A Minimal Element Distortion Strategy for Computational Mesh Dynamics

Ezequiel J. López[1], Norberto M. Nigro[1,*], Mario A. Storti[1] and Jorge A. Toth[2]

[1] *Centro Internacional de Métodos Computacionales en Ingeniería (CIMEC),*
INTEC-CONICET-UNL, Güemes 3450, 3000 Santa Fe, ARGENTINA
[2] *Universidad Nacional del Comahue, Buenos Aires 1400*
8300 Neuquén, ARGENTINA

SUMMARY

Mesh motion strategy is one of the key points in many fluid-structure interaction (FSI) problems. Due to the increasing application of FSI to solve the current challenging engineering problems this topic has deserved a highlight interest. There are several different strategies to solve this problem, some of them using a discrete and lumped spring-mass system to propagate the boundary motion into the volume mesh and many others using an elastostatic problem to deform the mesh. In all these strategies there is always a risk of producing an invalid mesh, a mesh with some elements inverted. Normally this condition is irreversible and once an invalid mesh is obtained it is difficult to continue.

In this paper the mesh motion strategy is defined as an optimization problem. By its definition this strategy may be classified as a particular case of an elastostatic problem where the material constitutive law is defined in terms of the minimization of certain energy functional that takes into account the degree of element distortion. Some advantages of this strategy is its natural tendency to high quality meshes, its robustness and its straightforward extension to 3D problems. Several examples included in this paper show these capabilities.

Even though this strategy seems to be very robust it is not able to recover a valid mesh starting from an invalid one. This improvement is left for future work.
Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: ALE, computational mesh dynamics, computational fluid dynamics, optimal mesh

1. Introduction

Over the last decades technological development in several engineering fields has placed emphasis on fluid-structure interaction (FSI) problems. As a consequence, numerical simulation of coupled hydrodynamic/structural problems has become one of the most challenging problems in computational fluid dynamics [1]. This kind of problems requires to solve both

---

*Correspondence to: Dr. Norberto M. Nigro
Centro Internacional de Métodos Computacionales en Ingeniería (CIMEC),
INTEC-CONICET-UNL, Güemes 3450, 3000 Santa Fe, ARGENTINA

coupled physical domain, the fluid and the structure problems in such a way that the flow problem dictates the generalized forces acting over the structure and the structural problem determines the geometry variation due to its deformation and also its translational and rotational motion. Currently efficient solution of fluid-structure interaction problems with large wall displacement is still a challenging problem in computational mechanics. Generally these problems are solved iteratively. First the fluid solution is obtained while treating the structure as rigid using a CFD (Computational Fluid Dynamics) code. The pressure and possibly the shear stresses are stored at each point of the fluid-structure interface. Then these time-dependent loads are transferred to the structural code, which finds the deformation of the structure and the displacement of the fluid-structure interface in time using a CSD (Computational Structure Dynamics) code. Then a new fluid solution is found by prescribing the motion of the structure. An alternative to this approach is to reformulate the problem as a coupled fluid-structure problem, and numerically solve the system of governing equations simultaneously. In fluid-structure interaction coupling is a very important but difficult problem, which is likely to give rise to a host of interesting questions concerning stability, well posedness of the numerical schemes used and their accuracy.

In relation with the amount of papers published in this area and concerning with the stages above listed there are three challenge topics that deserves special attention in FSI problems:

- the mesh motion strategy,

- the coupling between the fluid and the structure meshes at the interface and its consequence over the shared variables, [2]

- the time integration strategy or the partitioned scheme used to guarantee time accuracy [3, 4, 5].

The first one, associated with the moving mesh, is commonly named as Computational Mesh Dynamics (CMD), that is needed when ALE-based approaches are adopted. Although the moving mesh is only an artificial field in the coupled three-field FSI problem (CFD+CSD+CMD) it strongly influences the performance, robustness and also the accuracy of the overall approach. Its importance is also proven through the number of approaches that have been proposed in recent literature [6, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14]. Some criteria for a good CMD approach are the following:

- avoid re-meshing as long as possible;
- keep a good mesh quality over the whole simulation time;
- be able to deal with heavily inhomogeneous and anisotropic meshes;
- contain no problem dependent parameter;
- works for 2D as well as for 3D.

In this paper the mesh motion strategy is only addressed leaving the others two items for future work. Being the goal of this work to assess the performance of this new strategy of mesh movement and in order to reduce the presentation without loosing generality in the conclusions the structural problem is oversimplified and only a user defined structural motion is used. So the three field problem is reduced to a two field one because the computational structure

dynamics is omitted. One of the engineering problems of interest that may be simulated with these assumptions are in-cylinder flow in internal combustion engines where the piston and the valves motion are not mostly altered by the gas flow.

The mesh motion strategy presented here is based on an optimization problem using some sort of mesh quality indicator as the objective function to be maximized. The mesh quality is evaluated by two measures, the first one, a shape factor, takes into account the element distortion. The last one is the volume dilatation, an isotropic size factor needed to avoid the shrinking of elements satisfying the first requirement. Both indicators may be included in the functional with some weights according to the problem. The element distortion may be represented in different ways, one of them may be to take the maximum difference of the eigenvalues of the metric tensor built between the real geometry and an idealized one. Other equivalent way, used in this paper, is to take some definition of the dimensionless element aspect ratio, built as the ratio between the element volume and the corresponding power of the edges lengths. The former has the disadvantage of being expensive numerically because it requires the computation of the derivatives of the eigenvalues of the metric tensor in order to compute the jacobian matrix for the optimization problem. Doing an analogy with elasticity, the optimization problem used to move the mesh presented in this paper may be thought as the equivalent elastic energy minimization defined to reach the equilibrium elastic solution. While for the elasticity case the solution obtained depends on the known material constitutive law, for the mesh movement this law is not so obvious. For example, mesh-moving methods based on using different ratios of the Lame constants were introduced in [15] with results reported as a function of that ratio, and mesh-moving methods based on using different values for the "stiffening power" involved in the Jacobian-based stiffening approach were reported in [16] as a function of that stiffening power. The computational mesh dynamics strategy presented here is now included within an object oriented multi-physics code that is able to solve incompressible and compressible, [17, 18, 19] single phase and multi-phase [20], viscous fluid flow moving grid problems (http://venus.ceride.gov.ar/petscfem) [21]. Currently free surfaces problems are also addressed. The CFD code is based on a stabilized SUPG-PSPG finite element formulation [22, 23] for the incompressible flow case and a shock capturing SUPG finite element formulation for compressible one [24, 25]. The nonlinear system is solved by a Newton-like scheme and a preconditioned domain decomposition technique is used to solve the linear system arising at each Newton iteration [26, 27]. The paper is organized as follows: the next section is devoted to the continuum mechanics problem definition with the balance equations and the boundary conditions for the incompressible flow case. Extensions to compressible flow and multi-phase flow problems are straightforward and they are not included in this work. The following section presents the numerical scheme used and some details about the stabilization technique employed. The next section deals with the mesh motion strategy and finally there is a section with several numerical examples. Because the main goal of this paper is to assess the strategy for the mesh dynamics most of the examples presented involve only the mesh dynamic without coupling with any other field. These examples are well known benchmarks available at the literature. At the end of this section there are other examples where the fluid flow in a variable domain is solved. Conclusions and future work end the paper.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2000; **00**:1–6

2. A two field fluid-structure interaction problem. Governing equations

This section deals with the mathematical modeling of the turbulent incompressible fluid flow defined over a variable domain using a Large Eddy Simulation (LES) turbulence model.

This is organized in the following form: it starts with the mathematical description of the continuity and momentum equations, follows with the description of the turbulence model and finishes with the boundary conditions used for such an applications.

2.1. Mass and momentum equations

Consider an incompressible fluid immersed inside a variable geometry domain defined through a user prescribed boundary motion. This class of problems may be classified as the simplest Fluid-Structure Interaction one because the coupling is only in one way. The fluid motion does not produce any change in the structure motion. The motion of the fluid is governed by the incompressible Navier-Stokes equations that may be written following an ALE description in the following way:

$$\nabla \cdot \mathbf{u} = \mathbf{0} \quad in \quad \Omega(t) \times (0, T)$$
$$\rho(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla \mathbf{u}) - \nabla \cdot \sigma = \mathbf{0} \quad in \quad \Omega(t) \times (0, T), \tag{1}$$

with ρ and $\mathbf{u}$ the density and velocity of the fluid and σ the stress tensor, given by

$$\sigma = -p\mathbf{I} + 2\mu_{eff}\epsilon(\mathbf{u})$$
$$\epsilon(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + (\nabla \mathbf{u})^t) \tag{2}$$

where p and $\mu_{eff} = \rho \nu_{eff}$ are pressure and the effective dynamic viscosity proportional to the effective kinematic viscosity defined below, $\mathbf{I}$ represents the identity matrix and ϵ the deformation tensor.

$\hat{\mathbf{u}}$ represents the mesh velocity due to the boundary motion. This velocity arises from the computation of the mesh motion to be explained in detail later.

2.2. Turbulence modeling via Large Eddy Simulation

The LES form a family of turbulence models being the Smagorinsky one of the most popular choices. In this eddy viscosity model we define the turbulent kinetic viscosity as:

$$\nu_t = (C_s \Delta h)^2 \sqrt{2\epsilon(\mathbf{u}) : \epsilon(\mathbf{u})} \tag{3}$$

with $C_s = 0.18$ the Smagorinsky constant, Δ a damping function to reduce the amount of turbulent viscosity in the vecinity of solid objects inmersed in the fluid flow. h is the grid size, a parameter that divide the size of vortices being resolved and the sizes of vortices being modeled. One the main advantages of LES models is that the added viscosity tends to reduce when the grid is refined and in the limit of direct numerical simulation (DNS) the viscosity tends to zero.

Finally $\sqrt{\epsilon(\mathbf{u}) : \epsilon(\mathbf{u})}$ represents the trace of the strain velocity tensor making the eddy viscosity a local parameter.

2.3. Boundary conditions

The continuum formulation is completed by the initial and boundary conditions. Relative to the boundary conditions, except for the pressure, we may split the whole boundary in parts, imposing in each part a Dirichlet condition, a Neumann condition or a near wall boundary condition. For the velocity field we have

$$
\begin{aligned}
\Gamma &= \Gamma_g \cup \Gamma_h \cup \Gamma_{wall} \\
\Gamma_g \cap \Gamma_h \cap \Gamma_{wall} &= \emptyset \\
\mathbf{u} &= \mathbf{g} \qquad \text{on } \Gamma_g \\
\mathbf{n} \cdot \sigma &= \mathbf{h} \qquad \text{on } \Gamma_h \\
\mathbf{n} \cdot \sigma &= \mathbf{h}_{wall}(u_*(\mathbf{u})) \qquad \text{on } \Gamma_{wall}
\end{aligned}
\tag{4}
$$

The pressure should be fixed at a reference value in at least one node in order to remove the corresponding rigid mode in the numerical computation.

For the initial conditions we initialize the computation with a particular field for each unknown that depends on the simulation.

3. Finite element formulation

Even though finite volume is an standard for CFD during the last years there have been a lot of work where finite elements has been chosen as the numerical method with success. Currently both of them are well established within the CFD community. In this work finite element was adopted and in the following section its mathematical formulation is presented in the context of ALE problems.

3.1. Finite element formulation for incompressible Navier-Stokes equations

The functional spaces for the weight and interpolation functions are defined as follow:

$$
\begin{aligned}
S_{\mathbf{u}}^h &= \{\mathbf{u}^h | \mathbf{u}^h \in (H^{1h})^{nsd}, \mathbf{u}^h \dot{=} \mathbf{g}^h \text{ in } \Gamma_g\} \\
V_{\mathbf{u}}^h &= \{\mathbf{N}^h | \mathbf{N}^h \in (H^{1h})^{nsd}, \mathbf{N}^h \dot{=} \mathbf{0} \text{ in } \Gamma_g\} \\
S_p^h &= \{q^h | q^h \in H^{1h}\} \\
V_p^h &= \{p^h | p^h \in H^{1h}\}
\end{aligned}
\tag{5}
$$

with

$$
H^{1h} = \left\{ \phi^h | \phi^h \in C^0(\overline{\Omega}), \phi^h |_{\Omega^e} \in P^1, \forall \Omega^e \in \mathcal{E} \right\}
$$

the Sobolev space, P^1 is a first order polynomial set, nsd is the dimension of the physical domain $\Omega = \cup \Omega^e$, $\mathcal{E}$ means the discrete partition of the physical domain being Ω^e the part of this partition corresponding to the element e.

