

A GENERAL STABILIZED FORMULATION FOR INCOMPRESSIBLE FLUID FLOW USING FINITE CALCULUS AND THE FINITE ELEMENT METHOD

E. Oñate, J. García, G. Bugada and S.R. Idelsohn

International Centre for Numerical Methods in Engineering
Universidad Politécnica de Cataluña
Gran Capitán s/n, 08034 Barcelona, Spain
E-mail: onate@cimne.upc.es
Web page: www.cimne.upc.es

Abstract. We present a general formulation for incompressible fluid flow analysis using the finite element method (FEM). The necessary stabilization for dealing with convective effects and the incompressibility condition are introduced via the so called finite calculus (FIC) method. The extension of the standard eulerian form of the equations to an arbitrary lagrangian-eulerian (ALE) frame adequate for treating fluid-structure interaction problems is presented. The fully lagrangian form is also discussed. Details of an effective mesh updating procedure are presented together with a method for dealing with free surface effects of importance for ship hydrodynamic analysis and many other fluid flow problems. Examples of application of the eulerian, the ALE and the fully lagrangian flow descriptions are presented.

Key words: Stabilized formulation, incompressible fluid flow, finite calculus, finite element method, ship hydrodynamics.

1 INTRODUCTION

The development of efficient and robust numerical methods for analysis of incompressible flows has been a subject of intensive research in last decades. Much effort has been spent in developing the so called stabilized numerical methods overcoming the two main sources of instability in incompressible flow analysis, namely those originated by the high values of the convective terms and those induced by the difficulty in satisfying the incompressibility conditions.

The solution of above problems in the context of the finite element method (FEM) has been attempted in a number of ways [1]. The underdiffusive character of the Galerkin FEM for high convection flows (which incidentally also occurs for the central finite difference (FD) and finite volume (FV) methods [2]) has been corrected by adding some kind of artificial viscosity terms to the standard Galerkin equations.

A popular way to overcome the problems with the incompressibility constraint is by introducing a pseudo-compressibility in the flow and using implicit and explicit algorithms developed for this kind of problems such as artificial compressibility

schemes [4–6] and preconditioning techniques [7]. Other FEM schemes with good stabilization properties for the convective and incompressibility terms are based in Petrov-Galerkin (PG) techniques. The background of PG methods are the non-centred (upwind) schemes for computing the first derivatives of the convective operator in FD and FV methods [2,8]. More recently a general class of Galerkin FEM has been developed where the standard Galerkin variational form is extended with adequate residual-based terms in order to achieve a stabilized numerical scheme. Among the many methods of this kind in the finite element universe we can name the Streamline Upwind Petrov Galerkin (SUPG) method [1,9–18], the Galerkin Least Square (GLS) method [19,20], the Taylor-Galerkin method [21], the Characteristic Galerkin method [22–24] and its variant the Characteristic Based Split (CBS) method [25,26], pressure gradient operator methods [27] and the Subgrid Scale (SS) method [28–30]. A good review of these methods can be found in [31].

In this paper a stabilized finite element formulation for incompressible flows is derived in a different manner. The starting point are the modified governing differential equations of the fluid flow problem formulated via a finite calculus (FIC) approach [32]. The FIC method is based in invoking the balance of fluxes in a fluid domain of finite size. This introduces naturally additional terms in the classical differential equations of infinitesimal fluid mechanics which are a function of the balance domain dimensions. The new terms in the modified governing equations provide the necessary stabilization to the discrete equations obtained via the standard Galerkin finite element method [33–39].

The layout of the chapter is the following. In the next section, the main concepts of the FIC approach are introduced via a simple 1D convection-diffusion model problem. Then the basic FIC equations for incompressible flow problems are presented. The finite element discretization is introduced and the resulting matrix formulation is detailed. Both monolithic and fractional step schemes for the transient solution are presented.

The basic formulation is extended to account for free surface wave effects by using an arbitrary eulerian-lagrangian (ALE) frame and introducing the free surface boundary conditions. Here the numerical treatment of the free surface equation using the FIC method is presented. The analysis of fluid-structure interaction problems involving the movement of floating or submerged solids in a fluid is also discussed. These problems require the displacement of the mesh nodes in accordance with the motion of the structure or the free surface and here a simple and effective algorithm for updating the mesh nodes is described. In the last part of the chapter the fully lagrangian formulation for fluid flow analysis is presented as a particular case of the ALE form. The lagrangian description has many advantages for tracking the displacement of fluid particles in flows where large motions of the fluid surface occur such in the case of breaking waves, splashing of water, filling of moulds, etc. A positive feature of the lagrangian formulation is that the convective terms disappear in the governing equations of the fluid flow; in return the updating of the mesh at almost every time step is now a necessity and efficient algorithms for mesh generation must be used.

The examples show the efficiency of the eulerian, ALE and fully lagrangian formulations to solve classical fluid flow problems, as well as fluid-structure interaction situations involving contact with moving solids, waves around ships and large motions of the free surface, among others.

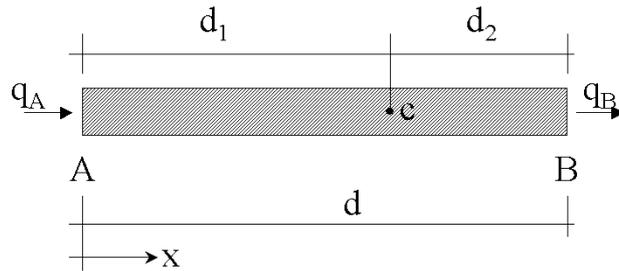


Figure 1: Equilibrium of fluxes in a balance domain of finite size

2 THE FINITE CALCULUS METHOD

We will consider a convection-diffusion problem in a 1D domain Ω of length L . The equation of balance of fluxes in a subdomain of size d belonging to Ω (Figure 1) is written as

$$q_A - q_B = 0 \quad (1)$$

where q_A and q_B are the incoming and outgoing fluxes at points A and B , respectively. The flux q includes both convective and diffusive terms; i.e. $q = u\phi - k\frac{d\phi}{dx}$, where ϕ is the transported variable, v is the velocity and k is the diffusivity of the material.

Let us express now the fluxes q_A and q_B in terms of the flux at an arbitrary point C within the balance domain (Figure 1). Expanding q_A and q_B in Taylor series around point C up to second order terms gives

$$q_A = q_C - d_1 \frac{dq}{dx}|_C + \frac{d_1^2}{2} \frac{d^2q}{dx^2}|_C + O(d_1^3) \quad , \quad q_B = q_C + d_2 \frac{dq}{dx}|_C + \frac{d_2^2}{2} \frac{d^2q}{dx^2}|_C + O(d_2^3) \quad (2)$$

Substituting eq.(2) into eq.(1) gives after simplification

$$\frac{dq}{dx} - \frac{h}{2} \frac{d^2q}{dx^2} = 0 \quad (3)$$

where $h = d_1 - d_2$ and all derivatives are computed at point C .

Standard calculus theory assumes that the domain d is of infinitesimal size and the resulting balance equation is simply $\frac{dq}{dx} = 0$. We will relax this assumption and allow the balance domain to have a *finite size*. The new balance equation (3) incorporates now the underlined term which introduces the *characteristic length* h . Obviously, accounting for higher order terms in eq.(2) would lead to new terms in eq.(3) involving higher powers of h .

Distance h in eq.(3) can be interpreted as a free parameter depending, of course, on the location of point C (note that $-d \leq h \leq d$). However, the fact that eq.(3) is the exact balance equation (up to second order terms) for any 1D domain of finite size and that the position of point C is arbitrary, can be used to derive numerical schemes with enhanced properties simply by computing the characteristic length parameter from an adequate “optimality” rule.

Consider, for instance, the modified equation (3) applied to the convection-diffusion problem. Neglecting third order derivatives of ϕ , eq.(3) can be written in an explicit form as

$$-u \frac{d\phi}{dx} + \left(k + \frac{uh}{2} \right) \frac{d^2\phi}{dx^2} = 0 \quad (4)$$

We see that the modified equation via the FIC method introduces *naturally* an additional diffusion term into the standard convection-diffusion equation. This is the basis of the popular “artificial diffusion” procedure [1,2,8,36]. The characteristic length h is typically expressed as a function of the cell or element dimensions. The optimal or critical value of h for each cell or element can be computed from numerical stability conditions such as obtaining a physically meaningful solution, or even obtaining “exact” nodal values [32–39].

Equation (3) can be extended to account for source effects. The full stabilized equation can be then written in compact form as

$$r - \frac{h}{2} \frac{dr}{dx} = 0 \quad (5)$$

with

$$r = -u \frac{d\phi}{dx} + \frac{d}{dx} \left(k \frac{d\phi}{dx} \right) + Q \quad (6)$$

where Q is the external source. For consistency a “finite” form of the Neumann boundary condition should be used. This can be readily obtained by invoking balance of fluxes in a domain of finite size next to the boundary Γ_q where the external (diffusive) flux is prescribed to a value q_p . The modified Neumann boundary condition can be written as [32]

$$k \frac{d\phi}{dx} + q_p - \frac{h}{2} r = 0 \quad \text{at } \Gamma_q \quad (7)$$

The definition of the problem is completed with the standard Dirichlet condition prescribing the value of ϕ at the boundary Γ_ϕ .

The underlined terms in Eqs.(5) and (7) introduce the necessary stabilization in the discrete solution of the problem using whatever numerical scheme. For details see [32–41].

The time dimension can be introduced in the FIC method by considering the balance equation in a space-time slab domain [32,35,36]. Quite generally the FIC equation can be written for any problem in mechanics as [32]

$$r_i - \frac{h_j}{2} \frac{\partial r_i}{\partial x_j} - \frac{\delta}{2} \frac{\partial r_i}{\partial t} = 0 \quad , \quad \begin{matrix} i = 1, n_b \\ j = 1, n_d \end{matrix} \quad (8)$$

where r_i is the i th standard differential equation of the infinitesimal theory, h_j are characteristic length parameters, δ is a time stabilization parameter and t the time; n_b and n_d are respectively the number of balance equations and the number of dimension of the problem along which balance of fluxes or forces is invoked (i.e., $n_d = 2$ for 2D problems, etc.).

For example, in the case of the convection-diffusion problem $n_b = 1$, Eq.(8) is particularized as

$$r - \frac{h_j}{2} \frac{\partial r}{\partial x_j} - \frac{\delta}{2} \frac{\partial r}{\partial t} = 0 \quad , \quad j = 1, n_d \quad (9)$$

with $r := - \left(\frac{\partial \phi}{\partial t} + u_i \frac{\partial \phi}{\partial x_i} \right) + \frac{d}{dx_k} \left(k \frac{d\phi}{dx_k} \right) + Q$.

The modified Neumann boundary conditions in the FIC formulation can be expressed in the general case as

$$q_{ij}n_j - t_i - \underline{h_j n_j r_i} = 0 \quad \text{on } \Gamma_q \quad i = 1, n_d \quad (10)$$

where q_{ij} are the generalized ‘‘fluxes’’ (such as the heat fluxes in a heat transfer problem or the stresses in solid or fluid mechanics), t_i are the prescribed values of the boundary fluxes and n_j are the components of the outward normal to the Neumann boundary Γ_q . For the transient case the initial boundary condition should also be specified [32,35,36].

In Eqs.(8)–(10) we have underlined once more the terms introduced by the FIC approach which are essential for deriving stabilized numerical formulations.

The starting point in the next section are the FIC equation for a viscous incompressible fluid. A simplified version of the equation will be chosen neglecting the time stabilization term as this is not relevant for the purposes of this work.

3 GENERAL FIC EQUATIONS FOR VISCOUS INCOMPRESSIBLE FLOW

The FIC governing equations for a viscous incompressible fluid can be written as

Momentum

$$r_{m_i} - \underline{\frac{1}{2} h_j \frac{\partial r_{m_i}}{\partial x_j}} = 0 \quad (11)$$

Mass balance

$$r_d - \underline{\frac{1}{2} h_j \frac{\partial r_d}{\partial x_j}} = 0 \quad (12)$$

where

$$r_{m_i} = \rho \left(\frac{\partial u_i}{\partial t} + \frac{\partial (u_i u_j)}{\partial x_j} \right) + \frac{\partial p}{\partial x_i} - \frac{\partial s_{ij}}{\partial x_j} - b_i \quad (13)$$

$$r_d = \frac{\partial u_i}{\partial x_i} \quad i, j = 1, n_d \quad (14)$$

Above u_i is the velocity along the i th global axis, ρ is the (constant) density of the fluid, p is the absolute pressure (defined positive in compression), b_i are the body forces and s_{ij} are the viscous deviatoric stresses related to the viscosity μ by the standard expression

$$s_{ij} = 2\mu \left(\dot{\epsilon}_{ij} - \delta_{ij} \frac{1}{3} \frac{\partial u_k}{\partial x_k} \right) \quad (15)$$

where δ_{ij} is the Kronecker delta and the strain rates $\dot{\epsilon}_{ij}$ are

$$\dot{\epsilon}_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (16)$$

The FIC boundary conditions are

$$n_j \sigma_{ij} - t_i + \frac{1}{2} h_j n_j r_{m_i} = 0 \quad \text{on } \Gamma_t \quad (17)$$

$$u_j - u_j^p = 0 \quad \text{on } \Gamma_u \quad (18)$$

and the initial condition is $u_j = u_j^0$ for $t = t_0$.

In Eqs.(17) and (18) t_i and u_j^p are surface tractions and prescribed displacements on the boundaries Γ_t and Γ_u , respectively, n_j are the components of the unit normal vector to the boundary and σ_{ij} are the total stresses given by $\sigma_{ij} = s_{ij} - \delta_{ij}p$. The sign in front the stabilization term in Eq.(17) is positive due to the definition of r_{m_i} in Eq.(13).

The h_i 's in above equations are characteristic lengths of the domain where balance of momentum and mass is enforced. In Eq.(17) these lengths define the domain where equilibrium of boundary tractions is established [32].

Eqs.(11)–(18) are the starting point for deriving stabilized finite element methods for solving the incompressible Navier-Stokes equations using equal order interpolation for the velocity and pressure variables [37–39]. Application of the FIC formulations to meshless analysis of fluid flow problems using the finite point method can be found in [40,41].

We note that the ‘‘conservative’’ form of the convective terms in Eq.(13) and the presence of the volumetric strain rate in the constitutive equation (15) do not take advantage of the incompressibility condition. These forms are useful for obtaining the relationship between the derivative of the volumetric strain rate and the momentum equations as shown in the next section. However, the standard form of the governing equations for incompressible flows will be used for the final derivation of the discretized FEM equations.

3.1 Stabilized integral forms

From Eq.(11) it can be obtained (taking into account Eq.(15))

$$\frac{\partial r_d}{\partial x_i} = -\frac{1}{a_i} \left[\bar{r}_{m_i} - \frac{h_j}{2} \frac{\partial r_{m_i}}{\partial x_j} \right] - \frac{\rho u_i h_k}{2a_i} \frac{\partial r_d}{\partial x_k} \quad i, j = 1, n_d \quad , \quad k \neq i \quad (19)$$

where

$$a_i = \frac{2\mu}{3} + \frac{\rho u_i h_i}{2} \quad (20a)$$

and

$$\bar{r}_{m_i} = \rho \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) + \frac{\partial p}{\partial x_i} - \frac{\partial}{\partial x_j} (2\mu \dot{\epsilon}_{ij}) - b_i \quad (20b)$$

Substituting Eq.(19) into Eq.(12) and retaining the terms involving the derivatives of r_{m_i} with respect to x_i only, leads to the following expression for the stabilized mass balance equation

$$r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} = 0 \quad (21)$$

with

$$\tau_i = \left(\frac{8\mu}{3h_i^2} + \frac{2\rho u_i}{h_i} \right)^{-1} \quad (22)$$

The τ_i 's in Eq.(21) are termed in the stabilization literature *intrinsic time parameters*. It is interesting to note that these parameters take here the values of $\tau_i = \frac{3h_i^2}{8\mu}$ and $\tau_i = \frac{h_i}{2\rho u_i}$ for the viscous limit (Stokes flow) and the inviscid limit (Euler flow), respectively. Similar values for τ_i (usually $\tau_i = \tau$ is taken) are used in other works from ad-hoc extensions of the 1D advective-diffusive problem [12–31]. It is remarkable that the intrinsic time parameters have been deduced here from the general FIC formulation and this shows the possibilities of the method.

