European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2006
P. Wesseling, E. Oñate and J. Périaux (Eds)

FLUID-STRUCTURE ITERATIVE ALGORITHMS USING PRESSURE SEGREGATION METHODS AND THEIR APPLICATION TO BRIDGE AERODYNAMICS

Santiago Badia and Ramon Codina

International Center for Numerical Methods in Engineering (CIMNE),
Universitat Politècnica de Catalunya
Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain
e-mail: sbadia@cimne.upc.edu, ramon.codina@upc.edu

Key words: Fluid-Structure Interaction, Pressure Segregation, Aeroelasticity, Bridge Aerodynamics.

Abstract. *In this paper we suggest some algorithms for the fluid-structure interaction problem stated using a domain decomposition framework. These methods involve stabilized pressure segregation methods for the solution of the fluid problem and fixed point iterative algorithms for the fluid-structure coupling. These coupling algorithms are applied to the aeroelastic simulation of suspension bridges. We assess flexural and torsional frequencies for a given inflow velocity. Increasing this velocity we reach the value for which the flutter phenomenon appears.*

1 INTRODUCTION

The interaction between a fluid and a structure appears in a wide variety of fields. Probably, the most analyzed fluid-structure interaction problem is the aeroelastic one (specially for aeronautical applications), for instance in the simulation of the action of a fluid (air) over a structure (such as a wing or a bridge). Recently, an increasing interest in the simulation of haemodynamics has motivated a lot of research on fluid-structure algorithms appropriate for the blood-vessel system.

The implementation of a coupled problem can be done using two different global strategies. The *monolithic* strategy implies the solution of the coupled problems simultaneously. *Partitioned* methods are usually used in order to keep software modularity and to allow the use of the numerical methods developed for every field separately. When using pressure segregation methods for the fluid problem (as in this work) *partitioned* procedures are naturally adapted, since a global iterative scheme is already needed to couple the velocity and pressure calculations.

The use of explicit procedures for the coupling has been deeply studied by Farhat and co-authors in the framework of aeroelasticity[1]. On the other side, for haemodynamics

more elaborated algorithms are needed for the coupling. In order to obtain convergence, Newton and quasi-Newton algorithms have been suggested[2,3]. The relaxation of these methods is a key aspect in order to reach convergence when dealing with these problems, and some possibilities have been used[4,5]. For a simplified blood-vessel system is studied, the *added mass* effect and the big impact of the relaxation on the convergence has been nicely explained[6].

Let us list what we need in order to solve a fluid-structure problem. In these problems the displacement of the structure changes the domain of the fluid. Then, the fluid equations have to be able to deal with moving domains. With this aim we use an ALE (*Arbitrary Lagrangian Eulerian*) approach. Some ALE formulations have been analyzed[7,8]. The fluid solver for incompressible flows is a key point of the algorithm because it consumes most of the CPU time. The monolithic treatment of the Navier-Stokes equations is involved (for the system solver) and time consuming. In order to improve the situation, we suggest the use of pressure segregation methods in their fractional step and predictor-corrector forms[9]. On the other hand, we use the orthogonal subgrid scale stabilized finite element method[10] for the space discretization, that allows the use of equal velocity-pressure interpolation. Less attention is paid to the structure solver. The following exposition can be applied to any kind of structural problem, with linear or nonlinear material behavior.

We have organized the present work as follows. In Section 2 we state every field problem in its continuous level and some notation is introduced. We write the strong and weak form of the governing equations of the coupled problem. In Section 3 we write the interface equation associated to the problem under consideration, using a Domain Decomposition framework. Some methods have been listed. Finally, at the fully discrete level, we introduce the fluid solvers and appropriate coupling procedures (Section 4). In particular, pressure segregation methods are suggested. In Section 5 we justify the algorithms chosen for the numerical experimentation and applications. Section 6 is devoted to the application of these methods to the simulation of bridge aerodynamics. Section 7 concludes the paper by drawing some conclusions.

2 THE CONTINUOUS PROBLEM

In this section we introduce the fluid-structure problem at the continuous level. Firstly, we treat some aspects about the problem domain, the definition of its movement and its restriction to the fluid and structure, the domain velocity and the matching conditions that these restrictions satisfy on the interface. Secondly, we state the governing equations of the fluid and structure problems and suggest how to calculate the domain displacement. We conclude this section with the matching conditions (that is, continuity of some values) that have to be imposed over the interface between the fluid and the solid.

We denote by Ω_t the domain occupied by the heterogeneous mechanical system at a given time $t > 0$. This domain is divided into the structure domain Ω_t^s and its complement Ω_t^f occupied by the fluid. We denote by $\Sigma_t \equiv \partial\Omega_t^f \cap \partial\Omega_t^s$ the fluid-structure interface.

Further, $\boldsymbol{n}_f$ is the outward normal of Ω_t^f on Σ_t and $\boldsymbol{n}_s$ its counterpart for the structure.

The total domain Ω_t is defined at every time instant by a family of mappings $\mathcal{A}_t$

$$\mathcal{A}_t : \Omega_0 \longrightarrow \Omega_t,$$

where Ω_0 is the reference domain associated to $t = 0$.

Let us introduce some notation. Given a function $f : \Omega_t \times [0,T] \longrightarrow \mathbb{R}$ defined at the current domain we indicate by $\hat{f} = f \circ \mathcal{A}_t$ the corresponding function defined at the initial configuration,

$$\hat{f} : \Omega_0 \times [0,T] \longrightarrow \mathbb{R}, \quad \hat{f}(\boldsymbol{x}_0, t) = f(\mathcal{A}_t(\boldsymbol{x}_0), t).$$

Furthermore, the time derivatives at the initial configuration are defined as follows:

$$\left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{x}_0} : \Omega_t \times (0,T) \longrightarrow \mathbb{R}, \quad \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{x}_0} (\boldsymbol{x}, t) = \frac{\partial \hat{f}}{\partial t}(\boldsymbol{x}_0, t).$$

We denote by $\boldsymbol{d}(\boldsymbol{x}, t)$ the displacement of the domain evaluated at the current configuration. Then, we could write the mapping $\mathcal{A}_t$ as $\mathcal{A}_t(\boldsymbol{x}_0, t) = \boldsymbol{x}_0 + \widehat{\boldsymbol{d}}(\boldsymbol{x}_0, t)$. We split the domain displacement into its fluid and structure restriction as $\boldsymbol{d} = \mathcal{R}_{\Omega_t^s}\boldsymbol{d} + \mathcal{R}_{\Omega_t^f}\boldsymbol{d} =: \boldsymbol{d}^s + \boldsymbol{d}^f$. From the trace theorem we know that

$$\boldsymbol{d}^s|_{\Sigma_t} = \boldsymbol{d}^f|_{\Sigma_t} \tag{1}$$

has to be satisfied. Moreover, we define

$$\boldsymbol{w} = \left.\frac{\partial \boldsymbol{d}^f}{\partial t}\right|_{\boldsymbol{x}_0}, \tag{2}$$

which is the domain velocity that we will require in order to write the fluid equations in an ALE framework.

In the present work we assume a Newtonian incompressible fluid. We use the ALE formulation in order to write the Navier-Stokes equations on moving domains. In what follows we only consider the boundary conditions on Σ_t. The rest of boundary conditions are essential for the definition of the problem but do not affect the following exposition. For this reason we have omitted them for the sake of clarity. The Navier-Stokes equations that govern the fluid problem read as follows: find a velocity field $\boldsymbol{u}$ and a pressure field p such that

$$\rho_f \frac{\partial \boldsymbol{u}}{\partial t} - \mu \Delta \boldsymbol{u} + \rho_f \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \rho_f \boldsymbol{f}_f \qquad \text{in } \Omega_t^f \times (0,T), \tag{3a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega_t^f \times (0,T), \tag{3b}$$

where ρ_f is the density and μ the viscosity of the fluid. The Cauchy stress tensor for the fluid is $\boldsymbol{\sigma}^f = -p\boldsymbol{I} + 2\mu\boldsymbol{\epsilon}(\boldsymbol{u})$ where $\boldsymbol{\epsilon}(\boldsymbol{u}) = (\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T)/2$ is the strain rate tensor and $\boldsymbol{I}$ the identity matrix. We denote by $\boldsymbol{\sigma}^f_n := \boldsymbol{\sigma}^f|_{\Sigma_t} \cdot \boldsymbol{n}_f$ the normal stress on Σ_t.

