INTERNATIONAL CENTER FOR NUMERICAL METHODS IN ENGINEERING

On some Fluid-Structure Iterative Algorithms using Pressure Segregation Methods. Application to Aeroelasticity

S. Badia
R. Codina

Publication CIMNE Nº-287, May 2006

On some Fluid-Structure Iterative Algorithms using Pressure Segregation Methods. Application to Aeroelasticity

S. Badia
R. Codina

Publication CIMNE Nº-287, May 2006

International Center for Numerical Methods in Engineering
Gran Capitán s/n, 08034 Barcelona, Spain

On some fluid-structure iterative algorithms using pressure segregation methods. Application to aeroelasticity

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.
ramon.codina@upc.es, sbadia@cimne.upc.es

March 28, 2006

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 (see [2]). *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 numerical simulation of the fluid-structure coupled problem is complicated. It does not only inherit the difficulties associated to the fluid and solid simulations, but the coupling of these two systems is also cumbersome in many situations. The difficulties arising from this coupled system depend strongly on the physical properties of the case to be simulated. Thus, the choice of an appropriate algorithm that deals well with the coupling varies with the problem to solve. For instance, applications in aeroelasticity and haemodynamics have very different behavior for the same coupling algorithm. Whereas for aeroelastic problems there is a clear tendency to solve the coupled system using explicit procedures, these methods are not appropriate in most haemodynamics applications. In the last case, the use of special implicit procedures for the coupling are required in order to reach good convergence. This situation can be explained by the *added mass* effect (see [5]). When the structure density ρ_s is much larger than the fluid density ρ_f (as it happens in aeroelasticity), the coupling procedure is more stable. On the contrary, when the fluid and structure densities are of the same order (as in haemodynamics) the *added mass* introduced by the fluid over the structural problem makes the convergence of the coupling algorithm much more involved.

The use of explicit procedures for the coupling has been deeply studied by Farhat and co-authors in the framework of aeroelasticity in [16, 18, 19, 36]. Therein they suggest improved explicit procedures that minimize the virtual energy introduced by the explicit algorithm, and added emphasis is placed on paralelization. The application they have in mind is the interaction between an aircraft and a compressible flow surrounding it. Fluid-structure algorithms for aeroelastic problems have also been used in civil and mechanical engineering.

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 (see [20, 23]). In [42, 14] some methods motivated from a domain decomposition approach to the fluid-structure problem have been proposed. 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 (see [34, 13]). In [5] a simplified blood-vessel system is studied, giving a nice explanation of the *added mass* effect and the big impact of the relaxation on the convergence. Alternatively, some kind of relaxation can be introduced with a pseudo-compressibility. The introduction of a pseudo-compressibility that vanishes when the convergence of the coupling is reached has been used in [38] for the simulation of a fluid in an elastic cavity.

We can say that, for a given time step size, explicit procedures are cheaper than implicit procedures. However, explicit procedures are also less accurate. Moreover, when using explicit techniques we are restricted to *small enough* time step sizes or otherwise the solution explodes. Implicit procedures allow larger time step sizes. But, depending on the problem, convergence can be a delicate aspect and involved implicit procedures can be required.

Herein we want to obtain appropriate algorithms for the simulation of fluid-structure problems using finite element methods. The interpretation of the coupling of the fluid and structure as a domain decomposition method without overlapping used in [37] is adopted. Further, the *linearization* of the Steklov-Poincaré operator associated to the fluid is suggested and exploited. We apply these algorithms to the aeroelastic analysis of bridges. We assume a Newtonian and incompressible fluid. The structure, as it is usually done in the analysis of these problems, is considered a rigid body with elastic coefficients in the rigid body motion degrees of freedom.

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 in [21, 24, 3, 1]. Comments about the relationship between the stability of these methods and the *geometric conservation law* can be found in [35, 17]. The ALE scheme has an intrinsic error in time that can spoil the accuracy of the fluid solver in fixed domains. For this reason, an appropriate ALE scheme depends on the time accuracy of the fluid solver for fixed domains. The ALE approach involves the movement of the domain (mesh) with appropriate Dirichlet boundary conditions. This movement is defined by a mesh displacement. Different techniques have been proposed for its computation. The most widely used is the harmonic extension of the Dirichlet functions on the boundaries, being this methodology the one adopted in the present work.

