INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2004; **44**:673–693 (DOI: 10.1002/fld.667)

A stabilized finite element procedure for turbulent fluid–structure interaction using adaptive time–space refinement

Paulo A. B. de Sampaio[1,*,†], Patrícia H. Hallak[2,‡], Alvaro L. G. A. Coutinho[2,§] and Michéle S. Pfeil[2,¶]

[1] *Divisão de Reatores, Instituto de Engenharia Nuclear-CNEN, CP 68550-CEP 21945-970, Rio de Janeiro, RJ, Brazil*
[2] *Programa de Engenharia Civil, Instituto Alberto Luiz Coimbra-COPPE-UFRJ, CP 68506-CEP 21945-970, Rio de Janeiro, RJ, Brazil*

SUMMARY

This paper presents our effort to addressing fluid–structure interaction (FSI) problems by means of computational mechanics. A stabilized finite element formulation is used to solve the incompressible Navier–Stokes equations written in primitive variables. The structure is modelled using rigid-body dynamic equations solved using a Runge–Kutta method. The distinctive feature of our approach is the combination of large eddy simulation (LES)—based on implicit turbulence modelling—with time–space adaptive techniques in arbitrary Lagrangian Eulerian co-ordinates (ALE). Three representative numerical examples are presented. The first one is the simulation of turbulent vortex shedding around a fixed obstacle, comparing our two-dimensional (2D) LES results with experiments and more refined three-dimensional (3D) numerical solutions. The second example presents an ALE computation with moving boundaries, where we were able to detect the lock-in phenomenon for an oscillating cylinder driven by periodic vortex shedding. Our final example is the FSI problem associated to the flow around a dominant central span section of the Rio-Niterói bridge, where we compare our results with experiments performed in a wind tunnel. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: stabilized formulations; large eddy simulation; fluid–structure interaction; adaptive methods

*Correspondence to: P. A. B. de Sampaio, Divisão de Reatores, Instituto de Engenharia Nuclear-CNEN, CP 68550-CEP 21945-970, Rio de Janeiro, RJ, Brazil.
†E-mail: sampaio@ien.gov.br
‡E-mail: patricia@labest.coc.ufrj.br
§E-mail: alvaro@nacad.ufrj.br
¶E-mail: michele@labest.coc.ufrj.br

Contract/grant sponsor: CNPq; contract/grant number: 350497/1994-0

Received 6 January 2003
Revised 31 August 2003

1. INTRODUCTION

Problems involving fluid–structure interaction (FSI) are important in many branches of engineering. Examples include the vibration of steam generator tubes in nuclear power plants, the aeroelastic behaviour of bridges and other civil engineering structures and the movement of risers and mooring lines in offshore oil production systems. Not long ago, the only approach possible for these problems was an experimental one, based on costly tests performed in wind or water facilities. More recently, with the development of computers and the area of computational fluid dynamics (CFD), engineers and analysts have gained access to complementary numerical tools for understanding, modelling and designing solutions for FSI problems.

As far as a complete and direct simulation of all relevant physical aspects is concerned, the complexity of the problems mentioned above exceeds the computing capacity of the fastest computers available. In the foreseeable future, engineers will still have to resort to ingenuity to combine numerical and experimental techniques in order to face the challenges these problems pose. Nonetheless, considerable progress is being made in the simulation of turbulence with large eddy simulation (LES), in the development of high performance parallel codes and in the use of adaptive techniques to optimize the accuracy of discretizations that are constrained by computational cost.

In this paper, we present our effort to addressing FSI problems by means of computational mechanics. Our approach involves LES with implicit turbulence modelling of the so-called sub-grid scales [1, 2]. This is combined with time–space adaptive techniques based on remeshing and on the use of suitable local time steps [3, 4]. The FSI problems are described using arbitrary Lagrangian–Eulerian co-ordinates to facilitate the implementation of dynamic and kinematic compatibility conditions between fluid and structure [5].

For the sake of clarity, we start presenting the CFD techniques in a Eulerian framework in Section 2. We describe the stabilized finite element formulation and the adaptive techniques employed in the computations. In Section 3, we discuss our particular LES procedure, where no explicit modelling of sub-grid scales is needed [1, 2]. A rigid-body structure and the corresponding equations for rigid-body motion are introduced in Section 4, where our CFD techniques are generalized to arbitrary Lagrangian–Eulerian co-ordinates in order to deal with fluid–structure interaction problems [5]. The numerical examples presented in Section 5 demonstrate the usefulness of the computational techniques developed. Finally, Section 6 contains our concluding remarks.

2. CFD TECHNIQUES

In this section we present the CFD techniques used in the computations. These include our stabilized finite element formulation and the time–space adaptive procedure based on remeshing and on the choice of suitable local time steps.

We start setting the problem in the *continuum* using a Eulerian description of the fluid flow. The problem is defined on the open bounded domain Ω, with boundary Γ, contained in the *nsd*-dimensional Euclidean space. The flow is modelled by the incompressible Navier–Stokes equations, written using the summation convention for $a = 1, \ldots, nsd$ and $b = 1, \ldots, nsd$, in

Cartesian co-ordinates:

$$\rho\left[\frac{\partial u_a}{\partial t}+u_b\frac{\partial u_a}{\partial x_b}\right]-\frac{\partial \tau_{ab}}{\partial x_b}+\frac{\partial p}{\partial x_a}=0 \tag{1}$$

$$\frac{\partial u_a}{\partial x_a}=0 \tag{2}$$

Note that the viscous stress is given by $\tau_{ab}=\mu(\partial u_a/\partial x_b+\partial u_b/\partial x_a)$, where μ is the fluid viscosity. The fluid density is denoted by ρ. The dependent variables are the velocity and pressure fields represented by u_a and p, respectively.

The Navier–Stokes equations are discretized using the stabilized finite element formulation of De Sampaio and Coutinho [4]. The derivation is only sketched here and the reader is referred to the original work for details [4]. The formulation is obtained from a least-squares minimization of the time-discretized momentum balance residual with respect to the velocity and pressure degrees of freedom. In our two-dimensional (2D) computations, we use linear triangles to approximate both velocity and pressure. A pressure-continuity equation, enforcing the mass balance, is obtained combining the standard Galerkin approximation of the continuity equation with the minimization of the momentum squared residual with respect to the pressure degrees of freedom. This has the form of a Poisson equation that avoids the restrictions associated to the Babuška–Brezzi stability condition, allowing the use of equal order interpolation for velocity and pressure in the formulation [4].

Considering that the discretized fields are $\hat{u}_a^{n+1}=N_j u_{aj}^{n+1}$ and $\hat{p}^{n+1}=N_j p_j^{n+1}$, where u_{aj}^{n+1} and p_j^{n+1} are the velocity and pressure nodal values at time-level $n+1$ and N_i are the shape functions, the method takes the form of a Petrov–Galerkin weighted residual approximation of the momentum equation

$$\int_\Omega\left(N_i+\frac{\Delta t}{2}\hat{u}_b^n\frac{\partial N_i}{\partial x_b}\right)\hat{R}_a\,\mathrm{d}\Omega=0\quad\forall\ \text{free}\ u_{ai}^{n+1} \tag{3}$$

combined with the pressure-continuity equation

$$\int_\Omega \Delta t\frac{\partial N_i}{\partial x_a}\hat{R}_a\,\mathrm{d}\Omega+\int_\Omega N_i\rho\frac{\partial \hat{u}_a^{n+1}}{\partial x_a}\,\mathrm{d}\Omega=0\quad\forall\ \text{free}\ p_i^{n+1} \tag{4}$$

where $\hat{R}_a$ denotes the Cartesian components of the residual of momentum:

$$\hat{R}_a=\rho\left(\frac{\hat{u}_a^{n+1}-\hat{u}_a^n}{\Delta t}+\hat{u}_b^n\frac{\partial \hat{u}_a^{n+1/2}}{\partial x_b}\right)-\frac{\partial \hat{\tau}_{ab}^n}{\partial x_b}+\frac{\partial \hat{p}^{n+1/2}}{\partial x_a} \tag{5}$$

2.1. Adaptive methods

Adaptive methods aim to optimize the accuracy of the computation for a given cost [4] or, conversely, to minimize the cost for attaining a required solution quality [6]. In this work, we combine a remeshing scheme with a local time-stepping algorithm for transient problems [4].