The *SUPG-PSPG* formulation of (1) is written from [28, 29] as:
Find $\mathbf{u}^h \in S_u^h$ and $p^h \in S_p^h$ satisfying

$$
\int_\Omega \mathbf{N}^h \cdot \rho\Big(\frac{\partial \mathbf{u}^h}{\partial t} + (\mathbf{u}^h - \hat{\mathbf{u}}^h) \cdot \nabla \mathbf{u}^h\Big) + \int_\Omega \epsilon(\mathbf{N}^h) : \sigma^h d\Omega +
$$

$$
+ \underbrace{\sum_{e=1}^{nel} \int_\Omega \delta^h \cdot \Big[\rho(\frac{\partial \mathbf{u}^h}{\partial t} + (\mathbf{u}^h - \hat{\mathbf{u}}) \cdot \nabla \mathbf{u}^h) - \nabla \cdot \sigma^h\Big] d\Omega}_{(\text{SUPG})} +
$$

$$
+ \underbrace{\sum_{e=1}^{nel} \int_\Omega \epsilon^h \cdot \Big[\rho(\frac{\partial \mathbf{u}^h}{\partial t} + (\mathbf{u}^h - \hat{\mathbf{u}}^h) \cdot \nabla \mathbf{u}^h) - \nabla \cdot \sigma^h\Big] d\Omega}_{(\text{PSPG})} + \qquad (6)
$$

$$
+ \sum_{e=1}^{nel} \int_\Omega \nu_{\text{LSIC}} \nabla \cdot \mathbf{N}^h \rho \nabla \cdot \mathbf{u}^h d\Omega + \int_\Omega q^h \nabla \cdot \mathbf{u}^h d\Omega = \int_{\Gamma_h} \mathbf{N}^h \cdot \mathbf{h}^h d\Gamma
$$

$$
\forall \mathbf{N}^h \in V_u^h\,,\,\forall q^h \in V_p^h
$$

The stabilization parameters are defined as:

$$
\delta^h = \tau_{\text{SUPG}}((\mathbf{u}^h - \hat{\mathbf{u}}^h) \cdot \nabla)\mathbf{N}^h\,,
$$

$$
\epsilon^h = \tau_{\text{PSPG}}\frac{1}{\rho}\nabla q^h\ ,
$$

$$
\tau_{\text{SUPG}} = \frac{h}{2||\mathbf{u}^h - \hat{\mathbf{u}}^h||}z(Re_u) \qquad (7)
$$

$$
\tau_{\text{PSPG}} = \tau_{\text{SUPG}}
$$

$$
\nu_{\text{LSIC}} = \frac{h}{2}||\mathbf{u}^h - \hat{\mathbf{u}}^h||z(Re_u)
$$

with Re_u and Re_U are the Reynolds numbers based on the element parameters, i.e.:

$$
Re_u = \frac{||\mathbf{u}^h - \hat{\mathbf{u}}^h||h}{2\nu} \qquad (8)
$$

The element length h is computed by:

$$
h = 2\Big(\sum_{a=1}^{nen} |\mathbf{s} \cdot \nabla w_a|\Big)^{-1} \qquad (9)
$$

with w_a the shape function associated with the a node, nen the number of nodes in the element, and $\mathbf{s}$ a unit normalized velocity vector relative to the mesh velocity. The function $z(Re)$ used in (8) is defined as:

$$
z(Re) = \begin{cases} Re/3 & 0 \le Re < 3, \\ 1 & 3 \le Re \end{cases} \qquad (10)
$$

The spatial discretization leads to the following system of equations:

$$(\mathbf{M} + \mathbf{M}_\delta)\mathbf{a} + \mathbf{N}(\mathbf{v}) + \mathbf{N}_\delta(\mathbf{v}) + (\mathbf{K} + \mathbf{K}_\delta)\mathbf{v} - (\mathbf{G} - \mathbf{G}_\delta)\mathbf{p} = \mathbf{F} + \mathbf{F}_\delta$$
$$\mathbf{G}^t\mathbf{v} + \mathbf{M}_\epsilon\mathbf{a} + \mathbf{N}_\epsilon(\mathbf{v}) + \mathbf{K}_\epsilon\mathbf{v} + \mathbf{G}_\epsilon\mathbf{p} = \mathbf{E} + \mathbf{E}_\epsilon \tag{11}$$

where $\mathbf{M}, \mathbf{M}_\delta$ and $\mathbf{M}_\epsilon$ are the Galerkin, SUPG and PSPG mass matrices, $\mathbf{N}(\mathbf{v}), \mathbf{N}_\delta(\mathbf{v})$ and $\mathbf{N}_\epsilon(\mathbf{v})$ are the corresponding nonlinear convective matrices, $\mathbf{K}, \mathbf{K}_\delta$ and $\mathbf{K}_\epsilon$ are the physical and numerical diffusive matrices and $\mathbf{G}, \mathbf{G}_\delta$ and $\mathbf{G}_\epsilon$ are the matrices for the gradient discrete version. $\mathbf{a}$ and $\mathbf{v}$ are the acceleration and velocities nodal vector respectively. Finally $\mathbf{E}, \mathbf{E}_\epsilon$ and $\mathbf{F}, \mathbf{F}_\delta$ are the source and boundary conditions terms for mass and momentum balance equations.

4. Mesh motion strategy

In the simulation of fluid-structure interaction the mesh is deformed when the boundaries are moved by any reason. One reason may be that the structure that play the role of fluid boundary is deformed by forces exerted by the fluid or is rigidly moved by the same forces. Other reason may be that some mechanical devices control the fluid boundary motion imposing a priori its time evolution, like intake and exhaust valves in an internal combustion engine. Regardless of the reason of mesh movement, the solution of the governing Navier-Stokes equations requires a new grid with updated nodal coordinates and the corresponding nodal speed at every time step. Regeneration of the grid is computationally expensive but mainly a source of accuracy degradation for the numerical solution because it requires a projection. For small relative displacements there are a lot of different mesh motion algorithms that may be used but when the displacements are large or the time step adopted is large relative to the structure dynamics some of them fails giving invalid meshes. Therefore, it is necessary to design the mesh motion strategy as robust as possible allowing to give also good quality meshes.

4.1. The optimization problem

In this work a new mesh motion strategy is presented. It is based on the solution of an optimization problem where the cost function is defined in terms of some indicator of the mesh quality.

It is assumed that the topology of the mesh (element connectivity) is kept fixed and the coordinates of the nodes are found by minimizing some functional. In its more general form the functional adopted is written as

$$F = F(\{x_j^\alpha\}) \tag{12}$$

where x_j^α is the α coordinate of the j node, and $\{x_j^\alpha\}$ represents the whole set of mesh coordinates. Meshes that are computed using pseudo-elastic material strategies can be put in this form, where F is the elastic energy functional. However there is enough freedom to *design* the functional in order to obtain better meshes. Some design conditions for F are

- F should be computed from element contributions (as in a normal finite element assembly process).

- The minimum of F should give the best quality mesh.
- F should be well behaved enough so as to be able to solve the minimization problem with Newton-like methods. In general, it will require F to have continuous first derivatives.
- F should be convex in order to guarantee unicity of the minimum and positivity of the *stiffness* matrices (the Hessian of the functional).

The first item requires that the functional may be computed as

$$F = g(F_1(\{x_j^\alpha\}_1), F_2(\{x_j^\alpha\}_2), \dots) \tag{13}$$

where $F_e(\{x_j^\alpha\}_e)$ is the functional for element e which is a function of the coordinates of its nodes and $g()$ some associative function that preserves convexity, for instance the sum or the maximum

$$\begin{aligned}
g(F_1, F_2, ...) &= F_1 + F_2 + ... \\
g(F_1, F_2, ...) &= \max(F_1, F_2, ...)
\end{aligned} \tag{14}$$

The second requirement is somewhat in conflict with the third and four ones. In the ideal case the F_e would be some indicator of the badness of the element, and $g()$ would be the maximum of its arguments, so that minimizing the mesh functional is equivalent to search for the mesh whose badness (i.e. the badness of the worst element) is minimum. However, using the maximum for $g()$ leads to functionals with non differentiable first derivatives, so functionals of the form

$$F = \sum_e |F_e(\{x_j^\alpha\}_e)|^p, \tag{15}$$

will be consider, preserving regularity, while for $p \to \infty$ the maximum (L_∞) criterion is recovered.

Figure 1. Invariance under dilatation.

Regarding the design of the element functional F_e itself, at first sight it should be a function of its deformation only, so that it should be invariant under dilatation, translation or rotation. However, the corresponding functional would be non-convex. Consider, for instance, a one dimensional problem covering the interval $[0, 1]$ with two linear elements, as in figure 1. There are three nodes, from which the position of nodes 1 and 3 are fixed by the boundary conditions and the only unknown is the position of node 2. Now, if the functional is invariant under dilatation, then the functional of all 1D elements would be the same, and the position of the node 2 would be undetermined. In order to regularize the problem a term depending on the volume is added to the functional as

$$F = C_v \left(\frac{V}{V_{ref}} - 1 \right)^m + C_q q^n \tag{16}$$

where V is the element volume, V_{ref} is the target element volume, q is any element-wise quality indicator, C_q and C_v are weight coefficients for shape and size terms in the functional. m must

be even and $n < 0$ in order to put the optimization problem as a minimization problem giving more weight in the functional to those elements with larger distortion ($q \to 0$). Normally, C_v should be kept as small as possible, while keeping the well posedness of the problem. m and n allow to use different discrete norms to measure the element distortion and the element size change, i.e. $n = 2$ means Euclidean norm and $n \to \infty$ means maximum norm. Now the quality factor q can honour the mentioned invariance.
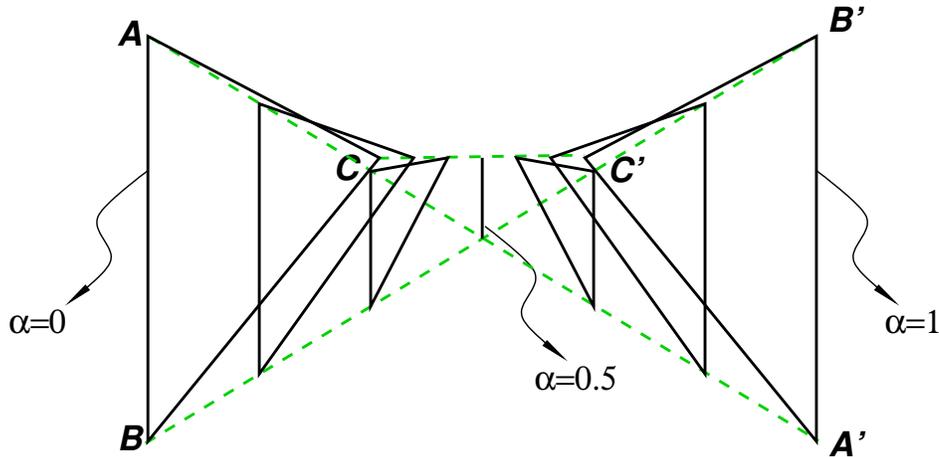
Figure 2. Preservation of convexity.

Regarding the convexity of the functional, this is perhaps the most difficult restriction to accomplish, and in fact it is in contradiction with the invariance under rotations. Consider for instance the case of the two triangles $T = ABC$ and $T' = A'B'C'$ in figure 2. Both of them are well oriented (counterclockwise) and are not to far from an equilateral one, so that they should have a relatively low functional value (badness). Consider the linear path that connects both of them, i.e. the family of triangles $T(\alpha)$ that are formed by linear interpolation of the coordinates of both of them as, for instance for the A vertex

$$x_A(\alpha) = (1 - \alpha) x_A + \alpha x_{A'}. \tag{17}$$

The triangles for $\alpha = 0, 0.25, 0.5, 0.75$ and 1 are shown in the figure. If the functional were convex, it should satisfy the inequality

$$F(T(\alpha)) \leq (1 - \alpha)F(T) + \alpha F(T'), \tag{18}$$

so that the badness of the interpolated triangles for $0 < \alpha < 1$ should be lower than the extreme ones, i.e. they should be *nicer*. But as it is seen in the figure, for $\alpha = 0.5$ the triangles collapses in a line, so that it seems that there is not an appropriate criterion of badness that would be convex. Then the convexity requirement on the functional is drooped. This means that it is not guaranteed the existence and unicity of the solution. It is possible to see, however, that the element functional chosen is convex in a neighborhood of the equilateral element.

In this way the coordinate α component of the j node (x_j^α) may be written as its initial grid coordinate plus a deformation u_j^α. Deriving the functional F respect to u_j^α the following expression for the residue is obtained:

Copyright © 2000 John Wiley & Sons, Ltd.
Prepared using **nmeauth.cls**

Int. J. Numer. Meth. Engng 2000; **00**:1–6

$$R_j^\alpha = \frac{\partial F}{\partial u_j^\alpha} = \frac{mC_v}{V_{ref}} \left(\frac{V}{V_{ref}} - 1 \right)^{m-1} \frac{\partial V}{\partial u_j^\alpha} + nC_q q^{n-1} \frac{\partial q}{\partial u_j^\alpha} \tag{19}$$

The jacobian matrix is given by:

$$K_{jk}^{\alpha\beta} = \frac{\partial}{\partial u_k^\beta} \left(\frac{\partial F}{\partial u_j^\alpha} \right) = \frac{mC_v}{V_{ref}} \left(\frac{V}{V_{ref}} - 1 \right)^{m-1} \left[\frac{m-1}{V_{ref}} \frac{\partial V}{\partial u_k^\beta} \frac{\partial V}{\partial u_j^\alpha} + \frac{\partial}{\partial u_k^\beta} \left(\frac{\partial V}{\partial u_j^\alpha} \right) \right] +$$
$$+ \quad nC_q q^{n-1} \left[\frac{n-1}{q} \frac{\partial q}{\partial u_k^\beta} \frac{\partial q}{\partial u_j^\alpha} + \frac{\partial}{\partial u_k^\beta} \left(\frac{\partial q}{\partial u_j^\alpha} \right) \right] \tag{20}$$

In this paper the quality indicator adopted for simplicial elements is the ratio between its volume and the sum of the edges length to a power given by the dimension of the real space nsd, scaled by a constant C taken in such a way so as to give a unit value for the ideal element. This indicator is dimensionless by definition and its final expression may be written as:

$$q = \frac{CV}{\sum_i l_i^{nsd}} \tag{21}$$

with $C = 4\sqrt{3}$ for triangles and $C = 36\sqrt{2}$ for tetrahedrons.

