Asociación Argentina
de Mecánica Computacional

Mecánica Computacional Vol XXV, pp. 169-183
Alberto Cardona, Norberto Nigro, Victorio Sonzogni, Mario Storti. (Eds.)
Santa Fe, Argentina, Noviembre 2006

NUMERICAL SIMULATION OF FREE SURFACE FLOWS WITH VOLUME CONTROL

Laura Battaglia, Gerardo J. Franck, Mario A. Storti and Jorge D'Elía

Centro Internacional de Métodos Computacionales en Ingeniería (CIMEC) , Instituto de Desarrollo Tecnológico para la Industria Química (INTEC) , Universidad Nacional del Litoral - CONICET , Güemes 3450, 3000-Santa Fe, Argentina , e-mail: lbattaglia@ceride.gov.ar, (gfranck,mstorti,jdelia)@intec.unl.edu.ar , web page: **http://www.cimec.org.ar**

Keywords: free surface flows, axisymmetric vertical vortex, mesh-movement, volume control, finite elements, parallel computing, fluid mechanics, multi-physics.

Abstract. In previous works (Battaglia et al, Mecánica Computacional, vol. XXIV, pp. 105-116, Buenos Aires, Argentina, Nov. 2005), the free surface movement of a fluid flow of an incompressible and viscous fluid was followed by a mesh-movement technique, where the update of the free surface location was smoothed in order to prevent numerical instability due to a fully explicit discretization of the differential equation that describes the free surface kinematics. Unlike fluid flows in closed domains, in cases where the movement of the free surface coexists with inflow and outflow sections with fixed parameters, the volume of the fluid could grow or decrease in an unexpected way, i.e., the initial parameters for incoming velocity and discharge pressure could not be appropriate for keeping bounded the volume of the fluid, leading to somewhat artificial free surface displacements. Then, a control technique is shown for modifying some parameters in order to balance the flow between successive steps during the time marching simulation, interacting with the multi-physics finite element code PETSc-FEM (http://www.cimec.org.ar/petscfem/). This control is achieved with an extension of the hooks technology presented in Battaglia et al. (Mecánica Computacional, vol. XXIII, pp. 3119-3132, Bariloche, Argentina, Nov. 2004). As a numerical example, the development of the free surface of an axisymmetric vertical vortex is simulated with this volume control strategy and a finite element (FE) computation.

1 INTRODUCTION

In previous presentations, e.g. see Battaglia et al. (2004, 2005, 2006), a Lagrangian method was used for solving free surface flows of viscous and incompressible fluids. Unlike these cases, where the fluid domains were chosen as closed, a numerical simulation of free surface flows with inflow and outflow sections presents an additional difficulty given by a proper selection of parameters to be imposed as boundary conditions, such as the flow rate or the pressure. For example, in the case of a draining tank, the following boundary conditions can be chosen: there is not fluid entrance and the pressure is known at the exit, giving a right free surface position. But if there is also an incoming flow rate, it can be not trivial to determinate its value for a flow regime in equilibrium, because its balance with the outgoing flow rate and net volume in the fluid domain is not known *a priori*, due to the effects of the free surface shape and vertical position (which can be raising or lowering) over the remaining flow field.

In the present work it is assumed that the pressure at outflow section is given, and the unknown parameter is the amount of fluid which must be admitted in the flow system to arrive at some equilibrium position for the free surface, that implies the same outgoing flow rate. Then, it is proposed to handle this situation through a control strategy which demands information from the CFD solver in a dynamic fashion, in such a way that the flow rate is automatically modified at each time step in order to achieve some target objective, such as the net volume in the instantaneous flow domain which, in turn, fix the position of the free surface. This mechanism of control is implemented under the scope of the finite element code PETSc-FEM, see Sonzogni et al. (2002), oriented to multiphysics, and it is based on the Message Passing Interface (MPI, **`http://www.mpi-forum.org/docs/docs.html`**) and the Portable Extensible Toolkit for Scientific Computations (PETSc, Balay et al. (2005), **`http://www-fp.mcs.anl.gov/petsc/`**).

2 IMPLEMENTATION

2.1 General Description

Free surface problems are solved by using separate modules of PETSc-FEM, which were programmed for rather specific applications, such as the Navier-Stokes (NS) solver or the moving mesh (MMV) one. They are used for solving the flow-fluid problem and the mesh-update one, respectively. In addition to the main modules, there are also C++ programs or scripts, called *hooks*, that control the exchange of information among the other ones. For the free surface process, there are so far the following C++ *hooks* of interest:

- *From NS solver to MMV*, that sends the free surface nodal displacements through a FIFO (First Input First Output);
- *From MMV to NS solver*, which returns the new nodal positions of the mesh nodes, also through a FIFO;
- *Surface Smoothing*, that performs a free surface stabilization with the objective of minimize numerical instabilities caused by the explicit formulation of the kinematic free surface condition between the NS and MMV solvers;
- *Control*, for cases which require the control over one or more parameters, such as boundary condition values. In this case, it is proposed a target volume for the domain, being controlled through the flow rate of fluid incoming to the domain.

Figure 1 shows a sketch of the computational interaction among the procedures involved in this proposal for treating fluid flows with a free surface.

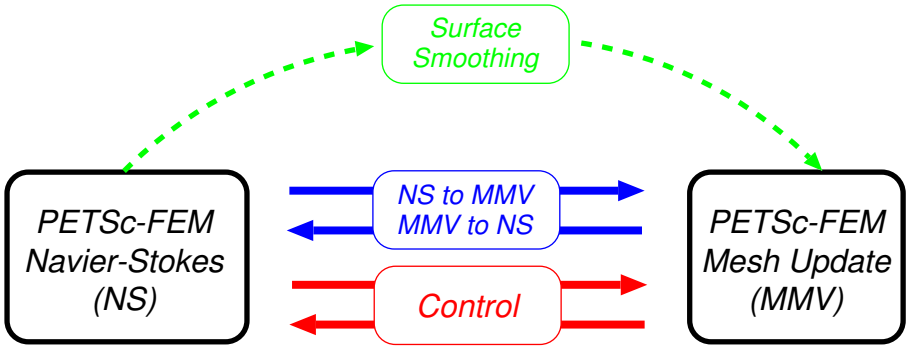

Figure 1: Computational interaction among modules in PETSc-FEM code for solving fluid flows with a free surface.