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 (see [12, 10, 11]). On the other hand, we use the orthogonal subgrid scale stabilized finite element method (see [8]) 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. Nevertheless, in the application we have considered, the structure is considered a rigid body. Thus, the computational cost of the structure is much lower than the computational cost of the fluid.

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 $t = 0$. We also define its restriction $\mathcal{A}_t^s := \mathcal{R}_{\Omega_t^s}(\mathcal{A}_t)$ over Ω_t^s and $\mathcal{A}_t^f := \mathcal{R}_{\Omega_t^f}(\mathcal{A}_t)$ over Ω_t^f, such that

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

being again Ω_0^s and Ω_0^f the fluid and structure domains at $t = 0$.

From the trace theorem (see [37]) applied to $\mathcal{A}_t$, we know that

$$\mathcal{A}_t^s|_{\Sigma_t} = \mathcal{A}_t^f|_{\Sigma_t},$$

where $|_{\Sigma_t}$ denotes the restriction to Σ_t. We stress the fact that $\mathcal{A}_t$ is arbitrary.

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 $\mathcal{A}_t(\boldsymbol{x}_0, t) = \boldsymbol{x}_0 + \widehat{\boldsymbol{d}}(\boldsymbol{x}_0, t)$. As before, we split the domain displacement into its fluid and structure restriction as $\boldsymbol{d} = \mathcal{R}_{\Omega_t^s}\boldsymbol{d} + \mathcal{R}_{\Omega_t^f}\boldsymbol{d} =: \boldsymbol{d}^s + \boldsymbol{d}^f$. Again, 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 $\boldsymbol{\sigma}_n^f := \boldsymbol{\sigma}^f|_{\Sigma_t} \cdot \boldsymbol{n}_f$ the normal stress on Σ_t.

Let us recall the *Reynolds transport formula*. Let $\psi(\boldsymbol{x}, t)$ be a function defined on Ω_t. Then, for any subdomain $V_t \subseteq \Omega_t$ such that $V_t = \mathcal{A}_t(V_0)$ with $V_0 \subseteq \Omega_0$ it holds that

$$\frac{d}{dt} \int_{V_t} \psi(\boldsymbol{x}, t) \, \mathrm{d}V = \int_{V_t} \left(\left. \frac{\partial \psi}{\partial t} \right|_{\boldsymbol{x}_0} + \psi \nabla \cdot \boldsymbol{w} \right) \mathrm{d}V. \tag{4}$$

At this point, using expression (4) 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

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

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega_t^f \times (0,T). \tag{5}$$

Remark 1 *We remark that formulations (3) and (5) are equivalent at the continuous level.*

The structure can easily handle with moving domains using a fully Lagrangian framework. For instance, if we consider an elastic structure, the problem that governs the displacement field on the structure is: find $\widehat{\boldsymbol{d}}^s(\boldsymbol{x}_0, t)$ such that

$$\rho_s \frac{\partial^2 \widehat{\boldsymbol{d}}^s}{\partial t^2} - \nabla|_{\boldsymbol{x}_0} \cdot (\widehat{\boldsymbol{\sigma}}^s(\widehat{\boldsymbol{d}}^s)) = \rho_s \boldsymbol{f}_s \qquad \text{in } \Omega_0^s \times (0,T), \tag{6}$$

where ρ_s is the solid density, $\boldsymbol{f}_s$ is the vector of body forces exerted on the solid and $\widehat{\boldsymbol{\sigma}}^s(\boldsymbol{x}_0, t)$ is the Piola-Kirchoff stress tensor for the solid at the reference configuration. We denote by $\boldsymbol{\sigma}_n^s := \boldsymbol{\sigma}^s|_{\Sigma_t} \cdot \boldsymbol{n}_s$ the normal stress on Σ_t.

The displacement of the structure domain $\boldsymbol{d}^s$ has been assumed equal to the structure displacement obtained from (6).

The introduction of boundary conditions on Σ_t for problems (5) and (6) in order for the heterogeneous problem to be well-posed are stated below.