2.1.1. Local time-stepping algorithm. In our procedure, we set local time steps according to the expression $\Delta t=\alpha h_e/\|\mathbf{u}^n\|$, where $\alpha=\coth(Re_h/2)-2/Re_h$, and $Re_h=\rho\|\mathbf{u}^n\|h_e/\mu$ is the

element Reynolds number based on the flow velocity $\mathbf{u}$ and on the local element size h_e. There are important reasons for such a choice. First, the weighting function used in the discretized momentum balance, Equation (3), becomes the SUPG weighting function of Hughes and Brooks [7] and a correct amount of *streamline upwinding* is introduced in the formulation. Indeed, the stabilization terms and weighting functions in Equations (3) and (4) have contributions proportional to Δt, and stable results can only be achieved through local definition of Δt according to local values of velocity, physical properties and mesh size. Further, note that the time step defined above is appropriate to follow the time evolution of the convection–diffusion processes resolvable on a mesh with local size h_e. For strong convection ($Re_h \gg 1$) we obtain $\Delta t \to h_e/\|\mathbf{u}^n\|$, whereas for pure diffusion ($Re_h = 0$) we obtain $\Delta t = \rho h_e^2/6\mu$. An obvious difficulty regarding the advance of the solution using local time steps is the fact that the solution at different locations is not synchronized. In this work, we use a simplified form of the algorithm introduced by De Sampaio [3], where the solution is advanced using the local time steps for *optimal* choice of weighting functions, but a time-interpolation procedure is used to synchronize the computation. The local time-stepping algorithm starts with the flow variables defined at time level t^n, i.e. $u_a^n = u_a(t^n)$ and $p^n = p(t^n)$, and then proceeds as follows:

(a) Set element time steps Δt_e using the corresponding element Reynolds number Re_h.
(b) Project the element time steps Δt_e to mesh nodes, obtaining a piecewise linear time-step distribution $\Delta t = \Delta t(\mathbf{x}) = N_j \Delta t_j$, where Δt_j denotes nodal time steps.
(c) Solve the governing equations using the local time-step distribution $\Delta t = \Delta t(\mathbf{x})$. The solution, obtained at time $\tilde{t}(\mathbf{x}) = t^n + \Delta t(\mathbf{x})$, is given by $\tilde{u}_a = u_a(\tilde{t})$ and $\tilde{p} = p(\tilde{t})$. The use of the local time steps guarantees a correct stabilization of the discretized equations used to obtain $\tilde{u}_a = u_a(\tilde{t})$ and $\tilde{p} = p(\tilde{t})$.
(d) Choose the interpolation time level $\Delta t_{\text{int}} = 0.999 \Delta t_{\min}$, where $\Delta t_{\min}$ is the minimum nodal time-step value.
(e) Synchronize the solution at the time level $t^{n+1} = t^n + \Delta t_{\text{int}}$, through interpolation on the time domain, using the solution at $t = t^n$, i.e. $u_a^n = u_a(t^n)$ and $p^n = p(t^n)$, and the solution at $\tilde{t}(\mathbf{x}) = t^n + \Delta t(\mathbf{x})$, given by $\tilde{u}_a = u_a(\tilde{t})$ and $\tilde{p} = p(\tilde{t})$. Thus, the synchronized solution at $t^{n+1} = t^n + \Delta t_{\text{int}}$ is given by $u_a(t^{n+1}) = (1-\gamma) u_a(t^n) + \gamma u_a(\tilde{t})$ and $p(t^{n+1}) = (1-\gamma) p(t^n) + \gamma p(\tilde{t})$, where $\gamma = \Delta t_{\text{int}}/\Delta t$ (note that $0 < \gamma < 1$).
(f) Update variables ($t^{n+1} \to t^n$) and return to step (a) to continue the transient computation.

In our procedure, Δt_{int} plays the same role as the usual Δt in orthodox time-advance schemes, in the sense that we have a synchronized solution whenever $t^{n+1} = t^n + \Delta t_{\text{int}}$.

2.1.2. Adaptive remeshing. Much progress has been achieved in the field of finite element mesh adaptivity. The rapid development is both a consequence of the research on error estimates [8] and of the availability of mesh generator routines capable of using the error data to build improved meshes. In this work, the *a posteriori* error estimator of Zienkiewicz and Zhu [8] is used to estimate the viscous stress error and to guide a remeshing procedure based on equally distributing the estimated error among the finite elements. The strategy, presented in detail in Reference [4], requires the user to specify the aimed number of elements and the minimum element size ($h_{\min}$) to be employed in the computation. Besides limiting the number of elements required to cover any given portion of the domain, the choice of the minimum element size indirectly limits the minimum time-step size, as space and time discretizations

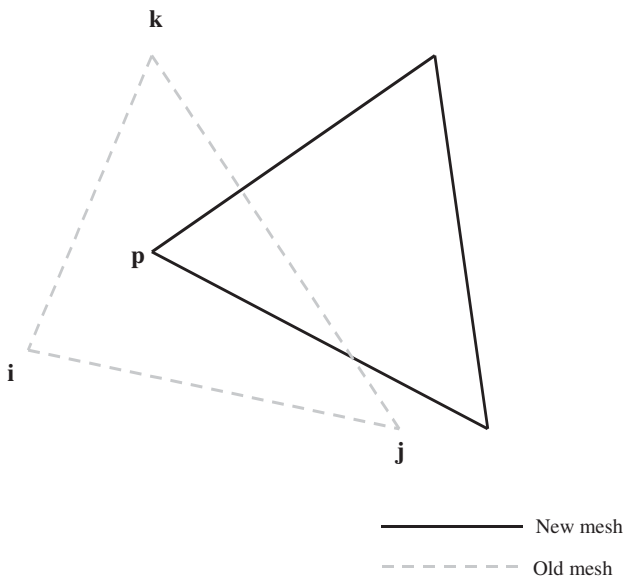


Figure 1. Field values at the new mesh node **p** are interpolated using data from old mesh nodes **i**, **j**, **k**.

are related by the formula $\Delta t = \alpha h_e / \|\mathbf{u}^n\|$. Remeshing is based on the use a coarse background grid that is locally refined with the creation of new points as required. The new points are connected to the nodes of the original background grid using Bowyer's algorithm [9].

The remeshing procedure is fully automatic: a new mesh is generated whenever the estimated viscous stress error varies more than 1% during the transient analysis. This means that the adequacy between mesh and solution is constantly evaluated as the computation proceeds. In practice, a mesh generated according to the solution at a given time step may be used for a number of subsequent time steps, depending on how fast the estimated error changes as the flow departs from the solution to which that mesh has been originally adjusted.

2.1.3. Time–space adaptive procedure. It is important to remark that the local time-stepping algorithm is used in conjunction with the remeshing scheme. This permits linking the spatial and time-step refinement through the equation that defines the local time step, i.e. $\Delta t = \alpha h_e / \|\mathbf{u}^n\|$, resulting in a time–space adaptive procedure. Thus, whenever the remeshing scheme creates some local mesh refinement to better resolve a particular flow feature, the time step is also adapted accordingly.

2.2. Transferring data to a new mesh

An important aspect in a remeshing procedure concerns the interpolation of field data between two consecutive meshes. The problem of transferring data between meshes is illustrated in Figure 1, where the determination of field values at a node on the new mesh requires the interpolation of variable values that are only available on the old mesh.

Referring to Figure 1, let u be a generic variable. Then, the simplest way to compute u at point **p** is through a linear interpolation based on the old mesh shape functions, i.e.

$\hat{u}_p = N_1(\mathbf{p})u_1 + N_2(\mathbf{p})u_2 + N_3(\mathbf{p})u_3$, where u_1, u_2, u_3 are nodal values on the old mesh and N_1, N_2, N_3 are the corresponding shape functions evaluated at point $\mathbf{p}$. However, the repeated use of this linear interpolation to transfer field data introduces dissipation in the numerical solution [6]. Therefore, we employ the interpolation scheme introduced by De Sampaio *et al.* [6], where first derivatives of u, also stored on the old mesh nodes, are used to obtain a correction term Δu_p. The interpolation scheme we use to transfer data between meshes takes then the general form $\tilde{u}_p = \hat{u}_p + \varepsilon \Delta u_p$. For $\varepsilon = 0$ we have linear interpolation, whilst for $\varepsilon = 1$ the interpolation becomes second-order accurate.

The linear interpolation scheme is used when interpolation is required within an element whose size is $h_{\min}$. This is performed because the $h_{\min}$ value is imposed only to render the computation affordable and, most probably, more refinement would be required to capture solution details on regions covered by elements with the minimum size. The choice of the linear interpolation scheme in this case is motivated by the need of damping contributions from spatial scales that cannot be resolved by the computational meshes used in the simulation. On the other hand, the second-order accurate interpolation is used when interpolating within elements larger than the prescribed minimum size $h_{\min}$.

3. LARGE EDDY SIMULATION (LES)

In an LES the large turbulence scales are resolved by the discretization while the small sub-grid scales are taken into account using the so-called sub-grid models [10]. The governing equations are obtained by formally applying a filter function on the original Navier–Stokes system. The filtering operation is shown in Equation (6), where we apply the filter to the velocity field $u_a(\mathbf{x}, t)$. The resulting filtered velocity field $\bar{u}_a(\mathbf{x}, t)$ is given by

$$\bar{u}_a(\mathbf{x}, t) = \int G(\mathbf{x} - \mathbf{x}') u_a(\mathbf{x}', t)\, \mathrm{d}\mathbf{x}' \tag{6}$$

The filter function $G(\mathbf{r})$ integrates to 1 and decays to zero outside a range $|\mathbf{r}| < \mathrm{O}(\Delta)$. The parameter Δ is referred to as the filter width [10].