At this point, using the Reynolds formula for the time derivative, we can write the fluid equations (3) in the ALE framework as follows: find a velocity $\boldsymbol{u}$ and a pressure p such that

$$\rho_f \left.\frac{\partial \boldsymbol{u}}{\partial t}\right|_{\boldsymbol{x}_0} - \mu\Delta\boldsymbol{u} + \rho_f(\boldsymbol{u}-\boldsymbol{w})\cdot\nabla\boldsymbol{u} + \nabla p = \rho_f \boldsymbol{f}_f \qquad \text{in } \Omega^f_t \times (0,T), \tag{4a}$$

$$\nabla\cdot\boldsymbol{u} = 0 \qquad \text{in } \Omega^f_t \times (0,T). \tag{4b}$$

The structure can easily handle with moving domains using a fully Lagrangian framework. For instance, we consider an elastic structure. We denote by $\boldsymbol{\sigma}^s_n := \boldsymbol{\sigma}^s|_{\Sigma_t} \cdot \boldsymbol{n}_s$ the normal stress on Σ_t. The displacement of the structure domain $\boldsymbol{d}^s$ is assumed equal to the structure displacement.

The fluid displacement $\boldsymbol{d}^f$ is arbitrary but has to satisfy condition (1). Thus, we can write $\boldsymbol{d}^f$ as an arbitrary extension of $\boldsymbol{d}^s|_{\Sigma_t}$ into Ω^f_t, that we denote by $\boldsymbol{d}^f = \mathrm{Ext}\,(\boldsymbol{d}^s|_{\Sigma_t})$. Different choices of the lifting operator $\mathrm{Ext}(\cdot)$ have been proposed in the literature. Herein, we adopt an harmonic extension evaluated at the current domain Ω^f_t. In this case, $\boldsymbol{d}^f$ is solution of the Laplace problem.

At this point, suitable matching conditions have to be applied on the interface Σ_t. These are continuity of normal stresses (due to the action-reaction principle) and velocities (due to the perfect adherence of the fluid to the structure):

$$\boldsymbol{u} = \frac{\partial \widehat{\boldsymbol{d}}^s}{\partial t} \qquad \text{on } \Sigma_t \times (0,T), \tag{5}$$

$$\boldsymbol{\sigma}^f_n + \boldsymbol{\sigma}^s_n = \boldsymbol{0} \qquad \text{on } \Sigma_t \times (0,T). \tag{6}$$

Then, the fluid-structure coupled problem is completely defined by the fluid problem (4), the structure problem , the fluid domain displacement, and the interface matching conditions (1), (5) and (6). For the space discretization of the equations, let us to write the weak form of the system. Given $t \in (0,T)$, the functional spaces

$$\begin{aligned}
\mathcal{V}(\Omega^f_t) &:= \left\{ \boldsymbol{v} : \Omega^f_t \to \mathbb{R}^d, \boldsymbol{v} = \hat{\boldsymbol{v}} \circ (\mathcal{A}^f_t)^{-1}, \hat{\boldsymbol{v}} \in (H^1(\Omega^f_0))^d \right\}, \\
\mathcal{V}_0(\Omega^f_t) &:= \left\{ \boldsymbol{v} \in \mathcal{V}(\Omega^f_t) |\, \boldsymbol{v}|_{\Sigma_t} = \boldsymbol{0} \right\}, \\
\mathcal{Q}(\Omega^f_t) &:= \left\{ q : \Omega^f_t \to \mathbb{R}, q = \hat{q} \circ (\mathcal{A}^f_t)^{-1}, \hat{q} \in L^2(\Omega^f_0) \right\}, \\
\Gamma(\Sigma_t) &:= \left\{ \boldsymbol{\gamma} : \Sigma_t \to \mathbb{R}^d, \boldsymbol{\gamma} = \hat{\boldsymbol{\gamma}} \circ (\mathcal{A}_t|_{\Sigma_t})^{-1}, \hat{\boldsymbol{\gamma}} \in (H^{1/2}(\Sigma_0))^d \right\},
\end{aligned}$$

will allow us to write the governing equations of the fluid (4) in their weak forms. The notation used here is as follows: $L^2(\omega)$ denotes the space of square integrable functions

in a spatial domain ω, $H^1(\omega)$ is the space of functions in $L^2(\omega)$ with first derivatives in $L^2(\omega)$, and $H^{1/2}(\sigma)$ is the space of functions defined on a $d-1$-manifold σ that are the trace of functions in $H^1(\omega)$, with $\sigma \subset \partial\omega$. For functions f and g defined on a d- or $d-1$-manifold, we write $\langle f, g\rangle_\omega := \int_\omega fg \, \mathrm{d}\omega$, omitting the subscript when ω is the domain where the problem under consideration is posed. For σ a $d-1$-manifold and $f \in H^{1/2}(\sigma)$, the space of functions g such that $\langle f, g\rangle_\sigma < \infty$ is denoted by $H^{-1/2}(\sigma)$. Finally, $(\cdot, \cdot)$ denotes the usual L^2 product in the domain where the problem considered is posed.

3 THE DOMAIN DECOMPOSITION APPROACH

In this section we reformulate the fluid-structure problem in a Domain Decomposition (DD onwards) framework[11]. First, the fluid problem is introduced in this framework, and after that, the structure problem. The resulting *interface equation* is written in different forms, in order to justify the use of different algorithms suggested in the literature for the fluid-structure problem.

Let us consider the time discretized version of (4) using backward-differencing formulas (BDF) for the time integration at time step $t^{n+1} = (n+1)\delta t$, $\delta t > 0$ being the time step size (assumed constant for simplicity). We denote the BDF-p operator as

$$D_p f^{n+1} = \frac{1}{\delta t \gamma_p} \sum_{i=0}^{p} \alpha_p^i f^{n+1-i} \tag{7}$$

where f is a generic time dependent function, f^n denotes its approximation at t^n, k is the order of accuracy of the scheme and γ_p and α_p^i are the parameters that define the BDF numerical integration. The first and second order BDF methods are defined as:

$$\begin{aligned} D_1 f^{n+1} &= f^{n+1} - f^n, \\ D_2 f^{n+1} &= \frac{3}{2}(f^{n+1} - \frac{4}{3} f^n + \frac{1}{3} f^{n-1}). \end{aligned}$$

At a fixed time step $n+1$, let us denote by $\boldsymbol{\lambda}$ the interface variable corresponding to the displacement on the fluid-structure interface, $\boldsymbol{d}|_{\Sigma_{t^{n+1}}}$. We denote by $\mathsf{FL}_{\delta t}$ the operator that gives the velocity and pressure field at t^{n+1} for a given $\boldsymbol{\lambda}$,

$$\begin{aligned} \mathsf{FL}_{\delta t} : \Gamma(\Sigma_{t^{n+1}}) &\to \mathcal{V}(\Omega^f_{t^{n+1}}) \times \mathcal{Q}(\Omega^f_{t^{n+1}}) \\ \boldsymbol{\lambda} &\mapsto (\boldsymbol{u}^{n+1}, p^{n+1}) \end{aligned}$$

There are multiple choices for the $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$ operator, corresponding to the different possibilities for the time approximation of the incompressible Navier-Stokes equations, such as the monolithic system or the fractional step version at the continuous level in space. Let us start with the monolithic scheme, denoted by $\mathsf{MN}_{\delta t}(\boldsymbol{\lambda})$. In this case, $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda}) = (\boldsymbol{u}^{n+1}, p^{n+1})$ is computed by solving the problem: given $\boldsymbol{\lambda} \in \Gamma(\Sigma_{t^{n+1}})$, find $\boldsymbol{u}^{n+1} \in$