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_t^f, that we denote by

$$\boldsymbol{d}^f = \mathrm{Ext}\left(\boldsymbol{d}^s|_{\Sigma_t}\right).$$

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_t^f. In this case, $\boldsymbol{d}^f$ is solution of the Laplace problem

$$\Delta \boldsymbol{d}^f = \boldsymbol{0} \qquad \text{in } \Omega_t^f \times (0,T), \tag{7a}$$

$$\boldsymbol{d}^f = \boldsymbol{d}^s \qquad \text{on } \Sigma_t \times (0,T). \tag{7b}$$

This extension is different from the harmonic extension evaluated at Ω_0^f, used, e.g., in [35].

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):

$$\boldsymbol{u} = \frac{\partial \widehat{\boldsymbol{d}}^s}{\partial t} \qquad \text{on } \Sigma_t \times (0,T), \tag{8}$$

$$\boldsymbol{\sigma}_n^f + \boldsymbol{\sigma}_n^s = \boldsymbol{0} \qquad \text{on } \Sigma_t \times (0,T). \tag{9}$$

Then, the fluid-structure coupled problem is completely defined by the fluid problem (5), the structure problem (6), the fluid domain displacement (7) and the interface matching conditions (1), (8) and (9). 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

$$\begin{aligned}
\mathcal{V}(\Omega_t^f) &:= \left\{ \boldsymbol{v} : \Omega_t^f \to \mathbb{R}^d, \boldsymbol{v} = \hat{\boldsymbol{v}} \circ (\mathcal{A}_t^f)^{-1}, \hat{\boldsymbol{v}} \in (H^1(\Omega_0^f))^d \right\}, \\
\mathcal{V}_0(\Omega_t^f) &:= \left\{ \boldsymbol{v} \in \mathcal{V}(\Omega_t^f) | \, \boldsymbol{v}|_{\Sigma_t} = \boldsymbol{0} \right\}, \\
\mathcal{Y}(\Omega_t^s) &:= \left\{ \boldsymbol{y} : \Omega_t^s \to \mathbb{R}^d, \boldsymbol{y} = \hat{\boldsymbol{y}} \circ (\mathcal{A}_t^s)^{-1}, \hat{\boldsymbol{y}} \in (H^1(\Omega_0^s))^d \right\}, \\
\mathcal{Q}(\Omega_t^f) &:= \left\{ q : \Omega_t^f \to \mathbb{R}, q = \hat{q} \circ (\mathcal{A}_t^f)^{-1}, \hat{q} \in L^2(\Omega_0^f) \right\}, \\
\Gamma(\Sigma_t) &:= \left\{ \boldsymbol{\gamma} : \Sigma_t \to \mathbb{R}^d, \boldsymbol{\gamma} = \hat{\boldsymbol{\gamma}} \circ (\mathcal{A}_t|_{\Sigma_t})^{-1}, \hat{\boldsymbol{\gamma}} \in (H^{1/2}(\Sigma_0))^d \right\},
\end{aligned}$$

will allow us to write the governing equations of the fluid (5) and structure (6) 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 $\langle f, g \rangle_\omega := \int_\omega f g \, \mathrm{d}\omega$, 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.

Assuming that $\boldsymbol{u}(t)$ is continuous in time for simplicity, the variational form of (5) for a given time value $t \in (0,T)$ reads: find $\boldsymbol{u}(t) \in \mathcal{V}(\Omega_t^f)$ and $p(t) \in \mathcal{Q}(\Omega_t^f)$ such that

$$\begin{aligned}
\rho_f \left(\frac{\partial \boldsymbol{u}}{\partial t}, \boldsymbol{v} \right) + \mu \left(\nabla \boldsymbol{u}, \nabla \boldsymbol{v} \right) + \rho_f \left((\boldsymbol{u} - \boldsymbol{w}) \cdot \nabla \boldsymbol{u}, \boldsymbol{v} \right) - (p, \nabla \cdot \boldsymbol{v}) &= \rho_f \langle \boldsymbol{f}_f, \boldsymbol{v} \rangle \quad \forall \boldsymbol{v} \in \mathcal{V}_0(\Omega_t^f), \\
(\nabla \cdot \boldsymbol{u}, q) &= 0 \qquad\qquad \forall q \in \mathcal{Q}(\Omega_t^f).
\end{aligned}$$