According to (19) for the computation of the residual it is necessary to have the expression of the derivative of the quality indicator with respect to the deformation. Therefore, deriving the expression 21 respect to u_j^α, it is obtained the following:

$$\frac{\partial q}{\partial u_j^\alpha} = \frac{C}{\left(\sum_i l_i^{nsd} \right)^2} \left[\frac{\partial V}{\partial u_j^\alpha} \sum_i l_i^{nsd} - V \frac{\partial}{\partial u_j^\alpha} \left(\sum_i l_i^{nsd} \right) \right] \tag{22}$$

Again, deriving the last expression with respect to u_k^β:

$$\frac{\partial}{\partial u_k^\beta} \left(\frac{\partial q}{\partial u_j^\alpha} \right) = -\frac{2}{\sum_i l_i^{nsd}} \frac{\partial}{\partial u_k^\beta} \left(\sum_i l_i^{nsd} \right) \frac{\partial q}{\partial u_j^\alpha} + \frac{C}{\left(\sum_i l_i^{nsd} \right)^2} \left\{ \frac{\partial}{\partial u_k^\beta} \left(\frac{\partial V}{\partial u_j^\alpha} \right) \sum_i l_i^{nsd} \; + \right.$$
$$+ \quad \frac{\partial}{\partial u_k^\beta} \left(\sum_i l_i^{nsd} \right) \frac{\partial V}{\partial u_j^\alpha} - \frac{\partial V}{\partial u_k^\beta} \frac{\partial}{\partial u_j^\alpha} \left(\sum_i l_i^{nsd} \right) \; - \tag{23}$$
$$- \quad V \frac{\partial}{\partial u_k^\beta} \left[\frac{\partial}{\partial u_j^\alpha} \left(\sum_i l_i^{nsd} \right) \right] \right\}$$

As V and $\sum_i l_i^{nsd}$ are functions of the element nodal coordinates, it is algebraically easier to adopt some local coordinate, for example using the nodal distance to a given element node taken as the origin. Adopting the first node as the reference node for every element in the mesh the local coordinates may be defined as:

$$w_j^\alpha = x_{j+1}^\alpha - x_1^\alpha \qquad (24)$$

with $j = 1, ..., nsd$.

This coordinate change allow to simplify the algebraic expressions for the computation of the element volume and the edges length with no increment in complexity for the derivatives expressions. As an example, for V, using the chain rule:

$$\frac{\partial V}{\partial u_j^\alpha} = \frac{\partial V}{\partial w_i^\gamma} \frac{\partial w_i^\gamma}{\partial u_j^\alpha} \qquad (25)$$

From the definition given by (24), it is found that the tensor $\frac{\partial w_i^\gamma}{\partial u_j^\alpha}$ is constant and given by:

$$\frac{\partial w_i^\gamma}{\partial u_j^\alpha} = (\delta_{i+1,j} - \delta_{1j})\delta_{\alpha\gamma} \qquad (26)$$

Applying the chain rule over the equation 25, the second derivatives of the element volume may be computed as:

$$\frac{\partial}{\partial u_k^\beta}\left(\frac{\partial V}{\partial u_j^\alpha}\right) = \frac{\partial}{\partial w_l^\mu}\left(\frac{\partial V}{\partial w_i^\gamma}\right)\frac{\partial w_l^\mu}{\partial u_k^\beta}\frac{\partial w_i^\gamma}{\partial u_j^\alpha} \qquad (27)$$

To get the expressions of the element volume derivatives respect to the local coordinates it is necessary to specify the kind of element to use.

For triangles the area may be computed as:

$$V = \frac{1}{2}\begin{vmatrix} x_1^1 & x_1^2 & 1 \\ x_2^1 & x_2^2 & 1 \\ x_3^1 & x_3^2 & 1 \end{vmatrix}$$

Writing the last expressions in local coordinates:

$$V = \frac{1}{2}\begin{vmatrix} 0 & 0 & 1 \\ x_2^1 - x_1^1 & x_2^2 - x_1^2 & 1 \\ x_3^1 - x_1^1 & x_3^2 - x_1^2 & 1 \end{vmatrix} = \frac{1}{2}\begin{vmatrix} w_1^1 & w_1^2 \\ w_2^1 & w_2^2 \end{vmatrix}$$

Therefore, the element volume expression is simplified to the following:

$$V = \frac{1}{2}(w_1^1 w_2^2 - w_1^2 w_2^1) \qquad (28)$$

Deriving the last equation respect to w_i^γ

$$\frac{\partial V}{\partial w_i^\gamma} = \frac{1}{2}\left[\delta_{\gamma 1}(w_2^2 \delta_{i1} - w_1^2 \delta_{i2}) + (w_1^1 \delta_{i2} - w_2^1 \delta_{i1})\right] \qquad (29)$$

The derivative of equation 29 respect to w_l^μ is written as:

$$\frac{\partial}{\partial w_l^\mu}\left(\frac{\partial V}{\partial w_i^\gamma}\right) = \frac{1}{2}(\delta_{l2}\delta_{i1} - \delta_{l1}\delta_{i2})(\delta_{\gamma1}\delta_{\mu2} - \delta_{\gamma2}\delta_{\mu1}) \tag{30}$$

The summation of the square of the edges length of the triangle in local coordinates looks like:

$$\sum_i l_i^2 = |\mathbf{w}_1|^2 + |\mathbf{w}_2|^2 + |\mathbf{w}_2 - \mathbf{w}_1|^2 \tag{31}$$

where

$$|\mathbf{w}_j|^2 = \sum_{\alpha=1}^{2}\left(w_j^\alpha\right)^2 \tag{32}$$

The derivatives of the equation 31 are given by the following expressions:

$$\frac{\partial}{\partial w_i^\gamma}\left(\sum_i l_i^2\right) = 2\left[w_i^\gamma + (-1)^i(w_2^\gamma - w_1^\gamma)\right] \tag{33}$$

$$\frac{\partial}{\partial w_l^\mu}\left[\frac{\partial}{\partial w_i^\gamma}\left(\sum_i l_i^2\right)\right] = 2\delta_{\mu\gamma}\left[\delta_{il} + (-1)^i(\delta_{2l} - \delta_{1l})\right] \tag{34}$$

For a tetrahedral element the volume may be computed as:

$$V = \frac{1}{6}\begin{vmatrix} x_1^1 & x_1^2 & x_1^3 & 1 \\ x_2^1 & x_2^2 & x_2^3 & 1 \\ x_3^1 & x_3^2 & x_3^3 & 1 \\ x_4^1 & x_4^2 & x_4^3 & 1 \end{vmatrix}$$

This expression is reduced to a third order determinant when local coordinates are introduced.

$$V = \frac{1}{6}\begin{vmatrix} 0 & 0 & 0 & 1 \\ x_2^1 - x_1^1 & x_2^2 - x_1^2 & x_2^3 - x_1^3 & 1 \\ x_3^1 - x_1^1 & x_3^2 - x_1^2 & x_3^3 - x_1^3 & 1 \\ x_4^1 - x_1^1 & x_4^2 - x_1^2 & x_4^3 - x_1^3 & 1 \end{vmatrix} = \frac{1}{6}\begin{vmatrix} w_1^1 & w_1^2 & w_1^3 \\ w_2^1 & w_2^2 & w_2^3 \\ w_3^1 & w_3^2 & w_3^3 \end{vmatrix}$$

The last determinant may be computed using the third order Levi-Civita tensor ϵ_{pqr}.

$$V = \frac{1}{6}\epsilon_{pqr}w_1^p w_2^q w_3^r \tag{35}$$

The summation over the edges in the denominator of the element quality indicator for tetrahedral in local coordinates may be expressed as:

$$\sum_i l_i^3 = |\mathbf{w}_1|^3 + |\mathbf{w}_2|^3 + |\mathbf{w}_3|^3 + |\mathbf{w}_2 - \mathbf{w}_1|^3 + |\mathbf{w}_3 - \mathbf{w}_2|^3 + |\mathbf{w}_1 - \mathbf{w}_3|^3 \tag{36}$$

The derivatives of the volume and the summation over the edges length are given by the following expressions:

$$\frac{\partial V}{\partial w_i^\gamma} = \frac{1}{6}(w_2^q w_3^r \delta_{1i}\delta_{p\gamma} + w_1^p w_3^r \delta_{2i}\delta_{q\gamma} + w_1^p w_2^q \delta_{3i}\delta_{r\gamma}) \tag{37}$$

$$
\begin{aligned}
\frac{\partial}{\partial w_l^\mu}\left(\frac{\partial V}{\partial w_i^\gamma}\right) &= \frac{1}{6}\epsilon_{pqr}[\delta_{1l}\delta_{p\mu}(w_3^r \delta_{2i}\delta_{q\gamma} + w_2^q \delta_{3i}\delta_{r\gamma}) + \delta_{2l}\delta_{q\mu}(w_3^r \delta_{1i}\delta_{q\gamma} + w_1^p \delta_{3i}\delta_{r\gamma}) + \\
&+ \delta_{3l}\delta_{r\mu}(w_2^q \delta_{1i}\delta_{p\gamma} + w_1^p \delta_{2i}\delta_{q\gamma})]
\end{aligned}
\tag{38}
$$

$$\frac{\partial}{\partial w_i^\gamma}\left(\sum_i l_i^3\right) = 3\left[|\mathbf{w}_i|w_i^\gamma - |\mathbf{w}_{i+1} - \mathbf{w}_i|(w_{i+1}^\gamma - w_i^\gamma) + |\mathbf{w}_i - \mathbf{w}_{i-1}|(w_i^\gamma - w_{i-1}^\gamma)\right] \tag{39}$$

$$
\begin{aligned}
\frac{\partial}{\partial w_l^\mu}\left[\frac{\partial}{\partial w_i^\gamma}\left(\sum_i l_i^3\right)\right] &= 3\Big\{\left[|\mathbf{w}_i|^{-1}w_i^\mu w_i^\gamma + |\mathbf{w}_i|\delta_{\mu\gamma}\right]\delta_{il} \;- \\
&- \left[|\mathbf{w}_{i+1} - \mathbf{w}_i|^{-1}(w_{i+1}^\mu - w_i^\mu)(w_{i+1}^\gamma - w_i^\gamma)\; + \right. \\
&+ \left. |\mathbf{w}_{i+1} - \mathbf{w}_i|\delta_{\mu\gamma}\right](\delta_{i+1,l} - \delta_{il}) \;+ \\
&+ \left[|\mathbf{w}_i - \mathbf{w}_{i-1}|^{-1}(w_i^\mu - w_{i-1}^\mu)(w_i^\gamma - w_{i-1}^\gamma)\; + \right. \\
&+ \left. |\mathbf{w}_i - \mathbf{w}_{i-1}|\delta_{\mu\gamma}\right](\delta_{il} - \delta_{i-1,l})\Big\}
\end{aligned}
\tag{40}
$$

4.2. The mesh dynamic problem definition

The equation 1 at the beginning of the governing equations section has included the mesh velocity $\hat{\mathbf{u}}$. For its computation it is necessary to know the mesh dynamics available solving the optimization problem presented in the last section. Therefore the mesh dynamics problem is finally defined as:

$$
\begin{aligned}
\delta\mathbf{x}_k^\beta &= (K_{j\,k}^{\alpha\,\beta})^{-1}R_j^\alpha \\
\delta\mathbf{x}|_\Gamma &= \delta\mathbf{x}_\Gamma(t)
\end{aligned}
\tag{41}
$$

with $\delta\mathbf{x}_\Gamma(t)$ a user defined boundary motion time law. Finally the mesh velocity is defined as

$$\hat{\mathbf{u}} = \frac{\delta\mathbf{x}}{\Delta t} \tag{42}$$

5. Numerical examples

The above presented mesh motion strategy was applied to several test problems, either 2 D or 3 D. A test is considered successful if it achieves a valid mesh at every time step. A valid mesh is defined as one that does not contain any element with negative volume.

The reference volume at the time step $k + 1$ was adopted as being $V_{ref}^e = (V^e)^k$. Values for $C_q = 1$ and $m = 2$ were adopted.

In the next sections the definition of each case together with their results are presented.

5.1. Benchmark 2 D

This first test was proposed by Farhat [6] and it consists of deforming a 2 D triangular mesh. The initial configuration is shown in figure 3. The mesh has 7 nodes and 9 elements.
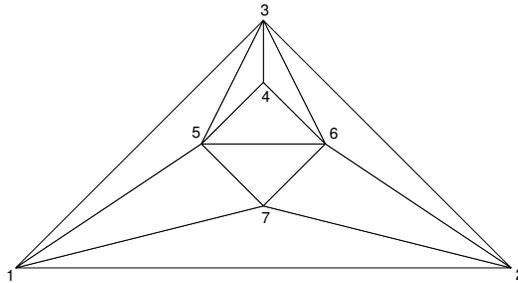
Figure 3. Benchmark 2D - Initial mesh.

