An Adaptive h-p Finite Element Method for Incompressible Free Surface Flows of Generalized Newtonian Fluids

Vincent Legat* J. Tinsley Oden†

Abstract

An adaptive hp-finite element method for free surface flows of incompressible viscous fluids is presented. Generalized Newtonian models and surface tension effects are included. Principal components of our method are a reliable discretization of the free surface, a moving grid algorithm, an *a posteriori* error estimator and an efficient adaptive strategy. A full Newton-Raphson iterative scheme and an adequate data structure are employed in the analysis. Numerical eperiments that illustrate applications of the method to model free surface flows are included.

1 Introduction

Recently, adaptive hp-finite element methods have been used for the analysis of significant classes of incompressible flows problems [1,2,3]. These techniques automatically adjust the parameters h and p (the mesh size and the degree of the polynomial) so as to deliver very high rates of convergence. The strategy can be compared with an optimal control paradigm in which the computational error in an appropriate norm is controlled by an adaptive scheme which orchestrates the distribution of h and p so as to deliver a preset level of accuracy. A rigorous *a posteriori* error estimate has been advanced [4,5] which provides a measure of the quantity to be controlled and a three-step adaptive strategy has been proposed [6] which can significantly reduce computer times required to adapt the mesh so as to achieve a target error.

In the present paper, these adaptive hp-methodologies are extended to free surface flows of Generalized Newtonian fluids. These extensions include the development of an *a posteriori*

*Postdoctoral Fellow, TICOM, The University of Texas at Austin, Austin, Texas, 78712, U.S.A., permanently at CESAME, Universite Catholique de Louvain, 4 avenue G. Lemaitre, B-1348, Louvain-la-Neuve, Belgium

†TICOM, The University of Texas at Austin, Austin, Texas, 78712, U.S.A.

error estimator for h-p finite element approximations of incompressible free surface flows of Generalized Newtonian fluids and a version of the so-called Texas-three-step adaptive strategy for producing good *h-p*-finite element meshes based on *a priori* and *a posteriori* error estimates.

Following this introduction, preliminary descriptions of the problem class under study are given. Discretization of the problem and a new moving grid algorithm are defined in Section 3. *A posteriori* estimates of the Navier-Stokes problem with free surfaces boundaries and the adaptive scheme are described in Section 4. Numerical results are collected in Section 5.

In this paper, we limit ourselves to two-dimensional steady flows, although extending the method to unsteady and three dimensional processes is straightforward. Features for considering three dimensional free surface flows, such as described in [7,8], are completely compatible with the present approach.

2 The Free Surface Problem

2.1 *Governing equations*

Let us consider the steady flow of a viscous incompressible Newtonian fluid in an open bounded domain $\Omega \subset \mathbb{R}^n$, $n = 2$ or 3. The partial differential equations governing the conservation of mass and momentum are the classical Navier-Stokes equations,

$$\left.\begin{array}{rcll} (\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u} - \boldsymbol{\nabla}\cdot\boldsymbol{\sigma}(\boldsymbol{u},p) &=& \boldsymbol{f} & \text{in } \Omega \\ \boldsymbol{\nabla}\cdot\boldsymbol{u} &=& 0 & \text{in } \Omega \end{array}\right\} \tag{1}$$

where $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x})$, $\boldsymbol{x} = (x_1, \cdots, x_n) \in \Omega$, is the velocity vector field, $\boldsymbol{f}$ is the body force and $\boldsymbol{\sigma}(\boldsymbol{u}, p)$ is the Cauchy stress tensor defined by

$$\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u},p) &= \boldsymbol{\tau}(\boldsymbol{u}) - p1 \\ \boldsymbol{\tau}(\boldsymbol{u}) &= 2\nu\boldsymbol{D}(\boldsymbol{u}) \\ \boldsymbol{D}(\boldsymbol{u}) &= \left(\boldsymbol{\nabla}\boldsymbol{u} + \boldsymbol{\nabla}\boldsymbol{u}^T\right)/2 \end{aligned}$$

where p is the pressure, 1 is the unit tensor, $\boldsymbol{\tau}(\boldsymbol{u})$ is the deviatoric part of the stress tensor, ν is the kinematic viscosity, and $\boldsymbol{D}(\boldsymbol{u})$ is the strain rate tensor.

In order to describe the boundary conditions for the free surface problem in Fig. 1, we define the following subsets of the boundary $\partial\Omega \subset \mathbb{R}^{n-1}$:

- $\partial\Omega_D$: the portion of the boundary on which Dirichlet boundary conditions are prescribed,

- $\partial\Omega_N$: the portion of the boundary on which Neumann boundary conditions are prescribed,
- $\partial\Omega_F$: the free surface.

Here, $\overline{\partial\Omega} = \overline{\partial\Omega_D \cup \partial\Omega_N \cup \partial\Omega_F}$, $\partial\Omega_F \cap \partial\Omega_D = \emptyset$ and $\partial\Omega_F \cap \partial\Omega_N = \emptyset$.

On the surfaces $\partial\Omega_D$ and $\partial\Omega_N$, the boundary conditions are specified in the usual way,

$$\left.\begin{array}{rcll} \boldsymbol{u} & = & \widehat{\boldsymbol{u}} & \text{on } \partial\Omega_D \\ \boldsymbol{\sigma}(\boldsymbol{u},p)\cdot\boldsymbol{n} & = & \widehat{\boldsymbol{g}} & \text{on } \partial\Omega_N \end{array}\right\} \tag{2}$$

where $\widehat{\boldsymbol{u}}$ is the prescribed velocity on $\partial\Omega_D$, $\widehat{\boldsymbol{g}}$ is the given traction on $\partial\Omega_N$ and $\boldsymbol{n}$ is the unit normal vector pointing out of the domain Ω. .

Considering the free surface $\partial\Omega_F$, it is convenient to parametrize this boundary by n-1 curvilinear parameters and introduce a local frame of reference. Using the symbol $\nabla*$ for the gradient along the free surface in this frame of reference, the kinematic and dynamic conditions can be written as

$$\left.\begin{array}{rcll} \boldsymbol{u}\cdot\boldsymbol{n} & = & 0 & \text{on } \partial\Omega_F \\ \boldsymbol{\sigma}(\boldsymbol{u},p)\cdot\boldsymbol{n} & = & -(p_{ext}+\gamma tr\nabla*\boldsymbol{n})\boldsymbol{n} & \text{on } \partial\Omega_F \end{array}\right\} \tag{3}$$

where p_{ext} is the exterior pressure and γ is the surface tension coefficient. The classical Euler-Lagrange formulation of the surface tension correlates the normal force with the main curvature of the free surface defined by $tr\nabla*\boldsymbol{n}$.

The Reynolds number Re and the capillary number Ca are respectively defined by

$$\begin{array}{rcl} Re & = & UL/\nu \\ Ca & = & \gamma/(\nu U) \end{array}$$

where U and L denote a characteristic velocity and length.

For a Newtonian fluid, the kinematic viscosity is a positive constant. However, most polymers present a shear-thinning behavior, which, within a range of shear rates can be described by a dependence of the viscosity upon the shear rate $\dot{\gamma}$. One of the most commonly used models for non-Newtonian viscosity is the power law model,

$$\eta(\dot{\gamma}) = K\dot{\gamma}^{m-1}$$

where K is a constant and m is a power law index. The shear rate $\dot{\gamma}$ is defined as the square of the second invariant of the strain rate tensor. The power law index lies between 0.6 and 0.2 for many polymers.

In order to demonstrate the robustness of our technique, we also consider Bingham fluids. In this case, the viscosity law is defined as follows :

$$\eta(\dot{\gamma}) = K + \frac{\tau_0}{\dot{\gamma}}, \qquad \text{when } \dot{\gamma} \geq \dot{\gamma}_{crit}$$

$$\eta(\dot{\gamma}) = K + \frac{\tau_0}{\dot{\gamma}}(2 + \frac{\dot{\gamma}}{\dot{\gamma}_{crit}}), \quad \text{when } \dot{\gamma} \leq \dot{\gamma}_{crit}$$

where $\dot{\gamma}_{crit}$ is a critical shear rate and τ_0 is the given yield stress.

2.2 *Characterization of motion of the free surface*

Before developing the weak formulation, it is first necessary to establish a procedure for characterizing the motion of the free surface. We begin by considering the boundary conditions on $\partial\Omega_F$ in more detail.