The weak form of the structure system (6) for a given time value $t \in (0,T)$ is: find $\widehat{\boldsymbol{d}}^s(t) \in \mathcal{Y}(\Omega_0^s)$ such that

$$\rho_s \left(\frac{\partial^2 \widehat{\boldsymbol{d}}^s}{\partial t^2}, \boldsymbol{y} \right) + \left((\widehat{\boldsymbol{\sigma}}^s(\widehat{\boldsymbol{d}}^s)), \nabla|_{\boldsymbol{x}_0} \boldsymbol{y} \right) = \rho_s \langle \boldsymbol{f}_s, \boldsymbol{y} \rangle + \langle \boldsymbol{\sigma}_n^s, \boldsymbol{y} \rangle_{\Sigma_t} \qquad \forall \boldsymbol{y} \in \mathcal{Y}(\Omega_0^s).$$

Finally, the boundary conditions that have to be imposed on Σ_t for the *weak* formulation of the coupled problem are:

$$\begin{aligned} \langle \boldsymbol{\sigma}_n^s, \boldsymbol{\gamma} \rangle_{\Sigma_t} + \langle \boldsymbol{\sigma}_n^f, \boldsymbol{\gamma} \rangle_{\Sigma_t} = 0 \qquad & \forall \boldsymbol{\gamma} \in \Gamma(\Sigma_t) \times (0,T), \\ \boldsymbol{u} = \left. \frac{\partial \boldsymbol{d}^s}{\partial t} \right|_{\Sigma_t} \qquad & \text{on } \Sigma_t \times (0,T), \end{aligned}$$

where $\boldsymbol{\sigma}_n^s$ and $\boldsymbol{\sigma}_n^f$ belong to $(H^{-1/2}(\Sigma_t))^d$.

3 THE DOMAIN DECOMPOSITION APPROACH

In this section we reformulate the fluid-structure problem in a Domain Decomposition (DD onwards) framework, as done in [42] and later works [14, 13]. 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 (5) 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

$$D_p f^{n+1} = \frac{1}{\delta t \gamma_p} \sum_{i=0}^{p} \alpha_p^i f^{n+1-i} \tag{10}$$

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 (see [28]). 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 (see [43]). 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

$$\frac{\rho_f}{\delta t}\left(D_k \boldsymbol{u}^{n+1}, \boldsymbol{v}\right) + \mu\left(\nabla \boldsymbol{u}^{n+1}, \nabla \boldsymbol{v}\right) + \rho_f\left((\boldsymbol{u}^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \boldsymbol{u}^{n+1}, \boldsymbol{v}\right)$$
$$-(p^{n+1}, \nabla \cdot \boldsymbol{v}) = \rho_f \langle \boldsymbol{f}_f^{n+1}, \boldsymbol{v}\rangle \qquad \forall \boldsymbol{v} \in \mathcal{V}_0(\Omega^f_{t^{n+1}}), \tag{11a}$$

$$\left(\nabla \cdot \boldsymbol{u}^{n+1}, q\right) = 0 \qquad \forall q \in \mathcal{Q}(\Omega^f_{t^{n+1}}), \tag{11b}$$

$$\boldsymbol{u}^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda} + \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}^{n-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}. \tag{11c}$$

Borrowing classical concepts from domain decomposition methods, we can define the *Steklov-Poincaré interface operator* (see [37]) for the fluid as follows: $\mathcal{S}_f$ is the Dirichlet-to-Neumann map in Ω^f_t such that

$$\begin{aligned} \mathcal{S}_f : H^{1/2}(\Sigma_t) &\to H^{-1/2}(\Sigma_t) \\ \boldsymbol{\lambda} &\mapsto \boldsymbol{\sigma}^f_n. \end{aligned} \tag{12}$$

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 $\boldsymbol{\sigma}^f_n$. Thus, this is a mapping between the trace of the displacement field $\boldsymbol{d}$ and the space of normal stresses exerted by the fluid. Obviously, this operator depends on the fluid solver used, $\mathsf{FL}_{\delta t}$.

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). This implies that the superposition of problems cannot be used and thus $\mathcal{S}_f$ has to deal also with forcing terms and non-homogeneous boundary conditions.