2.2 Fluid Problem

As described previously, the fluid part of the problem is solved by a NS module for the flow of a viscous and incompressible fluid, i.e. the partial differential equations on the flow domain $\Omega_t = \Omega(t)$ are written as

$$\begin{aligned} \rho\left(\partial_t \mathbf{v} + \mathbf{v}\cdot\nabla\mathbf{v} - \mathbf{f}\right) - \nabla\cdot\boldsymbol{\sigma} = 0\ ; \\ \nabla\cdot\mathbf{v} = 0\ ; \end{aligned} \tag{1}$$

at time t, with $t \in [0, T]$, being T some final time of simulation, $\mathbf{v}$ the fluid velocity, $\mathbf{f}$ the body force, ρ the fluid density, and the fluid stress tensor $\boldsymbol{\sigma}$ given by

$$\boldsymbol{\sigma} = -p\mathbf{I} + \mathbf{T}\ ; \tag{2}$$

as the sum of an isotropic $-p\mathbf{I}$ part due to the hydrostatic pressure p, $\mathbf{I}$ is the identity tensor and the deviatoric term

$$\mathbf{T} = 2\mu\boldsymbol{\epsilon} \quad ; \quad \boldsymbol{\epsilon} = \frac{1}{2}\left[\nabla\mathbf{v} + (\nabla\mathbf{v})^T\right]\ ; \tag{3}$$

which involves the dynamic and kinematic fluid viscosities μ and $\nu = \mu/\rho$, respectively, with $(...)^T$ indicating transposition. The boundary conditions at the flow boundaries Γ are split as

$$\begin{aligned} \mathbf{v} &= 0 \qquad \text{at } \Gamma_{wall}; \\ p &= P_{atm} \quad \text{at } \Gamma_{FS}; \\ \boldsymbol{\tau}\cdot\mathbf{n} &= 0 \qquad \text{at } \Gamma_{FS}; \end{aligned} \tag{4}$$

where Γ_{wall} is the solid walls and Γ_{FS} is the free surface. The last expression indicate that free surface is allowed to move in its instantaneous normal direction, represented by the unit normal $\mathbf{n}$.

Nodal velocities over the free surface are updated from one time step t^n to the following one t^{n+1} through the approximation

$$\mathbf{v}_j^{n+1} \approx \frac{\mathbf{x}_j^{n+1} - \mathbf{x}_j^n}{\Delta t}\ . \tag{5}$$

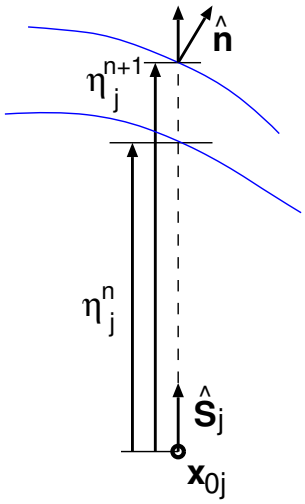

Figure 2: Directions and magnitudes for free surface nodes displacement.

The displacements of these nodes are restricted to a fixed direction, or "spine", represented by $\hat{\mathbf{s}}_j$, which in the scope of this study is chosen as vertical. The nodal positions are then calculated by means of

$$\mathbf{x}_j(t) = \mathbf{x}_{0,j} + \eta_j(t)\,\hat{\mathbf{s}}_j\ ; \tag{6}$$

and the scalar η_j is the magnitude of the displacement for the node whose initial position is $\mathbf{x}_{0,j}$, see Fig. 2. Considering Eq. (5), such displacement is

$$\Delta\eta_j^{n+1} = \eta_j^{n+1} - \eta_j^n = \Delta t \frac{\mathbf{v}_j^{n+1}\cdot\hat{\mathbf{n}}_j^n}{\hat{\mathbf{s}}_j\cdot\hat{\mathbf{n}}_j^n}\ . \tag{7}$$

For each node, the normal $\hat{\mathbf{n}}_j^n$ is computed each time step integrating over neighbor elements the approximate function $N_j(\mathbf{x})$.

In the vicinity of free surface, special boundary conditions on solid walls are proposed. This resource is applied because of large gradients generated in these zones by the non-slip condition, which is not only non-physical but also detrimental for the mesh update process. Then, either a perfect slip condition or the Navier slip condition are used, see Battaglia et al. (2004).

2.3 Mesh Update

After solving each time step of the NS problem, the updated free surface position can be determined either calculating explicitly the nodal displacements $\Delta\eta_j^{n+1}$, see Eq. (7), or through a surface smoothing process, once the velocity field is known.

In the PETSc-FEM code there are currently implemented two main strategies for computing the updated positions of the internal nodes keeping the topology unchanged: a pseudo-elastic update and a mesh motion strategy by a mesh-quality optimization, as reported in Battaglia et al. (2005).

2.3.1 Pseudo-elastic update

In this strategy, the new positions for the internal nodes, i.e., those which are not over the free surface, are determined by solving a pseudo-elastic problem over the domain Ω_0 with Dirichlet

boundary conditions. The standard formulation is

$$\begin{aligned} \sigma_{ij,j} &= 0 \ ; \\ \sigma_{ij} &= 2\tilde{\mu}\epsilon_{ij} + \tilde{\lambda}\delta_{ij}\epsilon_{kk} \ ; \\ \epsilon_{ij} &= \frac{1}{2}(u_{i,j} + u_{j,i}) \ ; \end{aligned} \tag{8}$$

being $\tilde{\mu}$ and $\tilde{\lambda}$ Lamé elastic constants adopted arbitrarily for the pseudo-material, with δ_{ij} as the Kronecker tensor. In practical situations, these parameters are replaced by the pseudo Poisson ratio $\tilde{\nu}$ and the pseudo elasticity modulus $\tilde{E}$, that are usually about 0.3 and 1.0, respectively. Notice that as the b.c.'s for the pseudo-elastic problem are all of Dirichlet type, the node displacements are independent of a multiplicative constant in $\tilde{E}$. In some cases, $\tilde{E}$ can be made artificially variable over the domain, increasing it near to critical points, like reentrant corners, in order to reduce mesh distortion in such places.

