in Engineering* 1991; **31**: 709.
21. Frayce D, Khayat RE. A dual reciprocity boundary element approach to three-dimensional transient heat conduction as applied to materials processing. *Numerical Heat Transfer A* 1996; **29**: 243.
22. Peraire M, Vahdati K, Morgan K, Zienkiewicz OC. Adaptive remeshing for compressive flow computation. *Journal of Computational Physics* 1987; **72**: 449.
23. Jin H, Wiberg NE. Two dimensional mesh generation, adaptive remeshing and refinement. *International Journal for Numerical Methods in Engineering* 1990; **29**: 1501.
24. Roberti P, Melkanoff MA. Self-adaptive analysis based on stress convergence. *International Journal for Numerical Methods in Engineering* 1987; **24**: 1973.
25. Rivara MC. Selective refinement/derefinement algorithms for sequences of nested triangulation. *International Journal for Numerical Methods in Engineering* 1989; **28**: 2889.
26. Bachmann PL, Witchen SL, Sheppard MS, Grice KR, Yerry MA. Robust, geometrically based, automatic two-dimensional mesh generation. *International Journal for Numerical Methods in Engineering* 1987; **24**: 1043.
27. Sheppard MS. Approaches to the automatic generation and control of finite element meshes. *Applied Mechanics Reviews* 1988; **41**: 169.
28. Dow JO, Byrd DE. Error estimation procedure for plate bending elements. *AIAA Journal* 1983; **21**: 685.
29. Batchelor GK. *Introduction to Fluid Dynamics*. Cambridge University Press: Cambridge, 1967.
30. Park WC, Homsy GM. Two-phase displacement in Hele–Shaw cells: theory. *Journal of Fluid Mechanics* 1984; **139**: 291.
31. Brebbia CA, Dominguez J. *Boundary Elements: An Introductory Course*. Computational Mechanics Publications and McGraw-Hill, 1992.
32. Nambiar RV, Valera RS, Lawrence KL, Morgan RB, Amil D. An algorithm for adaptive refinement of triangular element meshes. *International Journal for Numerical Methods in Engineering* 1993; **36**: 499.
33. Rosenberg IG, Stenger F. A lower bound on the angles of triangles constructed by bisecting the longest edge. *Mathematics and Computers* 1975; **29**: 390.
34. Tanner RI. *Engineering Rheology*. Oxford University Press: Oxford, 1986.
35. Khayat RE, Plaskos C, Genouvrier D. An adaptive boundary-element approach to 3D transient free-surface cavity flow, as applied to polymer processing. *International Journal for Numerical Methods in Engineering* 2000; **0**: 0.
36. Khayat RE, Elsin W, Kim K. An adaptive boundary-element approach to 2D moving-boundary flow of viscous fluids. *International Journal for Numerical Methods in Fluids* 2000; **33**: 847–868.

 