Analogously for the structure, we define the Steklov-Poincaré operator: $\mathcal{S}_s$ is the Dirichlet-to-Neumann map in Ω^s_t such that

$$\begin{aligned} \mathcal{S}_s : H^{1/2}(\Sigma_t) &\to H^{-1/2}(\Sigma_t) \\ \boldsymbol{\lambda} &\mapsto \boldsymbol{\sigma}^s_n. \end{aligned} \tag{13}$$

In this case, $\mathcal{S}_s$ 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

$$\begin{aligned} \mathcal{S}_s^{-1} : H^{-1/2}(\Sigma_t) &\to H^{1/2}(\Sigma_t) \\ \boldsymbol{\sigma}^s_n &\mapsto \boldsymbol{\lambda}. \end{aligned} \tag{14}$$

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 (9) 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

$$\mathcal{S}_f(\boldsymbol{\lambda}) + \mathcal{S}_s(\boldsymbol{\lambda}) = \mathbf{0}. \tag{15}$$

Thus, using the DD approach the initial coupled problem has been reduced to an interface equation.

An alternative form of the interface equation, obtained by applying the inverse of the Steklov-Poincaré operator $\mathcal{S}_s^{-1}$ in (15), reads as: find $\boldsymbol{\lambda} \in \Gamma(\Sigma_{t^{n+1}})$ such that

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda})) = \boldsymbol{\lambda}. \tag{16}$$

This expression motivates the use of the fixed point algorithm (see [6]). The iterative fixed point procedure can be written as: given $\boldsymbol{\lambda}^k$, with $k \geq 0$, find $\boldsymbol{\lambda}^{k+1}$ such that

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda}^k)) = \boldsymbol{\lambda}^{k+1}, \tag{17}$$

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.

Remark 2 *The solution of the fluid problem $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$ requires nonlinear iterations. Thus, algorithm (17) 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. (11), 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 (17) 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

$$-\mathcal{S}_s^{-1}(\widetilde{\mathcal{S}}_f(\boldsymbol{u}^{n+1,k}; \boldsymbol{\lambda}^k)) = \boldsymbol{\lambda}^{k+1}. \tag{18}$$

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.

Remark 3 *When using the algorithm (18) the same loop deals with the coupling of the fluid and structure systems and the nonlinearity of the fluid equations.*

Remark 4 *The* semi-linearized *fixed point algorithm (18) 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), as suggested in [15], in order to accelerate the iterative process.*

Alternative forms of the interface equation (15) motivate different iterative algorithms for the coupling. For instance, if we rewrite (15) as

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda})) - \boldsymbol{\lambda} = \mathbf{0}, \tag{19}$$

it motivates the use of a root finding technique. The use of the Newton algorithm in order to obtain the root of (19) has been explored in recent works (see e.g. [20]). It involves the computation of the tangent operators of $\mathcal{S}_f$ and $\mathcal{S}_s$. Again, these tangent operators account for the non-linearity of the fluid equations and the shape derivative. Its computation is an involved task. Approximate Jacobians invoking different approximations lead to a variety of Quasi-Newton methods (see [23, 32, 33]).

Deparis *et al.* in [14] use the approach adopted herein in order to motivate new algorithms. These algorithms, widely used as DD methods, are applied to the fluid-structure problem. They consider the preconditioned Richardson method to solve (15): given $\boldsymbol{\lambda}^k$, for $k > 0$, find $\boldsymbol{\lambda}^{k+1}$ such that

$$\mathcal{P}_k(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k) = -\mathcal{S}_f(\boldsymbol{\lambda}^k) - \mathcal{S}_s(\boldsymbol{\lambda}^k) \tag{20}$$

where $\mathcal{P}_k$ is a preconditioner of the Jacobian of $\mathcal{S}_f(\boldsymbol{\lambda}^k) + \mathcal{S}_s(\boldsymbol{\lambda}^k)$. Some alternative choices of $\mathcal{P}_k$ are suggested in [14].

Besides the iterative algorithm for the coupling, a relaxation method is advisable in order to improve the convergence properties of all the previous algorithms. The impact of the relaxation parameter on the convergence of the iterative algorithm for a simple test case has been analyzed in [5]. The Aitken acceleration method is the most widely used. The value of the optimal relaxation parameter for the Aitken technique has a known value for scalar equations. Different alternatives for the extension to the vector case have been proposed in [29, 13].