The free surface nodal displacements computed by $\mathbf{u}_j = \mathbf{x}_j^{n+1} - \mathbf{x}_j^0$, the slip ($\mathbf{u} \cdot \hat{\mathbf{n}} = 0$) and non-slip ($\mathbf{u} = 0$) over the rest of the frontiers, usually solid contours, are the boundary conditions that complete the set of needed data for solving the problem.

Briefly, following Xu and Accorsi (2004), the element stiffness matrix $\mathbf{K}_e$ for a pseudo-elastic problem can be calculated as follows,

$$\mathbf{K}_e = \int_{\Omega_t^e} \mathbf{B}^T \mathbf{D} \mathbf{B} \, |\mathbf{J}|^e \tau^e \, d\Omega_t^e \ ; \tag{9}$$

being $\mathbf{B}$ the derivative matrix of shape functions, $\mathbf{D}$ the constitutive matrix, $|\mathbf{J}|^e$ the elemental Jacobian and τ^e a factor for controlling stiffening, given by Tezduyar et al. (1993) and Stein et al. (2004) as

$$\tau^e = \left(\frac{|\mathbf{J}|^0}{|\mathbf{J}|^e} \right)^s \ ; \tag{10}$$

where the *stiffness exponent* s is a user-chosen parameter and $|\mathbf{J}|^0$ a scaling coefficient for consistency.

For linear pseudo elastic update $s = 0$ is the usual value, while for the non-linear alternative $s > 0$ is chosen, being $s < 0$ meaningless, regarding that $s \geq 0$ makes the smaller elements become more stiffened than the larger ones.

2.3.2 Mesh motion strategy by a mesh-quality optimization

This was designed for moving boundary problems, such as those with free surfaces, see López et al. (2006). The aim of this strategy is to get the best possible mesh at each time step under a selected quality criteria. The problem to be solved is an optimization one, where the functional to minimize could be written as

$$F = F(\{x_j^\alpha\}); \tag{11}$$

where x_j^α is the α-coordinate of j-node x_j and the set of mesh coordinates is $\{x_j^\alpha\}$. This functional must fulfill some requirements in order to be appropriate for the minimization process, and it is proposed as a sum of the element contributions

$$F = \sum_e F_e; \quad F_e = C_v \left(\frac{V_e}{V_{ref}} - 1 \right)^m + C_Q \, Q^n \ ; \tag{12}$$

being V_e the element volume, V_{ref} the target volume, C_v and C_Q weight coefficients where is allowed the user intervention, m, n norms to apply to size and shape measures, with certain restriction over its values, and Q a quality indicator for each element with the form of a quotient between the elemental volume V_e and the sum of its edge lengths l_i weighted to a the power p, coincident with the space dimension, that is,

$$Q = \frac{C\, V_e}{\sum_i l_i^p}\ ; \tag{13}$$

being the constant C chosen to get $0 < Q \leq 1$, where $Q = 1$ represents the equilateral element, i.e., in the case of the "simplices", $C = 4\sqrt{3}$ for triangles and $C = 36\sqrt{2}$ for tetrahedral elements. Other element shapes, like quadrangles or hexahedrals, can be taken into account by decomposing them in simplices and defining its quality as the quality of the worst simplex in that decomposition. The functional F given by Eq. (11) is computed from its element contributions, as a standard FE assembly process and it is submitted to an optimization process to get its minimum at each time step. The optimization process is performed with a global Newton-Raphson algorithm.

2.4 Control

The mechanism implemented so far consists of a volume control, which calculates the incoming flow rate q to impose by

$$\dot{q} = -C_{\text{vol}}\,(V - V_{\text{target}}); \tag{14}$$

where V is the volume of the fluid domain, V_{target} the objective volume, C_{vol} is a non-negative user chosen coefficient which allows the regulation of the variable term, and $\dot{q} = dq/dt$. Then, the flow rate decreases when $(V - V_{\text{target}}) > 0$, and it increases in the other case, in a proportion given by C_{vol}.

As well as the mentioned parameters, there are some restrictions over the incoming flow rate q, such as minimum and maximum values, the first to avoid the velocity turning outgoing instead ingoing in the inlet, and the second in order to keep bounded its growth.

Once q is calculated, as the inlet size can vary because of the mesh update process, the inlet radial velocity $\mathbf{v}_r$ is determined after reading the z-coordinate of the node which defines the height of the entrance, inside the same hook.

This control proposal counts on some disadvantages, the firs of them related to the fact that takes information from only one time step to determine the updated parameters for the following one. Besides, it requires some tests in order to reach an appropriate coefficient C_{vol} and it is sensible to the size of the time step. Other key subjects could be initial values for the variables considered, such as the initial flow rate or the target value, but these can be proposed from similar analysis developed without any control.

The fluid volume can be controlled through other parameters like gravity acceleration, atmospheric pressure or outlet pressure, but in practice the control of the incoming flow rate gave the best controlling characteristics, and was conceptually simpler.

In practical applications it was noticed that the proposed methodology could be improved in order to reach the objectives faster, leading to a lower computational cost. This could be made by including higher order terms in Eq. (14).

2.5 Free Surface Stabilization

It is a fact that explicit formulations of transport equations such as Eq. (7) lead to instabilities for high frequency, gravity waves or velocities relatively high in directions which are parallel to the interface. There are different studies made and treatments proposed for this subject, see for example García Espinosa (1999) or Güler et al. (1999). At this moment, PETSc-FEM is able to perform a smoothing step which avoid this problem, that consist on applying an operator $\mathcal{S}$ based on solving the heat equation over the free surface elevation by replacing Eq. (7) with

$$\Delta\eta_j^{n+1} = \mathcal{S}(\Delta\tilde{\eta}_j^{n+1}) \; ; \tag{15}$$

being $\Delta\tilde{\eta}_j^{n+1}$ the result of

$$\Delta\tilde{\eta}_j^{n+1} = \frac{\mathbf{v}_j^{n+1} \cdot \hat{\mathbf{n}}_j^n}{\hat{\mathbf{s}}_j \cdot \hat{\mathbf{n}}_j^n} \; . \tag{16}$$

For $\mathcal{S}$, the corresponding diffusivity is adjusted in order to get a spreading length of γh, with h the global mesh size and γ a parameter imposed by the user.