$\mathcal{V}(\Omega^f_{t^{n+1}})$ and $p^{n+1} \in \mathcal{Q}(\Omega^f_{t^{n+1}})$ such that

$$\frac{\rho_f}{\delta t}\left(D_k \boldsymbol{u}^{n+1}, \boldsymbol{v}\right) + \mu\left(\nabla \boldsymbol{u}^{n+1}, \nabla \boldsymbol{v}\right) + \rho_f\left((\boldsymbol{u}^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \boldsymbol{u}^{n+1}, \boldsymbol{v}\right)$$

$$-(p^{n+1}, \nabla \cdot \boldsymbol{v}) = \rho_f \langle \boldsymbol{f}_f^{n+1}, \boldsymbol{v}\rangle \qquad \forall \boldsymbol{v} \in \mathcal{V}_0(\Omega^f_{t^{n+1}}), \quad (8a)$$

$$\left(\nabla \cdot \boldsymbol{u}^{n+1}, q\right) = 0 \qquad \forall q \in \mathcal{Q}(\Omega^f_{t^{n+1}}), \quad (8b)$$

$$\boldsymbol{u}^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda} + \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}^{n-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}. \quad (8c)$$

Borrowing classical concepts from domain decomposition methods, we can define the *Steklov-Poincaré interface operator* for the fluid as follows: $\mathcal{S}_f$ is the Dirichlet-to-Neumann map in Ω^f_t such that

$$\begin{aligned} \mathcal{S}_f : H^{1/2}(\Sigma_t) &\to H^{-1/2}(\Sigma_t) \\ \boldsymbol{\lambda} &\mapsto \boldsymbol{\sigma}^f_n. \end{aligned} \quad (9)$$

This operator consists of solving the fluid problem given a value for the interface variable $\boldsymbol{\lambda}$, that is $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$, and recover the normal stress on the interface $\boldsymbol{\sigma}^f_n$.

We point out that the Steklov-Poincaré operator $\mathcal{S}_f$ for the fluid is nonlinear. It involves two different non-linearities: one associated to the convective term of the Navier-Stokes equations and a second one due to the fact that the fluid domain $\Omega^f_t \equiv \Omega^f_t(\boldsymbol{\lambda})$ does depend on the interface variable (shape non-linearity).

Analogously for the structure, we can define the Steklov-Poincaré operator $\mathcal{S}_s$, which in this case consists of solving the structure problem using $\boldsymbol{\lambda}$ as Dirichlet boundary condition for $\boldsymbol{d}^s$ on Σ_t and extract the value of the normal stress $\boldsymbol{\sigma}^s_n$ on Σ_t. Therefore, this is a mapping between the trace of the displacement field $\boldsymbol{d}$ and the space of normal stresses exerted by the structure. Again, this operator is nonlinear even for linear constitutive equations (as the elastic case considered) because of the *shape derivative* (the deformation of the solid domain). Let us introduce also $\mathcal{S}_s^{-1}$, which is the so called *Poincaré-Steklov interface operator*: $\mathcal{S}_s^{-1}$ is the Neumann-to-Dirichlet map in Ω^s_t such that

$$\begin{aligned} \mathcal{S}_s^{-1} : H^{-1/2}(\Sigma_t) &\to H^{1/2}(\Sigma_t) \\ \boldsymbol{\sigma}^s_n &\mapsto \boldsymbol{\lambda}. \end{aligned} \quad (10)$$

The operator $\mathcal{S}_s^{-1}$ consists of solving the structure problem using $\boldsymbol{\sigma}^s_n$ as Neumann boundary condition on Σ_t and recover $\boldsymbol{d}^s$ on the boundary. $\mathcal{S}_s^{-1}$ will be used for fixed point algorithms.

At this point the interface condition (6) that involves continuity of normal stresses on Σ_t can be easily rewritten as: find $\boldsymbol{\lambda} \in \Gamma(\Sigma_{t^{n+1}})$ such that

$$\mathcal{S}_f(\boldsymbol{\lambda}) + \mathcal{S}_s(\boldsymbol{\lambda}) = \boldsymbol{0}. \quad (11)$$

An alternative form of the interface equation, obtained by applying the inverse of the Steklov-Poincaré operator $\mathcal{S}_s^{-1}$ in (11), reads as: find $\boldsymbol{\lambda} \in \Gamma(\Sigma_{t^{n+1}})$ such that

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda})) = \boldsymbol{\lambda}. \tag{12}$$

This expression motivates the use of the fixed point algorithm. The iterative fixed point procedure can be written as: given $\boldsymbol{\lambda}^k$, with $k \geq 0$, find $\boldsymbol{\lambda}^{k+1}$ such that

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda}^k)) = \boldsymbol{\lambda}^{k+1}, \tag{13}$$

where $\mathcal{S}_f(\boldsymbol{\lambda})$ is associated to an appropriate semi-discrete fluid solver $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$. The initialization $\boldsymbol{\lambda}^0$ of the iterative process is treated in Section 5. Let us explain this equation: given a value for the interface displacement $\boldsymbol{\lambda}^k$, we solve the fluid problem for this $\boldsymbol{\lambda}^k$ using $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda}^k)$ and recover the normal stresses on the interface $\boldsymbol{\sigma}_n^f$, that is to say, we compute $\mathcal{S}_f(\boldsymbol{\lambda}^k)$. Then, we calculate the structure problem with $\boldsymbol{\sigma}_n^s = \boldsymbol{\sigma}_n^f$ as boundary condition on the fluid-structure interface. It gives a new value of the interface displacement, that now we call $\boldsymbol{\lambda}^{k+1}$. In this case we solve the *Neumann-to-Dirichlet* Poincaré-Steklov interface operator $-\mathcal{S}_s^{-1}(\boldsymbol{\sigma}_n^f)$. This procedure is repeated until convergence. The solution of the fluid problem $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$ requires nonlinear iterations. Thus, algorithm (13) involves the use of nested iterative loops.

We are also interested on a linearized version of $\mathcal{S}_f$. We denote by $\mathsf{FL}_{\delta t}(\boldsymbol{u}_*^{n+1}; \gamma)$ the linearized fluid operator that differs from the non-linearized version, i.e. (8), in the fact that the convective term in the momentum equation of the fluid has been replaced by $\boldsymbol{u}_*^{n+1} \cdot \nabla \boldsymbol{u}^{n+1}$ with $\boldsymbol{u}_*^{n+1}$ given. We also denote by $\widetilde{\mathcal{S}}_f(\boldsymbol{u}_*^{n+1})$ the linearization of $\mathcal{S}_f$ *around the point* $\boldsymbol{u}_*^{n+1}$, that is, involving the solution of the linearized fluid problem with $\mathsf{FL}_{\delta t}(\boldsymbol{u}_*^{n+1}; \gamma)$. In the next section we suggest the use of the *semi*-linear interface operator in some cases. We stress the fact that $\widetilde{\mathcal{S}}_f(\boldsymbol{u}_*^{n+1})$ is non linear due to the *shape derivative*.

A different version of the fixed point algorithm (13) is obtained when using the *semi-linearized* version of the interface operator $\mathcal{S}_f$ for the fluid. In this case the fixed point algorithm reads as follows: given $\boldsymbol{\lambda}^k$ and $\boldsymbol{u}^{n+1,k}$ with $k > 0$, compute $\boldsymbol{\lambda}^{k+1}$ by

$$-\mathcal{S}_s^{-1}(\widetilde{\mathcal{S}}_f(\boldsymbol{u}^{n+1,k}; \boldsymbol{\lambda}^k)) = \boldsymbol{\lambda}^{k+1}. \tag{14}$$

and obtain $\boldsymbol{u}^{n+1,k+1}$ from $\mathsf{FL}_{\delta t}(\boldsymbol{u}^{n+1,k}; \boldsymbol{\lambda}^k)$. The procedure is repeated until a selected norm of $\boldsymbol{u}^{n+1,k+1} - \boldsymbol{u}^{n+1,k}$ and (or) $\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k$ is below a threshold tolerance. When using the algorithm (14) the same loop deals with the coupling of the fluid and structure systems and the nonlinearity of the fluid equations.

The *semi-linearized* fixed point algorithm (14) involves the domain update at each iteration. This situation can be relaxed by using some criteria over $(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k)$ in order to decide to update or to freeze the domain at the current iteration (that is to say, to neglect or not the shape derivative). Alternatively, instead of freezing the domain, we can use a transpiration method (cheaper than the movement of the domain), in order to accelerate the iterative process.

Alternative forms of the interface equation (11) motivate different iterative algorithms for the coupling. Besides the iterative algorithm for the coupling, a relaxation method is advisable in order to improve the convergence properties of all the previous algorithms.