Nodes 1 and 2 are kept fixed and the node 3 is displaced vertically downward up to reach the horizontal line that join the first two nodes. The imposed displacement is linear in time. Figure 4 shows the mesh in several time steps. This result was obtained with $C_v = 2$ and $n = -2$. As it is noticeable the mesh motion strategy is able to cope all the deformation proposed giving valid meshes at every time step.

5.2. Benchmark 3 D

This test is a natural extension of the first 2 D test to the three-dimensional space. This test was solved in [7] and it proposes to exert a deformation of a tetrahedral mesh which has an initial configuration as shown in figure 5. The mesh contains 9 nodes and 18 elements.

In this case nodes 1, 2 and 3 are kept fixed and node 4 is displaced towards the plane defined by the first three nodes. The imposed displacement is linear in time. $C_v = 0$ and $n = -1$ values were adopted. As in its 2 D counterpart the strategy presented in this work allows to obtain valid meshes at every time step.

Figure 6 presents some mesh deformation steps. The evolution of the mean value and minimum value of the quality mesh indicator q versus the deformation is presented in figure 7.

Def: 25 %

Def: 51 %

Def: 75 %

Def: 90 %
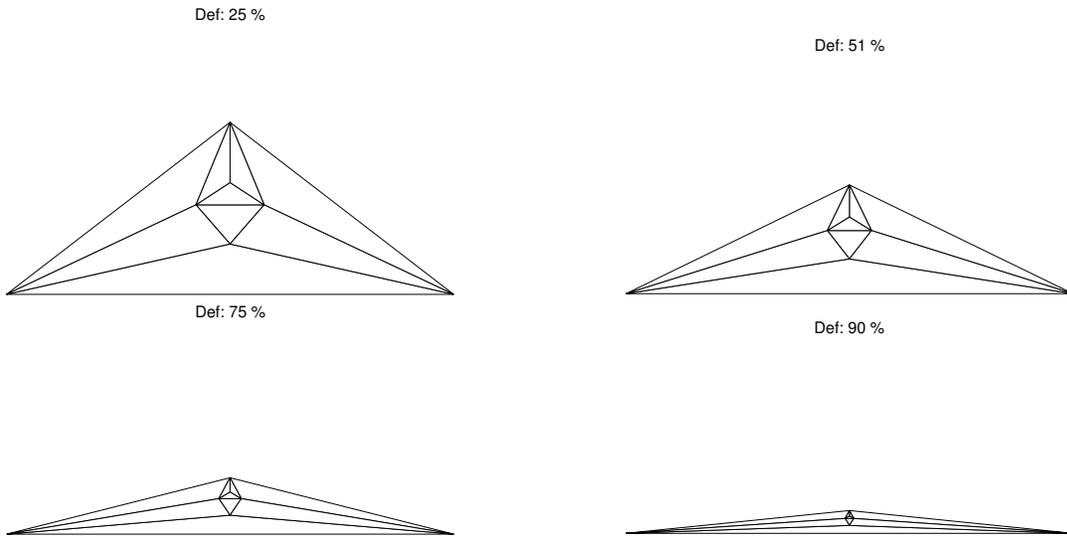
Figure 4. Benchmark 2 D - Deformed mesh.
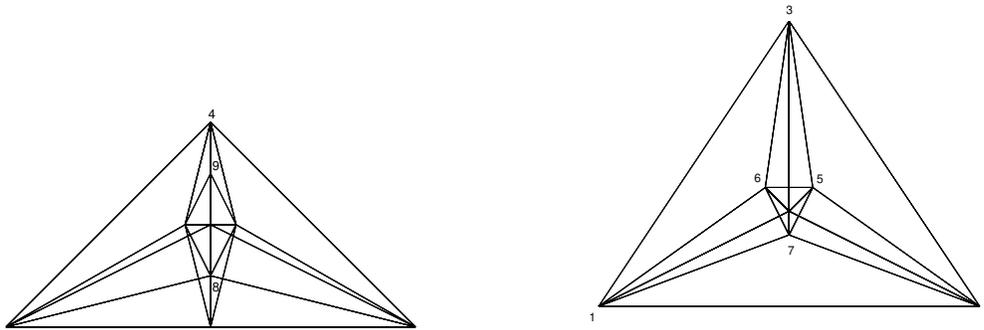
Figure 5. Benchmark 3D - Initial mesh.

5.3. Step 2 D

This example was taken from the gallery of tests of the distribution of *PETSc-FEM* code [21]. Figure 8 sketch the problem domain and the deformation sequence. In figure 9 several meshes obtained during the deformation sequence are shown using for this case the following values: $C_v = 2$ and $n = -1$. In figure 10 the mean value and the minimum value of the element quality indicator q as a function of the deformation were plot. In contrast with the application of standard linear elasticity where the mesh collapses at 50 % deformation, the application of this strategy to this problem makes possible to reach deformations larger than 99 %.

Def: 26 %

Def: 51 %

Def: 75 %

Def: 90 %
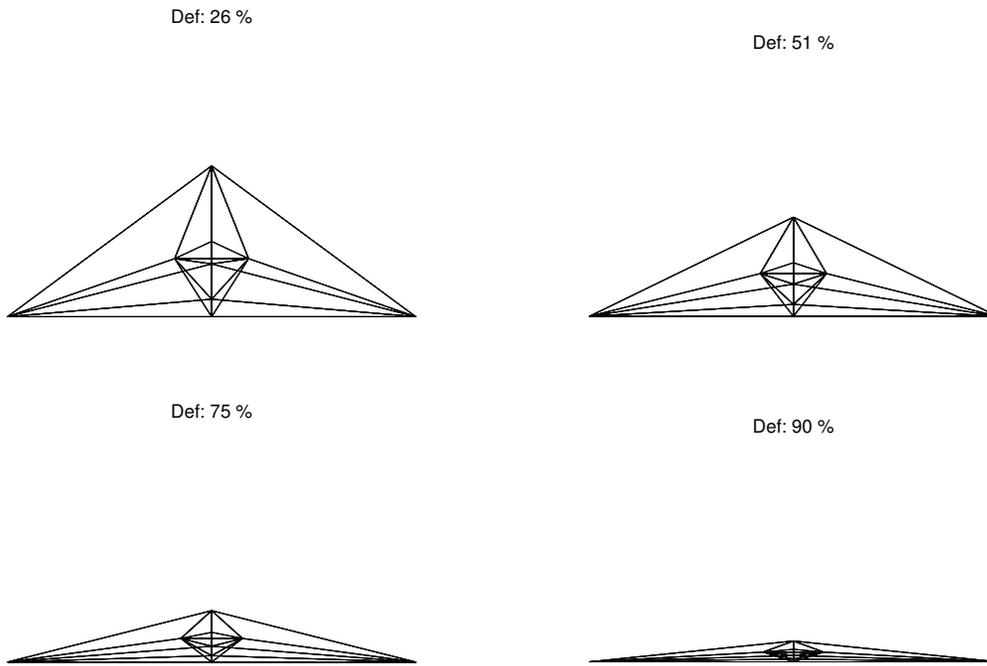
Figure 6. Benchmark 3 D.
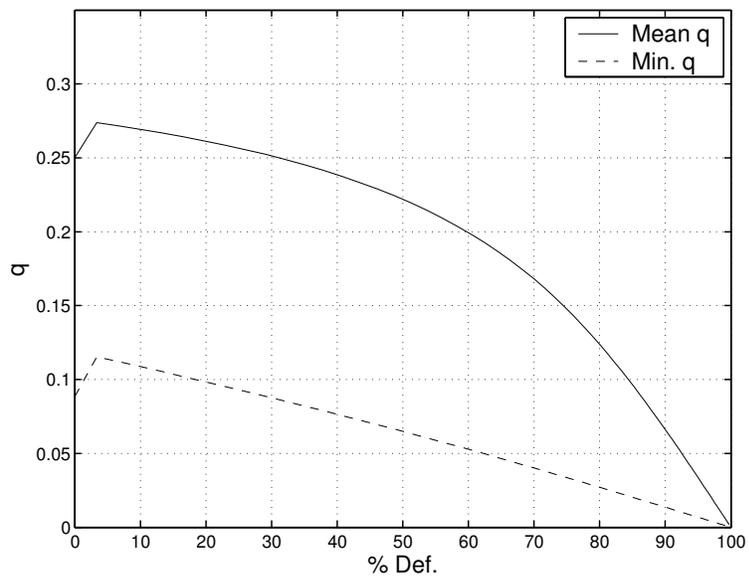
Figure 7. Mesh quality indicators for the 3 D case.
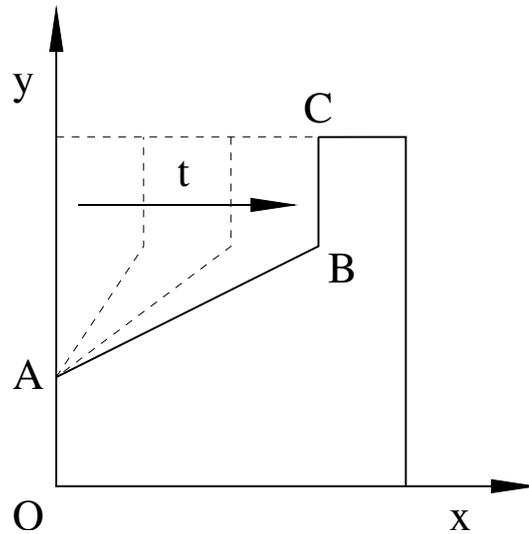
Figure 8. Step 2 D - Domain.

5.4. Step 3 D

This test is the extension of the last case to the three-dimensional space and it is also inside the distribution of *PETSc-FEM*.

This problem is clearly shown in figure 11. Initially the problem domain is defined by an undeformed cube.

The time dependent imposed deformation consists in the displacement of part of the top face of the cube along the vertical direction. This deformation is bounded by the outer region of a circle of radius R. During the time steps of the simulation the top face is transformed in two planes at different height joined by a truncated cone that has an upper radius r and a lower radius of R.

Figure 12 shows the surface mesh at three diferent time steps allowing to see how the moving mesh is transformed.

Employing $C_v = 0$ and $n = -1$ it is possible to reach a deformation of approximately 88 %, with the mean values and minimum values of q varying as shown in figure 13.

5.5. Square within a square test

This test contains two squares, one inside the another, initially centered as it is observed in figure 14.

The internal unit square is displaced towards one of the sides of the external square that has a three unit length without contact. The initial mesh is illustrated in figure 15.

Taking care of moving the boundary in such a way of producing an initial valid mesh for the nonlinear solver at each time step, it is possible to attain deformations larger than 99 %.

Figure 16 shows deformed meshes for different time steps using values for $C_v = 2$ and $n = -2$.

Def: 25 %

Def: 50 %

Def: 75 %

Def: 99 %
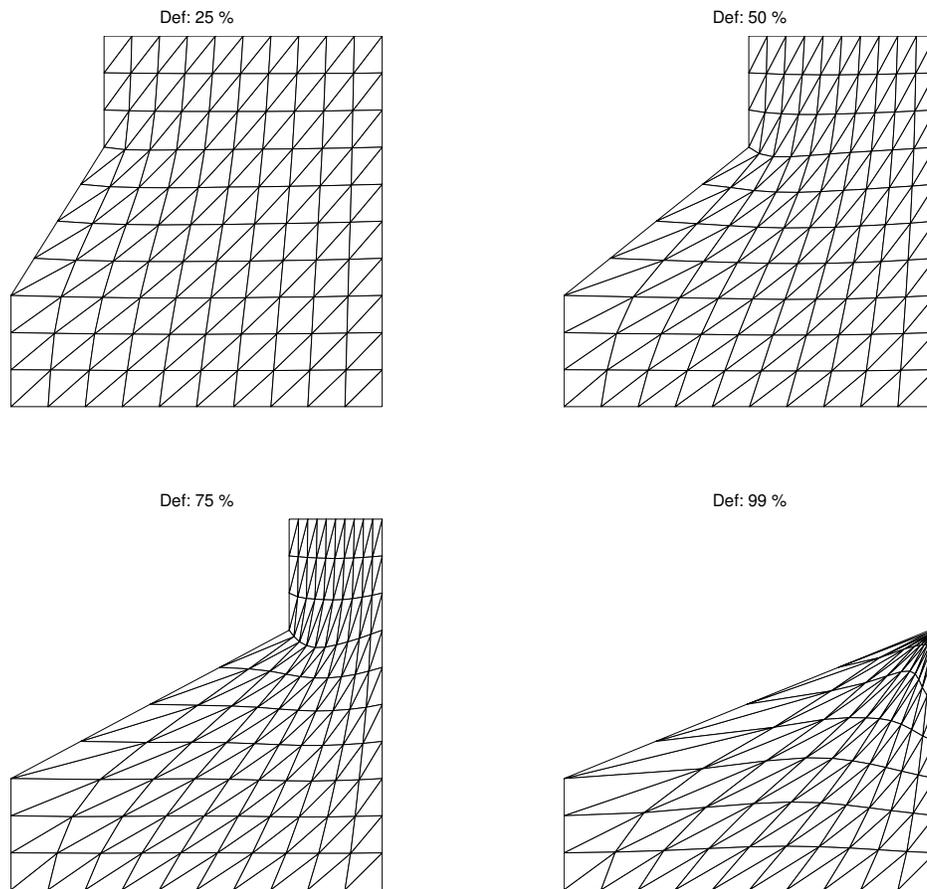
Figure 9. Step 2 D - Deformed mesh.