Starting with the Navier–Stokes equations in conservative form

$$\rho\left[\frac{\partial u_a}{\partial t} + \frac{\partial (u_a u_b)}{\partial x_b}\right] + \frac{\partial p}{\partial x_a} - \frac{\partial \tau_{ab}}{\partial x_b} = 0 \tag{7}$$

$$\frac{\partial u_a}{\partial x_a} = 0 \tag{8}$$

the filtered equations become

$$\rho\left[\frac{\partial \bar{u}_a}{\partial t} + \frac{\partial (\bar{u}_a \bar{u}_b)}{\partial x_b}\right] + \frac{\partial \bar{p}}{\partial x_a} - \frac{\partial \bar{\tau}_{ab}}{\partial x_b} - \frac{\partial \tau^{\mathrm{F}}_{ab}}{\partial x_b} = 0 \tag{9}$$

$$\frac{\partial \bar{u}_a}{\partial x_a} = 0 \tag{10}$$

Note that the filtered momentum equation, Equation (9), can be recast in convective form using the filtered mass balance, Equation (10),

$$\rho\left[\frac{\partial \bar{u}_a}{\partial t}+\bar{u}_b\frac{\partial \bar{u}_a}{\partial x_b}\right]-\frac{\partial \bar{\tau}_{ab}}{\partial x_b}+\frac{\partial \bar{p}}{\partial x_a}-\frac{\partial \tau^{\mathrm{F}}_{ab}}{\partial x_b}=0 \tag{11}$$

The filtering of the non-linear term of the momentum equation, Equation (7), gives rise to a closure problem for the extra stress term τ^{F}_{ab} that appears in Equation (9) or Equation (11). This term is analogous to the Reynolds-stress that arises in the derivation of the Reynolds-averaged Navier–Stokes (RANS) equations [10]. The stress term τ^{F}_{ab} is given by

$$\tau^{\mathrm{F}}_{ab}=\rho(\bar{u}_a\bar{u}_b-\overline{u_a u_b}) \tag{12}$$

At this point, the standard approach in LES is to introduce sub-grid models to express τ^{F}_{ab} in terms of the filtered flow variables. A number of sub-grid closures have been proposed such as the Smagorinsky model [10, 11], for instance. The reader is referred to the work of Sagaut [11] for a comprehensive account on the sub-grid scale models used in LES. A review on the advances in LES methodology for complex flows is presented in Reference [12].

Recently, though, some attempts have been made to let the numerical methods themselves (without resorting to explicit sub-grid models) to express the effect of the unresolvable sub-grid scales on the mean flow [1, 2, 13, 14]. De Sampaio and Coutinho [1] have called this *implicit sub-grid modelling*. What is important to bear in mind is the distinction between such an approach and a direct numerical simulation (DNS) of turbulence, where the time and space discretizations are fine enough to resolve all turbulence scales. Furthermore, note that the design of an LES with an *implicit sub-grid model* becomes the design of the numerical method itself. This includes not only the formulation used to obtain the discretized equations, but also the adaptive schemes and other algorithms that affect the way the unresolvable scales are treated (*implicitly modelled*) by the computation.

Next we shall investigate the relationship between our stabilized finite element formulation for incompressible flows and the more orthodox LES procedures. In particular, we shall highlight the implicit filtering and the implicit sub-grid scale modelling that are embedded in our stabilized formulation.

Let us consider the momentum equation written in conservative form, Equation (7), and a Taylor series to approximate the velocity field at time level $n+1$

$$u_a^{n+1}=u_a^n+\Delta t\frac{\partial u_a^n}{\partial t}+\frac{\Delta t^2}{2}\frac{\partial}{\partial t}\left(\frac{\partial u_a}{\partial t}\right)^{n+1/2} \tag{13}$$

Noting that Equation (7) permits substituting the terms $\partial u_a^n/\partial t$ and $\partial u_a^{n+1/2}/\partial t$ in the above Taylor series we obtain

$$\rho\left(\frac{u_a^{n+1}-u_a^n}{\Delta t}\right)+\rho\frac{\partial (u_a u_b)^n}{\partial x_b}+\frac{\partial p^n}{\partial x_a}-\frac{\partial \tau^n_{ab}}{\partial x_b}+\frac{\Delta t}{2}\frac{\partial}{\partial t}\left[\rho\frac{\partial (u_a u_b)}{\partial x_b}+\frac{\partial p}{\partial x_a}-\frac{\partial \tau_{ab}}{\partial x_b}\right]^{n+1/2}=0 \tag{14}$$

or

$$\rho\left(\frac{u_a^{n+1}-u_a^n}{\Delta t}\right)+\rho\frac{\partial (u_a u_b)^n}{\partial x_b}+\frac{\partial p^{n+1/2}}{\partial x_a}-\frac{\partial \tau^{n+1/2}_{ab}}{\partial x_b}+\frac{\partial}{\partial x_b}\left[\rho\frac{\Delta t}{2}\frac{\partial}{\partial t}(u_a u_b)\right]^{n+1/2}=0 \tag{15}$$

The above time discretization is the basis for the Lax–Wendroff and Taylor–Galerkin methods. In the Lax–Wendroff method Equation (15) is discretized in space with centred finite differences whereas in the Taylor–Galerkin scheme one proceeds discretizing Equation (15) with finite elements and the Galerkin method. In either case the last term on the left-hand side of Equation (15) is responsible for stabilizing the computation, i.e. preventing that errors arising from unresolvable scales destroy the simulation. Most importantly, Equation (15) can be interpreted as a particular time discretization of the filtered momentum equation, Equation (9), where the model chosen for the extra tensor τ_{ab}^{F} is

$$\tau_{ab}^{\mathrm{F}} = -\rho \frac{\Delta t}{2} \frac{\partial}{\partial t}(u_a u_b) \tag{16}$$

It is worth noting that the filtering operation that generates Equation (9) is merely a formal process that permits identifying a contribution τ_{ab}^{F}, arising from sub-grid scales, but that does not indicate possible closures for τ_{ab}^{F}. On the other hand, in the derivation of Equation (15), a Taylor series defines the form of the extra tensor τ_{ab}^{F}, whose presence in the formulation filters the numerical solution.

Introducing a finite element discretization and using the Galerkin method to approximate Equation (15), we obtain [5],

$$\int_{\Omega} \left(N_i + \frac{\Delta t}{2} \hat{u}_b^{n+1/2} \frac{\partial N_i}{\partial x_b} \right) \hat{R}_a^{n+1/2} \, \mathrm{d}\Omega = 0 \tag{17}$$

where

$$R_a^{n+1/2} = \frac{\rho}{\Delta t} (\hat{u}_a^{n+1} - \hat{u}_a^n) + \rho \hat{u}_b^{n+1/2} \frac{\partial \hat{u}_a^{n+1/2}}{\partial x_b} + \frac{\partial \hat{p}^{n+1/2}}{\partial x_a} - \frac{\partial \tau_{ab}^{n+1/2}}{\partial x_b} \tag{18}$$

Note that Equations (17) and (18) are essentially the same equations presented in Section 2, namely Equations (3) and (5). The only difference is the approximation, in our stabilized formulation, of $\hat{u}_a^{n+1/2}$ by $\hat{u}_a^n$. This shows that the stabilized finite element method presented in Section 2 embeds the model for τ_{ab}^{F} given by Equation (16).

It is important to remark that the scheme used to transfer data between meshes, described in Section 2, also plays a part in modelling the effects of the unresolvable scales. We recall that the linear interpolation scheme, used to interpolate data within the small elements with size $h_{\min}$, introduces some extra dissipation when transferring data that is available at the very limit of spatial resolution ($h_{\min}$). On the other hand, as we use a second-order scheme when interpolating data within elements larger than $h_{\min}$, we do not introduce any extra dissipation when dealing with the resolvable scales.

The main reason to adopt implicit sub-grid modelling is to avoid the excessive dissipation introduced on resolvable scales by explicit sub-grid models such as the Smagorinsky closure, for instance. In order to avoid excessive dissipation of resolvable flow features, Hughes *et al.* [15] proposed the *variational multiscale method*, with *a priori* separation between resolvable and unresolvable scales, and where the use of the Smagorinsky model is confined to the latter. In our case, the use of time steps adjusted to the local mesh resolution and physical properties, results in fully exploiting the time–space resolution available, whilst preventing that errors arising from unresolvable scales lead to numerical instability. Of course, the discretization that is required to suitably model a particular turbulent flow will be always problem dependent.