4 THE DISCRETE PROBLEM

This section is devoted to the fully discretized version of the coupling problem. We are focused on the discretization of the fluid. Three different sorts of methods are considered: monolithic, pressure-correction and predictor-corrector. Every method is introduced and stated. In the applications we consider the stabilized versions of these schemes using orthogonal subgrid scales. However, for the sake of clarity, we omit the stabilization terms in the formulation. We refer the reader to a set of articles that deal with stabilized pressure segregation methods[9,12]. The use of a stabilized space discretization allows us to use the same low-order finite element space for the interpolation of velocity and pressure. After the exposition of the alternative methods for the fluid problem, we state the discrete extension operator used for the calculation of the fluid domain movement. Finally, we suggest some coupling procedures taking into account the fluid solver used. These procedures are stated for being used in Section 6.

4.1 The discrete fluid problem

The fully discretized version of the monolithic scheme (8), denoted by $\mathsf{MN}_{\delta t,h}(\boldsymbol{\lambda}_h)$, reads as follows: for $n = 0, 1, 2, ...$, given $\boldsymbol{\lambda}_h \in \Gamma_h(\Sigma_{t^{n+1}})$ (understood as the displacement on the solid boundary at time step n), find $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that,

$$\frac{\rho_f}{\delta t}\left(D_k\boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h\right) + \mu\left(\nabla\boldsymbol{u}_h^{n+1}, \nabla\boldsymbol{v}_h\right) + \rho_f\left((\boldsymbol{u}_h^{n+1} - \boldsymbol{w}^{n+1})\cdot\nabla\boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h\right)$$

$$-(p_h^{n+1}, \nabla\cdot\boldsymbol{v}_h) = \rho_f\langle\boldsymbol{f}_f, \boldsymbol{v}_h\rangle \qquad \forall\boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \quad (15a)$$

$$\left(\nabla\cdot\boldsymbol{u}_h^{n+1}, q_h\right) = 0 \qquad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}), \quad (15b)$$

$$\boldsymbol{u}_h^{n+1} = \frac{1}{\delta t\gamma_p}\left(\boldsymbol{\lambda}_h + \sum_{i=1}^{p-1}\alpha_p^i\boldsymbol{d}_h^{n+1-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}, \quad (15c)$$

where $\Gamma_h(\Sigma_{t^{n+1}})$, $\mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $\mathcal{Q}_h(\Omega^f_{t^{n+1}})$ are classical finite element approximation spaces of the functional spaces $\Gamma(\Sigma_{t^{n+1}})$, $\mathcal{V}(\Omega^f_{t^{n+1}})$ and $\mathcal{Q}(\Omega^f_{t^{n+1}})$, respectively.

Again, we consider the linearized version of $\mathsf{MN}_{\delta t,h}(\boldsymbol{\lambda}_h)$ around $\boldsymbol{u}_{*,h}^{n+1}$, denoted by $\mathsf{MN}_{\delta t,h}(\boldsymbol{u}_{*,h}^{n+1}; \boldsymbol{\lambda}_h)$. $\mathsf{MN}_{\delta t,h}(\boldsymbol{\lambda}_h)$ implies the computation of velocities and pressure together. A substantial reduction of the computational cost is obtained when using a splitting technique. These techniques allow the uncoupling of velocity and pressure computation. Herein, we consider a pressure-correction method obtained at the discrete level[12].

We denote by $\mathsf{FS}_{\delta t,h}(\boldsymbol{\lambda}_h)$ the following problem: given $\boldsymbol{\lambda}_h \in \Gamma_h(\Sigma_{t^{n+1}})$, find $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ from the following scheme:

1. Find $\hat{\boldsymbol{u}}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ such that

$$\frac{\rho_f}{\gamma_p \delta t}\left(\hat{\boldsymbol{u}}_h^{n+1} - \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{u}_h^{n-i}, \boldsymbol{v}_h\right) + \mu\left(\nabla \hat{\boldsymbol{u}}_h^{n+1}, \nabla \boldsymbol{v}_h\right)$$
$$+\rho_f\left((\hat{\boldsymbol{u}}_h^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h\right)$$
$$-(\widetilde{p}_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = \rho_f \langle \boldsymbol{f}_f^{n+1}, \boldsymbol{v}_h \rangle \quad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{16a}$$

$$\hat{\boldsymbol{u}}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h - \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i}\right) \quad \text{on } \Sigma_{t^{n+1}}. \tag{16b}$$

2. Find $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that

$$-\gamma_p \delta t\left(\Pi_h\left(\nabla p_h^{n+1} - \nabla \widetilde{p}_h^{n+1}\right), \nabla q_h\right) = \rho_f\left(\hat{\boldsymbol{u}}_h^{n+1}, \nabla q_h\right) \quad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}). \tag{17}$$

3. Find $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ such that

$$\frac{\rho_f}{\delta t \gamma_p}\left(\boldsymbol{u}_h^{n+1} - \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h\right) - (p_h^{n+1} - \widetilde{p}_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = 0 \quad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{18a}$$

$$\boldsymbol{u}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h + \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i}\right) \quad \text{on } \Sigma_{t^{n+1}}. \tag{18b}$$

In step 2, $\widetilde{p}_h^{n+1}$ is *an appropriate approximation to* p_h^{n+1} and Π_h is the L^2 projection onto the velocity space. We consider an *incremental* fractional step method when $\widetilde{p}_h^{n+1} = p_h^n$. This method has an splitting error of order $\mathcal{O}(\delta t^2)$. The results are much better than for *total* projection methods, where $\widetilde{p}_h^{n+1} = 0$, without extra computational cost. Equation (17) of the second step of the method can be approximated by the pressure Poisson equation[12]: find $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that

$$-\gamma_p \delta t\left(\nabla p_h^{n+1} - \nabla \widetilde{p}_h^{n+1}, \nabla q_h\right) = \rho_f\left(\hat{\boldsymbol{u}}_h^{n+1}, \nabla q_h\right) \quad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}). \tag{19}$$

This approximation introduces the same artificial boundary condition that we find when we do the splitting at the continuous level, that is, $\partial p^{n+1}/\partial n = 0$ on the Dirichlet boundary of the velocity. This misbehavior is of special interest for fluid-structure interaction problems, due to the fact that the fluid-structure interface is a Dirichlet boundary. We defend the use of (17) if we want to avoid the artificial boundary conditions. The system matrix associated to (17) is cumbersome, but can be tackled when using an iterative solver.

For the pressure-correction method we only consider the fixed point iteration algorithm using nested loops, as justified below.

When we use an iterative implicit procedure for the coupling, the fluid problem is evaluated (at least) as many times as coupling iterations. Thus, it is natural to put in the momentum equation $\widetilde{p}_h^{n+1} = p_h^{n+1,k}$, $p_h^{n+1,k}$ being the pressure obtained at the previous iteration. In fact, if the resulting scheme converges, *the intermediate velocity* $\boldsymbol{u}_h^{n+1}$ *converges to the end-of-step velocity* $\hat{\boldsymbol{u}}_h^{n+1}$. Furthermore, $\boldsymbol{u}_h^{n+1}$ *converges to the solution of the monolithic fluid system.* Thus we do not need to distinguish between $\hat{\boldsymbol{u}}_h^{n+1}$ and $\boldsymbol{u}_h^{n+1}$ and (18) can be ignored. The final system to be solved at every coupling iteration is the following: given $\boldsymbol{\lambda}_h^k \in \Gamma_h(\Sigma_{t^{n+1}})$ and $p_h^{n+1,k} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$, find $\boldsymbol{u}_h^{n+1,k+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1,k+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that

$$\frac{\rho_f}{\delta t}\left(D_k\boldsymbol{u}_h^{n+1,k+1},\boldsymbol{v}_h\right)+\mu\left(\nabla\boldsymbol{u}_h^{n+1,k+1},\nabla\boldsymbol{v}_h\right)+\rho_f\left((\boldsymbol{u}_h^{n+1,k+1}-\boldsymbol{w}^{n+1})\cdot\nabla\boldsymbol{u}_h^{n+1,k+1},\boldsymbol{v}_h\right)$$
$$-(p_h^{n+1,k},\nabla\cdot\boldsymbol{v}_h)=\rho_f\langle\boldsymbol{f}_s,\boldsymbol{v}_h\rangle \qquad \forall\boldsymbol{v}_h\in\mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \quad (20a)$$

$$-\gamma_p\delta t\left(\Pi_h\left(\nabla p_h^{n+1,k+1}-\nabla p_h^{n+1,k}\right),\nabla q_h\right)=\rho_f(\boldsymbol{u}_h^{n+1,k+1},\nabla q_h) \quad \forall q_h\in\mathcal{Q}_h(\Omega^f_{t^{n+1}}), \quad (20b)$$

$$\boldsymbol{u}_h^{n+1,k+1}=\frac{1}{\delta t\gamma_p}\left(\boldsymbol{\lambda}_h+\sum_{i=0}^{p-1}\alpha_p^i\boldsymbol{d}_h^{n-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}. \quad (20c)$$