3 NUMERICAL EXAMPLE

Different authors proposed analytical and numerical solutions for problems as the classic *bath plug* or containers with an inferior drainage, see Forbes and Hocking (1995), where the free surface of the flow suffers the effects of the circumferential velocity, increasing to the vertical axis, showing the formation of the typical vortex, although the singularity over the vertical axis makes it a problem difficult to solve numerically speaking. Regarding that, a more convenient domain for studying is a portion of a reservoir for a suction duct, like the chosen one. As the inlet velocity is imposed with a predominant circumferential component, a vortex is generated in the fluid domain, leading to free surface deformation similar to those mentioned before. In the inviscid case without a radial velocity, in cylindrical coordinates r, ϕ, z, any circumferential velocity $v_\phi(r)$ is a solution for this problem. But when there is a small radial velocity $v_r(r)$, the circumferential fluid layers must keep its angular momentum and, as a result, the circumferential velocity $v_\phi(r)$ grows for $r \to 0$. The analytic solution shown in Sec. 3.1 corresponds to the limit for viscosity and radial velocity tending to zero. For the numerical experiment with a viscous fluid, it is necesary to impose some non-null radial velocity, because otherwise the fluid would decay to rest. Unfortunately, the solution for the inviscid case with circumferential and radial velocities was not found by the authors, but as an approximation to the null radial velocity case, a radial velocity component v_r small with respect to the circumferential one v_ϕ was adopted.

Thus, the example consists of a numerical simulation of an axisymmetrical vertical vortex with a free surface of a viscous and incompressible fluid under the action of a gravity field, with fixed pressure at the outlet and unknown incoming flow for a constant volume in the fluid domain. This problem was solved by the presented methodology and the simulated free surface profile is compared with the theoretical one in the inviscid limit.

3.1 Vertical potential vortex with a free surface under a gravity field

A vertical potential vortex with a free surface and under the action of the gravity field is considered, see sketch in Fig. 3. The vertical position $h = h(r)$ of its free surface respects to the hydrostatic equilibrium plane $z = 0$ is given by (e.g. see Appendix)

$$h(r) = \frac{v_0^2}{2g}\frac{r_0^2}{r^2} \; ; \tag{17}$$

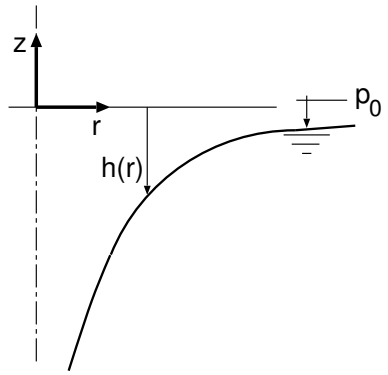

Figure 3: Sketch of a vertical potential vortex with a free surface.

where v_0 is the circumferential velocity at some radial distance r_0 from the vertical axisymmetric z-axis and g is the gravity acceleration.

3.2 FEM Simulation

3.2.1 Initial data

The numerical example consist of an axisymmetrical simulation of a vortex, with continuous fluid entrance given by the velocity in circumferential and radial directions, and mainly characterized by the rotation of fluid around the vertical axis, with a discharge proposed over a section of the inner cylindrical surface, see Fig. 4.

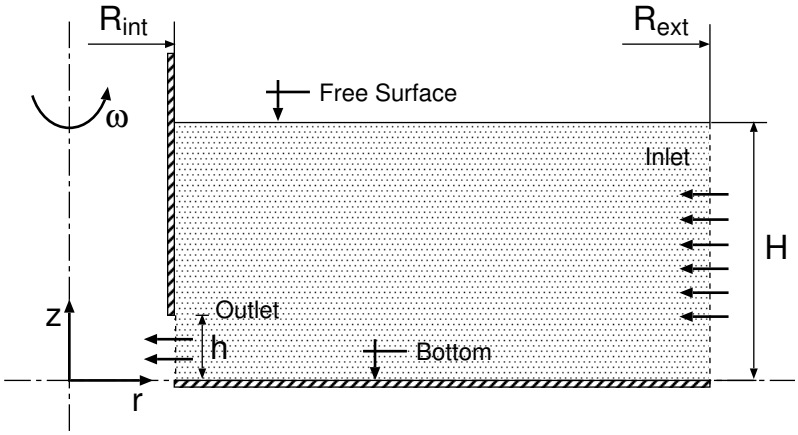

Figure 4: Geometry of the flow domain: a portion of a cylinder of annular section, limited by a free surface, a bottom and a rigid wall.

The geometry of this example is described with the help of Fig. 5. The flow domain is a cylinder of annular section, whose internal and external radius are $R_{\rm int} = 1.0$ m and $R_{\rm ext} = 6.0$ m respectively, an initial height $H = 1.40$ m, the same as the inlet section over the right, and outlet section $h = 0.20$ m height in the internal cylindrical surface. Over this outlet, there are conditions simulating the rigid wall which is assumed as part of the discharge system.

The fluid is adopted as viscous and incompressible, with kinematic viscosity $\nu = 0.001\ \mathrm{m}^2/\mathrm{s}$ and density $\rho = 1\ \mathrm{kg}/\mathrm{m}^3$. The gravity acceleration is $g = 0.16\ \mathrm{m}/\mathrm{s}^2$ in the $-z$ direction.