On one hand, assigning the surface force is a valid boundary condition for the Stokes' problem in a fixed geometry [9]. On the other hand, the kinematic condition $\boldsymbol{u} \cdot \boldsymbol{n} = 0$ may also be seen as a Dirichlet boundary condition in the normal direction. For a Stokes problem on a *fixed* domain, imposing (3.1) together with (3.2) would define an ill-posed problem. To define these flow boundary conditions simultaneously on $\partial\Omega_F$, we introduce a new unknown : the kinematic degree of freedom h, which describes the shape of the free surface. Finally, we correlate the incremental displacement of the free surface $\boldsymbol{w}$ with h. It is important to note that $\boldsymbol{w}$ only describes an arbitrary Eulerian stretching of the free surface; in other words, it defines a frame motion, but does not give the motion of material points. Therefore, for transient simulations, the frame velocity $\boldsymbol{w}/\Delta t$ has to be introduced in material derivatives of the conservative laws.

In our method, the incremental displacement of the free surface is defined by the equation,

$$\boldsymbol{w} = h \cdot \boldsymbol{d} \quad \text{on } \partial\Omega_F \tag{4}$$

where $\boldsymbol{d}$ is a given unit vector *non tangent* to $\partial\Omega_F$ and h is the amplitude of the displacement along this vector. This procedure is illustrated in Fig. 2. In other words, $\boldsymbol{d}$ can be selected a priori and can be fixed during an iterative process, as long as the free surface does not become tangent to this direction. If this happens, conditions on the force and the velocity in the normal direction can no longer be unambiguously prescribed. Selection of the direction $\boldsymbol{d}$ is the first step in the search for the solution. This vector may vary from node to node.

This vector can be chosen as the normal to the surface [10], the direction of the velocity across the free surface [11] or it can be oriented along given spines [12][13]. In this paper, $\boldsymbol{d}$ is calculated as the unit vector normal to the initial free surface and is fixed during the iterative process.

Finally, if γ does not vanish, some boundary conditions have to be defined for the geometric field h. First, let us consider the boundary $\partial\partial\Omega_F$ of the free surface itself. This contains respectively the extremities of $\partial\Omega_F$ or the curve bounding the free surface for $n = 2$ or $n = 3$. This space may be partitioned into two subsets,

- $\partial\partial\Omega_{FD}$: the portion of $\partial\partial\Omega_F$ on which the location of the free surface is imposed. A Dirichlet boundary condition is imposed for the geometric degree of freedom h.
- $\partial\partial\Omega_{FN}$: the portion of $\partial\partial\Omega_F$ on which the direction of the vector tangent to the free surface (and normal to the curve $\partial\partial\Omega_F$ if n=3) is imposed. This type of boundary condition is frequently called a *contact angles* condition and may be interpreted as a Neumann boundary condition for the field h.

Note that as the capillary number Ca vanishes, the contact angles can no longer be imposed.

2.3 *Weak formulation of the problem*

Let V, Q and M denote respectively the spaces of admissible velocities, admissible pressures and admissible degrees of freedom.

$$\left.\begin{array}{rcl} V & = & \left\{\boldsymbol{v} \in (H^1(\Omega))^n : \boldsymbol{v} = \hat{\boldsymbol{u}} \ \ a.e. \ \text{on} \ \ \partial\Omega_D\right\} \\ Q & = & \left\{q \in L^2(\Omega) : \int_\Omega q\, dx = 0\right\} \\ M & = & \{m \in H^1(\partial\Omega_F) : m = 0 \ \ a.e. \ \text{on} \ \ \partial\partial\Omega_{FD}\} \end{array}\right\} \tag{5}$$

By formally multiplying the momentum equation by an arbitrary test velocity $\boldsymbol{v}$ and integrating by parts, we obtain

$$\begin{aligned} &\int_\Omega (\boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{u}\cdot\boldsymbol{v} + \boldsymbol{\tau}(\boldsymbol{u}) : \boldsymbol{D}(\boldsymbol{v}) - p\,\boldsymbol{\nabla}\cdot\boldsymbol{v} - \boldsymbol{f}\cdot\boldsymbol{v}\ )dx \\ &= \int_{\partial\Omega_N} \hat{\boldsymbol{g}}\cdot\boldsymbol{v}\, ds - \int_{\partial\Omega_F} p_{ext}\boldsymbol{n}\cdot\boldsymbol{v}\, ds - \int_{\partial\Omega_F} \gamma tr(\nabla * \boldsymbol{n})\boldsymbol{n}\cdot\boldsymbol{v}\, ds \end{aligned} \tag{6}$$

If we consider a two-dimensional domain Ω, use of the divergence theorem on the last term of the right hand side of (6) gives

$$\int_{\partial\Omega_F} \gamma tr(\nabla * \boldsymbol{n})\boldsymbol{n}\cdot\boldsymbol{v}\, ds = \int_{\partial\Omega_F} \gamma\nabla * \boldsymbol{v}\, ds - [\![\gamma\widehat{\boldsymbol{m}}\cdot\boldsymbol{v}]\!]_{\partial\Omega_F}$$

where the unit vector $\widehat{\boldsymbol{m}}$ is tangent to the free surface and is outwardly directed at the extremities of the curve $\partial\Omega_F$ and $[\![\ ]\!]_{\partial\Omega_F}$ denotes the jump on $\partial\Omega_F$. For the three dimensional case [16], the divergence theorem introduces a line integral which involves the action of surface tension along the curve $\partial\partial\Omega_F$ bounding the free surface:

$$\int_{\partial\Omega_F} \gamma tr(\nabla * \boldsymbol{n})\boldsymbol{n}\cdot\boldsymbol{v}\, ds = \int_{\partial\Omega_F} \gamma\nabla * \boldsymbol{v}\, ds - \int_{\partial\partial\Omega_F} \gamma\widehat{\boldsymbol{m}}\cdot\boldsymbol{v}\, ds$$

The unit vector $\widehat{\boldsymbol{m}}$ is now tangent to the free surface $\partial\Omega_F$, normal to the curve $\partial\partial\Omega_F$ and outwardly directed.

Finally the two-dimensional weak form of the boundary-value problem for determining $(\boldsymbol{u}, p, h)$ is as follows:

Given body forces $\boldsymbol{f} \in V^*$ and surface tractions $\boldsymbol{g} \in H_{00}^{1/2}(\partial\Omega_N)$,
find $(\boldsymbol{u}, p, h) \in V \times Q \times M$ such that

$$\begin{aligned}
\int_\Omega (\boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{u}\cdot\boldsymbol{v} + \boldsymbol{\tau}(\boldsymbol{u}) : \boldsymbol{D}(\boldsymbol{v}) - p\,\boldsymbol{\nabla}\cdot\boldsymbol{v} - \boldsymbol{f}\cdot\boldsymbol{v})dx & \\
= \int_{\partial\Omega_N} \widehat{\boldsymbol{g}}\cdot\boldsymbol{v}\, ds + \int_{\partial\Omega_F} \gamma\nabla * \boldsymbol{v}\, ds - [\![\gamma\widehat{\boldsymbol{m}}\cdot\boldsymbol{v}]\!]_{\partial\Omega_F} & \quad \forall\ \boldsymbol{v} \in V, \\
\int_\Omega q\,\boldsymbol{\nabla}\cdot\boldsymbol{u}\, dx = 0 & \quad \forall\ q \in Q, \\
\int_{\partial\Omega_F} \boldsymbol{u}\cdot\boldsymbol{n}\, m\, ds = 0 & \quad \forall\ m \in M.
\end{aligned} \tag{7}$$

Remark 2.1. Two points should be noted concerning the final weak form of the Navier-Stokes equations. First, the *angle* conditions along the interface boundary may be automatically introduced by prescribing the direction of the vector $\widehat{\boldsymbol{m}}$. Second, only first partial derivatives of velocity and surface position with respect to $\boldsymbol{x}$ appear. ∎

Remark 2.2. An inherent difficulty remains in this *classical* weak formulation. It is impossible to impose simultaneously a Dirichlet velocity boundary condition and a natural angle condition.

In order to illustrate this difficulty, suppose that we want to calculate the location A of the intersection between the interface and a wall in Fig. 3. On one hand, A being a stagnation point, the velocity components become very small near it and the coefficients of the weak kinematic condition (7.3) tend to vanish. The position of A *cannot be determined* from the kinematic condition. On the other hand, imposing the essential velocity boundary conditions along the wall prevents us from applying a natural angle condition on A.