This problem is denoted by $\mathsf{PC}_{\delta t,h}(p_h^{n+1,k};\boldsymbol{\lambda}_h^k)$. We remark that in the case presented nested loops are needed: an internal loop to deal with the nonlinearity of the convective term and an external for the convergence to the monolithic fluid system (for fluid problems) or the *monolithic* coupling system (for fluid-structure problems). Again, there is the possibility to use one loop for everything. In this case the fluid solver is denoted by $\mathsf{PC}_{\delta t,h}(\boldsymbol{u}_h^{n+1,k},p_h^{n+1,k};\boldsymbol{\lambda}^k)$.

Method (20) (and obviously its modified version using one loop) are *predictor-corrector* schemes. These methods have been introduced[9] without the fluid-structure motivation. In these references the stabilization terms omitted in the present exposition are carefully treated.

Let us remark that, along this section we have considered $\boldsymbol{w}^{n+1}$ independent of the iterative process for the sake of clarity. However, this is not the general case.

4.2 The discrete fluid domain movement

As commented in the previous section, we use a harmonic extension operator on Ω_t^f in order to obtain $\boldsymbol{d}_h^f$. The discrete problem reads as follows: given $\boldsymbol{\lambda}_h^k \in \Gamma_h(\Sigma_{t^{n+1}})$, find $(\boldsymbol{d}_h^f)^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ such that

$$\left(\nabla(\boldsymbol{d}_h^f)^{n+1},\nabla\boldsymbol{v}_h\right)=0 \qquad \forall\boldsymbol{v}_h\in\mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \quad (21a)$$

$$(\boldsymbol{d}_h^f)^{n+1}=\boldsymbol{\lambda}_h. \quad (21b)$$

We call $(\boldsymbol{d}_h^f)^{n+1}=\mathrm{Ext}_h(\boldsymbol{\lambda}_h)$. The harmonic operator is applied on Ω_t^f because it allows to solve this problem using the same mesh that we use to compute the fluid problem.

4.3 Coupling algorithms for the discrete problem

In this section we propose three different coupling procedures for the pressure segregation methods listed in Section 3, exploiting their properties. We only consider the fixed point algorithms (13) and (14) for the coupling. Let us start with the pressure-correction method. As commented above the use of this method will be restricted to cases where an explicit procedure is used for the coupling. In this case the resulting iterative algorithm is: given $\widetilde{\boldsymbol{\lambda}}_h^{n+1}$, find $\boldsymbol{\lambda}_h^{n+1}$ such that

$$\boldsymbol{\lambda}_h^{n+1} = -\mathcal{S}_s^{-1}(\mathcal{S}_f(\widetilde{\boldsymbol{\lambda}}_h^{n+1})) \tag{22}$$

and $(\boldsymbol{u}_h^{n+1}, p_h^{n+1}) = \mathsf{FS}_{\delta t,h}(\widetilde{\boldsymbol{\lambda}}_h^{n+1})$. Here, $\widetilde{\boldsymbol{\lambda}}_h^{n+1}$ is an appropriate approximation of $\boldsymbol{\lambda}_h^{n+1}$. Different alternatives have been suggested in the literature. A first order approximation in time is $\widetilde{\boldsymbol{\lambda}}_h^{n+1} = \boldsymbol{\lambda}_h^n$. We can consider a more accurate second order approximation that reduces the artificial energy introduced to the system[1]. However, numerical instabilities occur much earlier with the second order predictor (see the numerical experimentation[4]. In this work we have adopted as initial condition

$$\widetilde{\boldsymbol{\lambda}}^{n+1} = -\mathcal{S}_s^{-1}((\boldsymbol{\sigma}_n^f)^n), \tag{23}$$

that is, we solve the structure problem at t^{n+1} using as Neumann boundary condition the normal stress $(\boldsymbol{\sigma}_n^f)^n$ exerted by the fluid at the previous time step. A second order method of this type is

$$\widetilde{\boldsymbol{\lambda}}^{n+1} = -\mathcal{S}_s^{-1}(2(\boldsymbol{\sigma}_n^f)^n - (\boldsymbol{\sigma}_n^f)^{n-1}). \tag{24}$$

A stability analysis of an aeroelastic test case using (23) and (24) together with explicit procedures has been developed[13] with good results.

When using implicit procedures for the coupling we have claimed that predictor-corrector schemes are superior. As commented above, there are some possibilities for the iterative process. Let us start with the one-loop algorithm. For every coupling iteration $k \geq 0$, the problem to be solved is: given $\boldsymbol{\lambda}_h^{n+1,k}$, $\boldsymbol{u}_h^{n+1,k}$ and $p_h^{n+1,k}$, find $\boldsymbol{\lambda}_h^{n+1,k+1}$ such that

$$\boldsymbol{\lambda}_h^{n+1,k+1} = -\mathcal{S}_s^{-1}(\widetilde{\mathcal{S}_f}(\boldsymbol{u}_h^{n+1,k}; \boldsymbol{\lambda}_h^{n+1,k})) \tag{25}$$

with $(\boldsymbol{u}_h^{n+1,k+1}, p_h^{n+1,k+1}) = \mathsf{PC}_{\delta t,h}(\boldsymbol{u}_h^{n+1,k}, p_h^{n+1,k}; \boldsymbol{\lambda}_h^{n+1,k})$. Thus, in the implicit coupling process, we have to solve (25) until convergence. In this method the same loop deals with the non-linearity of the convective term and the convergence to the monolithic system. Some other alternatives for the treatment of the iterations are possible. For instance, the use of nested loops, one for the coupling and one for the non-linearity. This case is similar to (25) but using $\mathcal{S}_f(\boldsymbol{\lambda}_h^{n+1,k})$ together with the fluid solver (20). Further, a third algorithm

could be used. At every coupling iteration k we could iterate over the predictor-corrector method until convergence to the monolithic fluid system. However, for simplicity, we only use (22) and (25) in the numerical experimentation. Alternative versions of (25) can be tested for every application in order to identify which is faster.

5 ON SOME ALGORITHMS FOR AEROELASTICITY

As explained above, the appropriate algorithm for the solution of the coupled system depends on the kind of problem to be solved. In this paper we have in mind aeroelastic problems. Let us draw some features about this sort of applications:

- The fluid solver consumes much more CPU time than the structure. For this reason, the number of fluid evaluations has to be minimized in order to optimize the computational cost.

- The convergence of the coupling iterative process is *easy*. As explained in Section 1, this behavior is associated to the fact that the structure density is much larger than the fluid density.

The use of Newton and Quasi-Newton methods, or the use of preconditioners together with a Richardson iterative process are justified in some cases (for instance haemodynamics) when the convergence of fixed point algorithms is very slow. However, for aeroelastic problems the convergence rate of the last method is good. That, together with the fact that the fixed point algorithm minimizes the number of fluid evaluations per iteration, has motivated its choice for the application to bridge aerodynamics.

Besides, the bottle neck of the coupling method is the fluid solver. In order to reduce the computational cost associated to the fluid solver we suggest the use of fractional step methods. Pressure-correction methods (16)-(18) and predictor-corrector methods (20) are considered. The pressure-correction scheme (in its implicit, semi-implicit or explicit version) is a good choice when using explicit procedures for the coupling. The coupling problem to be solved in this case is the one defined in (22). When using implicit procedures, as explained above for the fully discrete problem, the use of a predictor-corrector scheme is more appropriate, because we can profit from the coupling iterations in order for the fluid solver to tend to the monolithic system (decreasing the splitting error). The nice property of this method is that, when reaching convergence, the solution is the same as that obtained by the *monolithic* approach to the coupled problem. In (25) we have stated the algorithm using only a single loop for the fluid and the coupling.