Furthermore, special boundary conditions, incorporating the so-called universal law of the wall, may be required to model turbulent boundary layers. Nonetheless, the method proposed herein is capable of following the evolution of the large/slow features resolvable in a *given discretization*. In this sense, it is an LES method where no explicit sub-grid modelling is introduced.

4. FLUID–STRUCTURE INTERACTION (FSI)

We consider here the rigid-body motion of a structure that is driven by lift forces to oscillate in the direction transverse to the flow. This FSI problem is described using arbitrary Lagrangian–Eulerian co-ordinates in order to facilitate the implementation of dynamic and kinematic compatibility conditions between fluid and structure [5]. The momentum equation, Equation (1), is generalized to the ALE framework as follows:

$$\rho\left[\frac{\mathrm{d}u_a}{\mathrm{d}t}+(u_b-w_b)\frac{\partial u_a}{\partial x_b}\right]-\frac{\partial \tau_{ab}}{\partial x_b}+\frac{\partial p}{\partial x_a}=0 \tag{19}$$

where we have introduced the reference frame velocity $\mathbf{w}$. In particular, note that the time derivative in Equation (19) is the time derivative as viewed by an observer moving with the reference frame velocity $\mathbf{w}$. In practice, $\mathbf{w}$ will be defined conveniently in order to adjust the problem reference frame, from Eulerian ($\mathbf{w}=\mathbf{0}$) far from the moving body, to Lagrangian ($\mathbf{w}=\mathbf{u}$) on the fluid–solid interface. The fluid is considered to be attached to the body (non-slip and impermeability conditions), thus enforcing kinematic compatibility.

The structure exposed to the cross flow is free to move in the vertical direction y, but its movement is restrained in the x-direction. Rotation is also restrained. Thus, the structure one-degree-of-freedom dynamics is described by the equation:

$$m\frac{\mathrm{d}^2y}{\mathrm{d}t^2}+c\frac{\mathrm{d}y}{\mathrm{d}t}+ky=F_{\mathrm{L}} \tag{20}$$

where m, c and k are the mass, damping and stiffness parameters by unit length, respectively. The force F_{L} is responsible for the dynamic coupling between the fluid and the structure rigid-body motion. It is computed from the flow field as

$$F_{\mathrm{L}}=-\int_{\Gamma_c}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)n_x+2\mu\frac{\partial v}{\partial y}n_y-pn_y\right]\mathrm{d}\Gamma \tag{21}$$

where $\mathbf{n}$ is the unit normal vector on the fluid–solid interface $\mathbf{\Gamma}_c$ (pointing from fluid to the solid). Note that in Equation (21) we used $x_1=x$, $x_2=y$, $u_1=u$ and $u_2=v$. Although not used in the dynamic equation, Equation (20), we also compute the drag force acting on the structure. The drag force F_{D} is obtained replacing u by v and x by y (and *vice versa*) in Equation (21).

The method applied to the incompressible Navier–Stokes equations in a Eulerian reference frame, in Section 2, is generalized here to the ALE description using the apparent convective velocity $\mathbf{v}=\mathbf{u}-\mathbf{w}$ to replace $\mathbf{u}\cdot\nabla\mathbf{u}$ by $\mathbf{v}\cdot\nabla\mathbf{u}$ in the balance of momentum. Thus, the discretized

equations, Equations (3) and (4), become

$$\int_\Omega \left(N_i + \frac{\Delta t}{2}\, \hat{v}_b \frac{\partial N_i}{\partial x_b} \right) \hat{R}_a \, \mathrm{d}\Omega = 0 \quad \forall \text{ free } u_{ai}^{n+1} \tag{22}$$

and

$$\int_\Omega \Delta t \frac{\partial N_i}{\partial x_a} \hat{R}_a \, \mathrm{d}\Omega + \int_\Omega N_i \rho \frac{\partial \hat{u}_a^{n+1}}{\partial x_a} \, \mathrm{d}\Omega = 0 \quad \forall \text{ free } p_i^{n+1} \tag{23}$$

where $\hat{R}_a$ represents the momentum residuals

$$\hat{R}_a = \rho \left(\frac{\hat{u}_a^{n+1} - \hat{u}_a^n}{\Delta t} + \hat{v}_b^n \frac{\partial \hat{u}_a^{n+1/2}}{\partial x_b} \right) - \frac{\partial \hat{\tau}_{ab}^n}{\partial x_b} + \frac{\partial \hat{p}^{n+1/2}}{\partial x_a} \tag{24}$$

The time–space adaptive techniques presented in Section 3 for a Eulerian reference frame are generalized for the ALE co-ordinates. Thus, the local time-step formula presented in Section 2 is redefined in terms of the apparent convective velocity $\mathbf{v} = \mathbf{u} - \mathbf{w}$, becoming $\Delta t = \alpha h_e / \|\mathbf{v}^n\|$. Again, the combination of remeshing with these local time steps leads to a time–space adaptive procedure where the spatial and time discretizations are simultaneously adjusted.

In order to accommodate the movement of the structure, more conveniently described in a Lagrangian reference frame, and the fluid flow, described in Eulerian co-ordinates in our original program [4], we resorted to an ALE description of the FSI problem. Here we follow the basic idea proposed by Nomura and Hughes [16], defining an external Eulerian region for the fluid far from the structure, a transition region, where the ALE reference frame is used, and a Lagrangian description at the fluid–solid interface. These regions are depicted in Plate 1.

Note that mesh velocity must vary from zero, in the Eulerian region, to the velocity of the structure at the fluid–solid interface. We use a Laplace equation to determine a smooth and gradual transition of mesh velocity (and position) in the ALE region. Here, though, because we are dealing with a remeshing strategy based on refining a coarse background mesh, we must move not only the current computational mesh, but also the background mesh that will be used for remeshing. The Laplace equation algorithm is used in both cases.

The time advance of the flow field follows a segregated scheme. Pressure is computed first, then the new velocity components are updated. All linear equation systems involved are symmetric and solved by a matrix-free element-by-element pre-conditioned conjugate gradient method optimized for parallel-vector performance via mesh colouring [4]. Once the new flow field is found, the update of the structure position is obtained integrating Equation (20), with F_L computed from Equation (21), using a Runge–Kutta method.

We remark that the time step used to advance the fluid–structure solution is the interpolation time step Δt_int discussed in Section 2.1.1. In the FSI problems examined in this work, Δt_int is typically about 10^3 times smaller than the periods of oscillation of the structures considered and of the time scale of vortex shedding. In such cases, the use of the segregated solution procedure and the explicit time integration of the rigid-body motion is adequate. Furthermore, for the same reason, it was not necessary to introduce sub-iterations between fluid/solid within each time step. The reader should be aware, though, that the procedure adopted here may not be suitable for problems involving structural and forcing frequencies of the order of $1/\Delta t_\mathrm{int}$ or higher.

5. NUMERICAL EXAMPLES

In this section, we present applications of the techniques described above in some representative problems.

The first example is the simulation of turbulent vortex shedding around a fixed obstacle. This application shows the use of our LES procedure, with *implicit modelling* of sub-grid scales, in a test case where we can compare our results with experiments and more refined 3D numerical solutions. The purpose of this example is to assess our 2D LES approach in a problem where we do not have to deal with issues related to the ALE techniques implemented in the code.

The second example demonstrates the use of adaptive remeshing in an ALE computation with moving boundaries. It involves a laminar flow, where a rigid-body cylinder oscillates driven by the periodic forces caused by vortex shedding. The computational procedure was successful in detecting the occurrence of lock-in, where the vortex-shedding frequency is captured by the natural frequency of the structure, for the conditions considered in the example. In this laminar flow problem turbulence modelling is not an issue, and we pay attention to the ALE techniques and to the adaptive moving meshes implemented in the code.

The FSI problem associated to the flow around a dominant central span section of the Rio-Niterói bridge is taken as our final example. This continuous steel twin box girders bridge exhibits vortex-induced oscillation in the first bending mode when subjected to cross winds of relatively low velocities. We compare our results for the central span oscillation with reduced scale experiments performed in wind tunnel [17]. Whilst in the previous examples, we assessed the LES and the ALE techniques separately, in this practical engineering problem both the LES and the ALE adaptive techniques are simultaneously used in the computation.

All analyses have been non-dimensionalized in terms of reference scales. The reference scale for length is either the height of the square cylinder, the diameter of the circular cylinder or the depth of the bridge section. In all cases we denote the reference length by d. The free stream velocity u_0 is the reference scale for velocity. The reference pressure scale is ρu_0^2 and time is non-dimensionalized by d/u_0, i.e. $t^* = u_0 t/d$. The simulations are parameterized by the global Reynolds number $Re = \rho u_0 d/\mu$.

Numerical results are assessed in terms of drag and lift forces, of vortex-shedding frequency, and of the mean velocity and kinetic energy of fluctuations behind the square cylinder. The drag and lift coefficients are, respectively, given by $C_D = 2F_D/\rho u_0^2 d$ and $C_L = 2F_L/\rho u_0^2 d$, where F_D and F_L are, respectively, the drag and lift forces per unit span. The vortex-shedding frequency f is given in the non-dimensional form by the Strouhal number $St = f d/u_0$.