Therefore, we must introduce a modified weak formulation. By combining (3.1) and the projection of (3.2) along the vector $\boldsymbol{d}$, and by multiplying by an arbitrary test function m, we can obtain

$$\begin{aligned} \int_{\partial\Omega_F} \boldsymbol{\sigma}(\boldsymbol{u},p)\cdot\boldsymbol{d}\; m ds &= \int_{\partial\Omega_F} \gamma\nabla * m \;\cdot\boldsymbol{d} ds - [\![\gamma\widehat{\boldsymbol{m}}\cdot\boldsymbol{d}\; m]\!]_{\partial\Omega_F} \\ &+ \int_{\partial\Omega_F} \boldsymbol{u}\cdot\boldsymbol{n}\; m\, ds \end{aligned} \tag{8}$$

This scalar equation includes the kinematic and the dynamic conditions and looks exactly as an advection-diffusion equation for the geometric field h. The first term on the right hand side is the diffusion term while the last term is the transport part. Finally, by replacing (7.3) by the present equation (8), we obtain the modified weak formulation.

We are so able to impose *natural* angle conditions at any point. The influence of the contact angle will decrease as a function of the surface tension. Such an effect is clearly in agreement with the physics and the solution without capillarity may be reached as the limit of solution with decreasing surface tension. It is important to note that the imposition of an *essential* contact angle (by constraining the linear system) cannot preserve the weighting by the surface tension coefficient and leads to incorrect results for small capillary number.

Both weak formulations differ only at contact points along wall, because (8) can be obtained by a linear combination of (7.1) and (7.2) and by the use of the divergence theorem. In practical calculations, we take advantage of this property for evaluating the discrete form of (8). ∎

3 Discretization of the Problem

3.1 *hp-Finite Element approximations*

We employ here the general hp-data structure with 1-irregular meshes introduced in [14] and exploited in [1,2] and elsewhere. For completeness, we record its basic properties. We begin, as usual, by introducing a family $\mathcal{P}$ of partitions of Ω such that

$$\overline{\Omega} = \bigcup_{K}^{N(\mathcal{P})} \left\{ \overline{\Omega}_K \, ; \, \Omega_K \in \mathcal{P} \right\}, \; \Omega_K \cap \Omega_L = \emptyset, \; K \neq L$$

We suppose the Ω and $\mathcal{P}$ are such that each element Ω_K is the image of a master element $\overline{\widehat{\Omega}} = [-1,1]^n$ under an invertible map. On the master element $\widehat{\Omega}$, 9 nodes are defined; these are classified as vertex nodes, edge nodes, and interior nodes (we assume $n = 2$ for simplicity). If the functions φ_k are given by

$$\varphi_k(s) = \sqrt{\frac{2k-1}{2}} \int_{-1}^{s} P_{k-1}(t) dt$$

where P_{k-1} is the Legendre polynomial of degree $k-1$, the shape functions are defined on the master element as follows :

- At the vertex nodes, we use standard bilinear shape functions :

$$\widehat{\chi}_i(\xi,\eta) = \frac{1}{4}(1 \pm \xi)(1 \pm \eta)$$

  with $i = 1,2,3,4$.

- For the midpoints of each edge, we define edge functions :

$$\widehat{\zeta}_{ij}(\xi,\eta) = \begin{cases} \dfrac{1}{2}(1 \pm \eta)\varphi_j(\xi) \\ \\ \dfrac{1}{2}(1 \pm \xi)\varphi_j(\eta) \end{cases}$$

  with $i = 1,2,3,4$ and $j = 2,3,\cdots,p_i$.

- For the interior node, we consider bubble functions :

$$\widehat{\xi}_{ij}(\xi,\eta) = \varphi_i(\xi)\varphi_j(\eta)$$

  with $i = 2,\cdots,p_{01}$ and $j = 2,\cdots,p_{02}$.

The element shape functions define a space

$$\widehat{M}\left(\widehat{\Omega}, p_1, p_2, p_3, p_4, p_{01}, p_{02}\right) = span\Big\{\widehat{\chi}_i, \widehat{\zeta}_{ij}, \widehat{\xi}_{kl} : \begin{array}{l} i = 1, \cdots, 4 \\ j = 2, \cdots, \max_i p_i \\ k, l = 2, \cdots, \max\left(p_{01}, p_{02}\right) \end{array}\Big\}$$

In general, for h-p finite element calculations, it is common to use a subparametric mapping. However, the final shape of the domain being unknown in free surface problem, a isoparametric mapping F_K from the master element $\widehat{\Omega}$ to element Ω_K is selected.

In order to develop an adaptive strategy, the following generalizations are introduced. In one hand, the h-refinement allows us 1-irregular refinements of the mesh. Interelement constraints are imposed to maintain continuity of basis functions across interelement boundaries. In the other hand, the p-adaptivity consists of using polynomials of differing degrees in the definition of edge functions and bubble functions. Continuity of the global basis functions is maintained by enriching the edge function to match highest-degree polynomial used on a common interelement boundary. This data structure allows us the construction of the local and global hp-spaces

$$\begin{aligned}
M_K &= \left\{v = v(\boldsymbol{x}) = \widehat{v} \circ F_K^{-1},\, \widehat{v} \in \widehat{M}(\widehat{\Omega}, p_1^K, p_2^K, p_2^K, p_{01}^K, p_{02}^K)\right\} \\
N_K &= \left\{v = v(\boldsymbol{x}) = \left(\widehat{v} \circ F_K^{-1}\right)\Big|_{\Omega_K \cup \partial\Omega_F},\, \widehat{v} \in \widehat{M}(\widehat{\Omega}, p_1^K, p_2^K, p_2^K, p_{01}^K, p_{02}^K)\right\} \\
S^{hp}(\Omega) &= \left\{v = v(\boldsymbol{x}) \in C^0\left(\overline{\Omega}\right) : v_K = v\Big|_{\Omega_K} \in M_K\right\} \\
S^{hp}(\partial\Omega_F) &= \left\{v = v(\boldsymbol{x}) \in C^0\left(\overline{\partial\Omega_F}\right) : v_K = v\Big|_{\Omega_K \cup \partial\Omega_F} \in N_K\right\}
\end{aligned}$$

The spaces $S^{hp}(\Omega)$ and $S^{hp}(\partial\Omega_F)$ possess standard interpolation properties of hp-finite element methods [15]. For element Ω_K, if h_K is the diagonal length and p_K is the highest degree of the complete polynomials contained in M_K, then we have the following local interpolation property,

$$\left\|u - \widetilde{u}^{hp}\right\|_{s,\Omega_K} \leq C\, \frac{h_K^{\min(p_K+1-s, r-s)}}{p_K^{r-s}} \|u\|_{r,\Omega_K} \tag{9}$$

where $\widetilde{u}^{hp}$ is an appropriate hp-interpolant of u and $\|\ \|_{s,\Omega_K}$, $s \geq 0$, denotes the usual Sobolev norms for Ω_K.

The hp-finite element approximation of (6) is now characterized by the following problem

Given body forces $\boldsymbol{f} \in V^*$ *and surface tractions* $\boldsymbol{g} \in H^{1/2}_{00}(\partial\Omega_N)$,
find $(\boldsymbol{u}^{hp}, p^{h\widetilde{p}}, h^{hp}) \in V^{hp} \times Q^{h\widetilde{p}} \times M^{hp}$ *such that*

$$\begin{aligned}
\int_\Omega \boldsymbol{u}^{hp} \cdot \boldsymbol{\nabla}\boldsymbol{u}^{hp} \cdot \boldsymbol{v} + \boldsymbol{\tau}(\boldsymbol{u}^{hp}) : \boldsymbol{D}(\boldsymbol{v}) - p^{h\widetilde{p}} \boldsymbol{\nabla}\cdot\boldsymbol{v} \; - \boldsymbol{f}\cdot\boldsymbol{v} \; dx & \\
= \int_{\partial\Omega_N} \widehat{\boldsymbol{g}} \cdot \boldsymbol{v} \, ds + \int_{\partial\Omega_F} \gamma \nabla * \boldsymbol{v} \, ds - [\![\gamma \widehat{\boldsymbol{m}} \cdot \boldsymbol{v}]\!]_{\partial\Omega_F} & \quad \forall\, \boldsymbol{v} \in V^{hp}, \\
\int_\Omega q \, \boldsymbol{\nabla}\cdot\boldsymbol{u}^{hp} \, dx = 0 & \quad \forall\, q \in Q^{h\widetilde{p}}, \\
\int_{\partial\Omega_F} \boldsymbol{u}^{hp} \cdot \boldsymbol{n} \; m \, ds = 0 & \quad \forall\, m \in M^{hp}.
\end{aligned} \tag{10}$$

where the spaces containing the hp-finite element approximations are

$$\begin{aligned}
V^{hp}(\Omega) &= \left\{ \boldsymbol{v}^{hp} \in \left(S^{hp}(\Omega) \right)^n \cap V \right\} \\
Q^{h\widetilde{p}}(\Omega) &= \left\{ q^{h\widetilde{p}} \in S^{h\widetilde{p}}(\Omega) \cap Q \right\} \\
M^{hp}(\partial\Omega_F) &= \left\{ m^{hp} \in S^{hp}(\partial\Omega_F) \cap M \right\}
\end{aligned}$$