6 APPLICATIONS

6.1 Bridge Aerodynamics

Among the different topologies of bridges, *suspension* bridges span the greatest distances. However, the bending moments acting on the deck sections of this sort of bridges

are relatively small. Even though the span between piles is very large, the distance between cables, that in fact are working as piles, is small. For this reason these structures are flexible and light. These features make suspension bridges very influenced by wind actions. While for other topologies the aeroelastic behavior is not considered important, for suspension bridges it implies a key aspect of the design process.

In this work we study the flutter phenomenon. This dynamic phenomenon is induced by the fluid-structure coupling (the energy transfer). The flutter happens when the damping induced by the fluid to the structure makes the overall structure damping negative.

The flutter analysis has been developed by experimentation in wind tunnels. We point out that in wind tunnels the flutter limit is not obtained directly, that is, increasing the inflow velocity of the wind tunnel until failure. This flutter limit is obtained evaluating the *aeroelastic derivatives*.

The increasing in the capability of computers together with the improvement of numerical methods have motivated in the last decade the use of computer methods for the analysis of bridge aerodynamics[14].

The present application is devoted to the evaluation of flexural and torsional frequencies of the Great Belt bridge for a given inflow velocity and the direct flutter simulation using the methods introduced in the previous sections. The finite element method together with stabilized predictor-corrector and pressure-correction fluid solvers for the coupling have been used. The ALE framework has allowed to formulate the flow problem in moving domains.

6.2 The bridge model

For the numerical aeroelastic analysis of bridges, the 3D problem is usually reduced to a 2D problem. In fact, this is also the usual procedure for wind tunnel tests. In order to simulate the correct natural frequencies in the fundamental symmetric flexural and torsional modes, spring stiffnesses are applied to the elastic center of the cross-section. Lumped mass and moment of inertia on the gravity center have been introduced to simulate the mass and moment of inertia per unit length. Furthermore, the 2D cross-section is considered a rigid body.

Thus, the linearized ordinary differential equation (ODE) that governs the displacement of the structure reads as follows: find the displacement vector $\boldsymbol{d}_s \in \mathbb{R}^3$ (for a $2d$ problem) such that

$$\boldsymbol{M}\ddot{\boldsymbol{d}}_s + \boldsymbol{C}\dot{\boldsymbol{d}}_s + \boldsymbol{K}\boldsymbol{d}_s = \boldsymbol{f}, \tag{26}$$

where $\boldsymbol{M}$ is the *mass* matrix, $\boldsymbol{C}$ is the *damping* matrix, $\boldsymbol{K}$ is the *stiffness* matrix and $\boldsymbol{f}$ is the external force exerted over the structure (including force and moment). We point out that this external force depends on the displacement of the structure and thus the problem is nonlinear. The damping coefficients are usually given as a *percentage logarithmic decrement*.

For the time integration of the ODE (27) we use the unconditional stable *constant-average-acceleration* scheme, also called *trapezoidal rule*, which is described by the following set of equations:

$$\left\{ \begin{array}{l} \boldsymbol{M}\ddot{\boldsymbol{d}}_s^{\,n+1} + \boldsymbol{C}\dot{\boldsymbol{d}}_s^{\,n+1} + \boldsymbol{K}\boldsymbol{d}_s^{n+1} = \boldsymbol{f}^{n+1}, \\ \boldsymbol{d}_s^{n+1} = \boldsymbol{d}_s^{n} + \delta t \dot{\boldsymbol{d}}_s^{\,n} + \frac{\delta t^2}{4}\left(\ddot{\boldsymbol{d}}_s^{\,n+1} + \ddot{\boldsymbol{d}}_s^{\,n}\right), \\ \dot{\boldsymbol{d}}_s^{\,n+1} = \dot{\boldsymbol{d}}_s^{\,n} + \frac{\delta t}{2}\left(\ddot{\boldsymbol{d}}_s^{\,n+1} + \ddot{\boldsymbol{d}}_s^{\,n}\right). \end{array} \right.$$

This second order accurate scheme is particularly appropriate for the case under consideration due to the fact that preserves the energy of the structure, given by

$$E_s = \frac{1}{2}\dot{\boldsymbol{d}}_s \cdot \boldsymbol{M}\dot{\boldsymbol{d}}_s + \frac{1}{2}\boldsymbol{d}_s \cdot \boldsymbol{K}\boldsymbol{d}_s, \tag{27}$$

which is an important feature when analyzing the aeroelastic stability of the structure.

6.3 The coupling model

In this section we describe the fluid solver on moving domains and the coupling procedure that will be used for the direct analysis of flutter. The coupling procedure that we use herein for the simulation of this phenomenon is implicit. As it is widely known, explicit procedures introduce artificial energy to the system that can lead to undesirable numerical instability[1,4]. Due to the fact that we want to assess the stability of the coupling problem, intimately related to the energy transfer between fluid and structure, it is justified the use of an implicit procedure that avoids this artificial energy. Further, the implicit procedure tends to the solution of the monolithic coupled system, eliminating the splitting error associated to staggered procedures.

Due to the complexity of external flows that appear in aeroelastic applications, and its highly transient behavior, the use of second order methods are worth it, and even more when no extra computational cost is introduced. We have used here the BDF-2 scheme, both for the time integration of the momentum equation and for the evaluation of the mesh velocity in the fluid domain. By doing this, and as it is proved[8] for the convection-diffusion equation, the ALE formulation does not spoil the second order of accuracy of the fluid solver. The movement of the fluid domain has been computed by solving the discrete problem (21).

The formulation for the fluid problem that will be used is a stabilized pressure segregation method. More specifically, a predictor-corrector method is considered because of the fact that we use an implicit procedure, as justified in Section 5. We use a fixed point iterative method to solve the nonlinear interface problem. More precisely, the method used here is the integral version of the iteration scheme stated in (25).

6.4 Assessment of frequencies and direct flutter simulation

This section is devoted to the numerical simulation of the flutter limit and the assessment of frequencies of the Great Belt bridge (Denmark). The parameters that define the problem have been extracted from the literature[14]. The problem domain and its finite element discretization is shown in Figure 6.4. We have used an unstructured mesh of 48453 linear triangles for this simulation. A time step size of 0.01 s has been considered. The horizontal movement is restricted, as it is usually assumed. We do not know which are the appropriate elastic coefficients when analyzing the real sized problem with the real inflow velocity. For this reason we have assumed the elastic coefficients used for the dimensionless approximation analyzed by Selvam *et al.*[14]. It has to be taken into account that this assumption affects the obtained results and complicates the comparison to wind tunnel experiments.

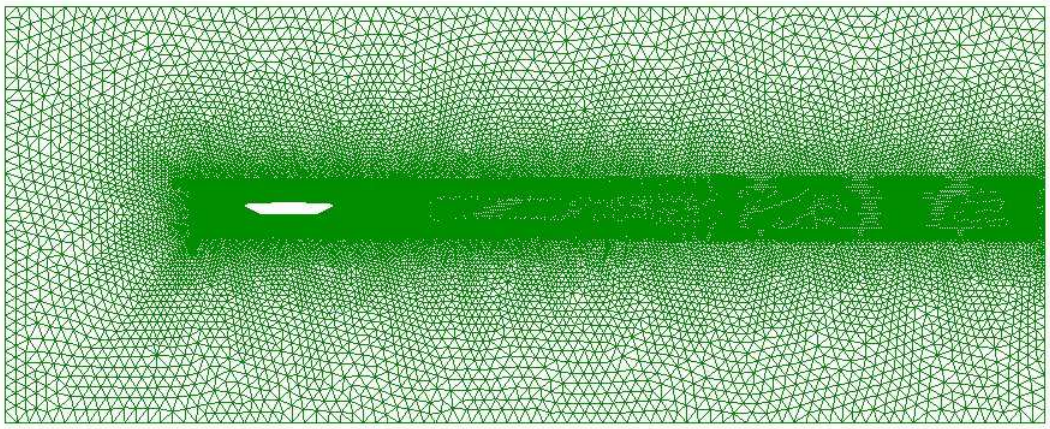
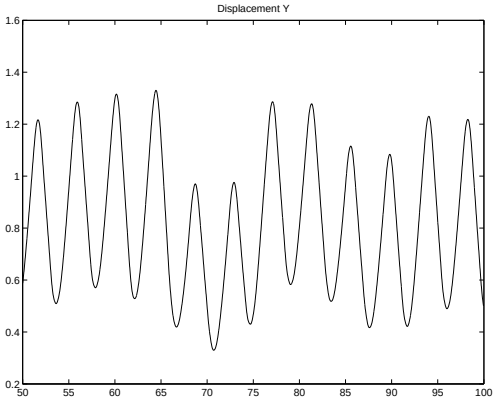