The weighted residual form of the momentum and mass balance equations (Eqs.(11) and (21)) is written as

Momentum

$$\int_{\Omega} \delta u_i \left[r_{m_i} - \frac{h_j}{2} \frac{\partial r_{m_i}}{\partial x_j} \right] + \int_{\Gamma_t} \delta u_i (\sigma_{ij} n_j - t_i + \frac{h_j}{2} n_j r_{m_i}) d\Gamma = 0 \quad (23)$$

Mass balance

$$\int_{\Omega} q \left[r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} \right] d\Omega = 0 \quad (24)$$

where δu_i and q are arbitrary weighting functions representing virtual velocity and virtual pressure fields. Integration by parts of the r_{m_i} terms leads to

$$\int_{\Omega} \delta u_i r_{m_i} + \int_{\Gamma_t} \delta u_i (\sigma_{ij} n_j - t_i) d\Gamma + \sum_e \int_{\Omega^e} \frac{h_j}{2} \frac{\partial \delta u_i}{\partial x_j} r_{m_i} d\Omega = 0 \quad (25a)$$

$$\int_{\Omega} q r_d d\Omega + \int_{\Omega} \left[\sum_{i=1}^{n_d} \tau_i \frac{\partial q}{\partial x_i} r_{m_i} \right] d\Omega - \int_{\Gamma} \left[\sum_{i=1}^{n_d} q \tau_i n_i r_{m_i} \right] d\Gamma = 0 \quad (25b)$$

The third integral in Eq.(25a) is expressed as a sum of the element contributions to allow for discontinuities in the derivatives of r_{m_i} along the element interfaces.

Also in Eq.(25b) we will neglect hereonwards the third integral by assuming that r_{m_i} is negligible on the boundaries. The deviatoric stresses and the pressure terms in the first integral of Eq.(25a) are integrated by parts in the usual manner. The resulting equations are

Momentum

$$\int_{\Omega} \left[\delta u_i \rho \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) + \delta \dot{\epsilon}_{ij} (s_{ij} - \delta_{ij} p) \right] d\Omega - \int_{\Omega} \delta u_i b_i d\Omega - \int_{\Gamma_t} \delta u_i t_i d\Gamma + \sum_e \int_{\Omega^e} \frac{h_j}{2} \frac{\partial \delta u_i}{\partial x_j} r_{m_i} d\Omega = 0 \quad (26)$$

Mass balance

$$\int_{\Omega} q \frac{\partial u_i}{\partial x_i} d\Omega + \int_{\Omega} \left[\sum_{i=1}^{n_d} \tau_i \frac{\partial q}{\partial x_i} r_{m_i} \right] d\Omega = 0 \quad (27)$$

We note that in Eq.(26) we use now the standard form of the convective operator for incompressible flows (i.e. neglecting the contribution from the volumetric strain rate $\frac{\partial u_i}{\partial x_i}$). Also in Eq.(26)

$$\delta \dot{\varepsilon}_{ij} = \frac{1}{2} \left(\frac{\partial \delta u_i}{\partial x_j} + \frac{\partial \delta u_j}{\partial x_i} \right)$$

3.2 Convective and pressure gradient projections

The computation of the residual terms can be simplified if we introduce now the convective and pressure gradient projections c_i and π_i , respectively defined as

$$\begin{aligned} c_i &= r_{m_i} - \rho u_j \frac{\partial u_i}{\partial x_j} \\ \pi_i &= r_{m_i} - \frac{\partial p}{\partial x_i} \end{aligned} \quad (28)$$

We can express r_{m_i} in Eqs.(26) and (27) in terms of c_i and π_i , respectively which then become additional variables. The system of integral equations is now augmented in the necessary number of additional equations by imposing that the residual r_{m_i} vanishes (in average sense) for both forms given by Eqs.(28). This gives the final system of governing equation as:

$$\begin{aligned} \int_{\Omega} \left[\delta u_i \rho \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) + \delta \dot{\varepsilon}_{ij} (s_{ij} - \delta_{ij} p) \right] d\Omega - \int_{\Omega} \delta u_i b_i d\Omega - \int_{\Gamma_t} \delta u_i t_i d\Gamma + \\ + \sum_e \int_{\Omega^e} \frac{h_k}{2} \frac{\partial (\delta u_i)}{\partial x_k} \left(\rho u_j \frac{\partial u_i}{\partial x_j} + c_i \right) d\Omega = 0 \end{aligned} \quad (29)$$

$$\int_{\Omega} q \frac{\partial u_i}{\partial x_i} d\Omega + \int_{\Omega} \sum_{i=1}^{n_d} \tau_i \frac{\partial q}{\partial x_i} \left(\frac{\partial p}{\partial x_i} + \pi_i \right) d\Omega = 0 \quad (30)$$

$$\int_{\Omega} \delta c_i \rho \left(\rho u_j \frac{\partial u_i}{\partial x_j} + c_i \right) d\Omega = 0 \quad \text{no sum in } i \quad (31)$$

$$\int_{\Omega} \delta \pi_i \tau_i \left(\frac{\partial p}{\partial x_i} + \pi_i \right) d\Omega = 0 \quad \text{no sum in } i \quad (32)$$

with $i, j, k = 1, n_d$. In Eqs.(31) and (32) δc_i and $\delta \pi_i$ are appropriate weighting functions and the ρ and τ_i weights are introduced for convenience.

4 FINITE ELEMENT DISCRETIZATION

We choose C^0 continuous linear interpolations of the velocities, the pressure, the convection projections c_i and the pressure gradient projections π_i over three node triangles (2D) and four node tetrahedra (3D). The linear interpolations are written as

$$\begin{aligned} u_i &= \sum_{j=1}^n N_j \bar{u}_i^j \quad , \quad p = \sum_{j=1}^n N_j \bar{p}^j \\ c_i &= \sum_{j=1}^n N_j \bar{c}_i^j \quad , \quad \pi_i = \sum_{j=1}^n N_j \bar{\pi}_i^j \end{aligned} \quad (33)$$

where $n = 3$ (4) for triangles (tetrahedra), $(\bar{\cdot})^j$ denotes nodal variables and N_j are the linear shape functions [1].

Substituting the approximations (33) into Eqs.(29–32) and choosing the Galerking form with $\delta u_i = q = \delta c_i = \delta \pi_i = N_i$ leads to following system of discretized equations

$$\mathbf{M}\dot{\bar{\mathbf{u}}} + (\mathbf{A} + \mathbf{K} + \hat{\mathbf{K}})\bar{\mathbf{u}} - \mathbf{G}\bar{\mathbf{p}} + \mathbf{C}\bar{\mathbf{c}} = \mathbf{f} \quad (34a)$$

$$\mathbf{G}^T \bar{\mathbf{u}} + \mathbf{L}\bar{\mathbf{p}} + \mathbf{Q}\bar{\boldsymbol{\pi}} = \mathbf{0} \quad (34b)$$

$$\hat{\mathbf{C}}\bar{\mathbf{u}} + \mathbf{M}\bar{\mathbf{c}} = \mathbf{0} \quad (34c)$$

$$\mathbf{Q}^T \bar{\mathbf{p}} + \hat{\mathbf{M}}\bar{\boldsymbol{\pi}} = \mathbf{0} \quad (34c)$$

where the element contributions are given by (for 2D problems)

$$\begin{aligned} M_{ij} &= \int_{\Omega^e} \rho N_i N_j d\Omega \quad , \quad A_{ij} = \int_{\Omega^e} N_i \rho \mathbf{u}^T \boldsymbol{\nabla} N_j d\Omega \\ \mathbf{K}_{ij} &= \int_{\Omega^e} \mathbf{B}_i^T \mathbf{D} \mathbf{B}_j d\Omega \quad , \quad \hat{\mathbf{K}}_{ij} = \int_{\Omega^e} \rho \frac{h_k u_l}{2} \frac{\partial N_i}{\partial x_k} \frac{\partial N_j}{\partial x_l} d\Omega \\ \mathbf{G}_{ij} &= \int_{\Omega^e} (\boldsymbol{\nabla} N_i) N_j d\Omega \quad , \quad \boldsymbol{\nabla} = \left[\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2} \right]^T \quad , \quad C_{ij} = \int_{\Omega^e} \frac{h_k}{2} \frac{\partial N_i}{\partial x_k} N_j d\Omega \\ L_{ij} &= \int_{\Omega^e} \boldsymbol{\nabla}^T N_i [\boldsymbol{\tau}] \boldsymbol{\nabla} N_j d\Omega \quad , \quad [\boldsymbol{\tau}] = \begin{bmatrix} \tau_1 & 0 \\ 0 & \tau_2 \end{bmatrix} \quad , \quad \hat{\mathbf{C}}_{ij} = \int_{\Omega^e} N_i \rho^2 u_k \frac{\partial N_j}{\partial x_k} d\Omega \\ \mathbf{Q} &= [\mathbf{Q}^1, \mathbf{Q}^2] \quad , \quad Q_{ij}^k = \int_{\Omega^e} \tau_k \frac{\partial N_i}{\partial x_k} N_j d\Omega \\ \hat{\mathbf{M}} &= \begin{bmatrix} \hat{\mathbf{M}}^1 & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{M}}^2 \end{bmatrix} \quad , \quad \hat{\mathbf{M}}_{ij} = \int_{\Omega^e} \tau_k N_i N_j d\Omega \\ \mathbf{f}_i &= \int_{\Omega^e} N_i \mathbf{b} d\Omega + \int_{\Gamma^e} N_i \mathbf{t} d\Gamma \quad , \quad \mathbf{b} = [b_1, b_2]^T \quad , \quad \mathbf{t} = [t_1, t_2]^T \end{aligned} \quad (35)$$

with $i, j = 1, n$ and $k, l = 1, n_d$.

In above \mathbf{B}_i is the standard strain rate matrix and \mathbf{D} the deviatoric constitutive matrix (assuming $\frac{\partial u_i}{\partial x_i} = 0$). For 2D problems

$$\mathbf{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x_1} & 0 \\ 0 & \frac{\partial N_i}{\partial x_2} \\ \frac{\partial N_i}{\partial x_2} & \frac{\partial N_i}{\partial x_1} \end{bmatrix}, \quad \mathbf{D} = \mu \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (36)$$

Note that the stabilization matrix $\hat{\mathbf{K}}$ adds an additional orthotropic diffusivity of value $\rho \frac{h_k u_l}{2}$.

4.1 Quasi-implicit scheme

It can be seen that matrices \mathbf{A} , $\hat{\mathbf{K}}$ and $\hat{\mathbf{C}}$ are dependent on the velocity field. The solution process can be advanced in time in a (quasi-nearly) implicit iterative manner using the following scheme.

Step 1

$$\bar{\mathbf{u}}^{n+1,i} = \bar{\mathbf{u}}^n - \Delta t \mathbf{M}^{-1} [(\mathbf{A}^{n+\theta_1,i-1} + \mathbf{K} + \hat{\mathbf{K}}^{n+\theta_1,i-1}) \bar{\mathbf{u}}^{n+\theta_1,i-1} - \mathbf{G} \bar{\mathbf{p}}^{n+\theta_2,i-1} + \mathbf{C} \bar{\mathbf{c}}^{n+\theta_3,i-1} - \mathbf{f}^{n+1}] \quad (37)$$

Step 2

$$\bar{\mathbf{p}}^{n+1,i} = -\mathbf{L}^{-1} [\mathbf{G}^T \bar{\mathbf{u}}^{n+1,i} + \mathbf{Q} \bar{\boldsymbol{\pi}}^{n+\theta_4,i-1}] \quad (38)$$

Step 3

$$\bar{\mathbf{c}}^{n+1,i} = -\mathbf{M}^{-1} \hat{\mathbf{C}}^{n+1,i} \bar{\mathbf{u}}^{n+1,i} \quad (39)$$

Step 4

$$\bar{\boldsymbol{\pi}}^{n+1,i} = -\hat{\mathbf{M}}^{-1} \mathbf{Q}^T \bar{\mathbf{p}}^{n+1,i} \quad (40)$$

where $0 \leq \theta_i \leq 1$.

In above $(\cdot)^{n,i}$ denote nodal values at the n th time step and the i th iteration. Note that $\mathbf{A}^{n+\theta_1,i-1} \equiv \mathbf{A}(\bar{\mathbf{u}}^{n+\theta_1,i-1})$ etc. Also $(\cdot)^{n+\theta_i,0} \equiv (\cdot)^n$ for the computations in step 1 at the onset of the iterations.

Steps 1, 3 and 4 can be solved explicitly by choosing a *lumped (diagonal) form* of matrices \mathbf{M} and $\hat{\mathbf{M}}$. In this manner the main computational cost is the solution of step 2 involving the inverse of a Laplacian matrix. This can be solved very effectively using an iterative method such as the conjugate gradient method or similar.

For $\theta_i \neq 0$ the iterative process is unavoidable. The iterations follow until convergence is reached in an adequate error norm in terms of the velocity and pressure variables, or the residuals r_{m_i} and r_d . Indeed some of the θ_i 's in Eqs.(37)–(40) can be made equal to zero. Note that for $\theta_2 = 0$ the algorithm is unconditionally unstable. A particularly interesting and simple semi-implicit form is obtained by making $\theta_1 = \theta_3 = \theta_4 = 0$. Now all steps can be solved explicitly with exception of

Step 2 for the pressure, which still requires the solution of a simultaneous system of equations.

Convergence of this solution scheme is however difficult for some problems. An enhanced version of the algorithm can be obtained by simply adding the term $\hat{\mathbf{L}}(\bar{\mathbf{p}}^{n+1,i} - \bar{\mathbf{p}}^{n+1,i-1})$ where $\hat{L}_{ij} = \Delta t \int_{\Omega^e} \nabla^T N_i \nabla N_j d\Omega$ to the equation for the computation of the pressure in the second step. The new term acts as a preconditioner of the pressure equation given now by

$$\bar{\mathbf{p}}^{n+1,i} = -[\mathbf{L} + \hat{\mathbf{L}}]^{-1}[\mathbf{G}^T \bar{\mathbf{u}}^{n+1,i} + \hat{\mathbf{L}} \bar{\mathbf{p}}^{n+1,i-1} + \mathbf{Q} \bar{\boldsymbol{\pi}}^{n+\theta_4,i-1}] \quad (41)$$

Note that the added term vanishes for the converged solution (i.e. when $\bar{\mathbf{p}}^{n+1,i} = \bar{\mathbf{p}}^{n+1,i-1}$).

An alternative to above algorithm is to use the fractional step method described in the nex section.