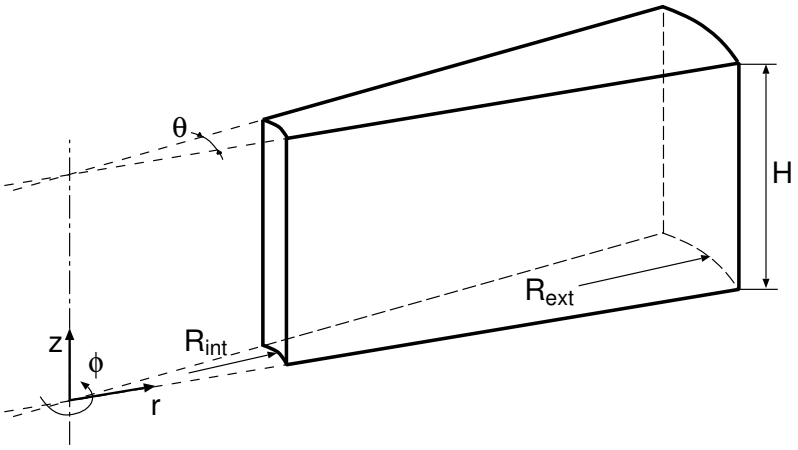

Figure 5: Sector of annular cylinder modeled for the example.

3.2.2 Model and boundary conditions

The axial symmetry of the example allowed the construction of the FE model, which consists of one layer of elements extruded from a two-dimensional rectangular mesh of quadrangles, see Fig. 6, around the z-axis, giving a sector of annular cylinder characterized by $\theta = 5^\circ$ of angular amplitude and 8-node hexahedral elements for the fluid model, although tetrahedral elements were used for the moving mesh instance through the mesh-quality optimization procedure, with $C_v = 0$, $C_Q = 1$ and $n = -1$ in Eq. (12).

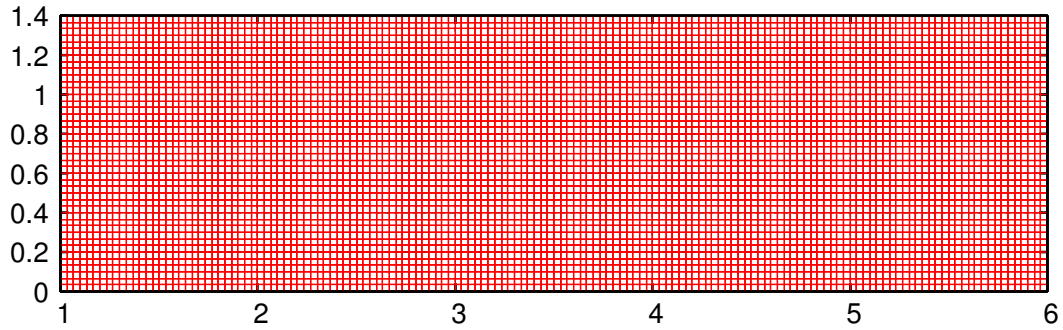

Figure 6: Two-dimensional mesh for generating the three-dimensional fluid model; scale in meters.

As the horizontal velocities were important in comparison with the free surface one, it was necessary to apply the smoothing operator given in Sec. 2.5 but with a low value for the influence coefficient, in this case $\gamma = 0.1$.

As said before, there are two main instances for the execution of a fluid flow with a free surface. The first of them is the NS one, where conditions are, following Fig. 7,

$$\begin{aligned} p &= P_{\text{atm}} && \text{over the free surface, CIJD;} \\ p &= p_{\text{out}} && \text{in the outlet, AGHB;} \\ \mathbf{v_r} &= \mathbf{v_t} = 0 && \text{over the inner surface, BHIC;} \\ \mathbf{v} \cdot \mathbf{n} &= 0 && \text{in the bottom, AFLG;} \\ \mathbf{v} \cdot \mathbf{n} &= 0 && \text{for the right slip section, KEDJ;} \\ \mathbf{v} &= \mathbf{v_i} && \text{in the inlet, LFEK;} \end{aligned} \tag{18}$$

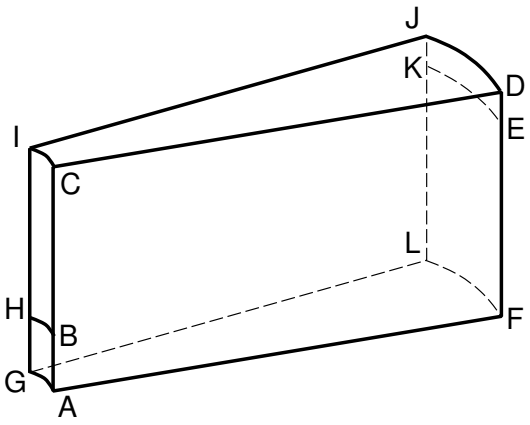

Figure 7: References for boundary conditions of the example.

being p_{out} the fixed pressure proposed for the outlet and $\mathbf{v_i}$ the incoming velocity, given by its magnitude in radial direction $|\mathbf{v_r}|$ and the ratio between this and the tangential velocity magnitude $|\mathbf{v_t}|$, i.e., the incidence angle, in this case of 100° measured from the r axis. The axisymmetry is imposed by periodic conditions among nodes of ACDF and LJIG faces, taking in account the angle θ between their planes. These restrictions imply that the radial, tangential and vertical components of $\mathbf{v}$ in each node of ACDF are the same as the corresponding ones in LJIG, as well as the pressure.

For the mesh movement procedure, the boundary conditions are null displacements in the outlet section (AGHB), null horizontal displacements for BHIC, LFDJ, ACDF and LJIG, allowing vertical movement for these nodes, and the free surface displaced by the NS solver through the corresponding hook. Notice that in this case not only the nodes over the free surface but all those over faces near it are moved in the direction given by the spines, which in this case is the vertical one.

The initial condition was the hydrostatic state for the free surface placed at the $z = 1.40$ m plane, at rest, giving an initial volume of $V = 2.14$ m^3, the time step was $\Delta t = 0.2$ s, atmospheric pressure $P_{\text{atm}} = 0$ and for the outlet $p_{\text{out}} = -0.045$ Pa.

The control applied to this example, described in Sec. 2.4, is characterized by a target volume of $V_{\text{target}} = 2.60$ m^3, with $C_{\text{vol}} = 1.0 \times 10^{-3}$ 1/s^2, initial flow rate $q_{\text{init}} = 1.0 \times 10^{-2}$ m^3/s, and $q_{\max} = 3.0 \times 10^{-2}$ m^3/s and $q_{\min} = 1.0 \times 10^{-4}$ m^3/s for upper and lower bounds, respectively. Initial values for q and V_{target} were obtained from previous tests made with the same model.