Figure 17 plots the variation of the mean value and the minimum value of the quality indicator as a function of the deformation.

5.6. Cube inside another cube test

This test case is proposed in [7] and is defined by a unit size cube inside another three unit size cube. Initially the cube centers are coincident. The interior cube is moved upwards without take contact with the external cube.

For this test values of 0 and -1 were adopted for C_v and n respectively. Figure 18 shows the variation of the mean value and minimum value of the quality indicator in terms of the deformation.
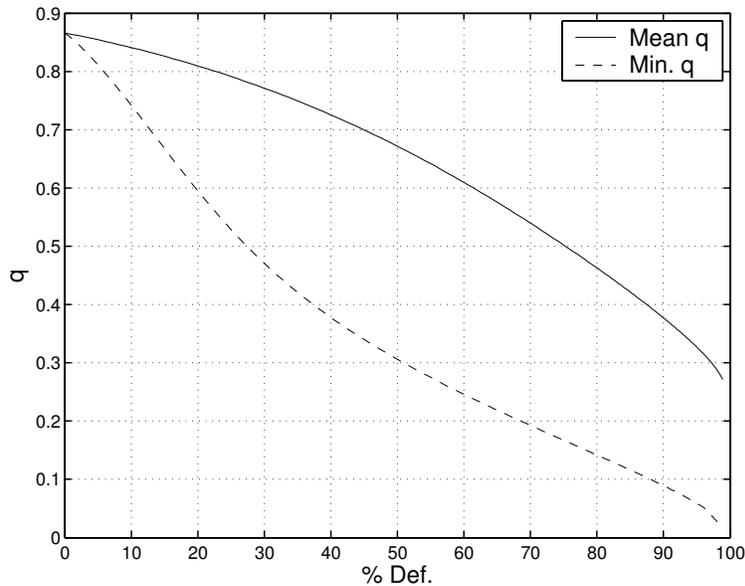
Figure 10. Step 2 D - Quality indicators.

5.7. Translation, rotation and bending 2 D test.

This test [8] concerns with the deformation of a two dimensional unstructured triangular mesh containing a very thin embedded structure, in this case a zero thickness plate was considered. The mesh domain is defined as $|x| \leq 1.0$ and $|y| \leq 1.0$ and the structure is placed in the region defined as $y = 0.0$ and $|x| \leq 0.5$. Around the structure the mesh has three structured layers of elements of width $l_y = 0.01$ with 50 elements along the span-wise of the structure ($l_x = 0.02$). Figure 19 shows the original mesh.

The test involves three different types of structure movements: a rigid translation along direction y, a rigid 2 D rotation around the origin and a prescribed bending. The maximum value for the first test was $\Delta y = 0.5$, for the rotation was $\Delta\theta = \pi/4$ and for the bending the structure adopt a half circle ($\theta = \pi$). For every cases the maximum deformation was adopted after 100 time steps using $C_v = 0$ and $n = -1$ values.

Since the strategy is based on optimizing the element quality in terms of its distortion it is not possible to keep the structured layers of elements around the structure that is normally a necessary condition for capturing accurate boundary layers flow patterns. Therefore, the nodes inside these structured layers were moved rigidly with the structure.

Figure 20 shows the meshes for the time step of maximum deformation for each case. Also it is included a detail of the region in the vicinity of the structure.

Figure 21 plots the evolution of the mean value and the minimum value of the mesh indicator q.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using **nmeauth.cls**

Int. J. Numer. Meth. Engng 2000; **00**:1–6
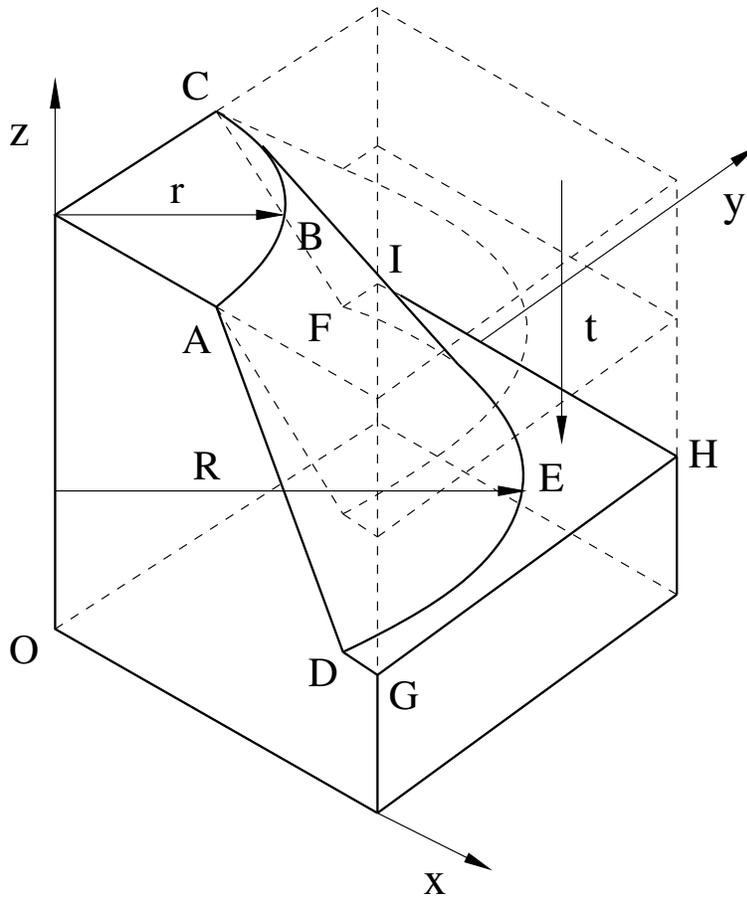
Figure 11. Step 3 D - Problem definition.

5.8. CFD-CMD coupling problem - Indentation test

In this section a two field (CFD-CMD) problem is presented in order to check the ability of the whole numerical scheme to treat flow field with moving boundaries. This problem is defined by a 2 D channel flow with a moving indentation, which has been studied experimentally by Pedley and Stephanoff [30] and numerically by Ralph and Pedley [31] and by Demirdžić and Perić [32]. Here, this example was firstly solved as it was originally defined in [32] and lately it was modified with a more severe indentation blockage in order to check the ability of the numerical solution to simulate the fully closed situation. This kind of situation is normally present in the fluid flow interaction with the valve motion in internal combustion engines.

5.8.1. Description of problem   Figure 22 shows the geometry of the flow domain, with the indentation shape taken from Pedley and Stephanoff [30] and approximated by the following function:
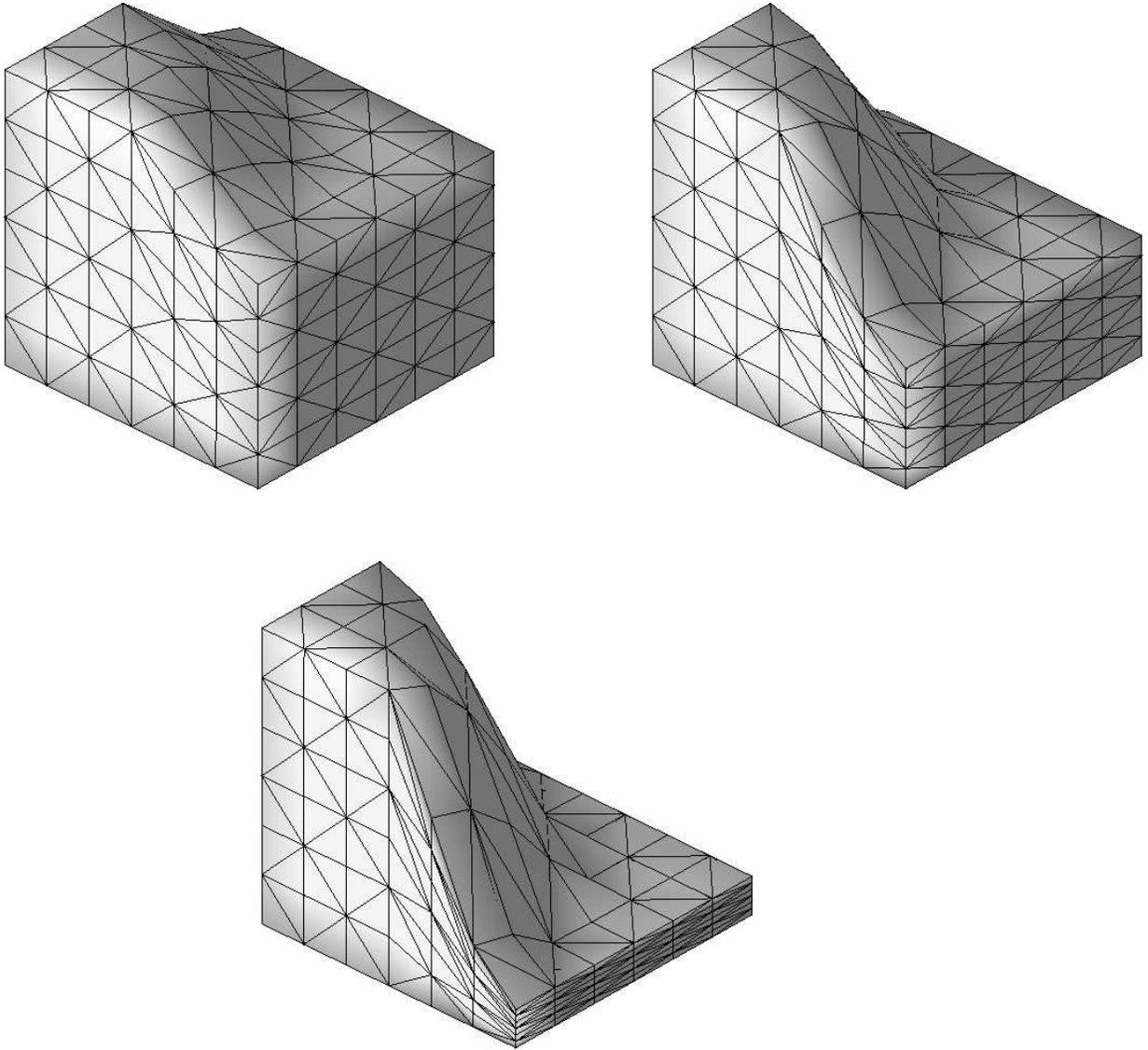
Figure 12. Step 3 D - Moving mesh deformation.

$$y(x) = \begin{cases} h & \text{for } 0 < x < x_1 \\ 0.5h\{1 - \tanh\left[a(x - x_2)\right]\} & \text{for } x_1 < x < x_3 \\ 0 & \text{for } x > x_3 \end{cases}$$

where $a = 4.14$, $x_1 = 4b$, $x_3 = 6.5b$, $x_2 = 0.5(x_1 + x_3)$ and

$$h = 0.5h_{max}\left[1 - \cos(2\pi t^*)\right] \tag{43}$$
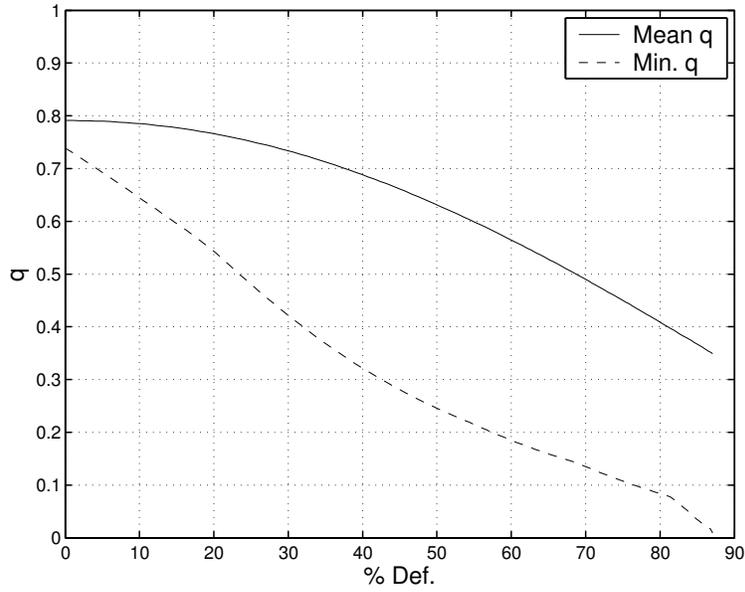
Figure 13. Step 3 D - Quality indicator.
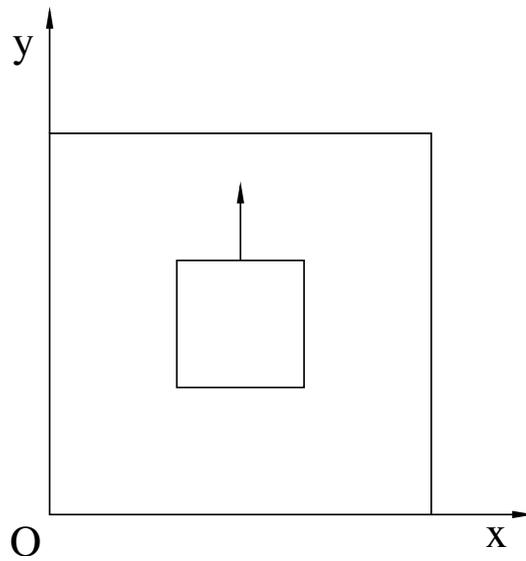
Figure 14. Square within another square test - Definition problem.

being $t^* = \frac{t-t_0}{T}$.