4.2 Fractional step method

An alternative algorithm can be obtained by splitting the pressure from the momentum equations as follow

$$\bar{\mathbf{u}}^* = \bar{\mathbf{u}}^n - \Delta t \mathbf{M}^{-1}[(\mathbf{A}^{n+\theta_1} + \mathbf{K} + \hat{\mathbf{K}}^{n+\theta_1})\bar{\mathbf{u}}^{n+\theta_1} - \alpha \mathbf{G} \bar{\mathbf{p}}^n + \mathbf{C} \bar{\mathbf{c}}^{n+\theta_3} - \mathbf{f}^{n+1}] \quad (42a)$$

$$\bar{\mathbf{u}}^{n+1} = \bar{\mathbf{u}}^* + \Delta t \mathbf{M}^{-1} \mathbf{G} \delta \bar{\mathbf{p}} \quad (42b)$$

In Eq.(42a) α is a variable taking values equal to zero or one. For $\alpha = 0$, $\delta p \equiv p^{n+1}$ and for $\alpha = 1$, $\delta p = \Delta p$. Note that in both cases the sum of Eqs.(42a) and (42b) gives the time discretization of the momentum equations with the pressures computed at t^{n+1} . The value of $\bar{\mathbf{u}}^{n+1}$ from Eq.(42b) is substituted now into Eq.(34b) to give

$$\mathbf{G}^T \bar{\mathbf{u}}^* + \Delta t \mathbf{G}^T \mathbf{M}^{-1} \mathbf{G} \delta \bar{\mathbf{p}} + \mathbf{L} \bar{\mathbf{p}}^{n+1} + \mathbf{Q} \bar{\boldsymbol{\pi}}^{n+\theta_4} = 0 \quad (43a)$$

The product $\mathbf{G}^T \mathbf{M}^{-1} \mathbf{G}$ can be approximated by a laplacian matrix, i.e.

$$\mathbf{G}^T \mathbf{M}^{-1} \mathbf{G} = \hat{\mathbf{L}} \quad \text{with} \quad \hat{\mathbf{L}} \simeq \int_{\Omega^e} \frac{1}{\rho} \nabla^T N_i \nabla N_j d\Omega \quad (43b)$$

A semi-implicit algorithm can thus be derived as follows.

Step 1 Compute $\bar{\mathbf{u}}^*$ explicitly from Eq.(41a) with $\mathbf{M} = \mathbf{M}_d$ where subscript d denotes hereonwards a diagonal matrix.

Step 2 Compute $\delta \bar{\mathbf{p}}$ from Eq.(43a) as

$$\delta \bar{\mathbf{p}} = -(\mathbf{L} + \Delta t \hat{\mathbf{L}})^{-1}[\mathbf{G}^T \bar{\mathbf{u}}^* + \mathbf{Q} \bar{\boldsymbol{\pi}}^{n+\theta_4} + \alpha \mathbf{L} \bar{\mathbf{p}}^n] \quad (44)$$

Step 3 Compute $\bar{\mathbf{u}}^{n+1}$ explicitly from Eq.(42b) with $\mathbf{M} = \mathbf{M}_d$

Step 4 Compute $\bar{\mathbf{c}}^{n+1}$ explicitly from Eq.(39) with

$$\bar{\mathbf{c}}^{n+1} = -\mathbf{M}_d^{-1} \hat{\mathbf{C}}^{n+1} \bar{\mathbf{u}}^{n+1} \quad (45)$$

Step 5 Compute $\bar{\boldsymbol{\pi}}^{n+1}$ explicitly from Eq.(40) as

$$\bar{\boldsymbol{\pi}}^{n+1} = -\hat{\mathbf{M}}_d^{-1} \mathbf{Q}^T \bar{\mathbf{p}}^{n+1} \quad (46)$$

This algorithm has an additional step than the iterative scheme of Section 4.1. The advantage is that now Steps 1 and 2 can be fully linearized by choosing $\theta_1 = \theta_3 = \theta_4 = 0$. Also the equation for the pressure variables in Step 2 has improved stabilization properties due to the additional laplacian matrix $\hat{\mathbf{L}}$.

The boundary conditions are applied as follow. No condition is applied in the computation of the fractional velocities \mathbf{u}^* in Eq.(44). The prescribed velocities at the boundary are applied when solving for $\bar{\mathbf{u}}^{n+1}$ in the step 3. The prescribed pressures at the boundary are imposed by making zero the pressure increments at the relevant boundary nodes and making $\bar{\mathbf{p}}^n$ equal to the prescribed pressure values.

4.3 Stokes flow

The formulation for a Stokes flow can be readily obtained simply by neglecting the convective terms in the general Navier-Stokes formulation. This also implies neglecting the convective stabilization terms in the momentum equations and, consequently, the convective projection variables are not larger necessary. Also the intrinsic time parameters τ_i take now the simpler form (see Eq.(22)):

$$\tau_i = \frac{3h_i^2}{8\mu} \quad (47)$$

The resulting discretized system of equations can be written as (see Eqs.(34))

$$\begin{aligned} \mathbf{M}\dot{\bar{\mathbf{u}}} + \mathbf{K}\bar{\mathbf{u}} - \mathbf{G}\bar{\mathbf{p}} &= \mathbf{f} \\ \mathbf{G}^T\bar{\mathbf{u}} + \mathbf{L}\bar{\mathbf{p}} + \mathbf{Q}\bar{\boldsymbol{\pi}} &= \mathbf{0} \\ \mathbf{Q}^T\bar{\mathbf{p}} + \hat{\mathbf{M}}\bar{\boldsymbol{\pi}} &= \mathbf{0} \end{aligned} \quad (48)$$

The iterative algorithm of Section 4.1 can now be implemented. Convergence is now faster due to the absence of the non linear convective terms in the momentum equation.

The steady-state form of Eqs.(48) can be expressed in matrix form as

$$\begin{bmatrix} \mathbf{K} & -\mathbf{G} & \mathbf{0} \\ -\mathbf{G}^T & -\mathbf{L} & -\mathbf{Q} \\ \mathbf{0} & -\mathbf{Q}^T & -\hat{\mathbf{M}} \end{bmatrix} \begin{Bmatrix} \bar{\mathbf{u}} \\ \bar{\mathbf{p}} \\ \bar{\boldsymbol{\pi}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \quad (49)$$

The system is symmetric and always positive definite and therefore leads to a non singular solution. We note that this property holds for *any interpolation function* chosen for $\bar{\mathbf{u}}$, $\bar{\mathbf{p}}$ and $\bar{\boldsymbol{\pi}}$, therefore overcoming the Babuška-Brezzi (BB) restrictions [1].

A reduced velocity-pressure formulation can be obtained by eliminating the pressure gradient projection variables $\bar{\boldsymbol{\pi}}$ from the last equation to give

$$\begin{bmatrix} \mathbf{K} & -\mathbf{G} \\ -\mathbf{G}^T & -(\mathbf{L} - \mathbf{Q}\hat{\mathbf{M}}^{-1}\mathbf{Q}^T) \end{bmatrix} \begin{Bmatrix} \bar{\mathbf{u}} \\ \bar{\mathbf{p}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix} \quad (50)$$

The reduction process is simplified by using a diagonal form of matrix $\hat{\mathbf{M}}$. Obviously above reduction is also applicable to the transient case.

5 FLUID-STRUCTURE INTERACTION. MESH UPDATING. ALE FORMULATION

5.1 General coupled solution scheme

The two algorithms of previous section can be readily extended for fluid-structure interaction analysis. The solution process in both cases includes the two additional steps.

Step A1. Solve for the movement of the structure due to the fluid flow forces

This implies solving the dynamic equations of motion for the structure written as

$$\mathbf{M}_s \ddot{\mathbf{d}} + \mathbf{K}_s \mathbf{d} = \mathbf{f}_{\text{ext}} \quad (51)$$

where \mathbf{d} and $\ddot{\mathbf{d}}$ are respectively the displacement and acceleration vectors of the nodes discretizing the structure, \mathbf{M}_s and \mathbf{K}_s are the mass and stiffness matrices of the structure and \mathbf{f}_{ext} is the vector of external nodal forces accounting for the fluid flow forces induced by the pressure and the viscous stresses. Clearly the main driving forces for the motion of the structure is the fluid pressure which acts in the form of a surface traction on the structure. Indeed Eq.(51) can be augmented with an appropriate damping term. The form of all the relevant matrices and vectors can be found in standard books on FEM for structural analysis [1].

Solution of Eq.(51) in time can be performed using implicit or fully explicit time integration algorithms. In both cases the values of the nodal displacement, velocities and accelerations at t^{n+1} are found.

Step A2. Compute the new position of the mesh nodes

Movement of a structure in a fluid originates a distortion in the mesh defining the control volume where the fluid equations are solved. Clearly a new mesh can be regenerated at each time step and this option is discussed in a later section dealing with lagrangian flows. A cheaper alternative is to update the position of the mesh nodes once the iterative process for the fluid and solid variables has converged. A simple algorithm for updating the mesh nodes is described in the next section.

5.2 A simple algorithm for updating the mesh nodes

Different techniques have been proposed for dealing with mesh updating in fluid-structure interaction problems. The general aim of all methods is to prevent element distortion during mesh deformation [42–44].

Chiandussi, Bugada and Oñate [45] have proposed a simple method for the movement of mesh nodes ensuring minimum element distortion. The method is based on the iterative solution of a fictitious linear elastic problem on the mesh domain. In order to minimize the mesh deformation the “elastic” properties of each mesh element are appropriately selected so that elements suffering greater movements are stiffer. The basis of the method is given below.

Let us consider an elastic domain with arbitrary homogeneous isotropic elastic properties characterized by the Young modulus \bar{E} and the Poisson coefficient ν . Once a discretized finite element problem has been solved using, for instance, standard C^0 linear triangles (in 2D) or linear tetrahedra (in 3D), the principal stresses

${}^1\sigma_i$ at the center of each element can be obtained. For 3D problems

$${}^1\sigma_i = \frac{\bar{E}(1-\nu)}{(1+\nu)(1-2\nu)} \left[\varepsilon_i + \frac{\nu}{1-\nu}(\varepsilon_j + \varepsilon_k) \right] \quad i, j, k = 1, 2, 3 \quad i \neq j \neq k \quad (52)$$

where ε_i are the principal strains. As the values of \bar{E} and ν are arbitrary it is useful to select $\nu = 0$. Eq.(32) simplifies in this case to $\sigma_i = \bar{E}\varepsilon_i$.

Let us assume now that a uniform strain field $\varepsilon_i = \bar{\varepsilon}$ throughout the mesh is sought. The principal stresses are then given by

$${}^2\sigma_i = E\bar{\varepsilon} \quad i = 1, 2, 3 \quad (53)$$

where E is the unknown Young modulus for the element.

A number of criteria can be now used to find the value of E . An effective approach found in [45] is to make equal the element strain energy densities in both analysis. Thus (for $\nu = 0$)

$$U_1 = \frac{{}^1\sigma_i\varepsilon_i}{2} = \frac{\bar{E}[(\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2)]}{2} \quad (54a)$$

$$U_2 = \frac{{}^2\sigma_i\varepsilon_i}{2} = \frac{3E\bar{\varepsilon}^2}{2} \quad (54b)$$

Equating eqs.(54a) and (54b) gives the sought Young modulus E as

$$E = \frac{\bar{E}}{3\bar{\varepsilon}^2}(\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2) \quad (55)$$

Note that the element Young modulus is proportional to the element deformation as desired. Also recall that both \bar{E} and $\bar{\varepsilon}$ are arbitrary constants for all elements in the mesh.

The solution process includes the following two steps.

Step 1. Consider the finite element mesh as a linear elastic solid with homogeneous material properties characterized by \bar{E} and arbitrary Young modulus \bar{E} and the Poisson ratio $\nu = 0$. Solve the corresponding elastic problem with imposed displacements at the mesh boundary.

Step 2. Compute the principal strains and the values of the new Young modulus in each element using Eq.(55) for a given value of $\bar{\varepsilon}$. Repeat the finite element solution of the linear elastic problem with prescribed boundary displacements using the new values of E for each element.

The movement of the mesh nodes obtained in the second step ensures a quasi uniform mesh distortion. Further details on this method including other alternatives for evaluating the Young modulus E can be found in [45].

The previous algorithm for movement of mesh nodes is able to treat the movement of the mesh due to changes in position of fully submerged and semi-submerged bodies. Note however that if the floating body intersects the free surface, the changes in the analysis domain geometry can be very important. From one time step to other emersion or immersion of significant parts of the body can occur.

A solution to this problem is to remesh the analysis domain. However, for most problems, a mapping of the moving surfaces linked to the mesh updating algorithm described above can avoid remeshing [38,46–49].

5.3 ALE formulation

The movement of the mesh defining the fluid domain requires accounting for the relative motion of the fluid particles with respect to the moving mesh. This can be dealt with by an arbitrary lagrangian-eulerian (ALE) formulation. This basically implies redefining the convective transport term in the momentum equation as

$$v_j \frac{\partial u_i}{\partial x_j} \quad \text{with } v_j = u_j - u_j^m \quad (56)$$

where v_j is the relative velocity between the moving mesh and the fluid point and u_j^m is the velocity of the mesh nodes. This velocity can be simply computed dividing by Δt the displacement vector of the nodes in the mesh obtained from the mesh updating algorithm previously described.

6 FREE SURFACE WAVE EFFECTS

Many problems of practical importance involve a free surface in the fluid. In general the position of such a free surface is unknown and has to be determined. Typical problems of this kind are water flow around ships, flow under and over water control structures, mould filling processes, etc.

On the free surface Γ_β we must ensure at all times that (1) the pressure (which approximate the normal traction) equals the atmospheric pressure p_a and the tangential tractions are zero (unless specific otherwise) and (2) that the material particles of the fluid belong to the free surface.

Condition (1) is simply fulfilled by imposing $p = p_a$ on Γ_β during the solution for the nodal pressures.

The free surface condition (2) can be written in the FIC formulation (neglecting time stabilization effects) as [46–49]

$$r_\beta - \frac{1}{2} h_{\beta_j} \frac{\partial r_\beta}{\partial x_j} = 0 \quad j = 1, 2 \quad (57)$$

where

$$r_\beta := \frac{\partial \beta}{\partial t} + v_i \frac{\partial \beta}{\partial x_i} - v_3 \quad i = 1, 2 \quad (58)$$

where β is the wave elevation (measured with respect to a reference surface of height β_{ref}) and v_i is the relative velocity defined in Eq.(56). The underlined term in Eq.(57) introduces the necessary stabilization for the solution of the highly convective (and non linear) equation defining the evolution of the wave elevation. Note that neglecting the stabilization term, the steady state form of Eq.(57) $\left(\frac{\partial \beta}{\partial t} = 0\right)$ simply states that the fluid particles move in the tangential direction to the free surface (in 2D: $\frac{\partial \beta}{\partial x} = \frac{u_2}{u_1} = \text{tg } \delta$ where δ is the angle which the velocity vector forms with the horizontal axis).

The solution in time of Eq.(57) can be expressed in terms of the nodal velocities computed from the flow solution, as

$$\beta^{n+1} = \beta^n - \Delta t \left[v_i^{n+1,i} \frac{\partial \beta^n}{\partial x_i} - v_3^{n+1,i} - \frac{h_{\beta_j}}{2} \frac{\partial r_\beta^n}{\partial x_j} \right] \quad (59)$$

Eq.(59) can now be discretized in space using the standard Galerkin method and solved *explicitly* to give the nodal wave heights at t^{n+1} [46–49]. This solution step should precede the computation of the structure motion in the case of a fluid-structure interaction problem. Typically the general algorithm will be as follows:

1. Solve for the nodal velocities $\bar{\mathbf{u}}^{n+1}$ and the pressures $\bar{\mathbf{p}}^{n+1}$ in the fluid domain using any of the algorithms of Section 4. When solving for the pressure variables impose $p^{n+1} = p_a$ at the free surface Γ_β .
2. Solve for the free surface elevation β^{n+1} via Eq.(59).
3. Compute the movement of the fully or semi-submerged or floating structure by solving the dynamic equations of motion of the structure (Eq.(51)).
4. Compute the new position of the mesh nodes in the fluid domain at time t^{n+1} . Alternatively, regenerate a new mesh.

The mesh updating proces can also include the free surface nodes, although this is not strictly necessary. An *hydrostatic adjustment* can be implemented once the new free surface elevation is computed by simple imposing the pressure at the nodes on the reference surface as

$$p^{n+1} = p_a + \rho|g|\Delta\beta \quad \text{with } \Delta\beta = \beta^{n+1} - \beta_{\text{ref}} \quad (60)$$

where g is the gravity constant. Eq.(60) allows to take into account the changes in the free surface without the need of updating the reference surface nodes. A higher accuracy in the solution of the flow problem can be obtained by updating the reference surface nodes after a number of time steps.