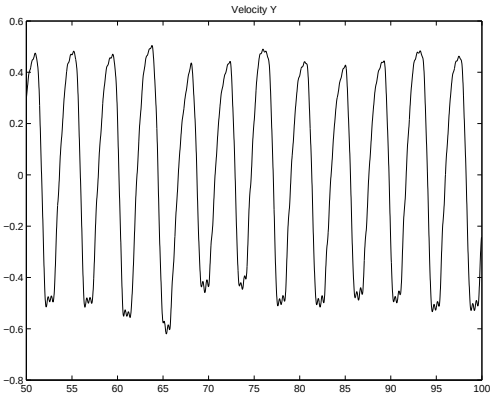
Figure 1: Space domain of analysis and mesh used for the simulation

Firstly, given an inflow velocity of $\boldsymbol{u}_{in} = (50, 0)$ m/s, we obtain the temporary response of the bridge. In figures 2(a), 2(b) and 2(c) we show the vertical displacement, velocity and acceleration. Figures 2(d), 2(e) and 2(f) show the rotation angle, angular velocity and angular acceleration. We plot the results after some time of computation. These plots prove the stability of the structure.

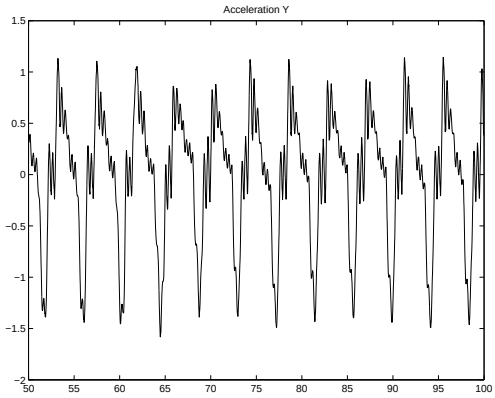
Using a *Fourier Fast Transform* we have obtained the frequencies associated to the vertical displacement (flexural frequency) and rotation angle (torsional frequency). We show these results in Figures 3(a) and 3(b). In both cases a clear dominant frequency governs the movement.



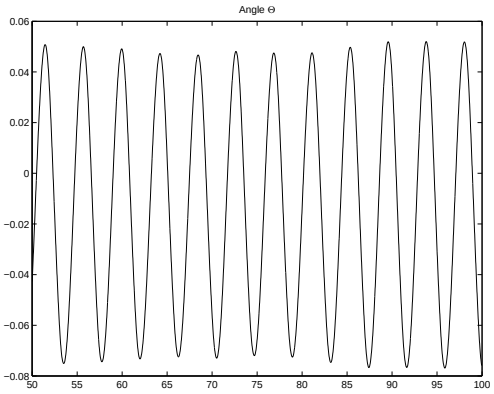
(a) Vertical displacement vs. time



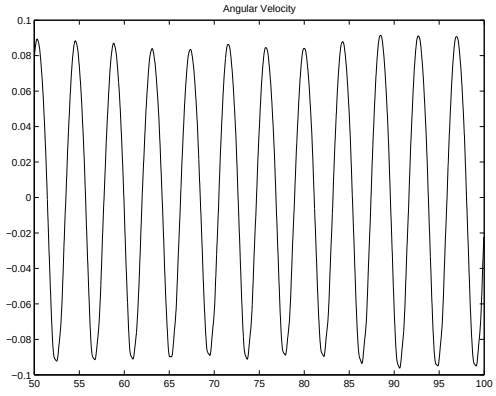
(b) Vertical velocity vs. time



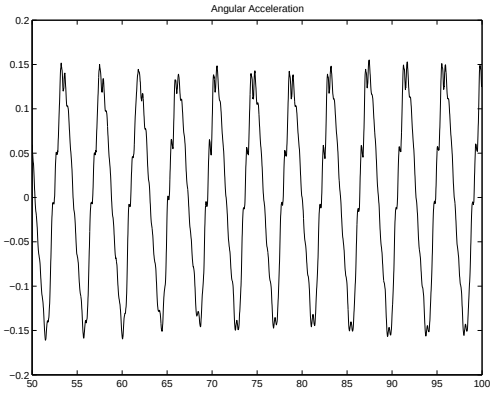
(c) Vertical acceleration vs. time



(d) Rotation angle vs. time

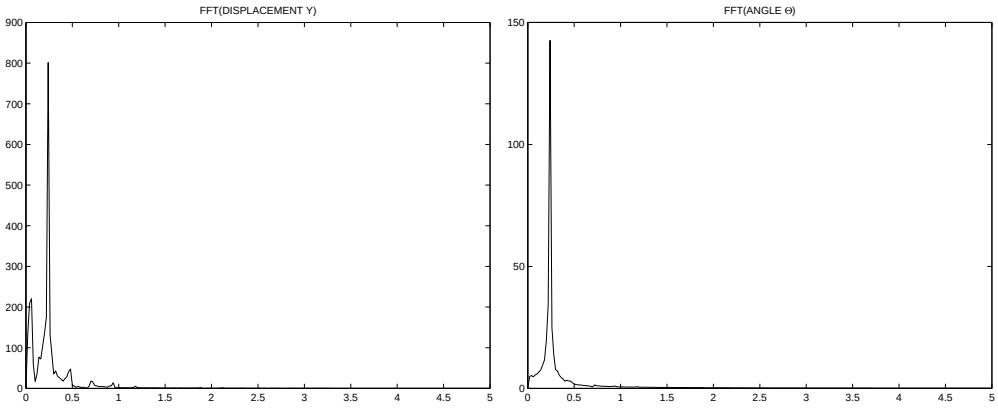


(e) Angular velocity vs. time



(f) Angular acceleration vs. time

Figure 2: Movement of the bridge for an inflow velocity $\boldsymbol{u}_{in} = (50, 0)$ m/s



(a) Vertical displacement: amplitude vs. frequency

(b) Rotation angle: amplitude vs. frequency

Figure 3: Fourier transform of vertical displacement and rotation angle of the bridge for inflow velocity $\boldsymbol{u}_{in} = (50, 0)$ m/s

The average number of iterations needed for the convergence of the integral version of method (25) to the monolithic system for a given time step is around *4 iterations per time step* for an inflow velocity of 50 m/s.

In a second step, we increase the inflow velocity until we reach the aeroelastic instability. The flutter phenomenon appears for an inflow velocity of 55 m/s. We plot the same values as before in figures 4- 5. We easily see in this case that the flutter instability appears for this velocity. In fact, the instability is translational and torsional (see Figures 4(a) and 4(d)). We plot velocities and accelerations for vertical displacement and rotation angle in Figures 4(b)- 4(c) and 4(e)- 4(f). The aerodynamic instability is clearly shown from the increase of the structure energy (Figure 5).