In general, the orders p and $\widetilde{p}$ of velocity and pressure approximations cannot be selected arbitrarily, but must be determined so that the well-known LBB condition (see, e.g. [16]) is satisfied to ensure stability of the scheme. In subsequent calculations, pressures are interpolated using $\widetilde{p} = p - 1$ while compatibility between $V^{hp}(\Omega)$ and $M^{hp}(\partial\Omega_F)$ does not seem to be a problem if equal order p's are used. Note there is no need to demand that the approximate pressures q^{hp} be continuous across interelement boundaries and we could also use $\widetilde{Q}^{hp}(\Omega) = \{q^{hp} \in Q : q^{hp}_K \in M_K\}$.

Remark 3.1. In the imposition of the kinematic condition, we note that without surface tension effects, the equation characterizing the free surface reduces to a first-order (hyperbolic) advection equation. In such cases, the standard Galerkin formulation may be unstable. We then resort to a characteristic approach involving marching by integrating along the velocity direction. If diffusion is introduced by capillarity forces in the global system, the Galerkin technique is generally adequate for handling the kinematic condition. ∎

3.2 *Moving grid algorithm*

Moving the free surface nodes without changing the position of interior nodes positions can lead to large and unacceptable element distortions in most situations. Then, we must propagate the motions of the boundary nodes into the domain by means of *a moving grid algorithm*. Its purpose is to avoid mesh distortion due to the free boundary motion and to maintain the original element density in the deformed mesh.

To describe such an algorithm, we introduce additional definitions. The boundary $\partial\Omega \subset \mathbb{R}^{n-1}$ is now partitioned into :

- $\partial\Omega_F$: the free surface on which the incremental nodal displacements are prescribed for satisfying simultaneously the kinematic and the dynamic conditions for the flow problem,
- $\partial\Omega_S$: the stationary part of $\partial\Omega$ on which the nodes are fixed,
- $\partial\Omega_T$: the portion of $\partial\Omega$ on which only the normal displacement of each node has to vanish. This allows the grid to stretch along this boundary and to better balance disturbances introduced by the motion of the free surface.

In general, a clear distinction between mesh generation algorithms based on PDE's and those on purely algebraic rules can be made. First, we consider one of the most popular algebraic rules for two-dimensional free surface problems [17,18]. The grid is sliced into one-dimensional segments which begin at a node of $\partial\Omega_S$ (whose coordinates are fixed) and which end at a node of $\partial\Omega_F$ (whose coordinates depend on h). These segments are called *spines* or *supports*. Interior nodes of the mesh lying on a support maintain their relative position on the support as its length changes with the motion of the interface nodes. Such linear interpolation provides an optimal monitoring of the mesh in many cases, but puts severe constraints on the mesh which has to be highly structured.

On the other hand, it is attractive to formulate the stretching algorithm as a boundary value problem. It is not surprising that the most efficient techniques that map a parent geometry to a deformed one are based on elliptic PDE generators (e.g. [19]). Unfortunately, the most efficient PDE which produce the best deformed meshes are highly non-linear and their use would dramatically increase the cost of the calculation. A good compromise between cost and efficiency consists in choosing a linear elliptic PDE generators.

One approach toward a reasonable linear elliptic moving grid algorithm is to use the Laplacian operator, which has certain smoothing properties [19,20,21]. The incremental

displacements $\boldsymbol{w}$ are obtained by solving the boundary value problems

$$\left.\begin{array}{rcll} \boldsymbol{\nabla}^2\boldsymbol{w} & = & 0 & \text{in } \Omega \\ \boldsymbol{w} & = & \widehat{h}\boldsymbol{d} & \text{on } \partial\Omega_F \\ \boldsymbol{w} & = & 0 & \text{on } \partial\Omega_S \\ \boldsymbol{w}\cdot\boldsymbol{n} & = & 0 & \text{on } \partial\Omega_T \end{array}\right\} \tag{11}$$

where $\widehat{h}$ is the prescribed amplitude of the incremental displacement along the given vector $\boldsymbol{d}$ for satisfying simultaneously the kinematic and dynamic conditions. For a *smooth* disturbance and a *smooth* boundary $\partial\Omega$, such a procedure is able to monitor the stretching of the mesh and allows us to handle unstructured grids. However, highly distorted grids may appear for relatively small disturbances. In other words, there is no constraint on the mesh, but the efficiency for maintaining smooth grids may be disappointing. Therefore, we propose a new stretching rule.

The basic idea consists in generating the linear operator which mimics the spine algorithm at a continuous level. The definition of h and $\boldsymbol{d}$ is extended to the whole domain Ω and the virtual displacement of each point of Ω is described as :

$$\boldsymbol{w} = h \cdot \boldsymbol{d} \quad \text{in } \Omega \tag{12}$$

where $\boldsymbol{d}$ is given unit vector and h is the amplitude of the displacement along this vector. The moving algorithm is based on both following strategy.

First, the following assumptions on the vector $\boldsymbol{d}$ are enforced :

- $\boldsymbol{d} = \boldsymbol{d}(\boldsymbol{x}) \in C^0\left(\overline{\Omega}\right)$,
- $\boldsymbol{d}$ cannot be tangent to $\partial\Omega_F$
- $\boldsymbol{d}$ must be tangent to $\partial\Omega_T$,
- $\boldsymbol{d}$ cannot vanish,

Second, the amplitudes h associated to each node are obtained by solving the following boundary value problem

$$\left.\begin{array}{rcll} (\boldsymbol{d}\cdot\boldsymbol{\nabla})^2 h & = & 0 & \text{in } \Omega \\ h & = & \widehat{h} & \text{on } \partial\Omega_F \\ h & = & 0 & \text{on } \partial\Omega_S \end{array}\right\} \tag{13}$$

The critical step is clearly the definition of the vector $\boldsymbol{d}$ that can be interpreted as the characteristic direction for the propagation of the disturbance from the free boundary into the grid interior. Note that for a suitable mesh, we can define $\boldsymbol{d}$ tangent to each spine and derive the spines algorithm. However, in the present approach, the characteristic direction for the disturbance propagation and the grid structure are defined as distinct concepts. Therefore, we are now able to provide an efficient moving algorithm for both structured and unstructured grids.

In other words, the present algorithm defines a one-dimensional stretching that appears to be much more robust than the two-dimensional stretching of the Laplacian operator. A comparison of results obtained by both techniques is shown in Fig 4.

3.3 *Iterative free surface flows solvers*

To compute $(\boldsymbol{u}^{hp}, p^{h\widetilde{p}}, h^{hp})$, we must also introduce an appropriate solver. Let us define, in a classical way, *explicit* and *implicit* schemes :

- In *explicit* schemes, the strategy for obtaining free surfaces is to decouple the flow equations from the calculation of the free boundary. More precisely, when a flow solution is obtained on an initial mesh, the velocity field is used for generating a new free surface on the basis of the kinematic condition. Such a strategy may be quite expensive in view of the introduction of an outer iteration above the flow solver itself. However, a fully explicit time marching algorithm which decouples all equations can be attractive. Since we are interested in steady state cases, a low-order temporal accuracy scheme may be adopted in order to reduce the cost of each iteration. Unfortunately, the number of steps becomes, in general, very important in view of the very low rate of convergence of such procedures [22,23].