7 LAGRANGIAN FLOW FORMULATION

The Lagrangian formulation is an effective (and relatively simple) procedure for modelling the flow of fluid particles undergoing severe distorsions such as water jets, high amplitude waves, water splashing, breaking waves, filling of cavities, etc. Indeed the lagrangian formulation seems to be an excellent procedure for treating fluid-structure interaction problems where the structure has large displacements. An obvious “a priori” advantage of the lagrangian formulation is that both the structure and the fluid motions are defined in the same frame of reference.

The lagrangian fluid flow equations can be simply obtained by noting that the velocity of the mesh nodes and that of the fluid particles are the same. Hence the relative velocity v_i is zero in Eq.(56) and the convective terms vanish in the momentum equations, while the rest of the fluid flow equations defined in Section 3 remain unchanged.

The FEM algorithms for solving the lagrangian flow equations is very similar to those for the eulerian or ALE description presented earlier and only the main differences will be given here. For preciseness we will focus in the semi-implicit fractional step algorithm of Section 4.2 (for $\theta_1 = \theta_4 = 0$ and $\alpha = 1$) accounting also for fluid-structure interaction effects.

Step 1 Compute explicitly a predicted value of the velocities \mathbf{u}^* as

$$\bar{\mathbf{u}}^* = \bar{\mathbf{u}}^n - \Delta t \mathbf{M}_d^{-1} [\mathbf{K} \bar{\mathbf{u}}^n - \mathbf{G} \bar{\mathbf{p}}^n - \mathbf{f}^{n+1}] \quad (61)$$

Step 2 Compute $\delta\bar{\mathbf{p}}$ from Eq.(44).

Step 3 Compute explicitly $\bar{\mathbf{u}}^{n+1}$ from Eq.(42b) with $\mathbf{M} = \mathbf{M}_d$.

Step 4 Compute $\bar{\boldsymbol{\pi}}^{n+1}$ explicitly from Eq.(40).

Step 5 Solve for the motion of the structure by integrating Eq.(51).

Step 6 Update the mesh nodes in a lagrangian manner as

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \bar{\mathbf{u}}_i^{n+1} \Delta t \quad (62)$$

Step 7 Generate a new mesh. This can be effectively performed using the extended Delaunay Tessellation described in [50]. Indeed the mesh regeneration can take place after a prescribed number of time steps or when the nodal displacements induce significant distortions in some element shapes.

Details of the treatment of the boundary conditions in the lagrangian flow formulation can be found in [51,52].

8 TURBULENCE MODELLING

The detailed discussion on the treatment of turbulent effects in the flow equation falls outside the objective of this chapter as any of the existing turbulence model is applicable.

In the examples presented next we have chosen a turbulence model based on the Reynolds averaged Navier-Stokes equations where the deviatoric stresses are computed as sum of the standard viscous contributions and the so called Reynold stresses. Here we have chosen the Boussinesq assumption leading to a modification of the viscosity in the standard Navier-Stokes equations as sum of the “physical” viscosity μ and a turbulent viscosity μ_T .

One of the simplest and more effective choices for μ_T is the Smagorinski LES model giving

$$\mu_T = C_l h^e (2\varepsilon_{ij}\varepsilon_{ij})^{1/2} \quad (63)$$

where h^e is the element size and C is a constant ($C \simeq 0.01$).

Indeed other many options are possible such as the one and two equations turbulence models (i.e. the k model and the $k - \varepsilon$ and $k - w$ models) and the algebraic stress models and the reader is referred to specialized books on this matter [53].

9 COMPUTATION OF THE CHARACTERISTIC LENGTHS

The evaluation of the stabilization parameters is one of the crucial issues in stabilized methods. Most of existing methods use expressions which are direct extensions of the values obtained for the simplest 1D case. It is also usual to accept the so called SUPG assumption, i.e. to admit that vector \mathbf{h} has the direction of the velocity field [32,37]. This unnecessary restriction leads to instabilities when sharp layers transversal to the velocity direction are present. This deficiency is usually corrected by adding a shock capturing or crosswind stabilization term [54,55]. Indeed, in the FIC formulation the components of \mathbf{h} introduce the necessary stabilization along both the streamline and transversal directions to the flow.

Excellent results have been obtained in all problems solved using linear tetrahedra with the same value of the characteristic length vector defined by

$$\mathbf{h} = h_s \frac{\mathbf{u}}{u} + h_c \frac{\nabla u}{|\nabla u|} \quad (64)$$

where $u = |\mathbf{u}|$ and h_s and h_c are the “streamline” and “cross wind” contributions given by

$$h_s = \max(\mathbf{l}_j^T \mathbf{u})/u \quad (65)$$

$$h_c = \max(\mathbf{l}_j^T \nabla u)/|\nabla u| \quad , \quad j = 1, n_s \quad (66)$$

where \mathbf{l}_j are the vectors defining the element sides ($n_s = 6$ for tetrahedra).

As for the free surface equation the following value of the characteristic length vector \mathbf{h}_β has been taken

$$\mathbf{h}_\beta = \bar{h}_s \frac{\mathbf{u}}{u} + \bar{h}_c \frac{\nabla \beta}{|\nabla \beta|} \quad (67)$$

The streamline parameter \bar{h}_s has been obtained by Eq.(65) using the value of the velocity vector \mathbf{u} over the 3 node triangles discretizing the free surface and $n_s = 3$.

The cross wind parameter \bar{h}_c has been computed by

$$\bar{h}_c = \max[\mathbf{l}_j^T \nabla \beta] \frac{1}{|\nabla \beta|} \quad , \quad j = 1, 2, 3 \quad (68)$$

The cross-wind terms in eqs.(64) and (67) account for the effect of the gradient of the solution in the stabilization parameters. This is a standard assumption in most “shock-capturing” stabilization procedures [54,55].

A more consistent evaluation of \mathbf{h} based on a diminishing residual technique can be found in [37].

10 EXAMPLES

The examples chosen show the applicability of the Eulerian, ALE and lagrangian formulations presented to solve fluid flow problems. Linear tetrahedra and triangles have been used in the 3D and 2D analysis shown. The fractional step algorithm of Section 4.2 for $\theta_1 = \theta_3 = \theta_4 = 0$ and $\alpha = 1$ has been used in all cases. The first example is the standard square cavity problem solved in 3D using an Euler formulation ($\mu = 0$). The second example is the flow past a submerged NACA 0012 profile. Here the free surface equation is solved together with the flow equations.

The next four examples fall within the category of fluid-structure interaction problems. The first is the analysis of a sphere falling in a tube filled with liquid where the mesh updating procedure of Section 5.2 has been used. Then three of ship hydrodynamics problems are solved including the analysis of a Wigley hull, a scale model of a commercial ship and an American Cup racing sail boat. Numerical results are compared with experimental data in all cases.

The last series of examples show applications of the Lagrangian formulation to the simulation of the collapse of a water column, a semi-submerged rotating water mill and a solid cube falling into a water recipient.

10.1 Square cavity problem

The purpose of this example is to test the stabilized formulation presented in the solution of a standard benchmark problem [17]. Figure 2 shows the definition of the problem solved with an unstructured 3D mesh of 7395 linear tetrahedra for a Reynolds number value of 1.

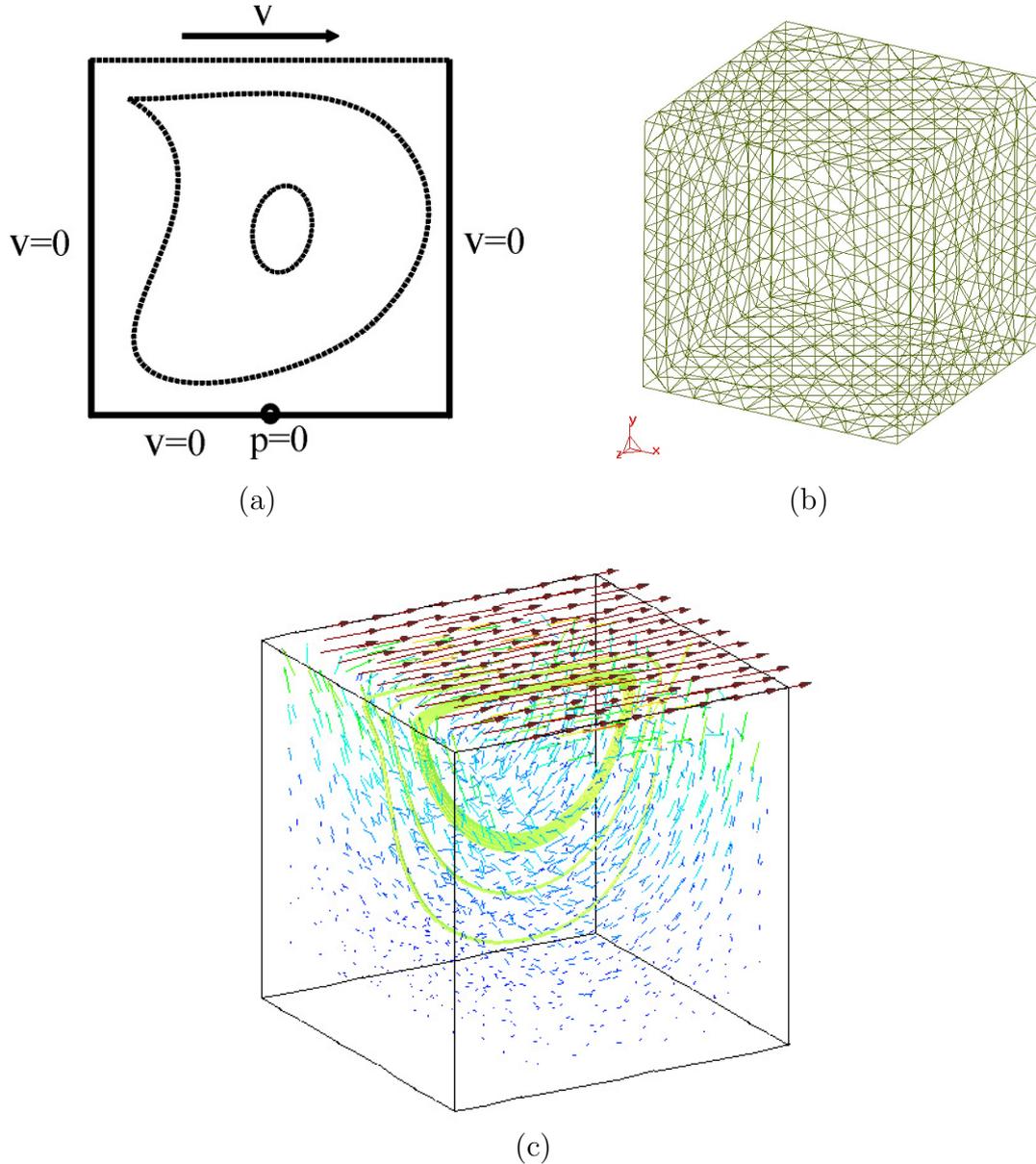


Figure 2: Square cavity problem. a) Problem definition. b) Unstructured mesh of 7395 linear tetrahedra. c) velocity field for $Re = 1$.

Results in Figure 3a,b are tabulated for the horizontal velocity along the vertical centerline of the mid-section and for vertical velocity and pressure along the horizontal centerline of the same section. Numerical results are fully stable and agree well with similar solutions reported [17]. The effect of the τ_i stabilization term in the pressure equation (see eq.(30)) is seen clearly in Figure 3c. The curves in this figure show the convergence towards steady state of the L_∞ norm of the nodal pressures with time. The curve listed as “standard” is obtained neglecting the τ_i stabilization

term, whereas the second curve shows the convergence when this term is taken into account. The difference between the two curves is noticeable as the error obtained with the fully stabilized solution is several orders of magnitude smaller than that obtained neglecting the τ_i term.

10.2 Submerged NACA 0012 profile

A 2D submerged NACA0012 profile at $\alpha = 5^\circ$ angle of attack is studied. This configuration was tested experimentally by Duncan [56] for high Reynolds numbers ($Re=400000$) and modelled numerically using the Euler equations by several authors [57–59]. The submerged depth of the airfoil is equal to the chord L . The Froude number for all the cases tested was set to $F = \frac{u}{\sqrt{gL}} = 0.5672$ where u is the incoming flow velocity at infinity.

The stationary free surface and the pressure distribution are shown in Figure 4. The non-dimensional wave heights compare well with the experimental results [56].

10.3 Sphere falling in a tube filled with water

The movement of a sphere falling by gravity in a cylindrical tube filled with water is studied. The relationship between the diameters of the sphere and the tube is 1:4. The Reynolds number for the stationary speed is 100. The mesh has 85765 elements with 13946 nodes (Figure 5).

Figures 5 and 6 show the mesh deformation and contours of the mesh deformation and of the velocity in the domain for different times, respectively. The evolution of the falling speed is shown in Figure 6c. Note the good agreement with the so called Stokes velocity computed by equaling the weight of the sphere with the resistance to the movement of the sphere expressed in terms of the velocity. Obviously, this value is slightly greater than the actual one as frictional effects are neglected.

A similar problem for a greater number of spheres has been solved by Johnson and Tezduyar [60].

10.4 Wigley hull

The next problem case considered here is the study of the hydrodynamics of the well known Wigley Hull.

The same configuration was tested experimentally in [61] and modelled numerically by several authors [58,59,62]. We use here an unstructured 3D finite element mesh of 65434 linear tetrahedra, with a reference surface of 7800 triangles, partially represented in Figure 7.

Figure 7 also shows the results of the viscous analysis of the Wigley model in three different cases for ($L = 6m, F = 0.316, \mu = 10^{-3}Kg/m.s$). In the first case the volume mesh was considered fixed, not allowing free surface nor ship movements. Secondly, the volume mesh was updated due to free surface movement, considering the model fixed. The third case corresponds to the analysis of a real free model including the mesh updating due to free surface evaluation and ship movement (sinkage and trim). A Smagorinsky turbulence model was used in the three cases.

Table 1 shows the obtained total resistance coefficient in the three cases studied compared with the experimental data.

In the study of the free model the numerical values of sinkage and trim were -0.1% and 0.035, respectively, while experiment gave -0.15% and 0.04.