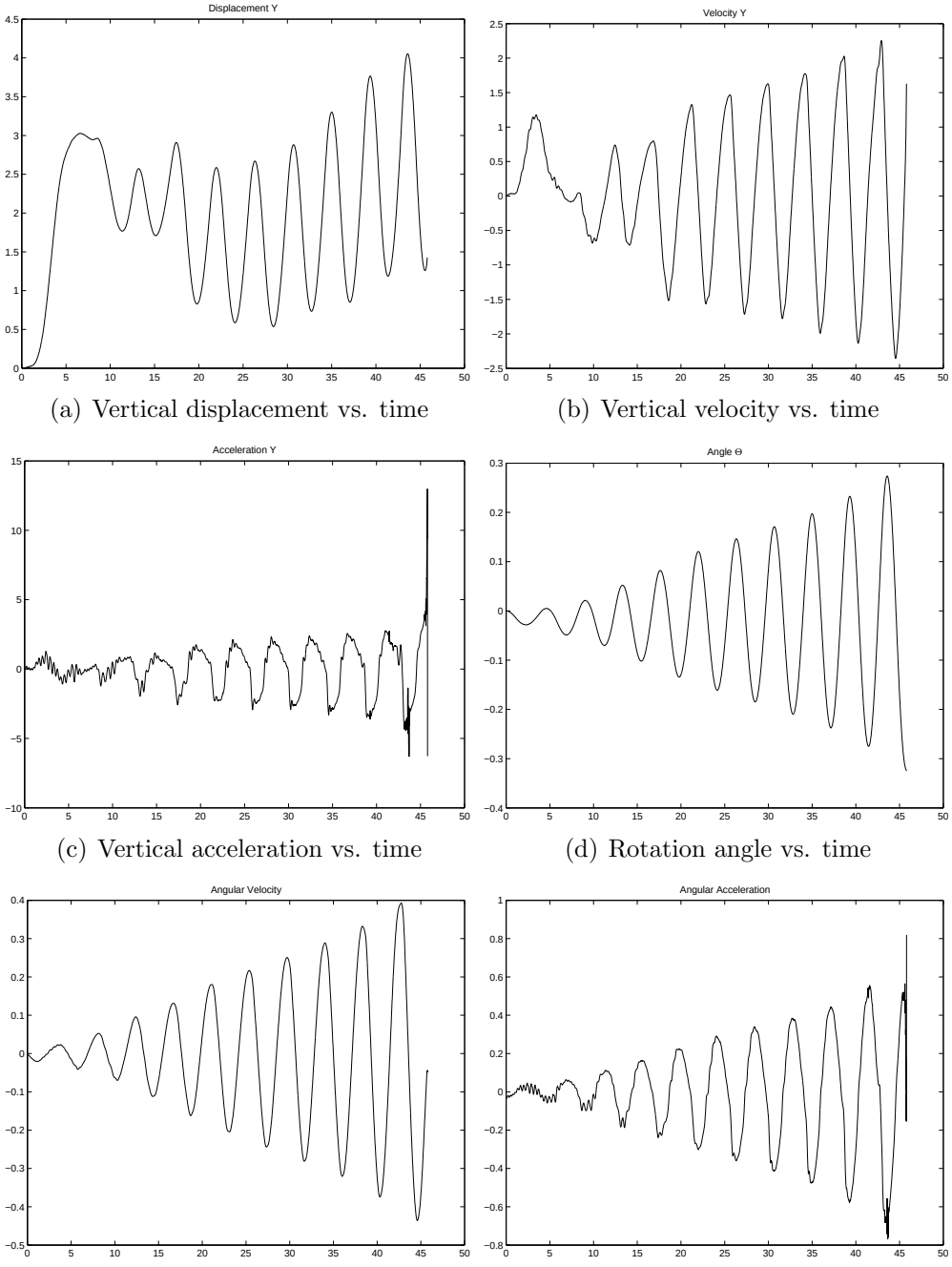
Obviously, the number of iterations needed for the inflow velocity of 55 m/s increases with the structure energy.

7 CONCLUSIONS

The coupling methods proposed herein for the calculation of the flutter limit have shown an excellent behavior in aeroelastic applications. The *coupling method proposed converges to the monolithic problem (for the predictor-corrector solver).* That is, the coupling process does not introduced any extra error (apart from tolerance stop criteria). Furthermore, this method shows a good convergence behavior for this kind of problems.

The other key point is the fact that *the present methods uncouple the velocity and pressure computation*, that implies a high reduction of the computational cost of the fluid problem, the bottle neck of aeroelastic simulations.

Summarizing, the key features of the formulation we propose are the use of second order stabilized pressure segregation methods (both pressure-correction and predictor-corrector versions) together with a second order ALE formulation, a second order structure



(a) Vertical displacement vs. time

(b) Vertical velocity vs. time

(c) Vertical acceleration vs. time

(d) Rotation angle vs. time

(e) Angular velocity vs. time

(f) Angular acceleration vs. time

Figure 4: Movement of the bridge for inflow velocity $\boldsymbol{u}_{in} = (55, 0)$ m/s

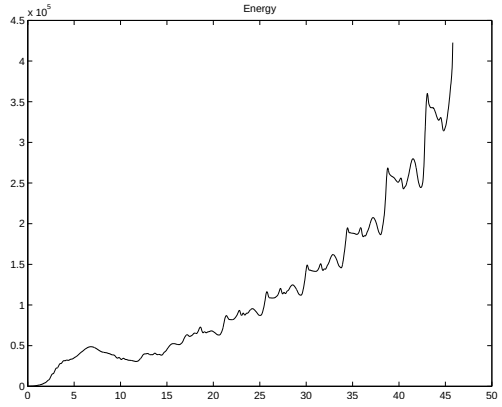


Figure 5: Bridge energy vs. time for inflow velocity $\boldsymbol{u}_{in} = (55, 0)$ m/s

solver and a coupling iterative procedure that tends to the monolithic system. Thus, *the overall fluid-structure coupling procedure* proposed herein *is second order accurate in time*, an important property for highly transient external flows that appear in aeroelastic applications.

We have applied this methods to the aeroelastic analysis of a bridge deck. The flutter velocity of 55 m/s obtained herein differs from the 65-70 m/s obtained from the aeroelastic derivatives assessed with wind tunnel tests. However, this gap could be expected, since the problems solved are different. It seems that the elastic coefficient that should be used for the direct analysis of flutter in dimensional form has to be higher than the one used for the scaled problem.

In fact, numerical experiments using the same method (even the same software) are in exceptional agreement with the wind tunnel results when assessing the aeroelastic derivatives[13]. This is even more relevant considering that in this reference an explicit coupling procedure has been used together with a pressure-correction method, thus introducing a splitting error and artificial energy.

REFERENCES

[1] S. Piperno and C. Farhat. Partitioned prodecures for the transient solution of coupled aeroelastic problems-Part II: energy transfer analysis and three-dimensional applications. *Computer Methods in Applied Mechanics and Engineering*, 190:3147–3170, 2001.

[2] M.A. Fernández and M. Moubachir. A Newton method using exact Jacobians for solving fluid-structure coupling. *Computers & Structures*, 83(2-3):127–142, 2005.

[3] J.F. Gerbeau and M. Vidrascu. Quasi-Newton algorithm based on a reduced model for fluid-structure interaction problems in blood flows. *Mathematical Modelling and Numerical Analysis*, 37(4):631–647, 2003.

[4] D.P. Mok and W.A. Wall. Partitioned analysis schemes for transient interaction of incompressible flows and nonlinear flexible structures. In *Trends in computational structural mechanics (W.A. Wall, K.U. Bletzinger and K. Schweizerhof, Eds.)*, CIMNE, Barcelona, Spain, 2001.

[5] S. Deparis. *Numerical analysis of axisymmetric flows and methods for fluid-structure interaction arising in blood flow simulation*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2004.

[6] P. Causin, J.F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4506–4527, 2001.

[7] L. Formaggia and F. Nobile. A stability analysis for the Arbitrary Lagrangian Eulerian formulation with finite elements. *East-West J. Num. Math.*, 7:105–132, 1999.

[8] S. Badia and R. Codina. Analysis of a stabilized finite element approximation of the transient convection-diffusion equation using an ALE framework. *SIAM Journal on Numerical Analysis*, submitted.

[9] R. Codina and O. Soto. Approximation of the incompressible Navier–Stokes equations using orthogonal–subscale stabilization and pressure segregation on anisotropic finite element meshes. *Computer Methods in Applied Mechanics and Engineering*, 193:1403–1419, 2004.

[10] R. Codina. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190:2681–2706, 2001.

[11] P. Le Tallec and J. Mouro. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering*, 190:3039–3067, 2001.

[12] R. Codina. Pressure stability in fractional step finite element methods for incompressible flows. *Journal of Computational Physics*, 170:112–140, 2001.

[13] R. Rossi. *Light weight Structures: Structural Analysis and Coupling Issues*. PhD thesis, Universitá di Bologna, 2005.

[14] R.P. Selvam and S. Govindaswamy. Aeroelasic analysis of bridge girder section using computer modelling. Technical Report, University of Arkansas, 2001.