5.1. *Cross flow past a square cylinder at high Reynolds number*

Simulations have been performed for $Re = 10^4$, 2.2×10^4, 10^5 and 10^6. Figure 2 presents the analysis domain. A uniform velocity field with $u_1 = u_0$ and $u_2 = 0$ is prescribed on the upstream face. At the faces parallel to the oncoming flow the condition $u_2 = 0$ is imposed and at the downstream face free traction and zero reference pressure are prescribed. On the body surface impermeability and non-slip velocity boundary conditions are introduced.

About 800 meshes, containing approximately 8000 elements each, were generated and used in transient analyses running from $t^* = 0$ to 120, with minimum element size of $0.02d$. Typi-

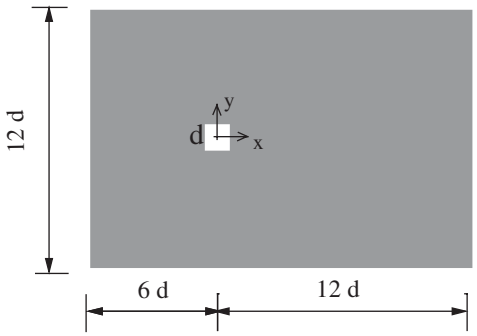


Figure 2. Analysis domain for the square cylinder in cross flow example.

cally, such analyses took about 18 000 time steps, involving the update of over 15 000 nodal variables per time step. Each transient ran for approximately 6 h on a 1.5 GHz P4 processor. The level of spatial refinement employed, though, is insufficient to resolve the thin boundary layers that characterize the cross flow at high Reynolds numbers. A limited investigation on the sensitivity of the results with respect to the choice of the minimum element size was performed for the $Re = 10^5$ example. No significant changes on the values of the mean force coefficients or the Strouhal number were observed when the minimum element size was reduced to $0.01d$. On the other hand, the computation took almost four times more CPU, partially because the increased number of elements, but mainly due to the time step reduction dictated by the expression $\Delta t = \alpha h_e / \|\mathbf{u}^n\|$. It must be said that the element size of $0.01d$ is still too large to capture boundary layer effects for the $Re = 10^5$ example. However, for problems involving bodies with sharp corners such as the square cylinder (and the Rio-Niterói bridge we shall consider in Section 5.3), the separation of boundary layers and the associated flow patterns are more strongly dictated by the sharp corners than by boundary layer mechanics. This explains the reasonably good results we shall present next, despite our insufficient resolution of boundary layers. On the other hand, boundary layer modelling (introduced by special boundary conditions based on near wall models) is required if we expect to simulate the so-called *Drag Crisis* that occurs for bluff bodies with smooth curvatures (such as a circular cylinder). The *Drag Crisis* is caused by a sudden change of the position of boundary layer separation. For cross flow past circular cylinders it occurs at about $Re = 3 \times 10^5$ and the resulting change in the flow pattern strongly affects the forces acting on the body.

Table I presents a comparison between our numerical results and some experimental and 3D LES data [21]. Note in Table I that the numerical time-averaged drag coefficient we obtained is in good agreement with the experimental value of $C_D \approx 2$ that characterizes the mean drag on square cylinders over a wide range of Reynolds numbers. As far as the *rms* fluctuation of C_L is concerned, the reported experimental results depend strongly on the level of turbulence present on the free stream flow [19]. For a smooth free stream such as the considered in this work, Vickery [19] reported the value of 1.32 for the fluctuating lift. Thus, the *rms* fluctuation of C_L obtained from the $Re = 10^5$ simulation is also in good agreement with the experimental observation. On the negative side, there is some discrepancy with regard to the Strouhal number. Indeed, Table I shows that our computation predicts a higher Strouhal number than that observed in experiments. However, in fairness to our 2D LES computations, note that the

Table I. Numerical and experimental results for Re ranging from 10^4 to 10^6. The over bar represents time-averaged values.

Experimental results	RE	$\bar{C}_L$	ΔC_{Lrms}	$\bar{C}_D$	ΔC_{Drms}	St
Lee [18]	1.75E+5	0	—	2.05	0.16–0.23	—
Vickery [19]	1.00E+5	—	1.32	2.05	0.17	0.12
Lyn *et al.* [20]	2.14E+4	—	—	2.10	—	0.13
3D LES results presented in a workshop [21]	RE	$\bar{C}_L$	ΔC_{Lrms}	$\bar{C}_D$	ΔC_{Drms}	St
Kawamura and Kawashima	2.2E+4	−0.005	1.33	2.58	0.27	0.15
Archambeau *et al.*	2.2E+4	0.0	—	2.02	—	0.09
Pourquié *et al.*	2.2E+4	−0.04	1.15	2.30	0.14	0.13
Tamura *et al.*	2.2E+4	−0.09	1.79	2.77	0.19	0.14
This work—2D LES	RE	$\bar{C}_L$	ΔC_{Lrms}	$\bar{C}_D$	ΔC_{Drms}	St
	1.0E+4	0.12	1.47	2.10	0.57	0.16
	2.2E+4	−0.005	1.75	2.03	0.60	0.16
	1.0E+5	−0.08	1.37	2.11	0.58	0.16
	1.0E+6	0.13	1.45	2.06	0.60	0.16

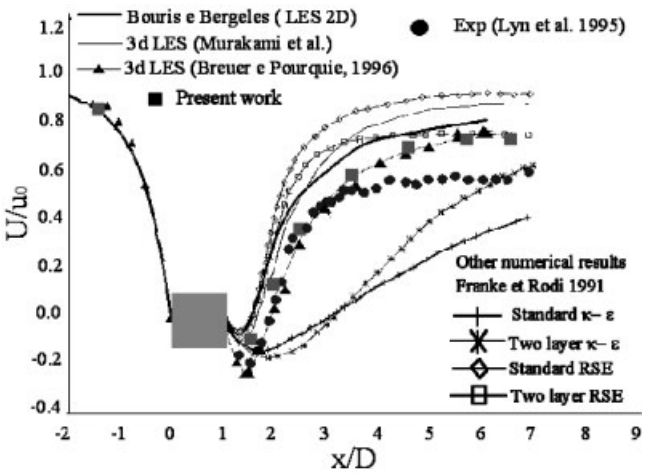


Figure 3. Mean velocity distribution as a function of x at $Re = 22\,000$. Comparison of the present work with other numerical and experimental data [20, 22–24].

results obtained using much more expensive 3D LES computations, also presented in Table I, are not significantly better than ours when compared with the experiments.

Next we show more comparisons, this time for $Re = 2.2 \times 10^4$, using the results presented by Bouris and Bergeles [22]. These authors have used the standard Smagorinsky sub-grid scale model in their 2D LES computations, comparing their results with experiments and 3D LES numerical data. In Figure 3 we see the time-averaged (mean) velocity obtained by several authors together with our own results with 2D LES based on *implicit sub-grid modelling*. Figure 3 shows that the centreline velocity we obtained is closer to the experimental data of Lyn *et al*. [20] than the centreline velocity obtained by Bouris and Bergeles in their 2D

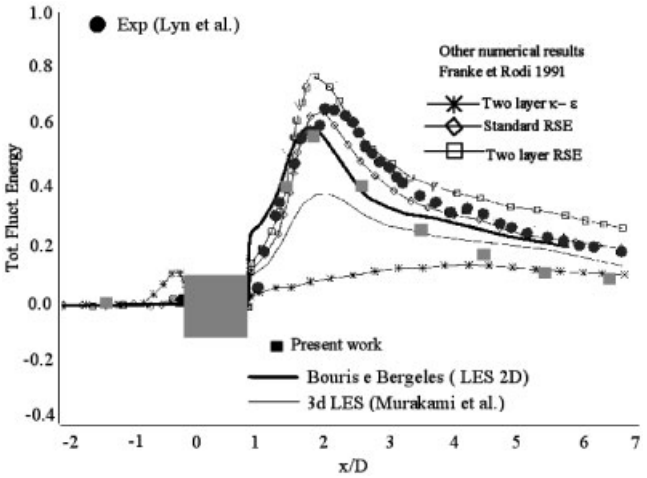


Figure 4. Total kinetic energy of fluctuations (periodic+turbulent) behind the cylinder at $Re = 22\,000$. Comparison of the present work with other numerical and experimental data [20, 22–24].

LES computation [22]. In particular, we obtained a better approximation of the size of the recirculation zone, although this is still smaller than that observed experimentally [20].