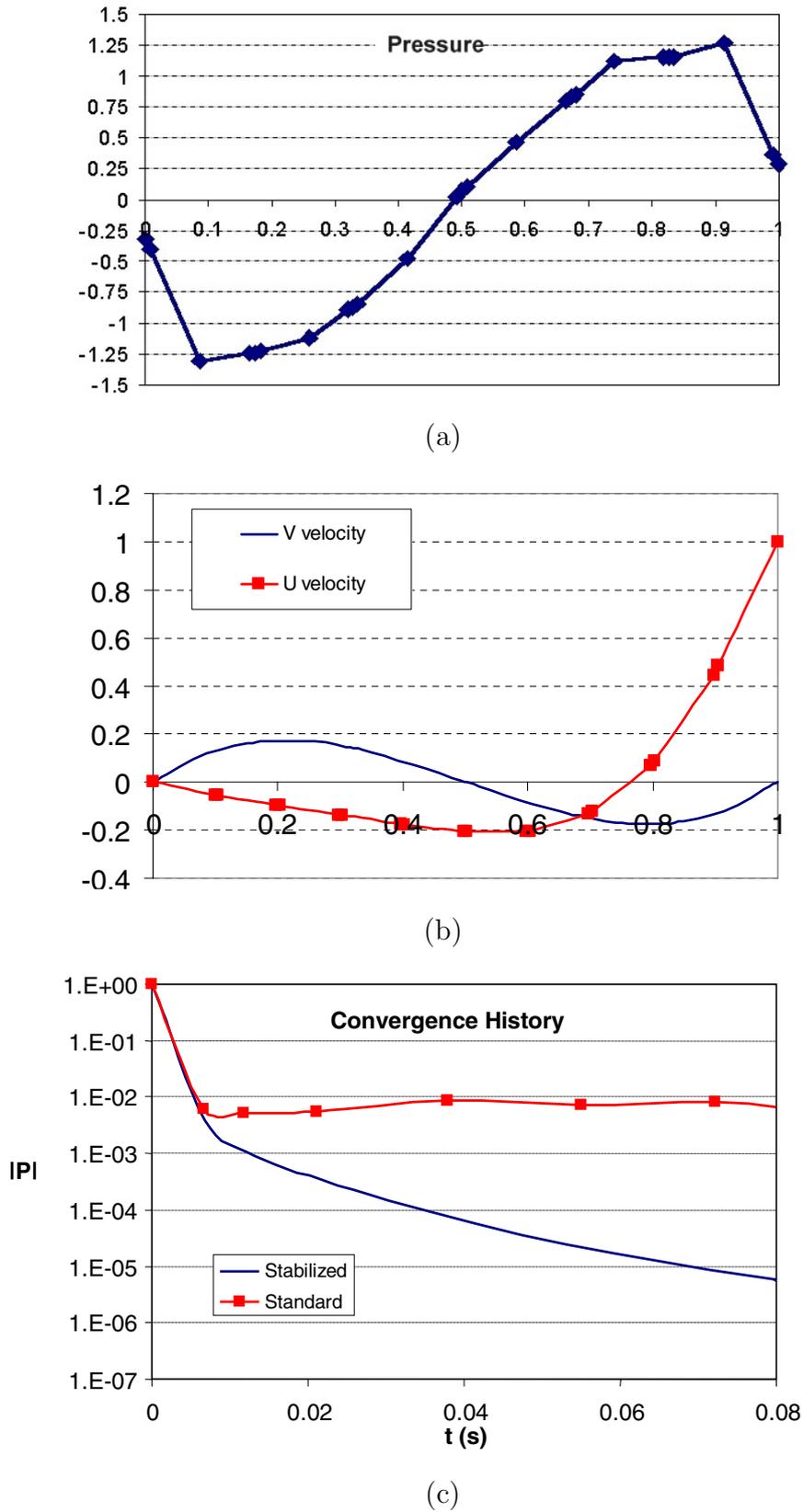


Figure 3: Square cavity. a) Distribution of pressure along horizontal centerline of mid-section. b) Distribution of velocity along horizontal centerline of mid-section. c) Convergence histories of the nodal pressure norm (L_∞) for the stabilised (accounting for τ_i) and the standard ($\tau_i = 0$) schemes.

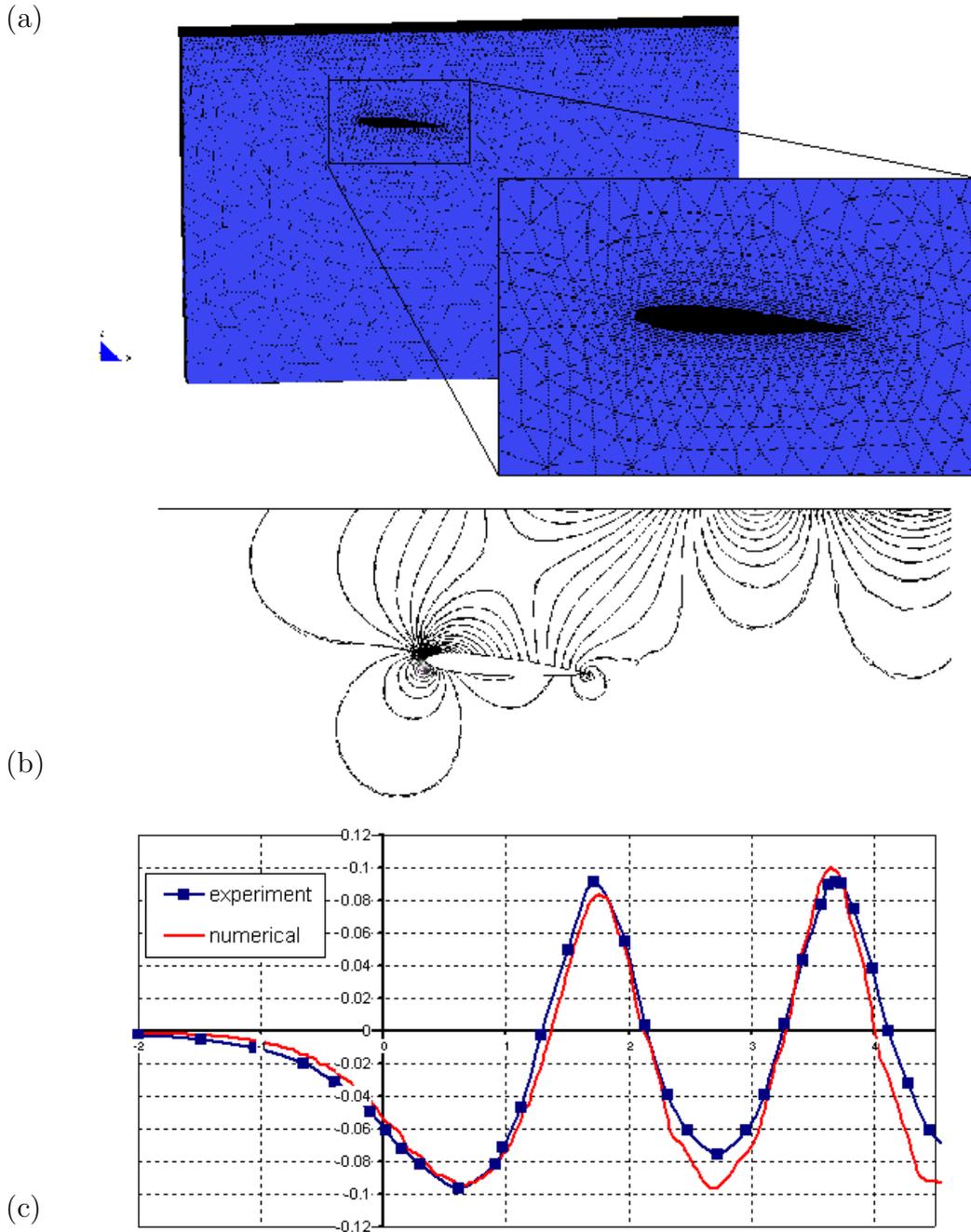


Figure 4: Submerged NACA0012 profile. a) Detail of the mesh of 70000 linear tetrahedra chosen. b) Pressure contours. c) Stationary wave profile.

Figure 7a shows the pressure distribution obtained near the Wigley hull for the free model. A number of streamlines have also been plotted in the figure. The obtained mesh deformation in this case is also presented in Figure 7b.

Comparisons of the obtained body wave profile with the experimental data for the free and fixed models are shown in Figure 7b. Significant differences are found close to stern in the case of the fixed model.

The free surface contours for the free ship motion are shown in Figure 7c.

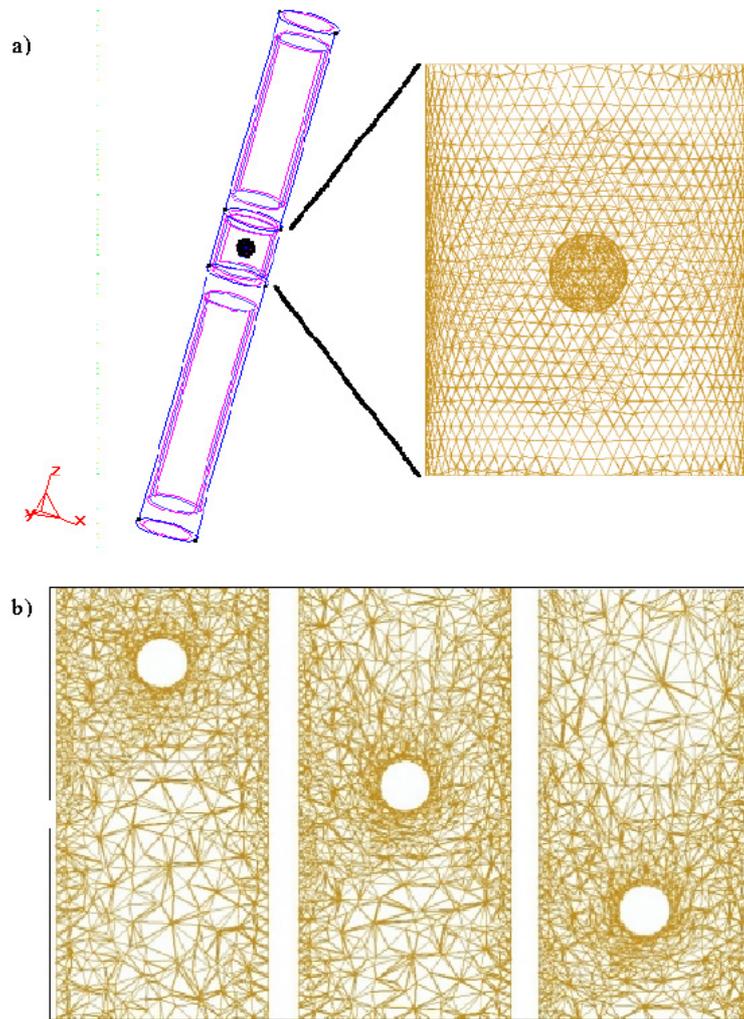


Figure 5: Sphere falling in a tube filled with liquid. a) Geometry definition and detail of the mesh of 85765 linear tetrahedra chosen. b) Mesh deformation during the falling of the sphere.

10.5 KVLCC2 hull model

The example is the analysis of the KVLCC2 benchmark model. Here a partially wetted transom stern is expected due to the low Froude number of the test. Figure 8 shows the NURBS geometry used obtained from the Hydrodynamic Performance Research team of Korea (KRISO). The obtained results are compared with the experimental data available in the KRISO database [63].

The smallest element size used was 0.001 m and the largest 0.50 m. The surface mesh chosen is shown in Figure 9. A total of 550.000 tetrahedra were used in the analysis. The transom stern flow model presented in the previous section was used.

Test 1.- Wave pattern calculation. The main characteristics of the analysis are listed below:

- Length: 5.52 m, Beam (at water plane): 0.82 m, Draught: 0.18 m, Wetted Surface: $8.08m^2$.
- Velocity: 1.05 m/seg, Froude Number: 0.142.
- Viscosity: $0.00126Kg/mseg$, Density: $1000Kg/m^3$, Reynolds number: 4.6310^6 .

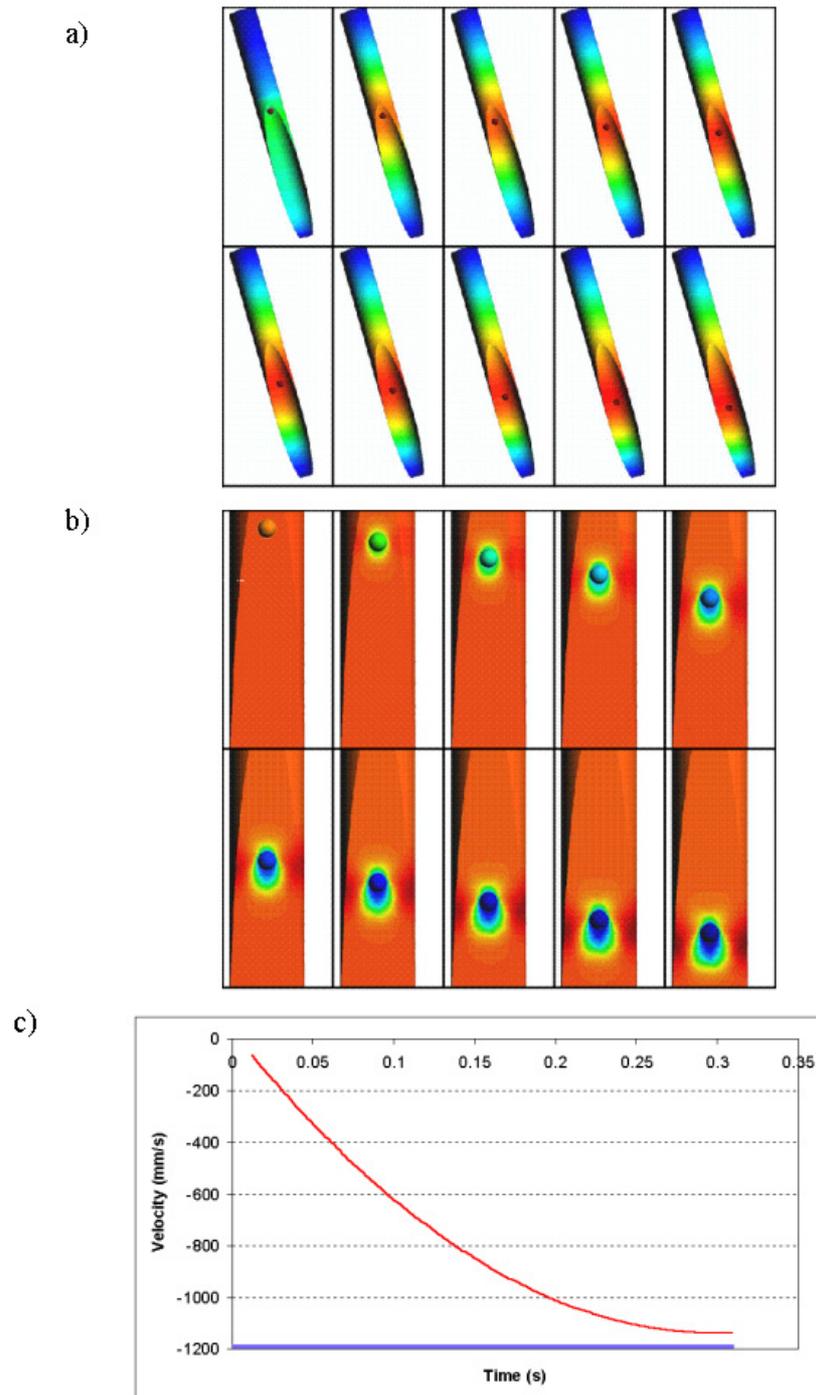
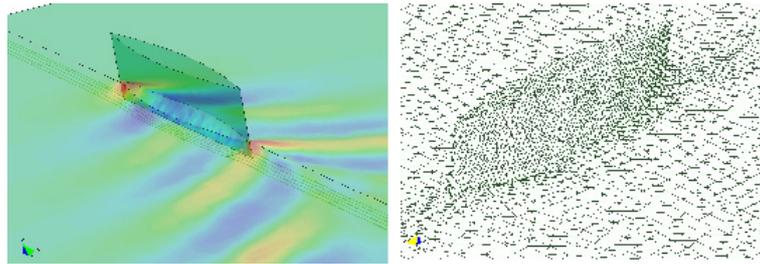
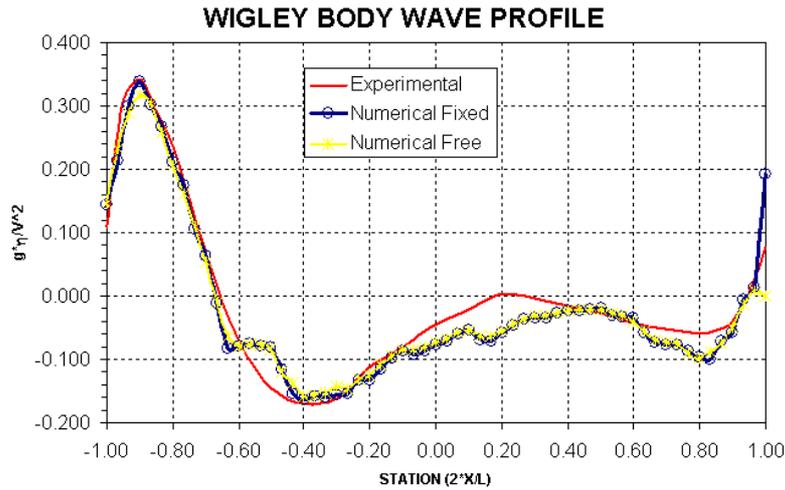


Figure 6: Sphere falling in a tube filled with liquid a) Evolution of contours of the mesh deformation. b) Evolution of contours of velocity module. c) Evolution of falling speed. Straight line indicates the theoretical Stokes speed (1.195 m/s).