- In *implicit* schemes, all equations are simultaneously solved. The kinematic degrees of freedom h are calculated together with all other field variables, while the equations of motion are supplemented by the kinematic condition. Now, the global system can be solved by a full Newton-Raphson iterative scheme. The method is especially efficient when it is necessary to use an implicit approach for the flow non-linearities. Indeed, solving the moving boundary problem requires essentially no more CPU time than the fixed boundary problem, because the number of additional variables is small. The method is advantageous in terms of CPU time (not in memory) with respect to most decoupled techniques. For example, the CPU time for a single iteration with the implicit method is 5% higher than the CPU time required for a fixed domain iteration. But, a full implicit scheme converges in 4 to 5 iterations (10^{-6} for relative variations),

where an explicit technique typically requires 20 to 30 fixed domain iterations to converge. Note also that the requirements on the initial guess are not so stringent as that for explicit schemes [13,18,7,8,24].

Here, we consider an implicit scheme and we derive a full Newton-Raphson scheme which allows us to obtain a quadratic rate of convergence. However, a good initial guess is always required for both formulations to converge, a piece of information which is not always available. Without a priori knowledge of the free surface location, it is clear that the initial free surface guess may be very far from the actual position. This is indeed a weak point for all methods published so far.

4 Error Estimates and Adaptivity

4.1 *A Posteriori Error Estimation with Equilibration*

We are now ready to consider the issue of error estimation for free surface flows. The aim is to compute rigorous, computable upper bounds to the approximation errors

$$\begin{aligned} \boldsymbol{u}^{error} &= \boldsymbol{u} - \boldsymbol{u}^{hp} \\ p^{error} &= p - p^{h\widetilde{p}} \\ h^{error} &= h - h^{hp} \end{aligned}$$

in appropriate norms. Complete details of this theory for the Navier-Stokes equations are given in a forthcoming paper [4]. If we can produce convergent *hp*-approximations and the rate of convergence is affected by the regularity of the solution, the theory merely demonstrates that we can provide a rigorous upper bound to the error. Here, confining our attention to the steady-state case, we generalize this approach to the free surface calculations.

First, we select an appropriate norm in which the error is to be estimated. Toward this end, we introduce the following norms :

$$\begin{aligned} |\boldsymbol{v}|_V^2 &= \sum_{K=1}^{N(\mathcal{P})} \int_{\Omega_K} 2\nu \boldsymbol{D}(\boldsymbol{v}) : \boldsymbol{D}(\boldsymbol{v}) dx \\ |q|_Q^2 &= \sum_{K=1}^{N(\mathcal{P})} \int_{\Omega_K} q^2 dx \\ |m|_M^2 &= \sum_{K=1}^{N(\mathcal{P})} \int_{\partial\Omega_F \cup \partial\Omega_K} \boldsymbol{\nabla} * m : \boldsymbol{\nabla} * m \, ds \end{aligned}$$

Corresponding to these choices, we have the energy–like norm,

$$|||(\boldsymbol{v}, q, m)|||^2 = |\boldsymbol{v}|_V^2 + |q|_Q^2 + |m|_M^2 \tag{14}$$

Without loss of generality, we restrict ourselves to the case without surface tension, although subsequent results may be generalized for the surface tension in a similar way. To obtain an upper bound to the error in this norm, we calculate the local error estimates $(\boldsymbol{u}_K^{est}, p_K^{est}, h_K^{est})$ by solving local elementwise boundary–value problems on each element "loaded" by the FEM residuals.

Find $(\boldsymbol{u}_K^{est}, p_K^{est}, h_K^{est}) \in V_K \times Q_K \times M_K$ *such that*

$$\begin{aligned}
\int_{\Omega_K} 2\nu \boldsymbol{D}(\boldsymbol{u}_K^{est}) : \boldsymbol{D}(\boldsymbol{v}_K) dx & \\
&= \int_{\Omega_K} -\boldsymbol{u}_K^{hp} \cdot \boldsymbol{\nabla} \boldsymbol{u}_K^{hp} \cdot \boldsymbol{v}_K - \boldsymbol{\tau}(\boldsymbol{u}_K^{hp}) : \boldsymbol{D}(\boldsymbol{v}_K)\ dx \\
&+ \int_{\Omega_K} p_K^{\widetilde{hp}}\, \boldsymbol{\nabla} \cdot \boldsymbol{v}_K + \boldsymbol{f} \cdot \boldsymbol{v}_K\ dx \\
&+ \int_{\partial\Omega_K} \left\langle \boldsymbol{n}_K \cdot \boldsymbol{\sigma}^{est}(\boldsymbol{u}_K^{hp}, p_K^{\widetilde{hp}}) \right\rangle \cdot \boldsymbol{v}_K ds \qquad \forall\ \boldsymbol{v}_K \in V_K, \\
\int_{\Omega_K} p_K^{est}\ q_K\ dx &= \int_{\Omega_K} q_K\, \boldsymbol{\nabla} \cdot \boldsymbol{u}_K^{hp}\, dx \qquad \forall\ q_K \in Q_K, \\
\int_{\partial\Omega_F \cup \partial\Omega_K} h_K^{est}\ m_K\ ds &= \int_{\partial\Omega_F \cup \partial\Omega_K} \boldsymbol{u}_K^{hp} \cdot \boldsymbol{n}_K\ m\, ds \qquad \forall\ m_K \in M_K, \\
& \qquad\qquad 1 \le K \le N(\mathcal{P}).
\end{aligned} \tag{15}$$

Here, subscripts K denote restrictions to element Ω_K. In particular, V_K, Q_K, M_K are spaces of restrictions of functions in V, Q, M to Ω_K. Note that M_K is nonempty only for the elements neighboring the moving boundary.

The last boundary term in (15.1) is of critical importance and is computed using a

equilibration technique [5]. For each pair of neighboring elements Ω_K and Ω_L, piecewise linear functions $\alpha_{KL}(s)$ are defined on the interelement boundaries $\partial\Omega_{KL} = \partial\Omega_K \cap \partial\Omega_L$ such that $\alpha_{KL}(s) + \alpha_{LK}(s) = 1$. The averaged approximate flux on $\partial\Omega_{KL}$ is then given by,

$$\left\langle \boldsymbol{n}_K \cdot \boldsymbol{\sigma}^{est}(\boldsymbol{u}_K^{hp}, p_K^{h\widetilde{p}}) \right\rangle = \boldsymbol{n}_K \left\{ \alpha_{LK}(s)\boldsymbol{\sigma}_K(\boldsymbol{u}_K^{hp}, p_K^{h\widetilde{p}}) + \alpha_{KL}(s)\boldsymbol{\sigma}_L(\boldsymbol{u}_K^{hp}, p_K^{h\widetilde{p}}) \right\} \tag{16}$$

where $s \in \partial\Omega_{KL}$ and $\boldsymbol{n}_K$ is a unit vector exterior and normal to $\partial\Omega_K$. The functions α_{KL} are then selected so that the element residual and the boundary residuals on the right hand side of (15.1) are balanced, rendering the local problem solvable.

In practical calculations, equation (15.1) is solved approximately over each element Ω_K using polynomial approximations of $\boldsymbol{u}_K^{est}$ and $\boldsymbol{v}_K$ which are, in general, of higher degree than those used in the approximation $\boldsymbol{u}^{hp}$. The resulting indicator is defined as follows :

$$\theta^2 = \sum_{K=1}^{N} \theta_K^2, \qquad \theta_K = |||(\boldsymbol{u}_K^{est}, div\, \boldsymbol{u}_K^{hp}, \boldsymbol{u}_K^{hp} \cdot \boldsymbol{n}_K)|||_K \tag{17}$$

Remark 4.1. It is worth noting that other choices could be used for the energy–like norm. For example, weighting coefficients can be introduced as follows

$$|||(\boldsymbol{v}, q, m)|||_*^2 = a_v|\boldsymbol{v}|_V^2 + a_p|q|_Q^2 + a_h|m|_M^2$$

Unit values for a_i leads to the norm $|||\ |||$. However for free surface problems, it may be advisable to bias the accuracy of the geometric degree of freedom and to increase the value a_h. Relative importance of the incompressibility constraint versus the error generated inside the momentum equations can also be controlled by adjusting the values of a_v and a_p.

It is obvious that considering a vanishing value of a_v will lead to a very cheap (but not accurate) estimate. Such procedure can, however, be useful for driving an adaptive strategy in very CPU expensive calculations.