Figure 4 shows the total (periodic+turbulent) kinetic energy of the fluctuations behind the body. The total kinetic energy of the fluctuations is given by

$$k = 0.5((u - \bar{u})^2 + (v - \bar{v})^2) \tag{25}$$

where $\bar{u}$ and $\bar{v}$ denote the time-averaged components of the velocity field. Our results match closely those of Bouris and Bergeles [22] up to $x/d = 3$ and the experimental data of Lyn *et al.* [20] up to $x/d = 2$. In this region, our results are surprisingly better than the 3D LES results of Murakami and Mochida [23]. However, the total kinetic energy we predicted becomes smaller than that observed experimentally, as we proceed further downstream. This can be explained by the fact that our procedure is based on automatic refinement of a coarse background mesh. In the present problem, refinement is concentrated close to the obstacle and the extent of the refined region downstream depends on the number of elements we specified for the automatic remeshing procedure. Our total kinetic energy result downstream would be improved if we allowed more elements to be used in the analysis, but, clearly, this would lead to a more expensive computation.

5.2. *Circular cylinder driven by periodic vortex shedding*

The relevant non-dimensional parameters are the Reynolds number $Re = \rho u_0 d/\mu$, the reduced mass $M^* = m/\rho d^2$, the reduced (or relative) damping $\xi = c/2\sqrt{km}$ and the reduced velocity $U_R = u_0/f_n d$, where $f_n = 1/2\pi\sqrt{k/m}$ is the cylinder natural frequency of oscillation in vacuum.

We present next results for a simulation with $Re = 200$, $M^* = 30$, $\xi = 0.01$ and $U_R = 5.5$. At time $t^* = 0$ the cylinder is placed on a uniform flow ($u = u_0, v = 0$) and kept at a fixed position until $t^* = 150$. Then, the cylinder is released to move along the vertical y-direction. Figure 5 shows the evolution of vertical displacement, vertical velocity and drag and lift coefficients.

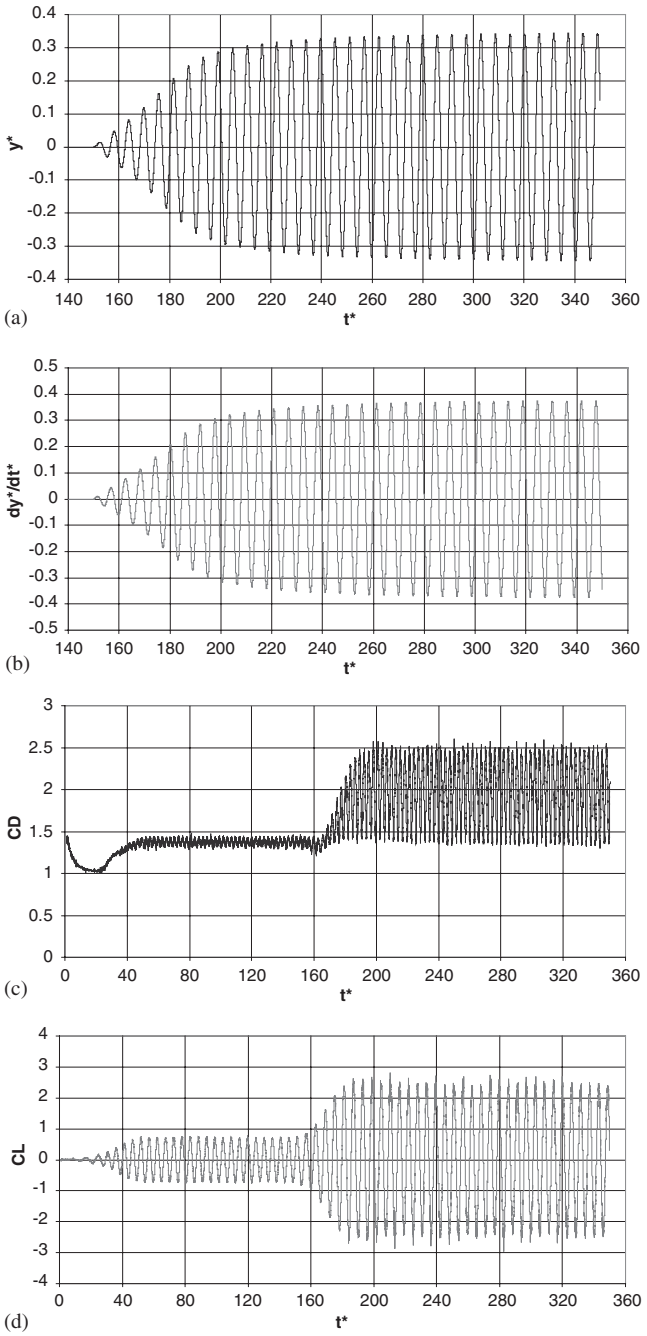


Figure 5. Time evolution of: (a) cylinder displacement, (b) cylinder velocity, (c) drag coefficient and (d) lift coefficient.

Table II. Results for a fixed circular cylinder in cross flow at $Re = 200$.

Source	C_D (mean)	C_L (rms)	St
Schlichting [25] (exp.)	1.3–1.4	—	0.18–0.20
Rosenfeld *et al.* [26] (num.)	1.31	0.460	0.20
Braza *et al.* [27] (num.)	1.3	0.544	0.20
Present work (num.)	1.382	0.499	0.195

Table III. Strouhal number and force coefficients before and after the cylinder is released to oscillate freely.

Parameter	Fixed cylinder	Released cylinder
St	0.195	0.176
C_D (mean)	1.382	1.979
C_D (rms)	0.034	0.391
C_L (rms)	0.499	1.618

Typical adaptive meshes, at $t^* = 300$ and 320, and details of the velocity and pressure fields at $t^* = 300$ are shown in Plate 2. The adaptive meshes contained about 5000 elements each, with minimum element size of $0.015d$. The minimum element size of $0.015d$ is fine enough to resolve the boundary layer for this laminar flow at $Re = 200$. The analysis took about 140 000 time steps, involving the update of over 10 000 nodal variables per time step. The transient ran for about a day on a 1.5 GHz P4 processor. The results for fixed cylinder compare well with experimental and numerical results [25–27], as shown in Table II.

A comparison between statistical data for fixed and released cylinder conditions is shown in Table III. A significant increase on the forces acting on the cylinder was observed in lock-in conditions.

Also, note the change of Strouhal number from $St = 0.195$, obtained when the cylinder is fixed, to the lock-in state at $St = 0.176$, obtained when the cylinder is released. When lock-in occurs, the frequency of oscillation f of the lift force is captured by the natural frequency of oscillation of the immersed cylinder, $f'_n = 1/2\pi\sqrt{k/(m + \Delta m)}$, where Δm is a small mass of fluid that is displaced as the cylinder moves. Thus, in lock-in conditions, the Strouhal number becomes $St = 1/U_R\sqrt{m/(m + \Delta m)}$. Neglecting Δm we would obtain a Strouhal number of 0.182 during lock-in, since $U_R = 5.5$ in this example. The slightly lower lock-in Strouhal number of 0.176 observed in the numerical simulation is due to the displaced mass of fluid Δm that moves with the cylinder, reducing the system oscillation frequency.

5.3. *Oscillation of the Rio-Niterói bridge*

In this section, we present results from a numerical simulation of fluid–structure interaction for the Rio-Niterói bridge, located in Rio de Janeiro, Brazil. This bridge is known to exhibit vertical oscillation in the first bending mode, caused by vortex shedding. Figure 6 shows the bridge cross-section exposed to the wind. The analysis domain is depicted in Figure 7. The structural data used in the analysis were mass per unit length $m = 24604.33$ kg/m, reduced

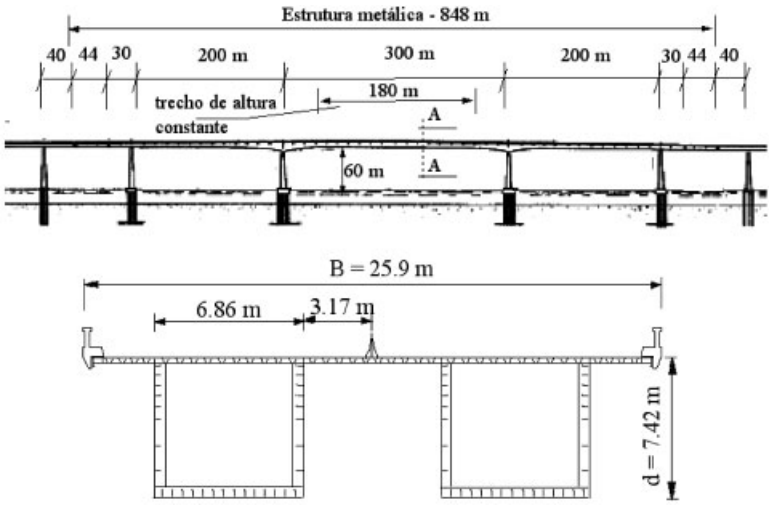


Figure 6. Detail of the Rio-Niterói bridge section analysed.