3.2.3 Results

The present results were obtained by using the mesh update process based on a mesh-quality optimization of Sec. 2.3.2, because it was the only available able to tolerate free surface displacements near the inner radius, circumstance that is illustrated through Fig. 8. The other alternative using the pseudo-elastic update of Sec. 2.3.1, was not able to solve the displacements registered in that region, in particular during the first time steps, when oscillations in the fluid level had higher amplitudes.

The volume and flow rate evolution during the first 10000 time steps of the analysis, plotted in Fig. 9, show how the control mechanism operates over these parameters. In this case, the initial volume was lower than V_{target}, which made the flow rate q keep growing until $q_{\max}$ was reached; then, the flow rate q kept its value until $V > V_{\text{target}}$, when the system begun to diminish

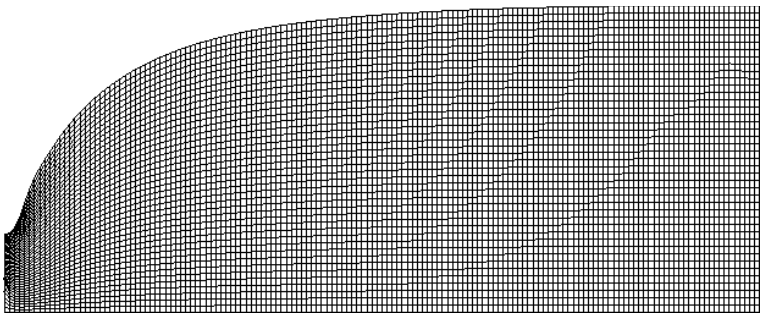

Figure 8: Deformed mesh for time $t = 40$ s; notice high displacements of free surface near the inner radius.

it. This behavior was repeated with lower amplitude of oscillation during the rest of the time, verifying that the task given to the control was accomplished. Notice that upper and lower values proposed for q were effective at the beginning of the study, saving important analysis time.

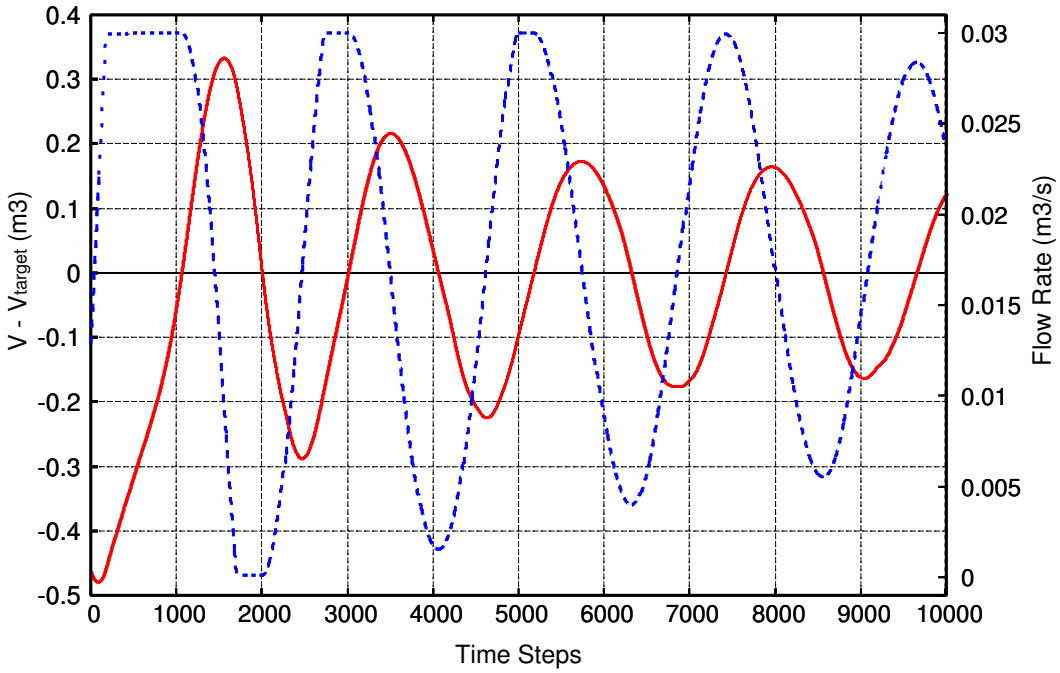

Figure 9: Curves of volume and flow rate variation for the example solved for the first 10000 time steps of analysis.

The displacements of the free surface were higher at the beginning of the study, according to the volume variations already shown. In Fig. 10, there are several free surface profiles for the times indicated, most of them coincident with maximum or minimum volume values. Then, it is possible to see how these displacements affected the mesh update process.

After 20000 time steps, the volume variation keeps under 1% of difference from the target volume, and the free surface exhibits a steady profile, shown in Fig. 11 along with the theoretical potential curve of the potential infinite vortex, from Sec. 3.1, being the incoming circumferential velocity determined at the inlet $v_0 = 0.0125$ m/s. The error between the curves compared are $e = 13\%$ at $r = 1$ m, but descends under 1% for $r > 2$ m. The high difference in the neighborhood of $r = 1$ m is attributed to the fact that the reference profile chosen is calculated for an inviscid flow with no rigid walls at the inner radius and null radial velocity.