4.2 *The Three–Step hp Adaptive Strategy*

Various adaptive strategies can be developed to control the numerical error by appropriate refinements and enrichments of the *hp* mesh. Here, we outline a three-step scheme that

attempts to minimize the computational effort required to reach a target error. The basic idea consists in solving the system on three meshes :

- First, we consider a mesh $\mathcal{P}^0$ that provides simply a rough and cheap approximation, but gives critical information needed for later refinements.
- Second, the intermediate mesh $\mathcal{P}^1$ is produced with only h adaptive refinements of the first mesh.
- Finally, the target mesh $\mathcal{P}^2$ is obtained by applying p adaptive enrichments.

To construct this scheme, we suppose that a global *a priori* estimate for a hp- approximation of (1)(2)(3) in the energy–like norm is (recall (9)) :

$$|||(\boldsymbol{u}^{error}, p^{error}, h^{error})|||^2 \leq \sum_{K=1}^{N(\mathcal{P})} \frac{h_K^{2\mu_K}}{p_K^{2\nu_K}} \Lambda_K^2$$

where h_K, p_K are respectively the size and the order of the element K for the velocity interpolation, Λ_K is a local unknown constant. The exponents μ_K, ν_K are also generally unknown and it is necessary to estimate them in some way. In the present algorithm, we do not distinguish between the order $\tilde{p} = p - 1$ of the pressure approximation and the order p of the velocities–coordinates and we user the velocity order p throughout.

Now, the three–step scheme introduces two major assumptions. This asymptotic estimate is treated as an equality and the actual error is available to sufficient accuracy through an *a posteriori* error estimate. Then we have

$$|||(\boldsymbol{u}^{est}, p^{est}, h^{est})|||^2 \approx \sum_{K=1}^{N(\mathcal{P})} \frac{h_K^{2\mu_K}}{p_K^{2\nu_K}} \Lambda_K^2 \tag{18}$$

We also introduce an error index :

$$\eta = \frac{|||(\boldsymbol{u}^{error}, p^{error}, h^{error})|||}{|||(\boldsymbol{u}, p, h)|||}$$

The starting point of the algorithm is the choice of a required target error index η^{tgt}.

- Introduce an initial mesh $\mathcal{P}^0$ of N^0 elements sufficiently fine to fall in the asymptotic part of the convergence curve for h-refinements. Solve the problem on this mesh. Calculate a local *a posteriori* error indicator θ_K^0 to estimate the local error.

  Now, setting respectively μ_K and ν_K to given μ,ν, we can estimate the constant Λ_K by using (18)

$$(\theta^0)^2 = \sum_{K=1}^{N^0} (\theta_K^0)^2 = \sum_{K=1}^{N^0} \frac{(h_K^0)^{2\mu}}{(p_K^0)^{2\nu}} (\Lambda_K)^2$$

and passing to the element level

$$\theta_K^0 \approx \frac{(h_K^0)^\mu}{(p_K^0)^\nu}\Lambda_K$$

From the orthogonality of the error to the space of approximation, we can estimate both the energy like norm of the solution and the initial error index

$$|||(\boldsymbol{u},p,h)|||^2 \approx |||(\boldsymbol{u}^0,p^0,h^0)|||^2 = |||(\boldsymbol{u}^{hp0},p^{hp0},h^{hp0})|||^2 + (\theta^0)^2$$

$$\eta^0 \approx \frac{\theta^0}{|||(\boldsymbol{u}^0,p^0,h^0)|||}$$

Select η^{int} such that $\eta^0 \leq \eta^{int} \leq \eta^{tgt}$.

- Calculate the number n_K of new sub-elements required in each element of $\mathcal{P}^0$ in order to obtain an optimal mesh $\mathcal{P}^1$ of N^1 elements achieving the required error index η^{int}.

  From (18), the predicted error estimate for the new mesh must satisfy

$$(\theta^{int})^2 = \sum_{K=1}^{N^0} n_K \ \frac{(h_K^1)^{2\mu}}{(p_K^0)^{2\nu}}(\Lambda_K)^2$$

  where h_K^1 is the mean size of new elements created inside the initial element K. If the error has also to be equidistributed, we obtain

$$(\theta^{int})^2 = N^1 \ \frac{(h_K^1)^{2\mu}}{(p_K^0)^{2\nu}}(\Lambda_K)^2$$

  For uniform refinements, the number of sub-elements can be correlated to their main size $n_K = (h_K^0/h_K^1)^{2/\beta}$ where $\beta = 2/n$ and n is the dimension of the problem. So, we have the following system which allows to compute n_K :

$$\left.\begin{array}{rcl} n_K & = & \left[N^1 \dfrac{(\theta_K^0)^2}{(\theta^{int})^2}\right]^{\frac{1}{\beta\mu}} \\ \displaystyle\sum_{K=1}^{N^0} n_K & = & N^1 \end{array}\right\} \qquad (19)$$

  where the global error θ^{int} is predicted by $\eta^{int}\ |||(\boldsymbol{u}^0,p^0,h^0)|||$. This non-linear system may easily be solved by an iterative scheme. Having n_K, we introduce h refinements to construct $\mathcal{P}^1$.

  Now, solve the problem on this second mesh and compute the local *a posteriori* error indicators θ_L^1.

- The third mesh $\mathcal{P}^2$ is constructed by calculating a distribution of polynomial degrees p_L for each element of $\mathcal{P}^1$ to reach the target error index η^{tgt}. At this stage, we have the error indicator θ^1 and we now estimate Λ_L on the h-adapted mesh $\mathcal{P}_1$ as

$$\theta_L^1 \approx \frac{(h_L^1)^\mu}{(p_L^0)^\nu}\Lambda_L$$

From eqn (18), the target equidistributed error indicator must satisfy

$$(\theta^{tgt})^2 = N^1 \frac{(h_L^1)^{2\mu}}{(p_L^2)^{2\nu}}(\Lambda_L)^2$$

Now, enrich p on each element to obtain $\mathcal{P}^2$. The final order of each element is given by

$$p_L = \left[N^1 \frac{(\theta_L^1)^2 (p_L^0)^{2\nu}}{(\theta^{tgt})^2} \right]^{\frac{1}{2\nu}} \tag{20}$$

where the global error θ^{tgt} is predicted by $\eta^{tgt}\ |||(\boldsymbol{u}^1, p^1, h^1)|||$.

Finally, solve the problem on $\mathcal{P}^2$ and compute an estimate of the final error index η^2. If $\eta^2 \leq \eta^{tgt}$ the computation is terminated; otherwise the whole procedure is repeated.

This technique leads to good but suboptimal meshes on model problems but exhibits very fast convergence characteristics with respect to CPU time.

Remark 4.2. However, the major weak points of the scheme is the selection of μ, ν and η^{int} which have a critical influence on the results. These, at present, are heuristic choices. The selection of η^{int} close to η^0 leads to an essentially p-adapted mesh while a larger η^{int} results in more h-adaptivity.

In our present numerical experiences, we selected the following values of $\mu = 1.5$ to 2 and $\nu = 0.5$ to 1. But, these values are clearly dependent on the problem. This selection has to be considered as a compromise between the robustness and the efficiency of the strategy.

Remark 4.3. In order to reduce the CPU time required for the error estimates θ^1 and θ^2, we introduce the following rule. The sets of uncritical elements of $\mathcal{P}^1$ and $\mathcal{P}^2$ are defined as follows

$$\mathcal{P}^1_{uncritical} = \left\{ \Omega_K \in \mathcal{P}^1 \ ; \ (\theta_K^0)^2 \leq \frac{(\theta^{tgt})^2}{N^1} \right\}$$

$$\mathcal{P}^2_{uncritical} = \left\{ \Omega_K \in \mathcal{P}^2 \ ; \ (\theta_L^1)^2 \leq \frac{(\theta^{tgt})^4}{(\theta^{int})^2 N^1} \right\}$$

In other words, the uncritical set contains all elements on which the error estimate is very low. These elements have not been modified at the previous step of the adaptive strategy

and we do not expect that they will be modified at the next step. So, we may safely consider the following approximations for the uncritical elements

$$\theta_K^1 \approx \theta_K^0$$

$$\theta_L^2 \approx \theta_L^1$$

when we are respectively estimating θ^1 and θ^2. On all others elements (called critical), the whole error estimate procedure is performed.