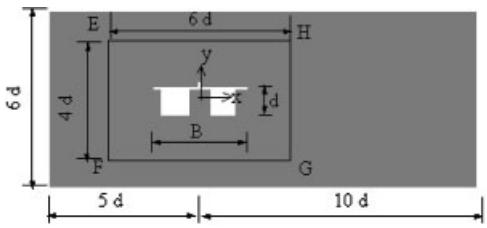


Figure 7. Analysis domain for the Rio-Niterói bridge example.

damping $\xi = 0.01$ and the first natural bending mode frequency $f_b = 0.32$ Hz. The air density and viscosity used were 1.25 kg/m^3 and 1.81×10^{-5} kg/ms, respectively. The dimensions of the bridge section studied are shown in Figure 6.

Simulations have been performed for wind velocities of $u_0 = 40, 54, 62$ and 70 km/h. The Reynolds number ranges from 5.69×10^6 to 1.03×10^7. The adaptive meshes contained about 25 000 elements each, with minimum element size of $0.02d$. The transient analyses took from 45 000 to 60 000 time steps, involving the update of about 50 000 nodal variables per time step. Each simulation took 2–3 days to complete on a 1.5 GHz P4 processor. Figure 8 presents two adaptive meshes for the example with $u_0 = 62$ km/h. Figure 9 shows the time evolution of vertical displacements for different wind velocities.

The mean amplitude of oscillation of the bridge was compared with results obtained in a reduced-scale experiment performed in a wind tunnel [17]. This comparison is shown in Figure 10. Note that the maximum amplitude in the numerical simulations is 210 mm, at the wind speed of $u_0 = 54$km/h This maximum amplitude value is close to the maximum amplitude of 260 mm obtained from the wind tunnel experiments. However, in the experiments the peak

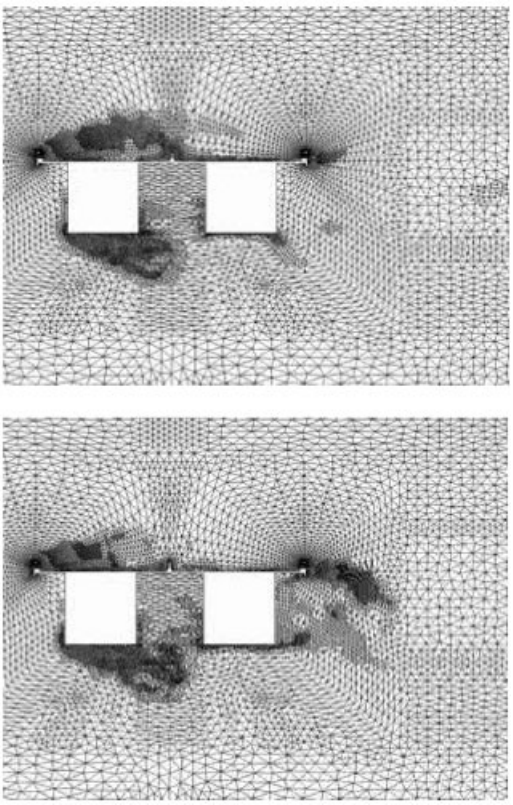


Figure 8. Typical adaptive meshes used during the simulation with $u_0 = 62\,\mathrm{km/h}$.

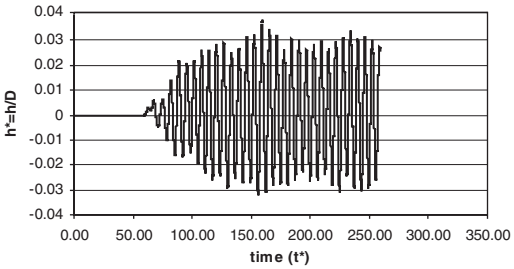
amplitude occurs at $u_0 = 62\,\mathrm{km/h}$ instead of $u_0 = 54\,\mathrm{km/h}$, revealing that we predicted a lower resonance wind velocity (U_{crit}) than that obtained in the wind tunnel.

In resonance the forcing frequency f is equal to first bending mode frequency f_{b}. Thus, the wind velocity at which resonance occurs can be written as

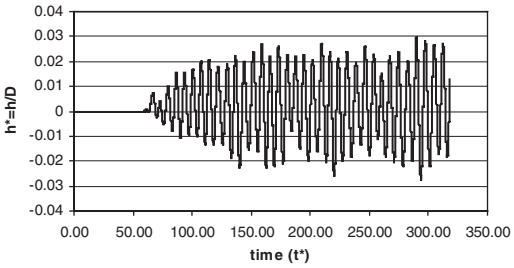
$$U_{\mathrm{crit}} = \frac{f_{\mathrm{b}} d}{S} \tag{26}$$

In the above equation, f_{b} and d are fixed data associated to the bridge. Thus, we conclude that we obtained a lower value for the wind velocity at resonance U_{crit} because in our simulations the Strouhal number S was overpredicted.

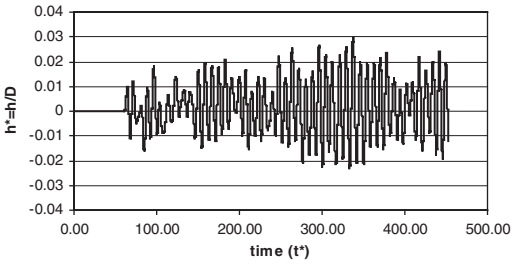
We recall that the overprediction of the Strouhal number also occurred in the example presented in Section 5.1, where we have addressed the problem of cross flow past a square cylinder at high Reynolds number. Further studies will be required to find out if the overprediction of the Strouhal number in our procedure can be cured by the use of special boundary conditions, based on boundary-layer physics, or if it is an intrinsic limitation of the 2D LES approach.



(a) Vertical displacement for $u_0 = 54$ Km/h



(b) Vertical displacement for u_0 ζ 62 Km/h



(c) Vertical displacement for $u_0 = 70$ Km/h

Figure 9. History of vertical displacement for different wind velocities.

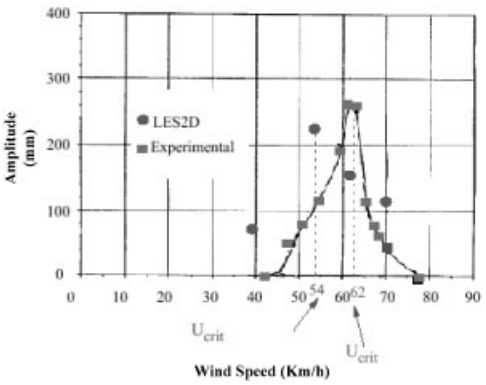


Figure 10. Mean amplitude of oscillation as a function of wind velocity.

6. CONCLUDING REMARKS

In this work we have shown the combination of time–space adaptive techniques, Large Eddy Simulation based on *implicit sub-grid modelling* and an arbitrary Lagrangian Eulerian (ALE) procedure in the study of representative fluid–structure interaction (FSI) problems.

Adaptive techniques permit optimizing the use of limited computer resources, focusing discretization refinement where it is most needed. Optimization of computer resources is particularly important for addressing the complex and demanding FSI problems found in engineering. The combination of spatial remeshing with local time steps, adjusted according to the time scales of the flow features resolvable by the mesh, leads to time–space adaptive computations and is of foremost importance for the simulation of transients.

In the first example, we considered the cross flow past a fixed square cylinder at high Reynolds (turbulent) conditions. The purpose of this example was to verify the performance of our 2D LES procedure with implicit sub-grid modelling. We obtained good results for the forces acting on the body. We also obtained a good qualitative prediction of the mean velocity and of the total (periodic+turbulent) kinetic energy of the fluctuations behind the square cylinder. However, the Strouhal number obtained was higher than that reported on experiments.

The second example involved the laminar cross flow past a circular cylinder and used a very fine time–space discretization, in such a way that we could validate our ALE and FSI algorithms, without dealing with turbulence modelling issues. The results obtained while the cylinder was maintained in a fixed position were in good agreement with other numerical and experimental data. After the cylinder has been released to oscillate freely, the simulation was capable of detecting the lock-in phenomenon, where the vortex-shedding frequency is captured by the natural frequency of the structure.

In the third example, we considered a civil engineering problem at a high Reynolds number: the flow-induced vertical oscillation of a dominant central span section of the Rio-Niterói bridge. The amplitude of oscillation predicted by the code was in reasonably good agreement with experimental results obtained in wind tunnel. However, the numerical simulation predicted resonance at a lower velocity than that observed in the wind tunnel experiments. It was shown that this is caused by a overprediction of the Strouhal number in the numerical simulation, a deficiency we have already observed in our first example, the cross flow past a square cylinder at high Reynolds numbers. Further research is required to establish if the Strouhal number prediction can be improved with the use of special boundary conditions, based on boundary layer physics, or if this deficiency is intrinsic to the 2D approach.