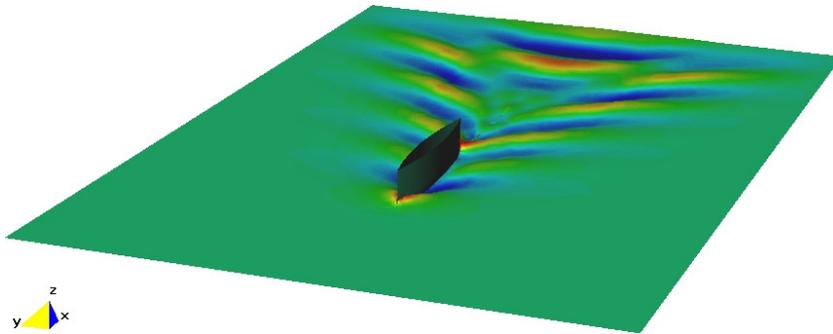
The turbulence model used in this case was the K model. Figures 10 and 11 show the wave profiles on the hull and in a cut at $y/L = 0.082$ obtained in Test 1, compared to the experimental data. The obtained results are quantitatively good close to the hull. A loss of accuracy is observed in the profiles away from the hull. This is probably due to the fact that the element sizes are not small enough in this area.



(a)



(b)



(c)

Figure 7: Wigley hull. a) Pressure distribution and mesh deformation of the wigley hull (free model). b) Numerical and experimental body wave profiles. c) Free surface contours for the truly free ship motion.

Test 2.- Wake analysis at different planes. Several turbulence models were used (*Smagorinsky*, K and $K - \epsilon$ model) in order to verify the quality of the results. Here, only the results from the $K - \epsilon$ model are shown. We note that the velocity maps obtained even for the simplest *Smagorinsky* model were qualitatively good, showing the accuracy of the fluid solver scheme used. The main characteristics of this analysis are listed below:

- Length: 2.76 m, Beam (at water plane): 0.41 m, Draught: 0.09 m, Wetted Surface: $2.02m^2$.
- Velocity: 25 m/seg.

	Experimental	Numerical
Test 1	$5.2 \cdot 10^{-3}$	$4.9 \cdot 10^{-3}$
Test 2	$5.2 \cdot 10^{-3}$	$5.3 \cdot 10^{-3}$
Test 3	$4.9 \cdot 10^{-3}$	$5.1 \cdot 10^{-3}$

Table 1: Wigley Hull. Total resistance coefficient

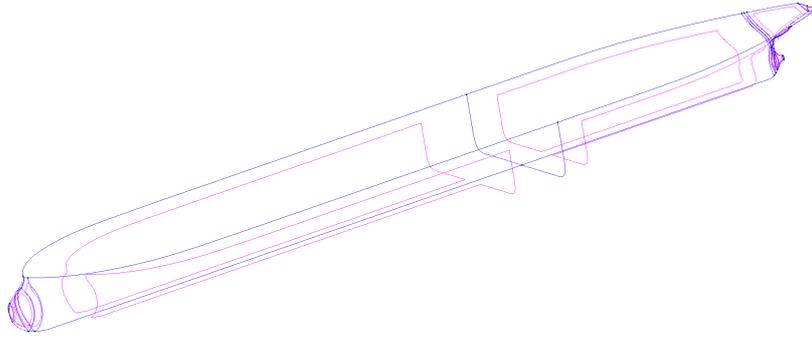


Figure 8: KVLCC2 model. Geometrical definition based on NURBS surfaces.

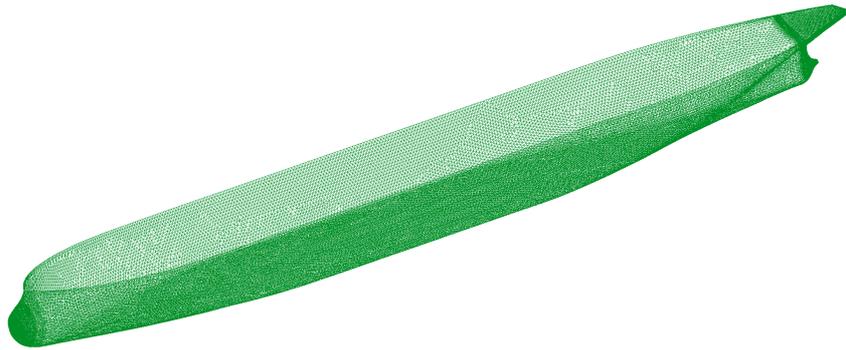


Figure 9: KVLCC2 model. Surface mesh used in the analysis.

- Viscosity: $3.0510^{-5} Kg/mseg$, Density: $1.01Kg/m^3$, Reynolds number: 4.6310^6 .

Figures 12–13 present results corresponding to the test 2. Figure 12 shows the contours of the axial (X) component of the velocity on a plane at 2.71 m from the orthogonal aft. Figure 13 shows the maps of the kinetic energy on this plane. Experimental results are shown for comparison in all cases. Further results for this problem can be found in [48].

10.6 American Cup BRAVO ESPAÑA Model

The next example is the analysis of the Spanish American Cup racing sail boat *Bravo España*. The finite element mesh used is shown in Figure 14. The results presented in Figures 14–17 correspond to the analysis of a non symmetrical case including appendages. Good comparison between the experimental data and the numerical results was again obtained.

Other results of the hydrodynamic analysis of American Cup racing boats carried out with the FEM formulation presented in the paper can be seen in [64].

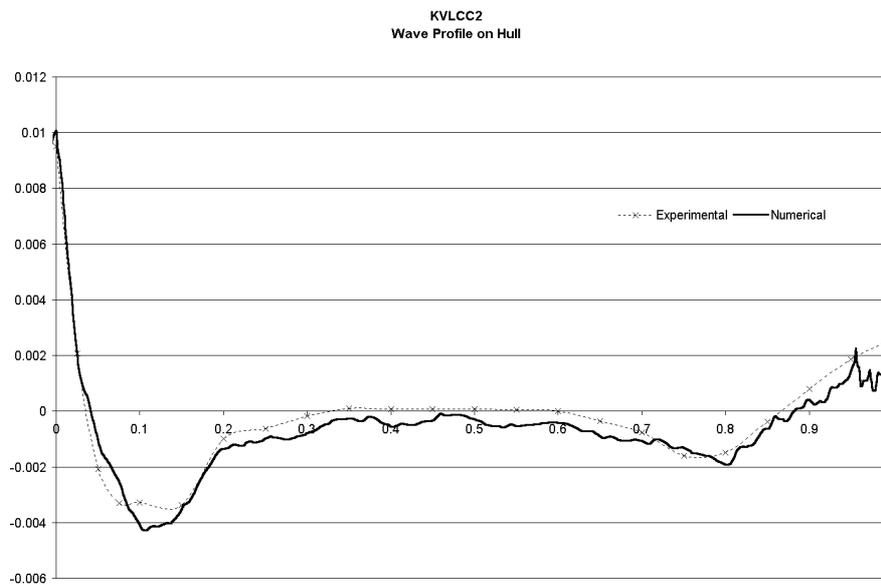


Figure 10: KVLCC2 model. Wave profile on the hull compared to experimental data. Thick line shows numerical results

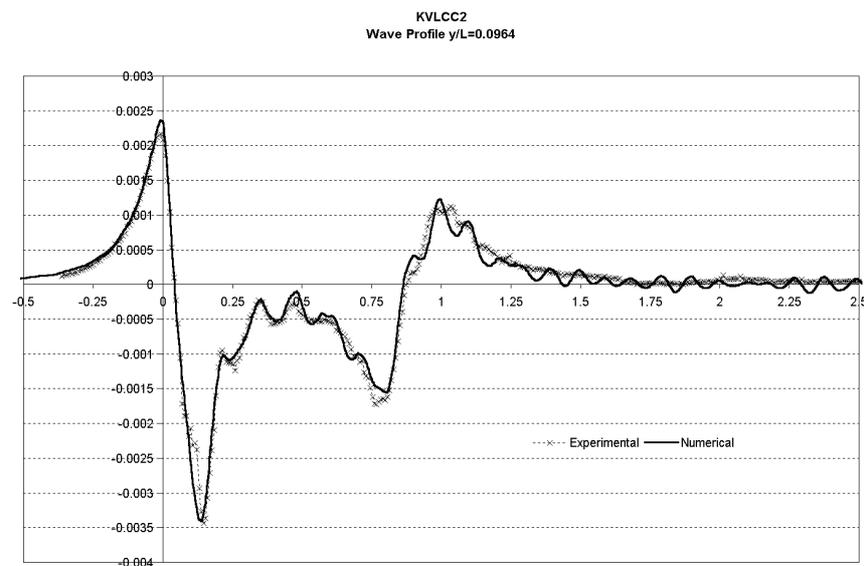


Figure 11: KVLCC2 model. Wave profile on a cut at $y/L=0.0964$ compared to experimental data [26]. Thick line shows numerical results

10.7 Lagrangian flow

The first problem solved with the lagrangian formulation is the study of the collapse of a water column. This problem was solved by Koshizu and Oka [65] both experimentally and numerically. It has become a classical example to test the validation of the lagrangian formulation in fluid flows. The water is initially located on the left supported by a removable board. The collapse starts at time $t = 0$, when the removable board is removed. Viscosity and surface tension are neglected. Figure 18 shows the point positions at different time steps. The dark points represent the free-surface detected with an alpha-shape algorithm [51,52].

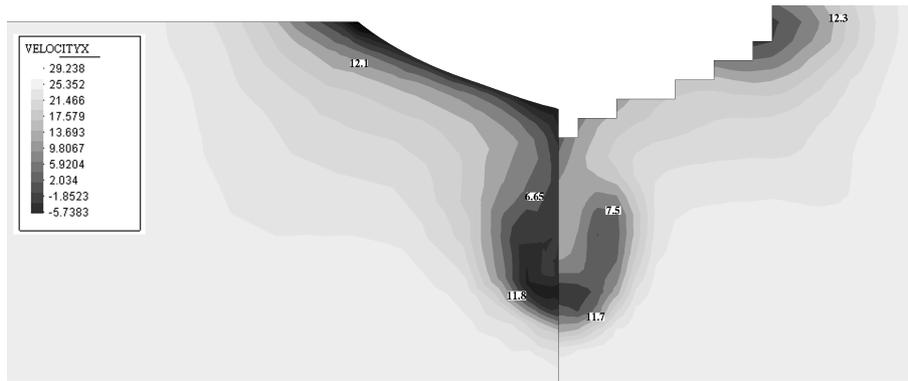


Figure 12: KVLCC2 model. Map of the X component of the velocity on a plane at 2.71 m from the orthogonal aft. Experimental results shown in the right figure.

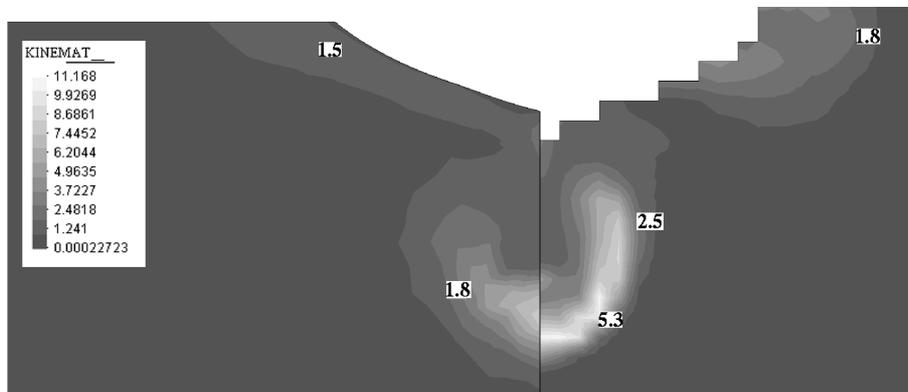


Figure 13: KVLCC2 model. Map of the eddy kinetic energy (K) on a plane at 2.71 m from the orthogonal aft. Experimental data shown in the right figure.

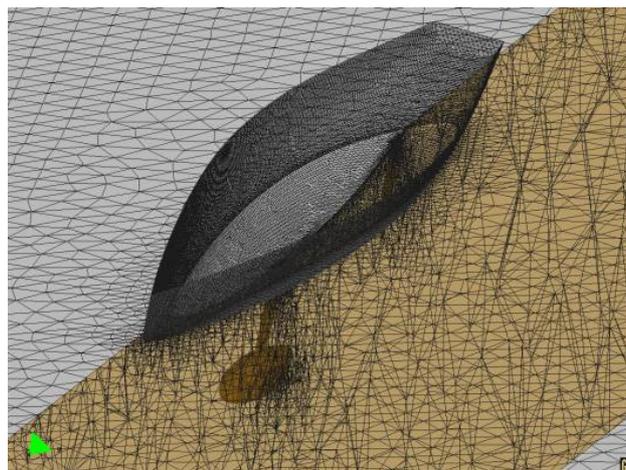


Figure 14: *Bravo España* sail racing boat. Mesh used in the analysis.

The internal points are gray and the fixed points are black.

The water is running on the bottom wall until, near 0.3 sec, it impinges on the right vertical wall. Breaking waves appear at 0.6 sec. Around 1 sec. the water

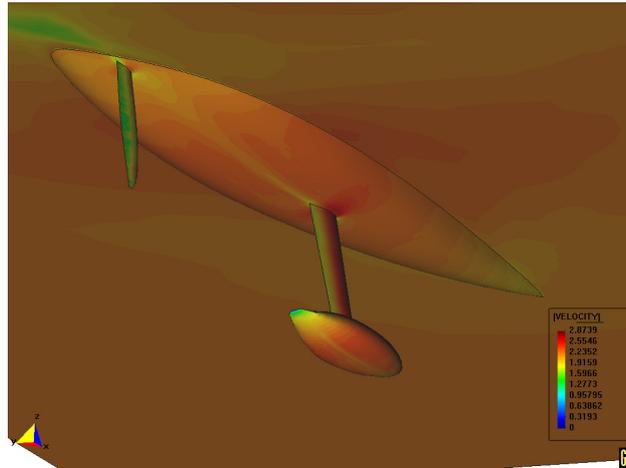


Figure 15: *Bravo España*. Velocity contours.

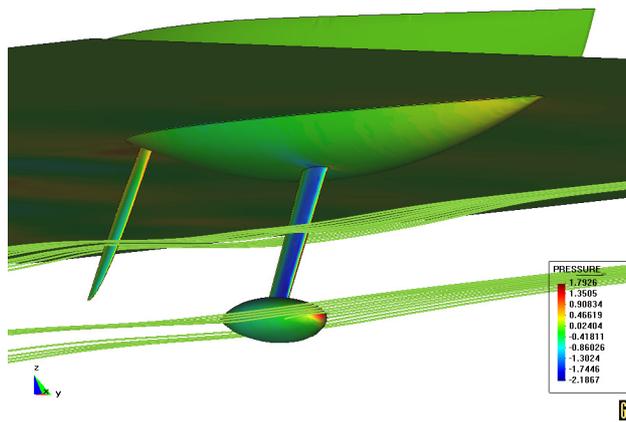


Figure 16: *Bravo España*. Streamlines.

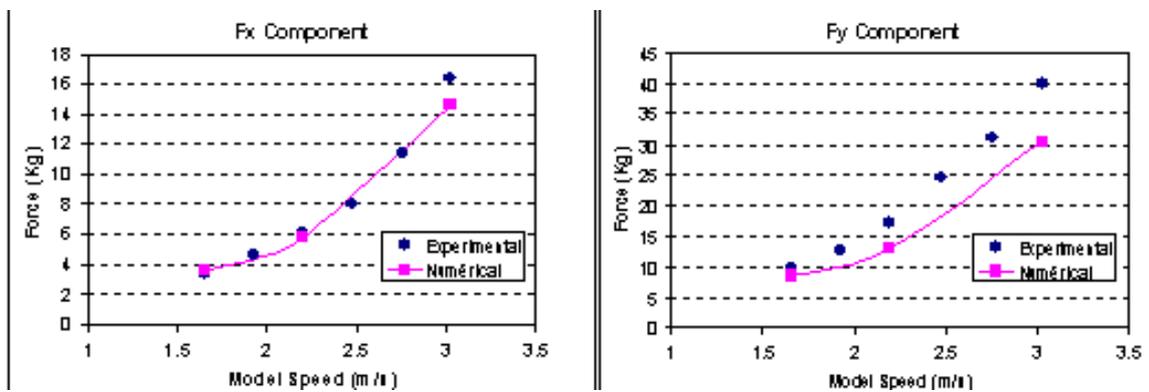


Figure 17: *Bravo España*. Resistance test. Comparison of numerical results with experimental data.

reaches the left wall. Agreement with the experimental results of [65] both in the shape of the free surface as well as in the time evolution are excellent.