In the last equations b is the channel height, T is the oscillation period and $h_{max} = 0.38b$ specifies the maximum blockage of the channel cross-section at $t^* = 0.5$. The channel geometry
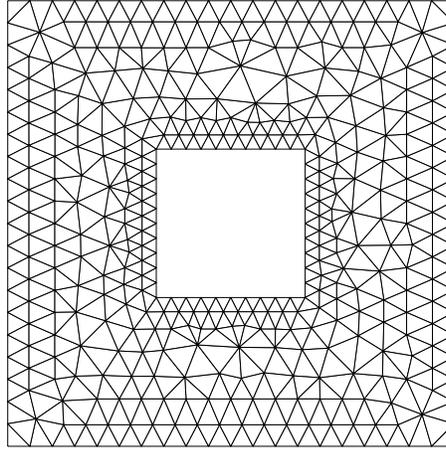
Figure 15. Square within another square test - Initial mesh.

is symmetric around $x = 0$. The Strouhal number, based on the channel height b, bulk velocity U and oscillation period T,

$$St = \frac{b}{UT}, \tag{44}$$

was 0.037 and the Reynolds number, based on the above reference quantities, was 507. At the initial state $t = t_0$ the flow is assumed to be fully developed (Poiseuille flow) with the maximum velocity $u_{max} = 1.5$ m/s. The velocity profile in the entrance section is assumed constant throughout the cycle, which is approximately in accordance with experiment. At the other channel end, zero gradient in the x-direction for both velocity components is taken as the boundary condition. Non-slip boundary conditions were imposed at the channel walls.

5.8.2. Maximum channel blockage: 38 % In the original problem the maximum blockage of the channel is defined as 38 % [32]. Figures 23, 24 and 25 show the pressure, the velocity vector field and the velocity magnitude respectively at $t^* = 0.2$, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 y 1.0. In these figures only the domain region downstream to the indentation is included due to the fact that the geometry aspect ratio is too large and the vortex dynamics upstream to the indentation is too low being the flow field almost unperturbed.

The maximum velocity (2.931 m/s) occurs at $t^* = 0.38$ and it exceeds the maximum velocity of the initial fully developed profile (1.5 m/s) by 95 %. This value is greater than that published by Demirdžić and Perić [32] in almost 20 %. The formation of the first vortex behind the indentation was detected at $t^* = 0.23$. The second vortex appears close to the opposite wall at $t^* = 0.35$, being the distance between these two vortices around two times the channel width. At $t^* = 0.42$ a third vortex arises upstream to the first one. As the time evolves more vortices are formed at both channel walls in an alternated way until approximately $t^* = 0.91$, after this time no more vortices were detected.

Figure 26 show the traction at both channel wall, allowing to correlate the peak values with the vortices magnitude and their motion along the wall.
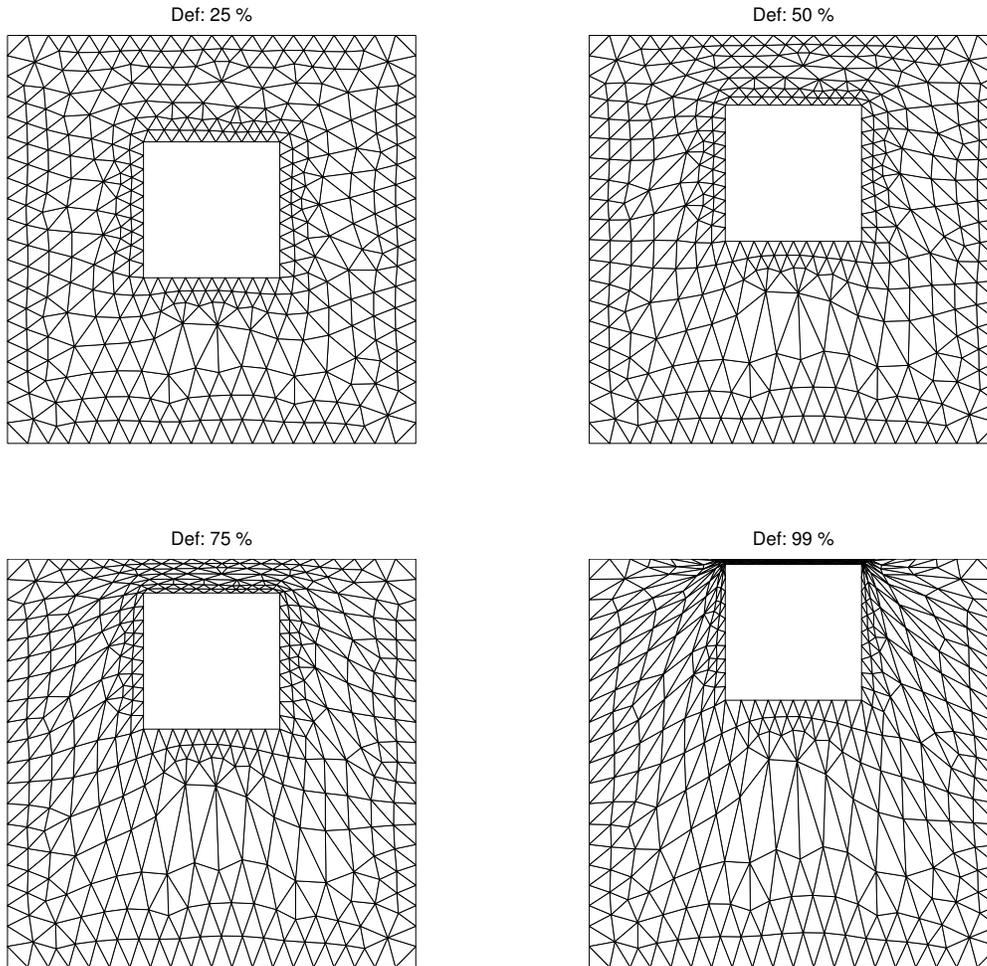
Figure 16. Square within another square test - Deformed mesh.

Figure 27 plots the more intense vortices center position compared with experimental results presented by Pedley and Stephanoff [30]. Following the reference, the x-axis of figure 27 is defined as:

$$x^* = \frac{(x - x_1)(10St)^{1/3}}{b}. \tag{45}$$

5.8.3. Maximum channel blockage: 99 % One of the main goals of every moving grid techniques is to be robust. This means that the technique gives answer in almost all the possible situations. One of the main difficulties in CMD is to generates grids between objects that approximate until their contact. In this case the elements that originally were placed
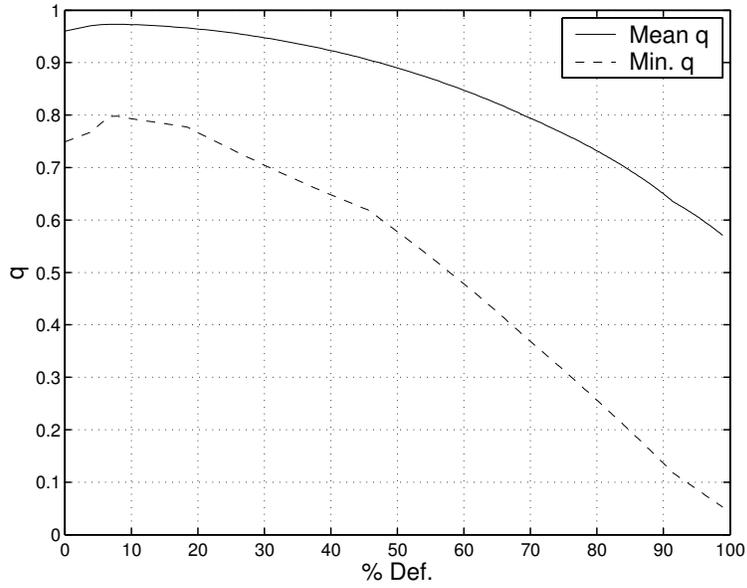
Figure 17. Square within another square test - Quality indicator statistics.
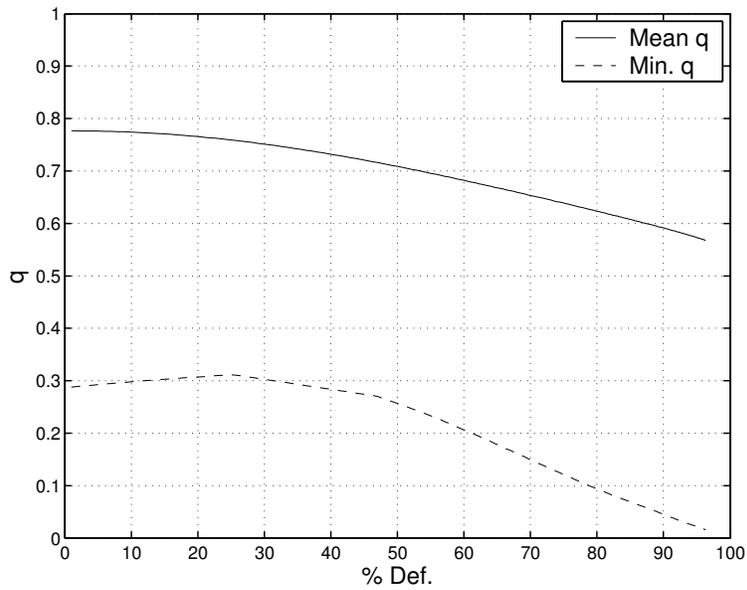
Figure 18. Cube inside another cube test - Quality indicator statistics.

between them should disappear. Numerically this situation is hard to solve and in the spirit of simplification one possible solution to this problem is to consider that the objects approximate

Copyright © 2000 John Wiley & Sons, Ltd.
Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2000; **00**:1–6
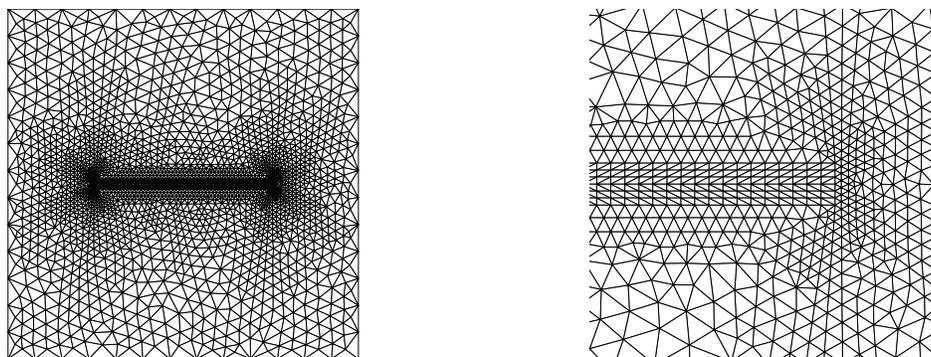
Figure 19. Initial mesh.

each other as much as possible with no contact. One example of our interest for this case is the closure of a valve in an internal combustion engine. This section presents an approximation for this case using the above cited example but with a blockage as large as to simulate the closure. In this example the maximum blockage reached 99 %.

As boundary conditions the pressure with a null tangential velocity at both channel ends were imposed. At the channel walls a non-slip boundary conditions were adopted. The initial conditions were the same as in the original example (see previous section).

The following figures (28) show some plots about the absolute value of velocity and (29) the mesh corresponding to several time steps.

The final figure 30 shows the flow rate versus time. It is noticeable that the closure condition is almost satisfied with this level of blockage.

6. Conclusions

In this paper an optimization based for moving mesh strategy was presented . The functional adopted here, based on some mesh quality indicator, in this case measured in terms of the element distortion, showed to be very robust to follow severe boundary deformations and it seems to be able to be implemented inside of a fluid-structure interaction code.

Even though this strategy seems to be very robust it is not able to recover a valid mesh starting from an invalid one. This improvement is left for future work.

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2000; **00**:1–6

Figure 20. Deformed mesh.