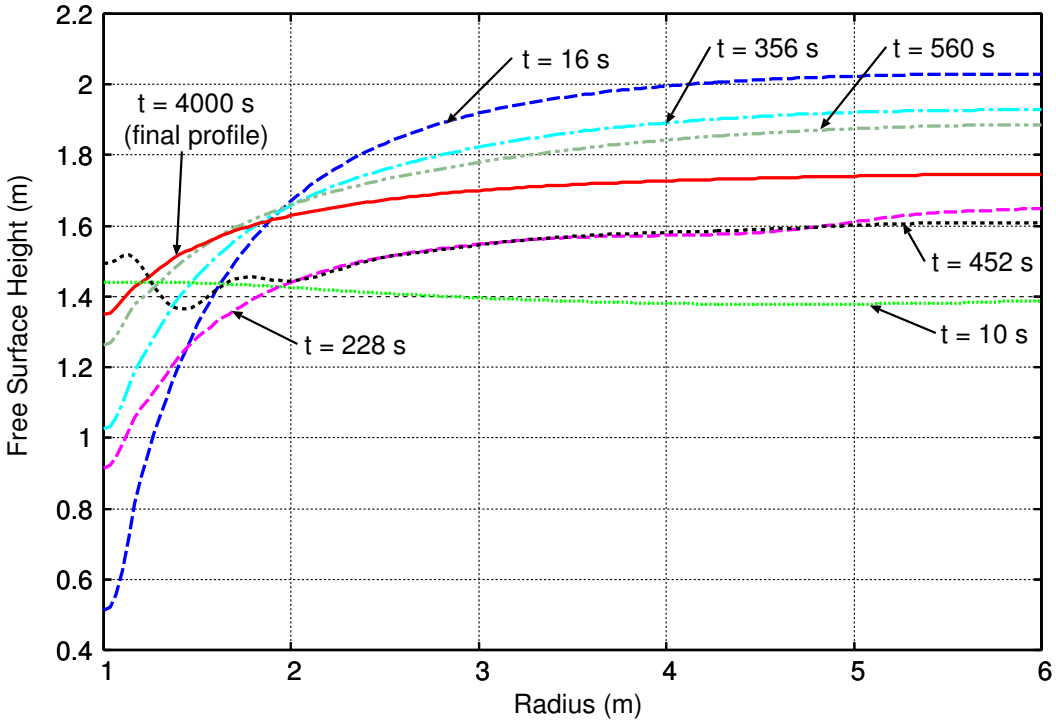

Figure 10: Free surface profiles calculated at several times.

Figure 12 shows the magnitude of $\mathbf{v}$ for an intermediate time step, where it is possible to see that the velocity increases in sectors closer to R_{int} because of the growth of the circumferential component and the small size of the outlet. The exception is given by the low values appreciated over the wall placed in the inner radius because of the boundary condition established there.

4 CONCLUSIONS

The control was able to lead the flow rate to a value that establishes the inflow/outflow balance for the given target volume, but its effectiveness could be higher by improving the mechanism, especially pointing to shorten the time of analysis. Furthermore, this proposal or a similar one is needed in order to assure that there are tools appropriate to solve this kind of problems, where inner and outer flows are not balanced from the beginning.

Besides, it was confirmed that there are some difficulties with the mesh update, especially in the case of large free surface displacements: in the presented example, only the mesh update based on mesh-quality optimization was able to solve the large displacements near the inner radius of the free surface, in particular during the first time steps, when oscillations in the fluid level had higher amplitudes.

5 APPENDIX

5.1 Analytical solution in the inviscid limit

The vertical position $h = h(r)$ of the free surface of a vertical potential vortex under the action of the gravity acceleration can be determined as follows, e.g. see Spurk (1997). The velocity potential in cylindrical coordinates r, ϕ, z is given by $\Phi = v_0\ r_0\ \phi$, where v_0 is the circumferential velocity at some radial distance r_0 from the vertical axisymmetric axis, named

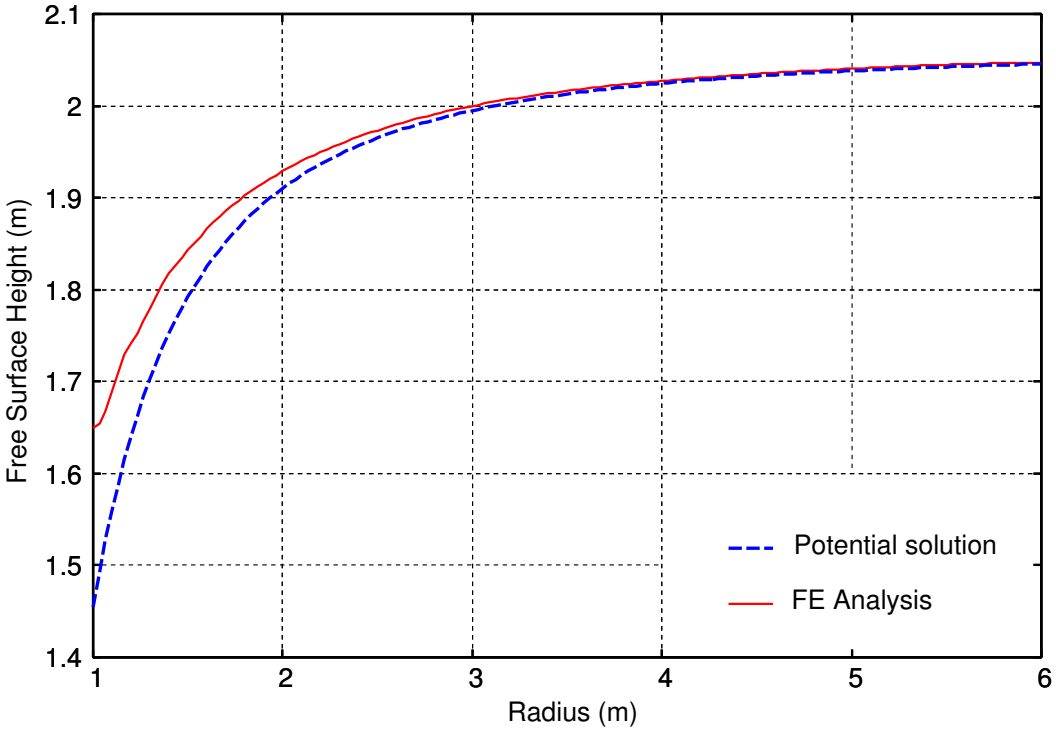

Figure 11: Final free surface profile from FE analysis, superimposed with the potential vortex solution.

z, see Fig. 3. Being $\mathbf{v} = (v_r, v_\phi, v_z)$ the velocity field, with

$$v_r = \frac{\partial \Phi}{\partial r} = 0 \quad ; \quad v_\phi = \frac{1}{r}\frac{\partial \Phi}{\partial \phi} = v_0 \frac{r_0}{r} \quad ; \quad v_z = \frac{\partial \Phi}{\partial z} = 0 \quad ; \tag{19}$$

from where $|\mathbf{v}| = v_\phi$. As the fluid is incompressible, the divergence of the velocity field $\nabla \cdot \mathbf{v}$ is null, the fluid density ρ is constant and, then, it is applicable the Bernoulli equation, which is imposed between two points over the free surface, the first one at $r = r_0$, far away enough from the origin, with $z = 0$ and $p = p_0$, and the other one at a generic position r with $z = -h(r)$ and the same pressure $p = p_0$. Then

$$p_0 + \frac{\rho}{2} v^2(r_0) = p_0 + \frac{\rho}{2} v^2(r) - \rho\, g\, h(r); \tag{20}$$

being g the gravity acceleration in the $-z$ direction, so

$$h(r) = \frac{v^2(r)}{2g} = \frac{v_0^2}{2g}\frac{r_0^2}{r^2}\ ; \tag{21}$$

which allows the calculation of free surface position as a function of r when the other parameters are fixed.