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 [7, 11, 12, 10, 9]. 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 (11), 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,

$$\frac{\rho_f}{\delta t}\left(D_k \boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h\right) + \mu\left(\nabla \boldsymbol{u}_h^{n+1}, \nabla \boldsymbol{v}_h\right) + \rho_f\left((\boldsymbol{u}_h^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h\right)$$
$$-(p_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = \rho_f \langle \boldsymbol{f}_f, \boldsymbol{v}_h \rangle \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{21a}$$

$$\left(\nabla \cdot \boldsymbol{u}_h^{n+1}, q_h\right) = 0 \qquad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}), \tag{21b}$$

$$\boldsymbol{u}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h + \sum_{i=1}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n+1-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}, \tag{21c}$$

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 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.

Let us introduce how we approximate the continuous spaces with finite dimensional subspaces that can be handled numerically. Let Θ_h^t be a finite element partition of the domain Ω_t^f in a family of elements $\{K_e\}_{e=1,...,n_{el}}$, n_{el} being the number of elements. We denote the diameter of the sphere that circumscribes element K by h_K and the diameter of the sphere inscribed in K by ϱ_K. We also call $h = \max_{K\in\Theta_h^t}(h_K)$ and $\varrho = \min_{K\in\Theta_h^t}(\varrho_K)$. We assume that all the element domains $K \in \Theta_h^t$ are the image of a reference element $\tilde{K}$ through polynomial mappings F_K, affine for simplicial elements, bilinear for quadrilaterals and trilinear for hexahedra. On $\tilde{K}$ we define the polynomial spaces $R_k(\tilde{K})$, where R_k is, for simplicial elements, the set of polynomials in $x_1, ..., x_d$ of degree less than or equal to k, called P_k. For quadrilaterals and hexahedra R_k consists of polynomials in $x_1, ..., x_d$ of degree less than or equal to k in each variable, called Q_k. We also introduce the finite element partition Ξ_h^t of the interface Σ_t, which is completely defined by Θ_h^t. For simplicity, we consider that the finite element partitions of the fluid and solid meshes match on Σ_t, or alternatively, the structure is considered a rigid-body, as in Section 6.

The finite element spaces introduced before and that we will use in the following are:

$$
\begin{aligned}
\mathcal{V}_h(\Omega_0^f) &= \{\widehat{\boldsymbol{v}}_h \in \mathcal{C}^0(\Omega_0^f) \mid \widehat{\boldsymbol{v}}_h|_K = \tilde{\boldsymbol{v}} \circ F_K^{-1},\ \tilde{\boldsymbol{v}} \in R_k(\tilde{K}),\ K \in \Theta_h^t\},\\
\mathcal{Q}_h(\Omega_0^f) &= \{\widehat{q}_h \in \mathcal{C}^0(\Omega_0^f) \mid \widehat{q}_h|_K = \tilde{q} \circ F_K^{-1},\ \tilde{q} \in R_k(\tilde{K}),\ K \in \Theta_h^t\},\\
\Gamma_h(\Sigma_0) &= \{\widehat{\gamma}_h \in \mathcal{C}^0(\Sigma_0) \mid \widehat{\gamma}_h|_K = \tilde{\gamma} \circ F_K^{-1},\ \tilde{\gamma} \in R_k(\tilde{K}),\ K \in \Xi_h^t\},\\
\mathcal{V}_h(\Omega_{t^{n+1}}^f) &= \{\boldsymbol{v}_h \in \mathcal{C}^0(\Omega_{t^{n+1}}^f) \mid \boldsymbol{v}_h = \widehat{\boldsymbol{v}}_h \circ \mathcal{A}_t^{-1},\ \widehat{\boldsymbol{v}}_h \in \mathcal{V}_h(\Omega_0^f)\},\\
\mathcal{V}_{h,0}(\Omega_{t^{n+1}}^f) &= \left\{\boldsymbol{v}_h \in \mathcal{V}_h(\Omega_{t^{n+1}}^f) | \ \boldsymbol{v}_h|_{\Sigma_{t^{n+1}}} = \boldsymbol{0}\right\},\\
\mathcal{Q}_h(\Omega_{t^{n+1}}^f) &= \{q_h \in \mathcal{C}^0(\Omega_{t^{n+1}}^f) \mid q_h = \widehat{q}_h \circ \mathcal{A}_t^{-1},\ \widehat{q}_h \in \mathcal{Q}_h(\Omega_0^f)\},\\
\Gamma_h(\Sigma_t) &= \{\gamma_h \in \mathcal{C}^0(\Sigma_t) \mid \gamma_h = \widehat{\gamma}_h \circ \mathcal{A}_t^{-1},\ \widehat{\gamma}_h \in \Gamma_h(\Sigma_0)\}.
\end{aligned}
$$