Translation test

Rotation test

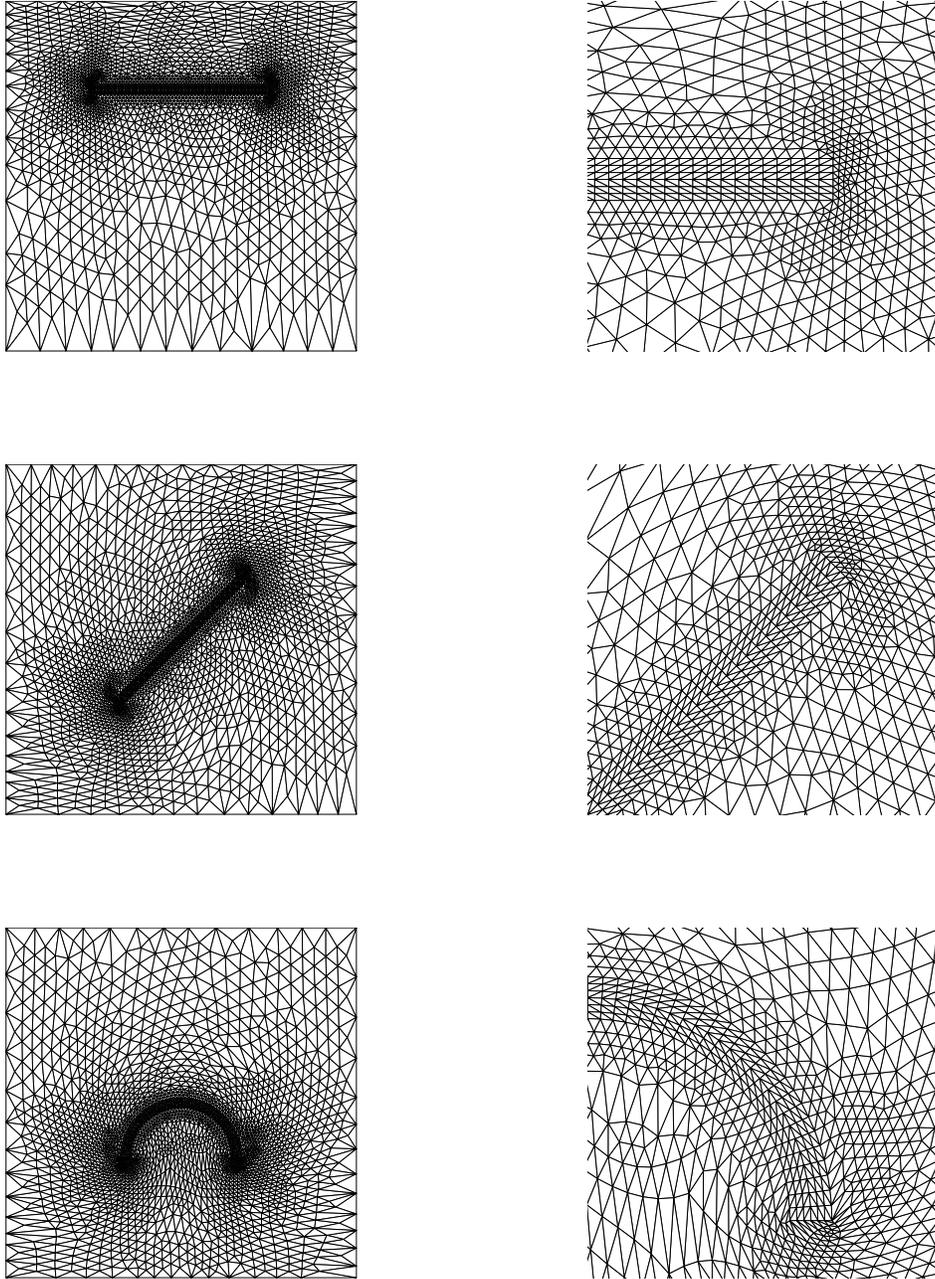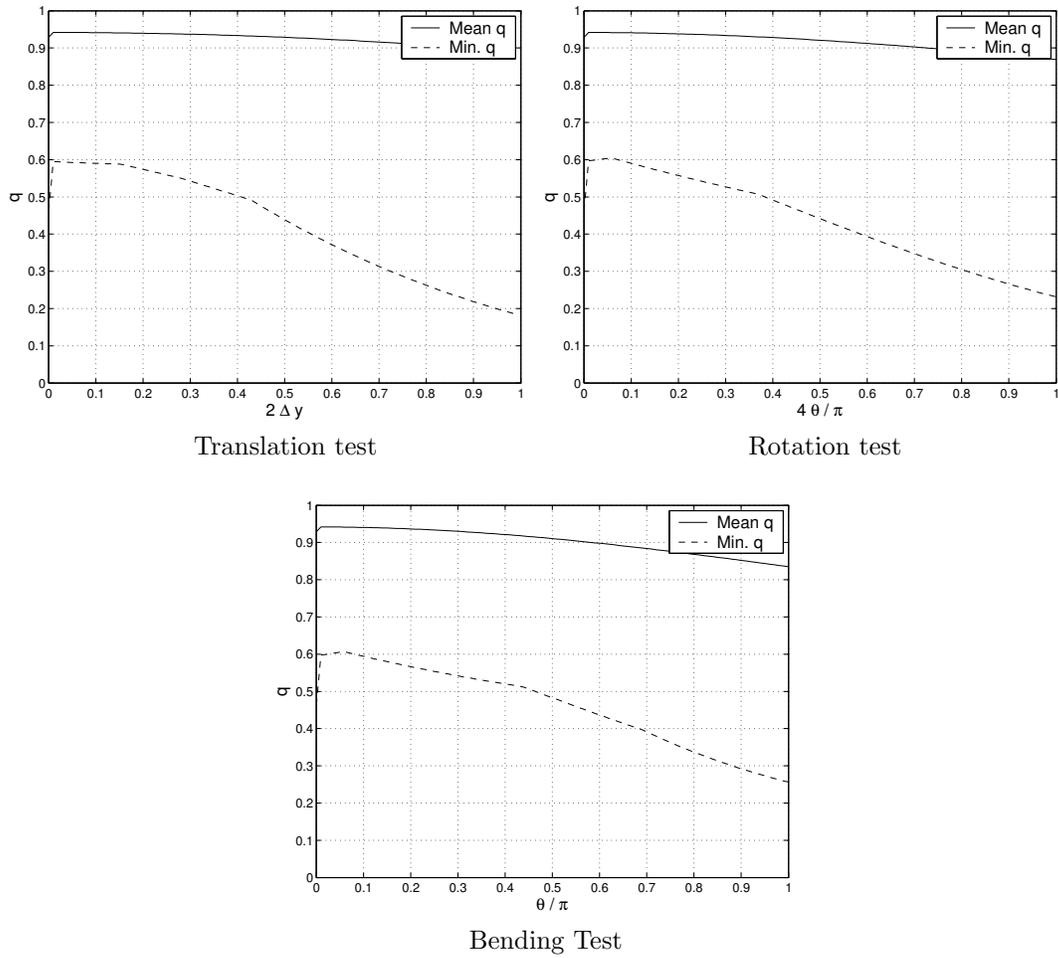Bending Test

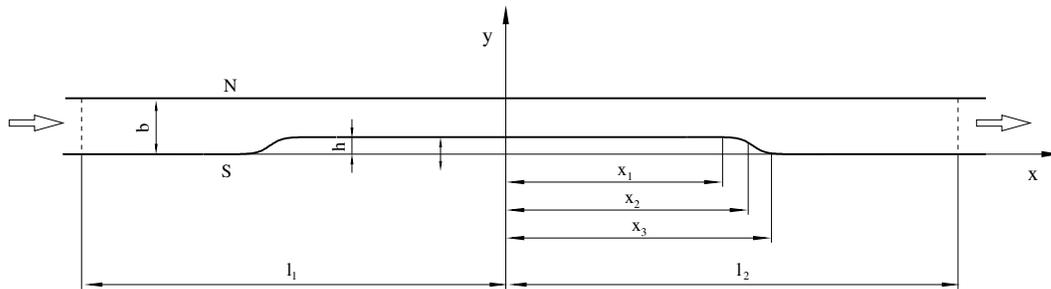Figure 21. Quality indicator.

Figure 22. Geometry of the flow domain (not to scale): $b = 1$ cm, $l_1 = 9.85$ cm, $l_2 = 18.0$ cm.
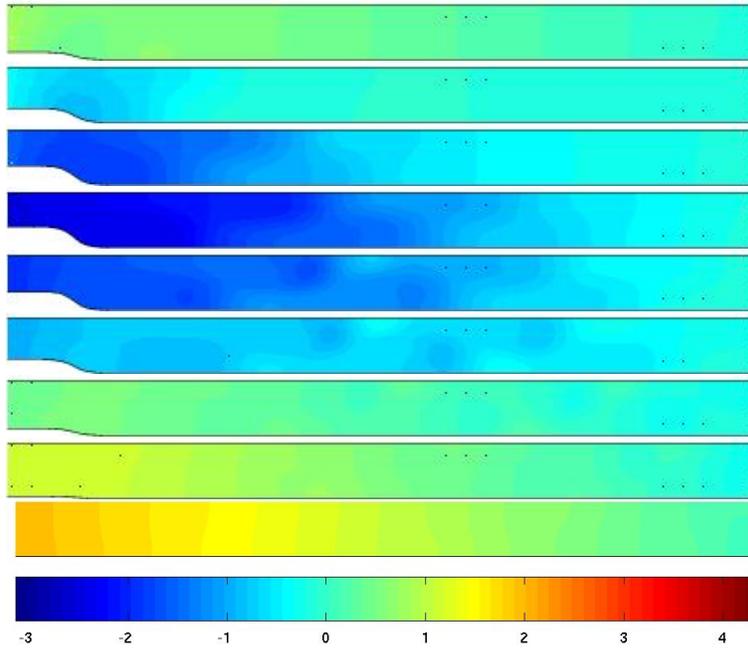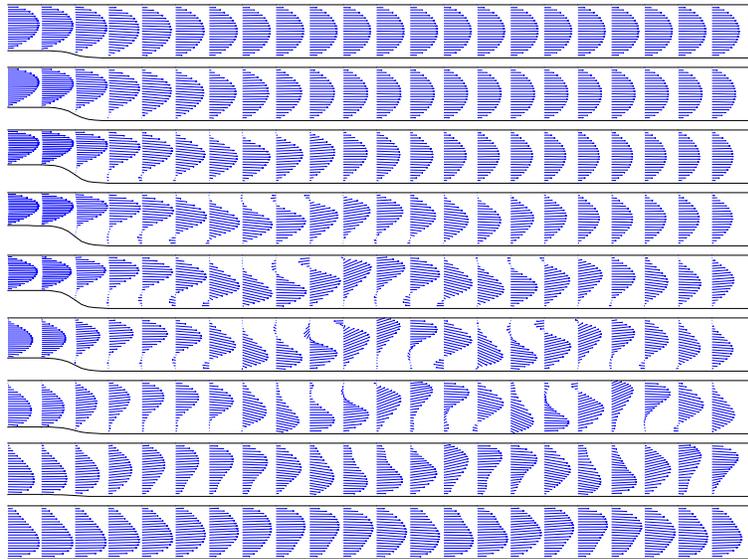
Figure 23. Pressure colormap.
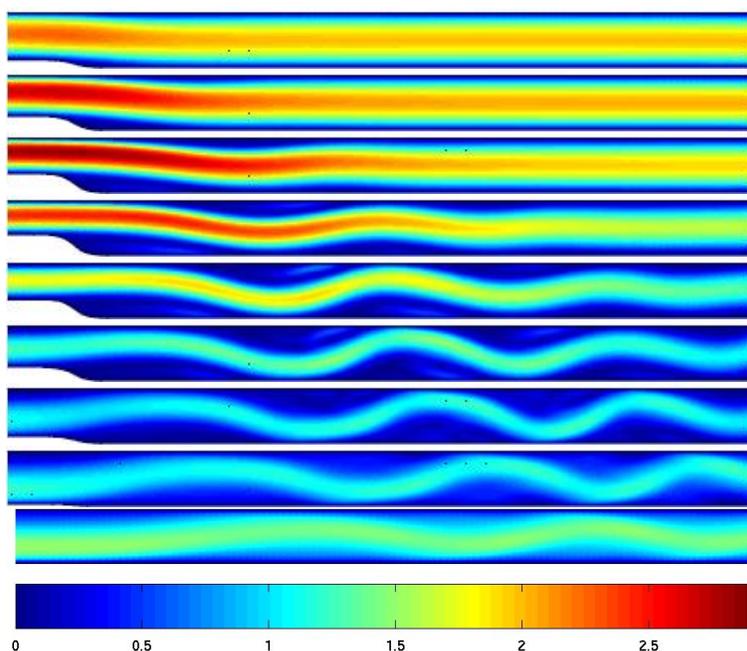

Figure 24. Velocity vector field.

REFERENCES

Figure 25. Velocity magnitude colormap.