Acknowledgments The authors thank Rodrigo Paz for his help with PETSc-FEM applications. This work has received financial support from Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET, Argentina, grants PIP 02552/00, PIP 5271/05), Universidad Nacional del Litoral (UNL, Argentina, grant CAI+D 2005-10-64) and Agencia Nacional de Promoción Científica y Tecnológica (ANPCyT, Argentina, grants PICT 12-14573/2003, PME 209/2003), and was partially performed with the *Free Software Foundation/GNU-Project* resources as GNU/Linux OS and GNU/Octave, as well another Open Source resources as PETSc, MPICH and OpenDX.

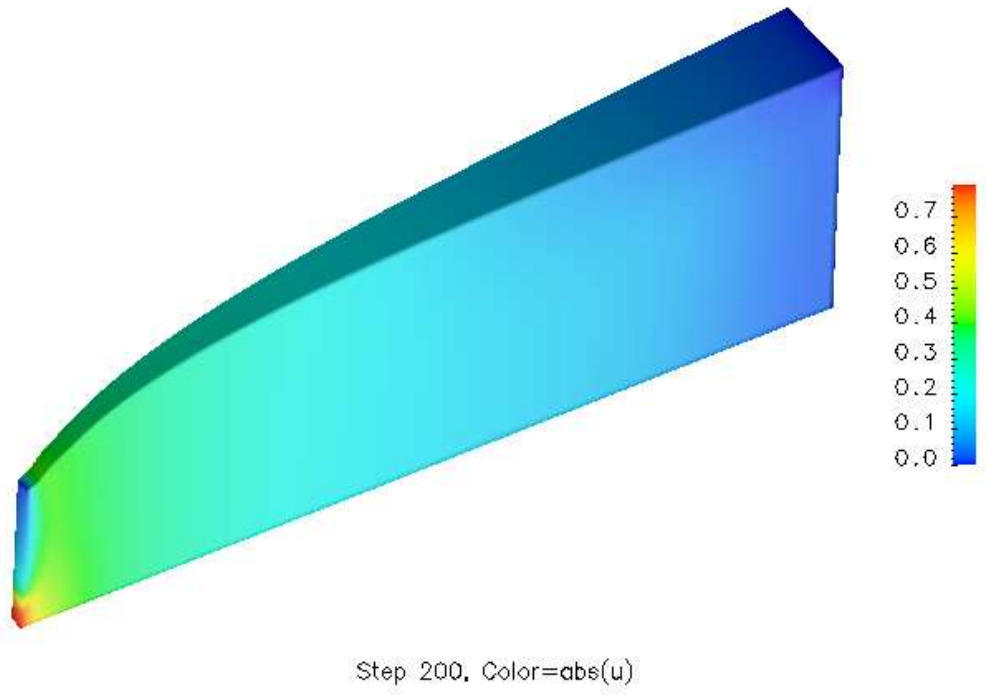

Figure 12: Magnitude of the velocity **v** in the domain analyzed for an intermediate time step.

REFERENCES

Balay S., Buschelman K., Eijkhout V., Gropp W., Kaushik D., Knepley M., McInnes L., Smith B., and Zhang H. Petsc 2.3.0 users manual. Technical Report UC-405, Argonne Nat. Lab., 2005.

Battaglia L., D'Elía J., Storti M., and Nigro N. Parallel implementations of free surface flows. In *Mecánica Computacional*, volume XXIII, pages 3119–3132. 2004.

Battaglia L., D'Elía J., Storti M., and Nigro N. Free-surface flows in a multi-physics programming paradigm. In *Mecánica Computacional*, volume XXIV, pages 105–116. 2005.

Battaglia L., D'Elía J., Storti M., and Nigro N. Numerical simulation of transient free surface flows using a moving mesh technique. *Journal of Applied Mechanics*, 2006. In press.

Forbes L.K. and Hocking G.C. The bath-plug vortex. *Journal of Fluid Mechanics*, 284:43–62, 1995.

García Espinosa J. *Un Método de Elementos Finitos para Análisis Hidrodinámico de Estructuras Navales*. Ph.D. thesis, Univerisdad Politécnica de Cataluña, 1999.

Güler I., Behr M., and Tezduyar T. Parallel finite element computation of free-surface flows. *Computational Mechanics*, 23(2):117–123, 1999.

López E., Nigro N., Storti M., and Toth J. A minimal element distortion strategy for computational mesh dynamics. *International Journal for Numerical Methods in Engineering*, 2006. In press.

Sonzogni V., Yommi A., Nigro N., and Storti M. A parallel finite element program on a Beowulf Cluster. *Advances in Engineering Software*, 33:427–443, 2002.

Spurk J. *Fluid mechanics: problems and solutions*. Springer-Verlag, Germany, 1997.

Stein K., Tezduyar T., and Benney R. Automatic mesh update with the solid-extension mesh

moving technique. *Comput. Meth. Appl. Mech. Engrg.*, 192:2019–2032, 2004.

Tezduyar T., Aliabadi S., Behr M., Johnson A., and Mittal S. Parallel finite-element computation of 3d flows. *Computer*, 26:27–36, 1993.

Xu Z. and Accorsi M. Finite element mesh update methods for fluid-structure interaction simulation. *Finite Element in Analysis and Design*, 40:1259–1269, 2004.