5 Numerical results

5.1 *Test problem 1 : 2D planar extrusion flow*

We first present the planar free surface calculation of the swelling problem. We consider the steady motion of an isothermal, incompressible Newtonian jet emerging from a slit as shown in Fig. 1. We impose no-slip conditions on the wall of the tube and a fully developed velocity profile in the entry section. Vanishing normal and tangential surface forces are imposed on the surface of the jet and in the last section. The lengths of the die and of the jet are respectively equal to 2 and 8.5 lengths of the inflow section.

From an initial mesh with 229 degrees of freedom (for a scalar unknown) and quadratic interpolation, the estimated error index is .17. Then, the 3-step adaptive strategy is used with an intermediate error index $\eta^{int} = 0.10$ and a target error index of $\eta^{tgt} = 0.05$. The three meshes and the error index evolution are shown in Fig. 5. It is noted that the element are h-refined near the singularity and that orders of $p = 4$ and $p = 3$ are assigned near this point. Shaded elements reflect nonuniform p-distribution in final mesh.

The local *equilibrated* error estimate ((α)-Estimated Error) is plotted in Fig. 6. Fig. 7. shows the local *unbalanced* error estimate ((0.5)-Estimated Error) which is not so expensive in CPU and provides almost similar results. Computed pressure and velocity components distribution are shown in Fig. 8. We also superpose the profiles of the swelling ratio along the first elements of the free surface for the three meshes of the adaptive procedure. The swelling ratio is defined as the relative increase of the extrudate to the inflow section. A dramatic improvement of the accuracy is produced by the adaptive strategy. We obtain a swelling ratio of 19.81% on the final mesh which is in excellent agreement with the results of the literature.

In order to illustrate the cost of the adaptive strategy, Table 1 provides the CPU time used for each part of the calculation. In particular, the equilibration doubles the cost of

Mesh	CPU for the solution	CPU for the error estimates		
		on all elements (α)	on all elements (0.5)	on uncritical elem. (0.5)
$\mathcal{P}^0$	672	109	55	55
$\mathcal{P}^1$	1543	359	173	160
$\mathcal{P}^2$	8375	662	418	203
Total	10590	1130	646	418
	100%	10%	6%	3%

Table 1: CPU time accounting for the planar extrusion problem

the error estimates although this cost can be divided by a factor 2 if the error is calculated only on critical elements. The numerical results reported in this work are obtained by a full Newton-Raphson scheme which allows us to reach a quadratic rate of convergence. For the examples shown, a direct frontal solver is used in each time step. The total number of iterations to reach the solution for a given mesh is approximately equal to 3 to 6 (relative variations of 10^{-9}).

Turning to the selection of η^{int}, Fig. 9. shows several alternative adaptive ways to reach η^{tgt}. The CPU time as a function of η^{int} is also provided in this figure. The average value is the total CPU time used for solving the problem and estimating an unbalanced error. The minimal value is the cost when the error is estimated only on critical elements and the maximal value is reached when we calculate equilibrated error estimates on all elements.

Surface tension and inertia have been successfully included. We consider the same problem with $Re = 100$ and $Ca = 1$. The presence of inertia effects requires that we modify the mesh : the length of the jet is now equal to 25 lengths of the inflow section. The final mesh is shown in Fig. 10. Closeup views of the 3 adaptive meshes, computed error distributions and results are shown in Fig. 11-12-13-14.

We also present result of the planar extrusion problem for a Power-Law fluid of index 0.2. The velocity distribution will be relatively flat across the die section with a large velocity gradient near the wall of the die. Such a velocity profile will reduces swelling effects. For very low values of the power law index (0.3 and lower), sharp velocity boundary layers require

Mesh	Number of DOF	Swelling ratio	Swelling ratio error	Error index η
$\mathcal{P}^0$	251	16.0%	25.2%	0.1049
$\mathcal{P}^1$	341	13.7%	7.0%	0.0632
$\mathcal{P}^2$	562	12.7%	0.1%	0.0361

Table 2: Axisymmetric extrusion problem : swelling ratio errors

finer meshes and a Picard scheme to obtain the results. The three meshes and the error index evolution are shown in Fig. 15. In Fig. 16-17, the distributions of calculated error estimates are given. Velocity contourlines and a pressure plot are shown in Fig 18.

5.2 *Test problem 2 : 2D axisymmetric extrusion flow*

Numerical simulations of the extrusion of a Newtonian fluid from a circular die have been available for some time [24,25,26]. For creeping Newtonian flow, the calculation gives a swelling ratio of 12.7% with a highly refined mesh. The swelling ratio is defined as the relative increase of the radius of the extrudate to the radius of the die. We solved the same problem with the 3-step strategy. The three meshes and the error index are shown in Fig.19. Critical improvements in the accuracy of the swelling ratio are obtained by using an adaptive strategy. We obtain very accurate results with fewer degrees of freedom. In Table 2, we compare the evolution of the error estimates and the true error of the swelling ratio.

Computed distributions of the error estimates and the results on the final mesh are indicated in Figs. 20-21-22.

5.3 *The rotating disk*

We consider the axisymmetric problem described in Figure 23. A disk is rotating in a cylindric tank. Inertia forces induce secondary motions in planes containing the axis of symmetry, while the shape of the free surface located at the top of the domain is deformed.

The present swirling problem has "2.5 dimensions", i.e. three velocity components are calculated in a domain which is two-dimensional (see the Appendix for detailed equations). It is assumed that the fluid does not slip along the wall nor along the rotating disk. On the free surface, capillarity forces and the kinematic condition are imposed. The usual conditions

are imposed along the axis of symmetry. Let us recall that the shape of the free surface is unknown. An horizontal angle conditions are imposed at both ends of the free surface. The results below have been obtained by a time marching procedure starting with the rest state and a flat surface.

Let R denote the radius of the disk and ω its rate of rotation. The Reynolds number for a Newtonian fluid, the Froude number and the Capillary number are respectively given by

$$Re = \frac{\omega\ R^2}{\nu}$$

$$Fr = \frac{\omega^2\ R}{g}$$

$$Ca = \frac{\gamma}{\nu\omega R}$$

In the present solution of this problem, the domain located above the disk in Fig. 23 is considered as a mobile domain, i.e. its elements are stretched when the free surface position is updated.

The flow features include the shape of the free surface and the secondary vortices generated in the tank together with the rotating motion. For $Re = 128, Fr = 7.34, Ca = 0.125$, the three deformed meshes produced by the 3-step adaptive strategy are shown on Fig. 24. Computed distributions of the error estimates are given in Figs. 25-26. In Fig. 27, one observes an S-shaped free surface while the velocity contourlines exhibits secondary vortices. The azimuthal velocity component which has its maximum on the outer boundary of the disk.

Acknowledgment: This work was performed at the Texas Institute for Computational Mechanics at the University of Texas at Austin. The support of DARPA of one author (J.T.Oden) under Contract #DABT63–92–0042 is gratefully aknowledged. V. Legat wishes to acknowledge the support from the Fonds National de la Recherche Scientifique (FNRS) and from a NATO Research Fellowship.

References

1. Oden, J. T.,Kennon S.R., Tworzydlo, W.W., Bass, J.M. and Berry, C., Progress on Adaptive *hp*-Finite Element Methods for the Incompressible Navier-Stokes Equations, *Journal of Computational Mechanics*, (in press), 1993.

2. Oden, J.T., Error Estimates and Control in Computational Fluid Dynamics,

Zienkiewicz Lecture of **MAFELAP VIII**, edited by J. Whiteman, J. Wiley and sons, Strasbourg, (to appear) , 1993.

3. Wu, W., *hp*-Adaptive Methods for Incompressible Viscous Flow Problems, Ph.D. Dissertation, The University of Texas, Austin, 1993

4. Oden, J. T., Wu, W., and Ainsworth, M., *A Posterori* Error Estimates for the Navier-Stokes Equations, TICOM Report 93-05, Austin and, to appear in *Computer Methods in Applied Mathematics and Engineering.*

5. Ainsworth, M., and Oden, J. T., A Unified Approach to *A Posteriori* Error Estimation Using Element Residual Methods, *Numerische Mathematik*, **54**, 1993.

6. Oden, J. T.,Patra, A. and Feng, Y., An *hp* Adaptive Strategy, **Adaptive, Multilevel and Hierarchical Computational Strategies** edited by A.K. Noor, AMD-Vol 157, ASME publications, pp. 23-46, 1993.

7. Legat, V. and Marchal, J.M., Prediction of Three-Dimensional Shape Extrudates by an Implicit Iterative Scheme, *International Journal for Numerical Methods in Fluids*, **14**, pp. 609–625, 1992.