1. Benson DJ. Computational methods in Lagrangian and Eulerian hydrocodes. *Computer Methods in Applied Mechanics and Engineering* 1992; **99**(2-3): 235–394.
2. Cebral JR. *Loose Coupling Algorithms for Fluid Structure Interaction.* Ph.D. thesis, Institute for Computational Sciences and Informatics, George Mason University, 1996.
3. Koobus B, Farhat C. Second order time-accurate and geometrically conservative implicit schemes for flow computations on unstructured dynamic meshes. *Computer Methods in Applied Mechanics and Engineering* 1999; **170**(1): 103–129.
4. Farhat C, Geuzaine P, Grandmont C. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *Journal of Computational Physics* 2001; **174**(2): 669–694.
5. Farhat C, Geuzaine P. Design and analysis of robust ALE time-integrators for the solution of unsteady flow problems on moving grids. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**: 4073–4095.
6. Farhat C, Degand C, Koobus B, Lesoinne M. Torsional springs for two-dimensional dynamic unstructured fluid meshes. *Computer Methods in Applied Mechanics and Engineering* 1998; **169**: 231–245.
7. Burg COE. A robust unstructured grid movement strategy using three-dimensional torsional springs. 34th *AIAA Fluid Dynamics Conference and Exhibit* 2004; AIAA Paper 2004-2529.
8. Stein K, Tezduyar TE, Benney R. Automatic mesh update with the solid-extension mesh moving technique. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**: 2019–2032.
9. Bar-Yoseph PZ, Mereu S, Chippada S, Kalro VJ. Automatic monitoring of element shape quality in 2-D and 3-D computational mesh dynamics. *Computational Mechanics* 2001; **27**(5): 378–395.
10. Blom FJ. Considerations on the spring analogy. *Numerical Methods in Fluids* 2000; **32**(6): 647–668.
11. Chiandussi G, Bugeda G, Oñate E. A simple method for automatic update of finite element meshes. *Communications in Numerical Methods in Engineering* 2000; **16**: 1–19.
12. Kjellgren P, Hyvärinen J. An arbitrary Lagrangian-Eulerian finite element method. *Computational Mechanics* 1998; **21**: 81–90.
13. Löhner R, Yang C. Improved ALE mesh velocities for moving bodies. *Communications in Numerical Methods in Engineering* 1996; **12**(10): 599–608.
14. Wall WA. *Fluid-Struktur Interaktion mit stabilisierten Finiten Elementen.* Ph.D. thesis (in German),
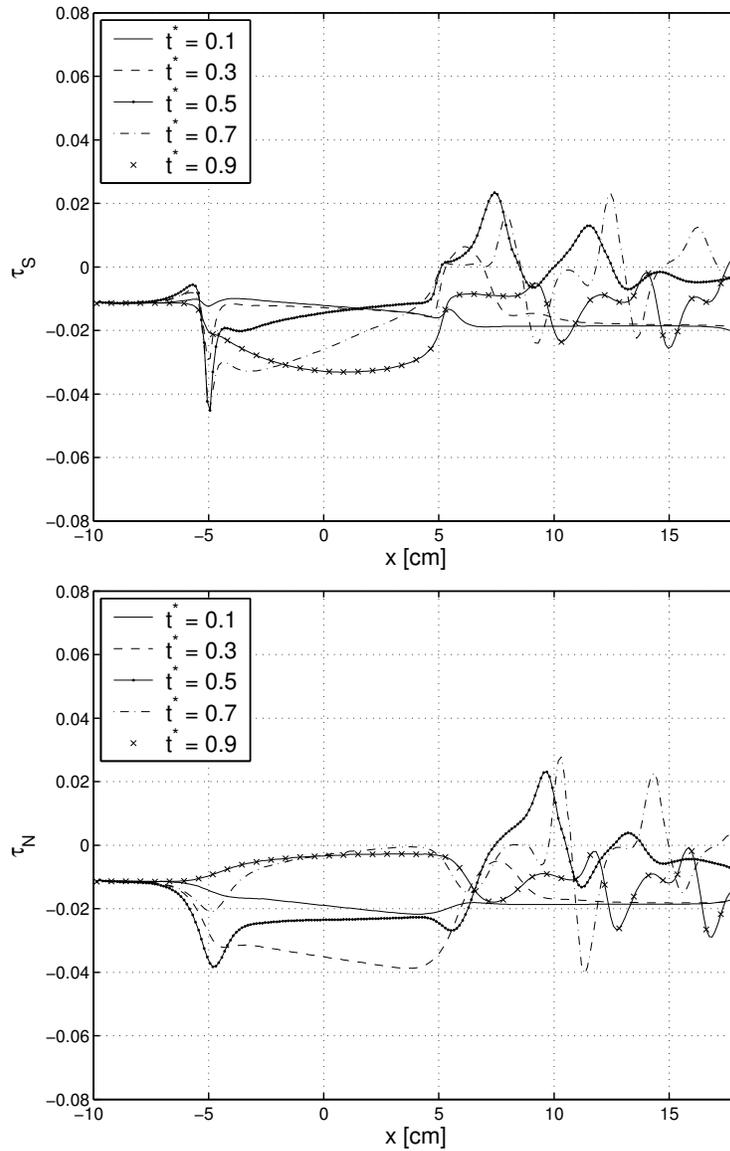
Figure 26. Traction at bottom channel wall (τ_S) and top channel wall (τ_N).

Report No 31, Institute of Structural Mechanics, University of Sttutgart, 1999.

15. Johnson AA, Tezduyar TE. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Computer Methods in Applied Mechanics and Engineering* 1994; **119**(1): 73–94.

16. Stein K, Tezduyar T, Benney. Mesh moving techniques for fluid-structure interactions with large displacements. *Journal of Applied Mechanics* 2003; **70**(1): 58–63.

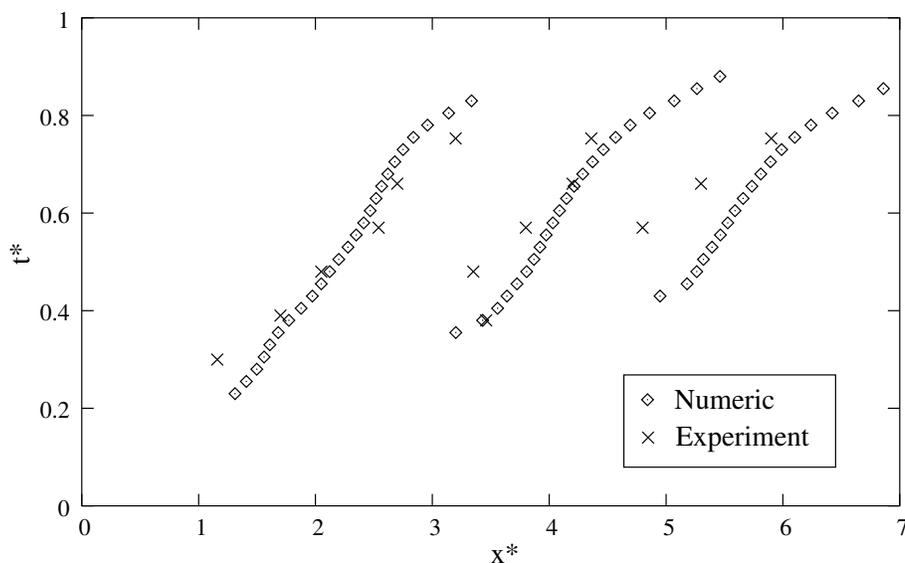17. Nigro N, Storti M, Idelsohn S, Tezduyar T. Physics based GMRES preconditioner for compressible and

Copyright © 2000 John Wiley & Sons, Ltd.

Prepared using nmeauth.cls

Int. J. Numer. Meth. Engng 2000; **00**:1–6

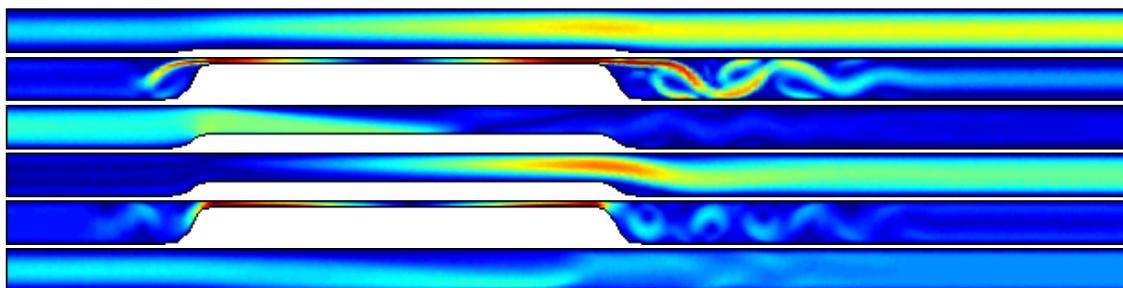Figure 27. Comparison of the vortices center position against the experimental ones.

Figure 28. Velocity magnitude colormap.

incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1998; **154**(3): 203–228.
18. Sonzogni VE, Yommi AM, Nigro NM, Storti MA. A parallel finite element program on a Beowulf cluster. *Advances in Engineering Software* 2002; **33**(7-10): 427–443.
19. Idelsohn S, Storti M, Nigro N, Buscaglia G. A Petrov-Galerkin formulation for advection-reaction-diffusion problems. *Computer Methods in Applied Mechanics and Engineering* 1996; **136**: 27–46.
20. Mendez CG, Nigro N, Cardona A. Drag and non-drag force influences in numerical simulations of metallurgical ladles. *Journal of Materials Processing Technology* 2005; **160**(3): 296–305.
21. Storti M, Nigro N, Paz R. *PETSc-FEM: A General Purpose, Parallel, Multi-Physics FEM Program.* 2004
22. Brooks AN, Hughes TJR. Streamline upwind Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**: 199–259.
23. Tezduyar TE, Mittal S, Ray SE, Shih R. Incompressible flow computations with stabilized bilinear and linear equal order interpolation velocity-pressure elements. *Computer Methods in Applied Mechanics and Engineering* 1992; **95**: 221–242.
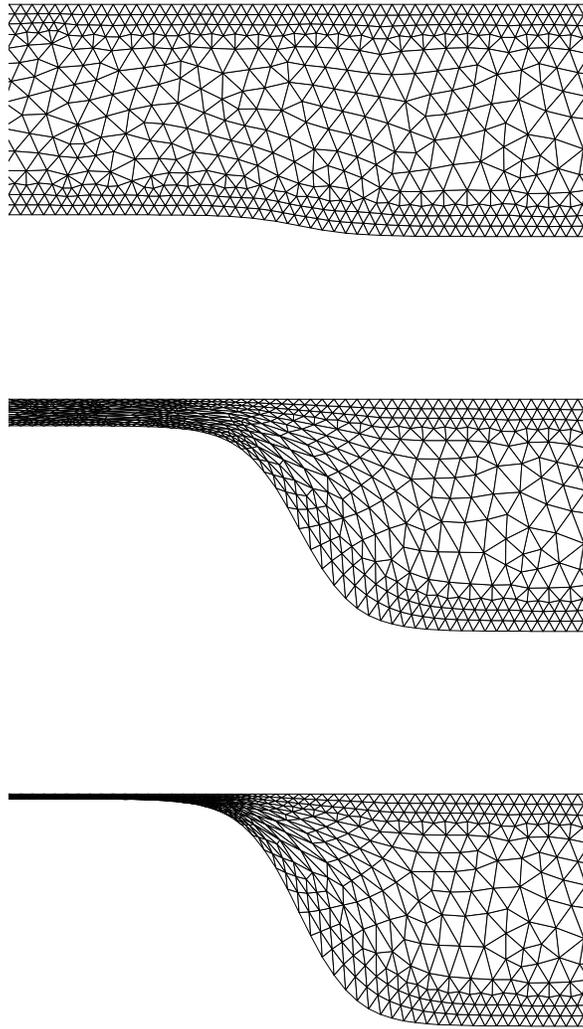
Figure 29. Mesh deformation.

24. Tezduyar TE, Hughes TJR. Finite element formulations for convection dominated flows with particular emphasis on the compressible Euler equations. *Tech. Rep. 0125, AIAA paper* 1983.
25. Hughes TJR, Tezduyar TE. Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations. *Computer Methods in Applied Mechanics and Engineering* 1984; **45**: 218–284.
26. Storti M, Dalcin L, Paz R, Yommi A, Sonzogni V, Nigro N. An interface strip preconditioner for domain
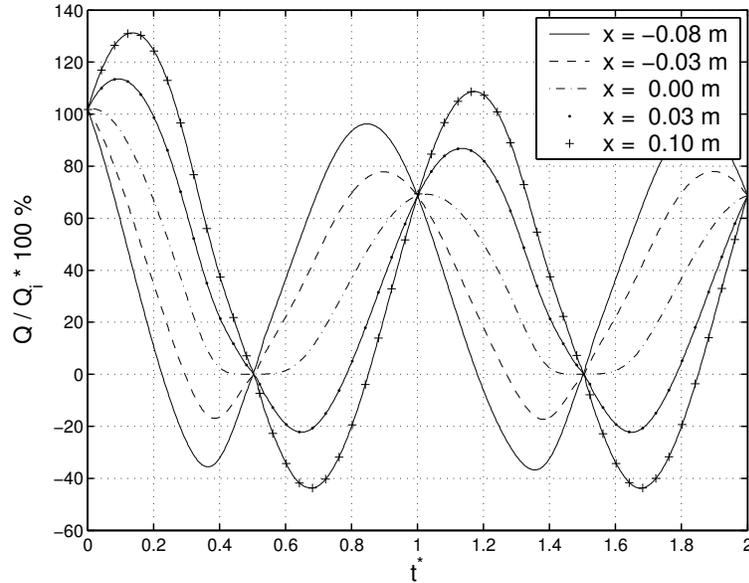
Figure 30. Flow rate at several x stations.

decomposition methods. *submitted to Advances in Engineering Software*; 2004

27. Storti MA, Paz RR. An interface strip preconditioner for domain decomposition methods: application to hydrology. *International Journal for Numerical Methods in Engineering* 2005; **62**(13): 1873–1894.
28. Tezduyar TE. Stabilized finite element formulations for incompressible flow computations. *Advances in Applied Mechanics* 1991; **28**: 1–44.
29. Tezduyar TE. Computation of moving boundaries and interfaces and stabilization parameters. *International Journal for Numerical Methods in Fluids* 2003; **43**(5): 555–575.
30. Pedley TJ, Stephanoff KD. Flow along a channel with a time-depent indentation in one wall: the generation of vorticity waves. *Journal of Fluid Mechanics* 1985; **160**: 337–367.
31. Ralph ME, Pedley TJ. Flow in a channel with a moving indentation. *Journal of Fluid Mechanics* 1988; **190**: 87–112.
32. Demirdžić I, Perić M. Finite volume method for prediction of fluid flow in arbitrarily shaped domains with moving boundaries. *International Journal for Numerical Methods in Fluids* 1990; **10**: 771–790.