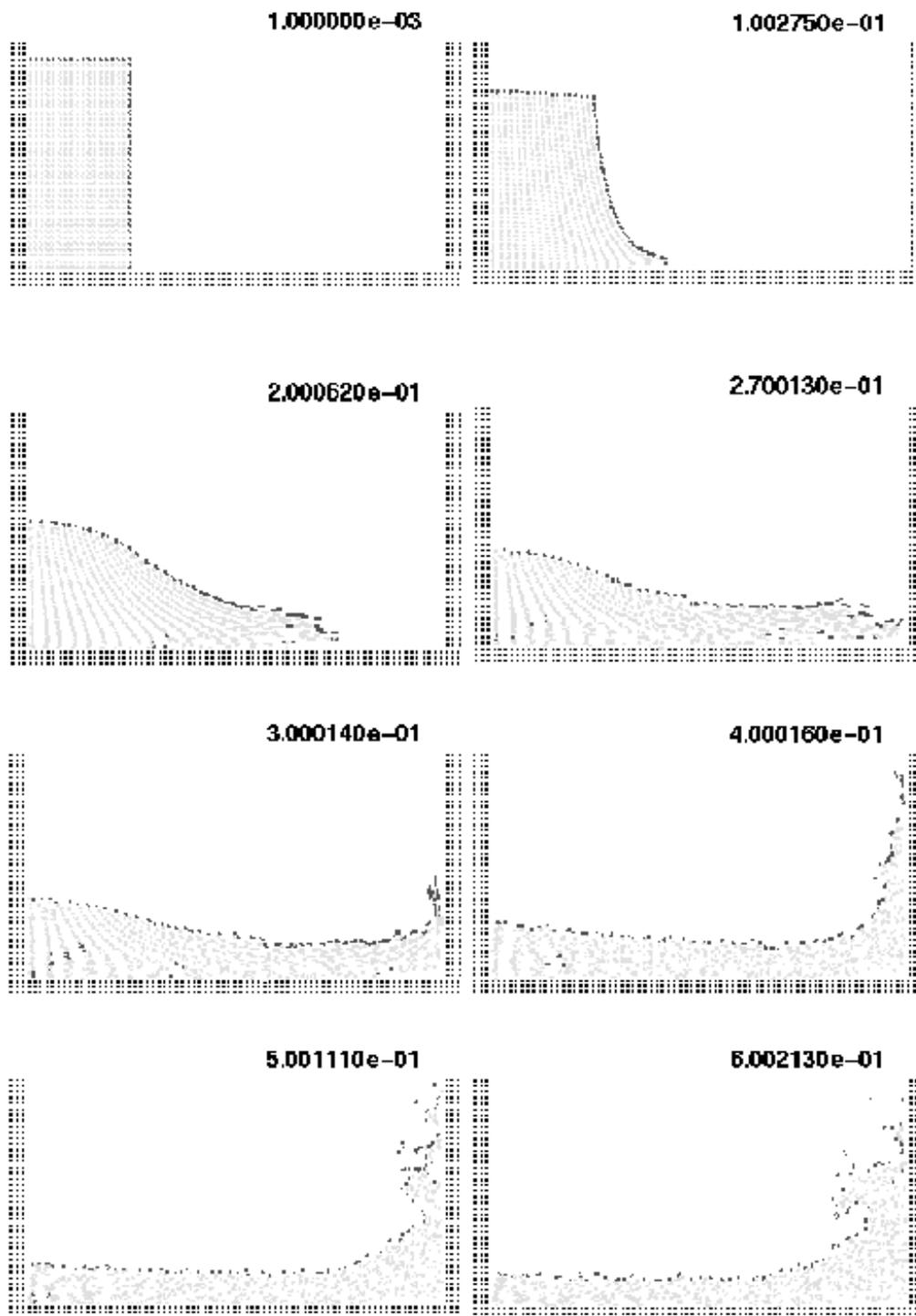


Figure 18: Water column collapse at different time steps.

The second example is the analysis of a rotating water mill semi submerged in water. A schematic representation of a water mill is presented in Figure 19. The blades of the mill have an imposed rotating velocity, while the water is initially in a stationary and flat position. Fluid structure interactions with free-surfaces and water fragmentation are well reproduced in this example.

The last example represents a free cube falling down into a recipient full of water.

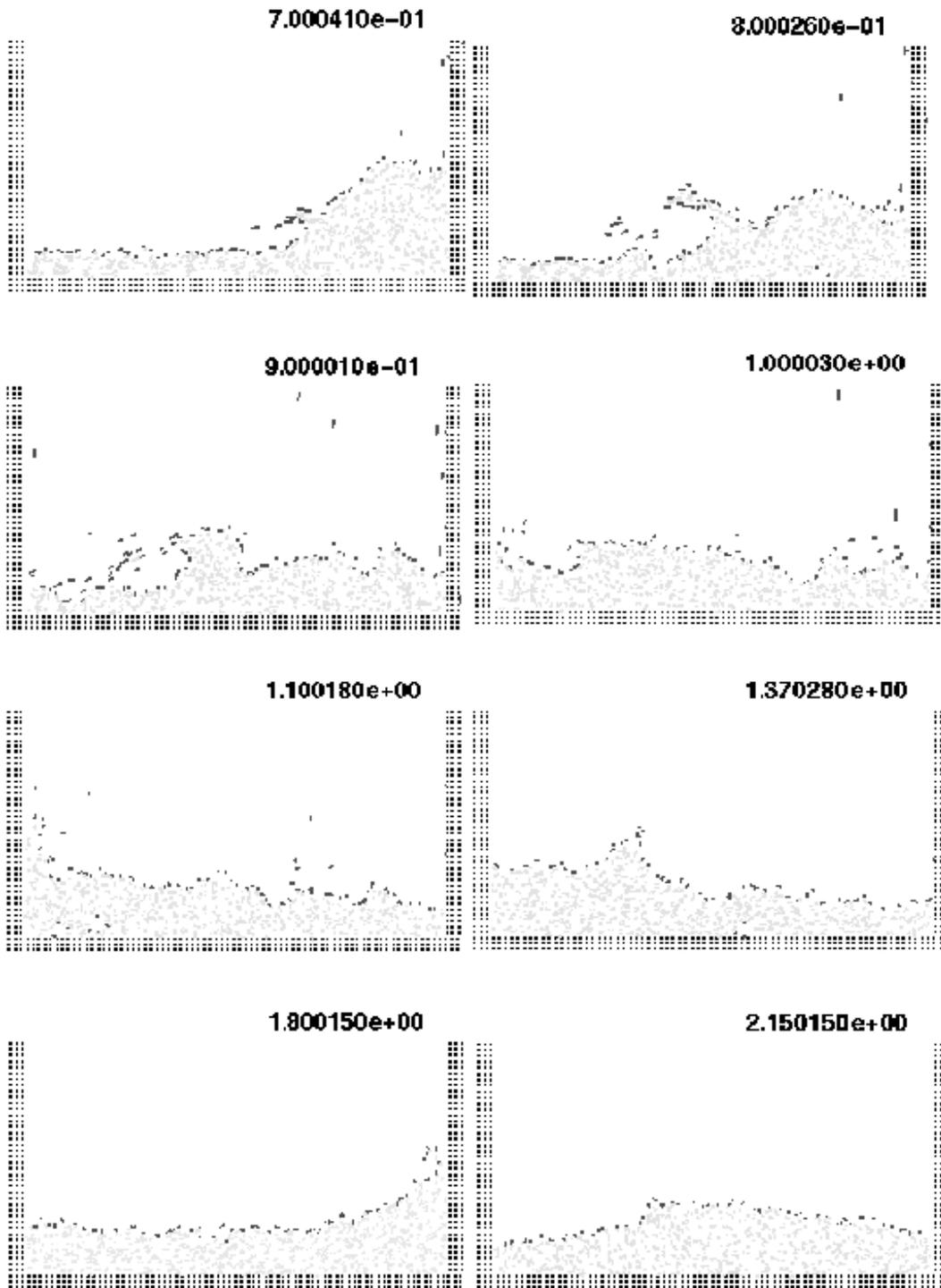


Figure 18: cont.

The solid cube was modeled by introducing a high viscosity parameter in the elements in the following way: all the polyhedral elements formed by nodes contained in the cube have a high viscosity value. The other elements are inviscid. The results of Figure 20 represent correctly the contact problem when the cube hits the water and also the speed during the sinking process.

More examples showing applications of the Lagrangian formulation previously described can be found in [51,52].

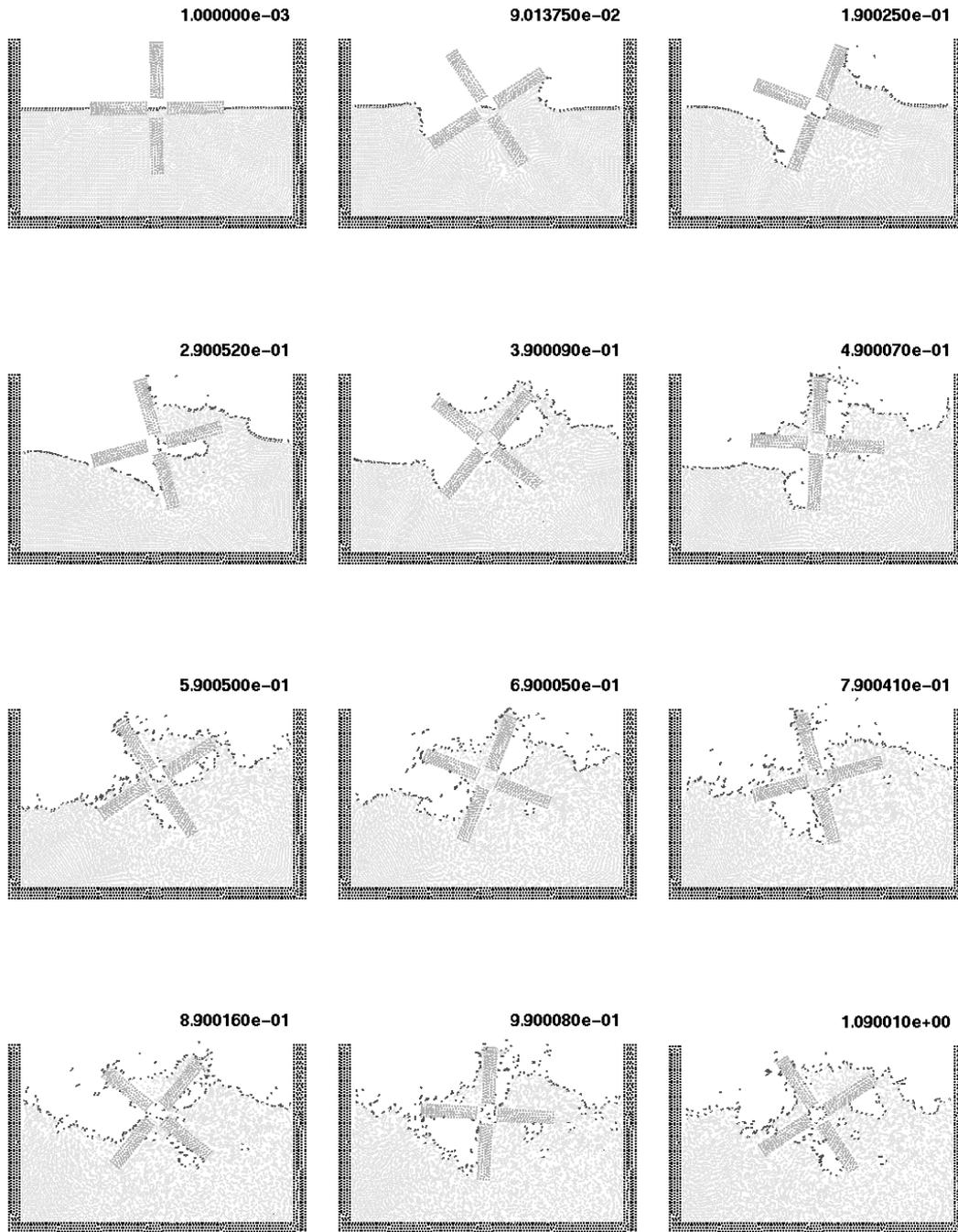


Figure 19: Rotating water mill.

11 CONCLUSIONS

The finite calculus form of the fluid mechanics equations is a good starting point for deriving stabilized finite algorithms for solving a variety of fluid flow problems using Euler, ALE and fully lagrangian descriptions. Both monolithic and fractional step algorithms with intrinsic stabilization properties can be readily derived as shown here. Free surface wave effects and fluid-structure interaction situations can be accounted for in a straight forward manner within the general flow solution schemes. The ALE formulation is particularly adequate for analysis of problems involving free

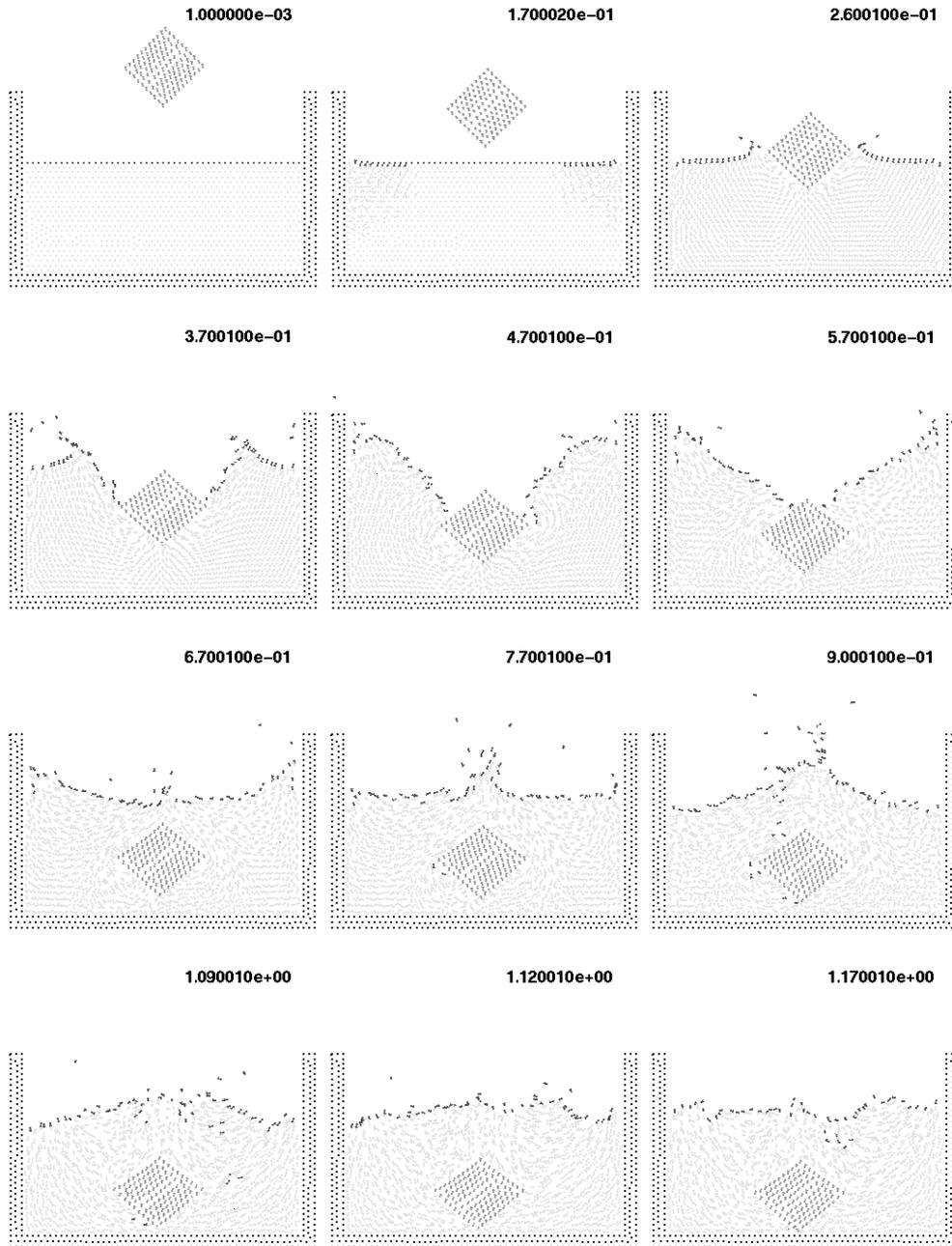


Figure 20: Solid cube falling into a recipient with water.

surface waves of moderate amplitude typical of ship hydrodynamics situations. The lagrangian formulation allows to solve in an effective manner fluid flow problems involving large motions of the free surface and complex fluid-structure interactions.