8. Legat, V. and Marchal, J.M., Die Design : an Implicit Formulation for the Inverse Problem, *International Journal for Numerical Methods in Fluids*, **16**, pp. 29–42, 1993.

9. Ladyzhenskaya, O.A., **The Mathematical Theory of Viscous Incompressible Flow**, 1969, Gordon and Breach Science Publishers, New-York.

10. Berghezan, D. and Dupret, F., Numerical Solution of Stratified Coating Flow by a Variational Method, *Journal of Computational Physics*, (in press), 1993.

11. Frederiksen, C.S. and Watts, A.M., Finite Element Method for Time–Dependent Incompressible Free Surface Flows, *Journal of Computational Physics*, **39**, pp. 282–304, 1981.

12. Ruschak, K.J., A Method for Incorporating Free Boundary with Surface Tension in Finite-Element Fluid-Flow Simulators, *International Journal for Numerical Methods in Engineering*, **15**, pp. 639–648, 1980.

13. Kistler, S.F. and Scriven, L.E., Coating Flows, in **Computational Analysis of Polymer Processing**, edited by J.R.A. Pearson and S.M. Richardson, Chap. 8, Applied Science Publishers, London, 1983.

14. Demkowicz, L. F., Oden, J. T., Rachowicz, W., and Hardy, O., 1989, Toward a Unified Approach to *hp* Adaptive Finite Element Methods, Part 1: Constrained Approximation, *Computer Methods in Applied Mechanics and Engineering*, **77**, No. 1, pp. 79–112.

15. Babuska, I. and Suri, M., The *p* and *hp* Versions of the Finite Element Method, an Overview, *Computer Methods in Applied Mechanics and Engineering*, **80**, pp. 5–26, 1990.

16. Carey, G.F. and Oden, J.T., **Finite Elements : A Second Course**, Prentice Hall, Englewood Cliffs, 1983.

17. Ruschak, K.J., A Three-Dimensional Linear Stability Analysis for two-Dimensional Free Boundary Flows by Finite Element Methods, *Computers and Fluids*, **11**, pp. 391–401, 1983.

18. Saito, H. and Scriven, L.E., Study of Coating Flow by the Finite Element Method, *Journal of Computational Physics*, **42**, pp. 53–??, 1981.

19. Thompson, J.F., Warsi, Z.U.A. and Mastin, C.W., **Numerical Grid Generation, Foundations and Applications**, North Holland, New York, 1985.

20. Tworzydlo, W.W., Huang, C.Y. and Oden, J.T., Adaptive Implicit–Explicit Finite Element Method for Compressible Viscous Flows, *Computer Methods in Applied Mechanics and Engineering*, **97**, pp. 245–288, 1992.

21. Huang, C.Y. and Oden, J.T., GAMMA2D A Multiregion–Multiblock Structured–Unstructured Grid Generation Package for Computational Mechanics, *Computers and Structures*, (in press) , 1993.

22. Wambersie, 0. and Crochet, M.J. Transient Finite Element Method for Calculating Steady State Three-Dimensional Free Surfaces, *International Journal for Numerical Methods in Fluids*, **14**, pp. 343–360, 1992.

23. Gresho, P. M., On the Theory of Semi–Implicit Projection Methods for Viscous Incompressible Flow and its Implementation via a Finite Element Methods that Also Introduces a Nearly Consistent Mars Matrix: Part 1. Theory. In *International Journal for Numerical Methods in Fluids*, **11**, pp. 587–620, 1990.

24. Georgiou, G.C., Schultz, W.W. and Olson, L.G., Singular Finite Element for the Sudden-Expansion and the Die-Swell Problems, *International Journal for Numerical Methods in Fluids*, **10**, pp. 357–372, 1990.

25. Nickell, R.E., Tanner, R.I. and Caswell, B. Finite Element Solution of Viscous Jet Flow with Surface Tension, *Journal of Fluid Mechanics* **65** pp. 189–206, 1974.

26. Han, C.T., Tsai, C.C., and Liu, T.J., A Finite Difference Technique for Solving Newtonian Jet Swell Problem, *International Journal for Numerical Methods in Fluids*, **15**, pp. 773–790, 1992.

Appendix In this section, we give the complete weak formulation of axisymmetric Navier-Stokes equations with three velocity components. Time derivatives are included for completeness.

In above formula, r and z denote the radial and axial coordinates. Along the boundary, we define s as the curvilinear coordinate. Using the comma for partial differentiation, we introduce the three velocity components u_r, u_z, u_θ and the pressure p and their differentiates $u_{r,t}, u_{r,r}, u_{r,z}$ ... By multiplying the momentum equation by an test function v and by dividing with a common factor 2π, the use of the divergence theorem (as described in section 2.3) leads to the following equations

$$
\begin{aligned}
\int_\Omega (u_{r,t} + r_{,t}u_{r,r} + z_{,t}u_{r,z})vrdrdz &= \int_\Omega ((u_r u_{r,r} + u_z u_{r,z})vr - u_\theta u_\theta v)drdz \\
&+ \int_\Omega \nu((2u_{r,r}v_r + (u_{r,z} + u_{z,r})v_z)r + 2\frac{u_r}{r}v)drdz \\
&- \int_\Omega (pv_{,r}r + pv)drdz - \int_\Omega f_r vrdrdz \\
&+ \int_{\partial\Omega_N} \widehat{g}_r vrds \\
&+ \int_{\partial\Omega_F} \gamma(v_{,s}r_{,s}r + v)ds - [\![\gamma\widehat{m}_r vr]\!]_{\partial\Omega_F} \\
\int_\Omega (u_{z,t} + r_{,t}u_{z,r} + z_{,t}u_{z,z})vrdrdz &= \int_\Omega (u_r u_{z,r} + u_z u_{z,z})vrdrdz \\
&+ \int_\Omega \nu((u_{r,z} + u_{z,r})v_r + 2u_{z,z}v_z)rdrdz \\
&- \int_\Omega pv_{,z}rdrdz - \int_\Omega f_z vrdrdz \\
&+ \int_{\partial\Omega_N} \widehat{g}_z vrds \\
&+ \int_{\partial\Omega_F} \gamma v_{,s}z_{,s}rds - [\![\gamma\widehat{m}_z vr]\!]_{\partial\Omega_F}
\end{aligned}
$$

$$
\begin{aligned}
\int_\Omega (u_{\theta,t} + r_{,t}u_{\theta,r} + z_{,t}u_{\theta,z})vrdrdz &= \int_\Omega ((u_r u_{\theta,r} + u_z u_{\theta,z})vr + u_r u_\theta v)drdz \\
&+ \int_\Omega \nu(((u_{\theta,r} - \frac{u_\theta}{r})v_r + u_{\theta,z}v_z)r - (u_{\theta,r} - \frac{u_\theta}{r})v)drdz \\
&+ \int_{\partial\Omega_N} \widehat{g}_\theta vrds
\end{aligned}
$$

where $\boldsymbol{f}$ and $\hat{\boldsymbol{g}}$ are respectively the body forces and the given traction on the Neumann portion of the boundary. The unit vectors $\widehat{\boldsymbol{m}}$ define the angle conditions for both ends of the free boundary. For transient processes, $r_{,t}$ and $x_{,t}$ are the velocity components of the frame motion (or grid motion) introduced by the motion of the free surface and the stretching algorithm. Finally, it should be noted that the axisymmetric formulation can be directly derived by eliminating the third equation and the third velocity equation in these formula.

Introducing an arbitrary pressure q and an arbitrary geometric degree of freedom m, the weak forms of the incompressibility constraint and the kinematic condition are given by

$$\int_{\Omega}((u_{r,r}+u_{z,z})qr+u_{r}q)drdz = 0$$

$$\int_{\partial\Omega_F}(u_{r}n_{r}+u_{z}n_{z})mrdrdz = 0$$

where $\boldsymbol{n}$ is the unit normal vector pointing out of the domain. We also give the components of the strain rate tensor which are required for the calculation of his second invariant for Non-Newtonian fluids.

$$d_{rr} = u_{r,r}$$

$$d_{rz} = d_{zr} = \frac{u_{r,z}+u_{z,r}}{2}$$

$$d_{r\theta} = d_{\theta r} = u_{\theta}-\frac{u_{\theta}}{2r}$$

$$d_{zz} = u_{z,z}$$

$$d_{\theta\theta} = \frac{u_{r}}{r}$$