Clearly, real turbulent flows are three dimensional and involve very small space and time scales. Apart from extremely fine DNS computations, not affordable in most practical problems, other approaches used to simulate turbulence in engineering, such as LES or RANS, depart from reality in one way or another. In such a context, there is no such a thing as an absolute turbulence model that can be considered without reference to what is expected from its practical use. This is the pragmatic position adopted in this work. Indeed, we share the view of Bouris and Bergeles [22] that although one should aim at performing realistic 3D LES computations, 2D LES is a compromise imposed by computer limitations that should not be dismissed so easily.

REFERENCES

1. De Sampaio PAB, Coutinho ALGA. Simulating vortex shedding at high Reynolds numbers. In *The Proceedings of the 10th International Offshore and Polar Engineering Conference*, Chung JS, Olagnon M, Kim CH (eds), vol. III. International Society of Offshore and Polar Engineers (ISOPE): Cupertino-CA, U.S.A., 2000; 461–466.
2. De Sampaio PAB, Coutinho ALGA, Hallak PH, Pfeil MS. Some large eddy simulations in wind engineering problems. *Proceedings of the 22nd Iberian Latin American Congress on Computational Methods in Engineering* (*CILAMCE*), Campinas-SP, Brazil, 2001.
3. De Sampaio PAB. Transient solutions of the incompressible Navier–Stokes equations in primitive variables employing optimal local time stepping. In *Proceedings of the 8th International Conference on Numerical Methods for Laminar and Turbulent Flow*, Taylor C (ed.). Pineridge Press: Swansea, 1993, 1493–1504.
4. De Sampaio PAB, Coutinho ALGA. Simulation of free and forced convection incompressible flows using an adaptive parallel/vector finite element procedure. *International Journal for Numerical Methods in Fluids* 1999; **29**:289–309.
5. Hallak PH. Aeroelastic parameters for bridges via computational fluid dynamics. *Ph.D. Thesis*, Federal University of Rio de Janeiro (UFRJ), Brazil, 2002 (in Portuguese).
6. De Sampaio PAB, Lyra PRM, Morgan K, Weatherill NP. Petrov–Galerkin solutions of the incompressible Navier–Stokes equations in primitive variables with adaptive remeshing. *Computer Methods in Applied Mechanics and Engineering* 1993; **106**:143–178.
7. Brooks A, 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.
8. Zienkiewicz OC, Zhu JC. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering* 1987; **24**:337–357.
9. Bowyer A. Computing Dirichlet tessellations. *Computer Journal* 1981; **24**:162–166.
10. Mathieu J, Scott J. *An Introduction to Turbulent Flow*. Cambridge University Press: Cambridge, MA, 2000.
11. Sagaut P. *Large Eddy Simulation for Incompressible Flows*. Springer: Berlin, 2001.
12. Moin P. Advances in large eddy simulation methodology for complex flows. *International Journal of Heat and Fluid Flow* 2002; **23**:710–720.
13. Drikakis D. Embedded turbulence model in numerical methods for hyperbolic conservation laws. *International Journal for Numerical Methods in Fluids* 2002; **39**:763–781.
14. Margolin LG, Rider WJ. A rationale for implicit turbulence modelling. *International Journal for Numerical Methods in Fluids* 2002; **39**:821–841.
15. Hughes TJR, Mazzei L, Jansen KE. Large eddy simulation and the variational multiscale method. *Computing and Visualization in Science* 2000; **3**:47–59.
16. Nomura T, Hughes TJR. An arbitrary Lagrangian–Eulerian finite element method for interaction of fluid and a rigid body. *Computer Methods in Applied Mechanics and Engineering* 1992; **95**:115–138.
17. Robinson R, Savage MG. Wind tunnel investigation of the President Costa e Silva Bridge, Rio de Janeiro, Brazil. *Laboratory Technical Report* LTR-LA-311, National Aeronautical Establishment, National Research Council Canada, Ottawa, 1989.
18. Lee BE. The effect of turbulence on the surface pressure field of a square prism. *Journal of Fluid Mechanics* 1974; **69**:263–282.
19. Vickery BJ. Fluctuating lift and drag on a long cylinder of square cross-section in a smooth and in a turbulent stream. *Journal of Fluid Mechanics* 1966; **25**:481–494.
20. Lyn DA, Einav S, Rodi W, Park J. A laser-Doppler velocimetry study of ensemble-averaged characteristics of the turbulent near wake of a square cylinder. *Journal of Fluid Mechanics* 1995; **304**:285–319.
21. Rodi W, Ferziger JH, Breuer M, Pourquié M. Status of large eddy simulation: results of a workshop. *Journal of Fluids Engineering* (*ASME*) 1997; **119**:248–262.
22. Bouris D, Bergeles G. 2D LES of vortex shedding from a square cylinder. *Journal of Wind Engineering and Industrial Aerodynamics* 1998; **80**:31–46.
23. Murakami S, Mochida. On turbulent vortex shedding flow past 2D square cylinder predicted by CFD. *Journal of Wind Engineering and Industrial Aerodynamics* 1995; **54–55**:191–211.
24. Franke R, Rodi W. Calculation of vortex shedding past a square cylinder with various turbulence models. In *Proceedings of the 8th Symposium on Turbulent Shear Flows*. Springer: Berlin, 1991; 189–204.
25. Schlichting H. *Boundary-Layer Theory*. McGraw-Hill: New York, 1979.
26. Rosenfeld M, Kwak D, Vinokur M. A fractional step solution method for the unsteady incompressible Navier–Stokes equations in generalized coordinate systems. *Journal of Computational Physics* 1991; **94**:102–137.
27. Braza M, Chassaing P, Minh H. Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder. *Journal of Fluid Mechanics* 1986; **165**:79–130.

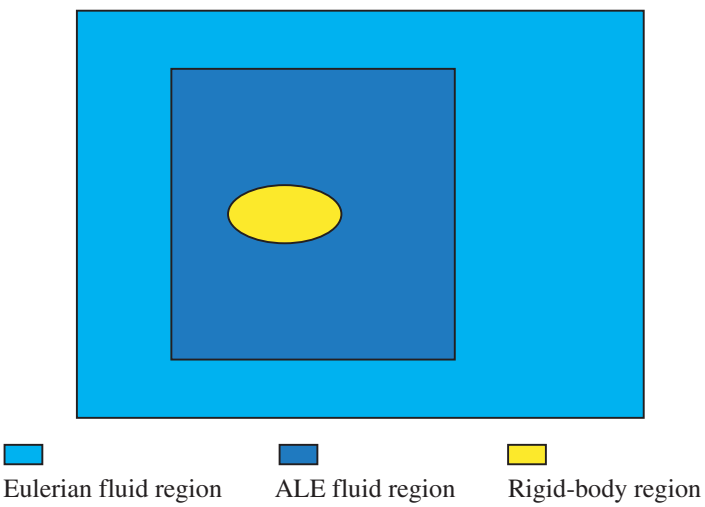


Plate 1. Splitting of the problem domain into Eulerian, ALE and rigid-body regions.

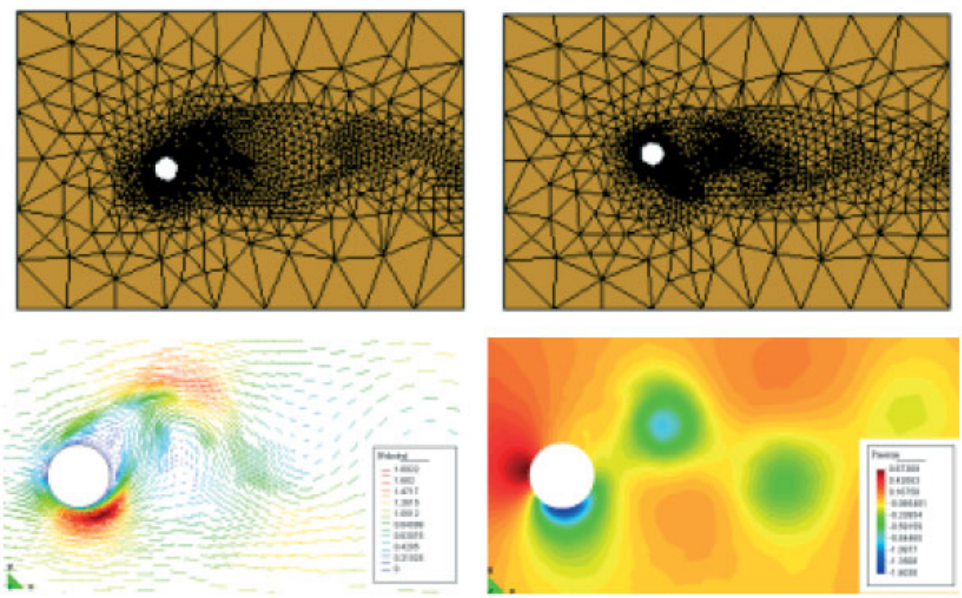


Plate 2. Adaptive meshes and details of the velocity and pressure fields close to the moving cylinder.