ACKNOWLEDGEMENTS

The authors are grateful to Copa America Desafio Español SA for providing the geometry and experimental data of the racing boat analyzed.

Examples 10.1–10.7 were analyzed with the finite element code Tdyn based on the FEM formulation here presented [66].

Thanks are also given to Dr. Roberto Flores for many useful discussions.

REFERENCES

- [1] O.C. Zienkiewicz and R.C. Taylor, *The finite element method*, 5th Edition, 3 Volumes, Butterworth–Heinemann, 2000.
- [2] C. Hirsch, *Numerical computation of internal and external flow*, J. Wiley, Vol. 1 1988, Vol. 2, 1990.
- [3] A.J. Chorin, “A numerical solution for solving incompressible viscous flow problems” *J. Comp. Phys.*, **2**, 12–26, 1967.
- [4] W.R. Briley, S.S. Neerarambam and D.L. Whitfield, “Multigrid algorithm for three-dimensional incompressible high-Reynolds number turbulent flows”, *AIAA Journal*, **33** (1), 2073–2079, 1995.
- [5] J. Peraire, K. Morgan and J. Peiro, “The simulation of 3D incompressible flows using unstructured grids”, In *Frontiers of Computational Fluid Dynamics*, Caughey DA and Hafez MM. (eds), Chapter 16, J. Wiley, 1994.
- [6] C. Sheng, L.K. Taylor and D.L. Whitfield, “Implicit lower-upper/approximate-factorization schemes for incompressible flows” *Journal of Computational Physics*, **128** (1), 32–42, 1996.
- [7] M. Storti, N. Nigro and S.R. Idelsohn, “Steady state incompressible flows using explicit schemes with an optimal local preconditioning”, *Computer Methods in Applied Mechanics and Engineering*, **124**, 231–252, 1995.
- [8] J.C. Heinrich, P.S. Hayakorn and O.C. Zienkiewicz, “An upwind finite element scheme for two dimensional convective transport equations”, *Int. J. Num. Meth. Engng.*, **11**, 131–143, 1977.
- [9] S.R. Idelsohn, M. Storti and N. Nigro, “Stability analysis of mixed finite element formulation with special mention to stabilized equal-order interpolations” *Int. J. for Num. Meth. in Fluids*, **20**, 1003–1022, 1995.
- [10] S.R. Idelsohn, N. Nigro, M. Storti and G. Buscaglia, “A Petrov-Galerkin formulation for advection-reaction-diffusion”, *Comput. Meth. Appl. Mech. Engrg.*, **136**, 27–46, 1996.
- [11] A. Brooks and T.J.R. Hughes, “Streamline upwind/Petrov-Galerkin formulation for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations”, *Comput. Methods Appl. Mech. Engrg*, **32**, 199–259, 1982.
- [12] T.J.R. Hughes and M. Mallet, “A new finite element formulations for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective-diffusive systems”, *Comput Methods Appl. Mech. Engrg.*, **58**, pp. 305–328, 1986.
- [13] P. Hansbo and a. Szepessy, “A velocity-pressure streamline diffusion finite element method for the incompressible Navier-Stokes equations”, *Comput. Methods Appl. Mech. Engrg.*, **84**, 175–192, 1990.
- [14] T.J.R. Hughes, L.P. Franca and M. Balestra, “A new finite element formulation for computational fluid dynamics. V Circumventing the Babuska-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accomodating equal order interpolations”, *Comput. Methods Appl. Mech. Engrg.*, **59**, 85–89, 1986.

- [15] L.P. Franca and S.L. Frey, “Stabilized finite element methods: II. The incompressible Navier-Stokes equations”, *Comput. Method Appl. Mech. Engrg.*, Vol. **99**, pp. 209–233, 1992.
- [16] T.J.R. Hughes, G. Hauke and K. Jansen, “Stabilized finite element methods in fluids: Inspirations, origins, status and recent developments”, in: *Recent Developments in Finite Element Analysis. A Book Dedicated to Robert L. Taylor*, T.J.R. Hughes, E. Oñate and O.C. Zienkiewicz (Eds.), (International Center for Numerical Methods in Engineering, Barcelona, Spain, pp. 272–292, 1994.
- [17] M.A. Cruchaga and E. Oñate, “A finite element formulation for incompressible flow problems using a generalized streamline operator”, *Comput. Methods in Appl. Mech. Engrg.*, **143**, 49–67, 1997.
- [18] M.A. Cruchaga and E. Oñate, “A generalized streamline finite element approach for the analysis of incompressible flow problems including moving surfaces”, *Comput. Methods in Appl. Mech. Engrg.*, **173**, 241–255, 1999.
- [19] T.J.R. Hughes, L.P. Franca and G.M. Hulbert, “A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations”, *Comput. Methods Appl. Mech. Engrg.*, **73**, pp. 173–189, 1989.
- [20] T.E. Tezduyar, S. Mittal, S.E. Ray and R. Shih, “Incompressible flow computations with stabilized bilinear and linear equal order interpolation velocity–pressure elements”, *Comput. Methods Appl. Mech. Engrg.*, **95**, 221–242, 1992.
- [21] J. Donea, “A Taylor-Galerkin method for convective transport problems”, *Int. J. Num. Meth. Engrg.*, **20**, 101–119, 1984.
- [22] J. Douglas, T.F. Russell, “Numerical methods for convection dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures”, *SIAM J. Numer. Anal.*, **19**, 871, 1982.
- [23] O. Pironneau, “On the transport-diffusion algorithm and its applications to the Navier-Stokes equations”, *Numer. Math.*, **38**, 309, 1982.
- [24] R. Löhner, K. Morgan, O.C. Zienkiewicz, “The solution of non-linear hyperbolic equation systems by the finite element method”, *Int. J. Num. Meth. in Fluids*, **4**, 1043, 1984.
- [25] R. Codina, M. Vazquez and O.C. Zienkiewicz, “A general algorithm for compressible and incompressible flow - Part III. The semi-implicit form” *Int. J. Num. Meth. in Fluids*, **27**, 13–32, 1998.
- [26] R. Codina and O.C. Zienkiewicz, “CBS versus GLS stabilization of the incompressible Navier-Stokes equations and the role of the time step as stabilization parameter”, *Communications in Numerical Methods in Engineering*, **18 (2)**, 99–112, 2002.
- [27] R. Codina and J. Blasco, “Stabilized finite element method for the transient Navier-Stokes equations based on a pressure gradient operator”, *Comput. Methods in Appl. Mech. Engrg.*, **182**, 277–301, 2000.
- [28] T.J.R. Hughes, “Multiscale phenomena: Green functions, subgrid scale models, bubbles and the origins of stabilized methods”, *Comput. Methods Appl. Mech. Engrg.*, Vol. **127**, pp. 387–401, 1995.

- [29] F. Brezzi, L.P. Franca, T.J.R. Hughes and A. Russo, “ $b = \int g$ ”, *Comput. Methods Appl. Mech. Engrg.*, **145**, 329–339, 1997.
- [30] R. Codina, “Stabilized finite element approximation of transient incompressible flows using orthogonal subscales”, *Comput. Methods Appl. Mech. Engrg.*, **191**, 4295–4321, 2002.
- [31] J. Donea and A. Huerta, “Finite element method for flow problems”, J. Wiley, 2003.
- [32] E. Oñate, *Derivation of stabilized equations for advective-diffusive transport and fluid flow problems*, *Comput. Meth. Appl. Mech. Engrg.*, Vol. **151**, pp. 233–267, (1998).
- [33] E. Oñate, J. García and S. Idelsohn, *Computation of the stabilization parameter for the finite element solution of advective-diffusive problems*, *Int. J. Num. Meth. Fluids*, Vol. **25**, pp. 1385–1407, (1997).
- [34] E. Oñate, J. García and S. Idelsohn, *An alpha-adaptive approach for stabilized finite element solution of advective-diffusive problems with sharp gradients*, *New Adv. in Adaptive Comp. Met. in Mech.*, P. Ladeveze and J.T. Oden (Eds.), Elsevier, (1998).
- [35] E. Oñate and M. Manzan, A general procedure for deriving stabilized space-time finite element methods for advective-diffusive problems, *Int. J. Num. Meth. Fluids*, **31**, 203–221, 1999.
- [36] E. Oñate and M. Manzan, “Stabilization techniques for finite element analysis of convection diffusion problems”, in *Computational Analysis of Heat Transfer*, G. Comini and B. Sunden (Eds.), WIT Press, Southampton, 2000.
- [37] E. Oñate, “A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation”, *Comp. Meth. Appl. Mech. Engrg.*, **182**, 1–2, 355–370, 2000.
- [38] E. Oñate and J. García, “A finite element method for fluid-structure interaction with surface waves using a finite calculus formulation”, *Comput. Methods Appl. Mech. Engrg.*, **191**, 635–660, (2001).
- [39] E. Oñate, “Possibilities of finite calculus in computational mechanics”, Submitted to *Int. J. Num. Meth. Engrg.*, 2002.
- [40] E. Oñate and S. Idelsohn, A mesh free finite point method for advective-diffusive transport and fluid flow problems,, *Computational Mechanics*, **21**, 283–292, 1988.
- [41] E. Oñate, C. Sacco and S. Idelsohn, “A finite point method for incompressible flow problems”, *Computing and Visualization in Science*, **2**, 67–75, 2000.
- [42] T.E. Tezduyar, M. Behr and J. Liou, “A new strategy for finite element computations involving moving boundaries and interfaces - the deforming-spatial-domain/space-time procedure: I. The concept and the preliminary tests”, *Comput. Methods in Appl. Mech. and Engrg.*, **94**, 339–351, 1992.
- [43] S. Mittal and T.E. Tezduyar, “Parallel finite element simulation of 3D incompressible flows - fluid structure interactions”, *Int. J. Num. Meth. Fluids*, **21**, 933–953, 1995.

- [44] N. Maman and C. Farhat, “Matching fluid and structure meshes for aeroelastic computations: A parallel approach”, *Computers and Structures*, **54**, 779–785, 1995.
- [45] G. Chiandusi, G. Bugeda and E. Oñate, “A simple method for update of finite element meshes”, *Commun. Numer. Meth. Engng.*, **16**, 1–9, 2000.
- [46] E. Oñate, S. Idelsohn, C. Sacco and J. García, “Stabilization of the numerical solution for the free surface wave equation in fluid dynamics”, ECCOMAS CFD98, K. Papaliou *et al.* (Eds.), J. Wiley, 1998.
- [47] J. García, *A finite element method for analysis of naval structures* (in Spanish), Ph.D. Thesis, Univ. Politècnica de Catalunya, December (1999).
- [48] J. García and E. Oñate, “An unstructured finite element solver for ship hydrodynamic problems”, Accepted for publication in *J. Appl. Mech.*, 2002.
- [49] E. Oñate, J. García and S.R. Idelsohn, “Ship hydrodynamics”, in *Encyclopedia of Computational Mechanics*, E. Stein, R. de Borst and T.J.R. Hughes (Eds), J. Wiley, 2004.
- [50] S.R. Idelsohn, E. Oñate, N. Calvo and F. del Pin, “The meshless finite element method”, Accepted for publication in *Int. J. Num. Meth. Engng.*, 2002.
- [51] S.R. Idelsohn, E. Oñate, F. Del Pin and N. Calvo, “Lagrangian formulation: the only way to solve some free-surface fluid mechanics problems”, *Fifth World Congress on Computational Mechanics*, Mang HA, Rammerstorfer FG and Eberhardsteiner J. (eds), July 7–12, Viena, Austria, 2002, Web...
- [52] S.R. Idelsohn, E. Oñate and F. Del Pin, “A lagrangian meshless finite element method applied to fluid-structure interaction problems”, Submitted to *Computer and Structures*, 2002.
- [53] D.C. Wilcox, *Turbulence modeling for CFD*, DCW Industries Inc., 1994.
- [54] T.J.R. Hughes and M. Mallet, “A new finite element formulations for computational fluid dynamics: IV. A discontinuity capturing operator for multidimensional advective-diffusive system”, *Comput. Methods Appl. Mech. Engrg.*, **58**, 329–336, 1986.
- [55] R. Codina, “A discontinuity-capturing crosswind dissipation for the finite element solution of the convection-diffusion equation”, *Comput. Methods Appl. Mech. Engrg.*, **110**, 325–342, 1993.
- [56] J.H. Duncan, “The breaking and non-breaking wave resistance of a two-dimensional hydrofoil”, *J. Fluid Mech.*, Vol. **126**, 1983.
- [57] T. Hino, L. Martinelli and A. Jameson, “A finite volume method with unstructured grid for free surface flow”, pp. 173-194 in *Proc. of the 6th Int. Conf. Num. Ship Hydrodynamics*, Iowa City, Iowa, 1993.
- [58] S.R. Idelsohn, E. Oñate and C. Sacco, *Finite element solution of free surface ship-wave problem*, *Int. J. Num. Meth. Engng.*, **45**, 503–508, (1999).
- [59] R. Löhner, C. Yang, E. Oñate and S. Idelsohn, *An unstructured grid-based parallel free surface solver*, *Appl. Num. Math.*, **31**, 271–293, (1999).
- [60] A.A. Johnson and T.E. Tezduyar, “3D simulation of fluid-particle interaction with the number of particles reaching 100”, *Comput. Methods Appl. Mech. Engrg.*, **145**, 301–321, 1997.

- [61] Proceedings of 2nd DTNSRDC Workshop on Ship Wave Resistance Computations. David Taylor Naval Ship Research and Development Center. Noblese, F. and McCarthy J.H. (Eds.), Maryland, USA, 1983.
- [62] J.R. Farmer, L. Martinelli and A. Jameson, “A fast multigrid method for solving incompressible hydrodynamic problems with free surfaces”, *AIAA J.*, **32**, 6, 1175-1182, 1993.
- [63] Korea Research Institute of Ships and Ocean Engineering (KRISO). <http://www.ihr.uiowa.edu/gothenburg2000/KVLCC/tanker.html>.
- [64] J. García, R. Luco-Salman, M. Salas, M. López-Rodríguez and E. Oñate E, “An advanced finite element method for fluid-dynamic analysis of America’s Cup boat”, *High Performance Yatch Design Conference*, Auckland, 4–6, December, 2002.
- [65] S. Koshizuka and Y. Oka, “Moving particle semi-implicit method for fragmentation of incompressible fluid”, *Nuclear Engineering Science*, **123**, 421-434, 1996.
- [66] *Tdyn. A finite element code for fluid-dynamic analysis*, COMPASS Ingeniería y Sistemas SA, www.compassis.com, (2002).