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 (see [7]).

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 $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_h(\Omega_{t^{n+1}}^f)$ and $p_h^{n+1} \in \mathcal{Q}_h(\Omega_{t^{n+1}}^f)$ from the following scheme:

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

$$
\begin{aligned}
\frac{\rho_f}{\gamma_p \delta t}\left(\hat{\boldsymbol{u}}_h^{n+1} - \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{u}_h^{n-i}, \boldsymbol{v}_h\right) + \mu\left(\nabla \hat{\boldsymbol{u}}_h^{n+1}, \nabla \boldsymbol{v}_h\right)&\\
+\rho_f\left((\hat{\boldsymbol{u}}_h^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h\right)&\\
-(\widetilde{p}_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = \rho_f \langle \boldsymbol{f}_f^{n+1}, \boldsymbol{v}_h\rangle \quad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega_{t^{n+1}}^f),& \qquad (22a)\\
\hat{\boldsymbol{u}}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h - \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i}\right) \quad \text{on } \Sigma_{t^{n+1}}.& \qquad (22b)
\end{aligned}
$$

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

$$
-\gamma_p \delta t\left(\Pi_h\left(\nabla p_h^{n+1} - \nabla \widetilde{p}_h^{n+1}\right), \nabla q_h\right) = \rho_f\left(\hat{\boldsymbol{u}}_h^{n+1}, \nabla q_h\right) \quad \forall q_h \in \mathcal{Q}_h(\Omega_{t^{n+1}}^f). \qquad (23)
$$

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

$$
\begin{aligned}
\frac{\rho_f}{\delta t \gamma_p}\left(\boldsymbol{u}_h^{n+1} - \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h\right) - (p_h^{n+1} - \widetilde{p}_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = 0 \quad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega_{t^{n+1}}^f),& \qquad (24a)\\
\boldsymbol{u}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h + \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i}\right) \quad \text{on } \Sigma_{t^{n+1}}.& \qquad (24b)
\end{aligned}
$$

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 (23) of the second step of the method can be approximated by the pressure Poisson equation (see [7]): find $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that

$$-\gamma_p \delta t \left(\nabla p_h^{n+1} - \nabla \widetilde{p}_h^{n+1}, \nabla q_h\right) = \rho_f \left(\hat{\boldsymbol{u}}_h^{n+1}, \nabla q_h\right) \qquad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}). \tag{25}$$

Remark 5 *This approximation introduces the same artificial boundary condition that we find when we do the splitting at the continuous level (see [43]), 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. Thus, an artificial boundary condition over the pressure is imposed on the surface where the pressure is integrated for the calculation of the stresses exerted by the fluid. We defend the use of (23) if we want to avoid the artificial boundary conditions. The system matrix associated to (23) is cumbersome, but can be tackled when using an iterative solver, case in which only matrix-vector products are needed. Furthermore, the use of a closed integration rule for approximating the Gramm (mass) matrix that appears in (23) reduces considerably the computational cost. Nevertheless, it would be interesting to assess the impact of the artificial boundary condition on fluid-structure problems.*

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 (24) 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

$$\frac{\rho_f}{\delta t}\left(D_k \boldsymbol{u}_h^{n+1,k+1}, \boldsymbol{v}_h\right) + \mu\left(\nabla \boldsymbol{u}_h^{n+1,k+1}, \nabla \boldsymbol{v}_h\right) + \rho_f\left((\boldsymbol{u}_h^{n+1,k+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \boldsymbol{u}_h^{n+1,k+1}, \boldsymbol{v}_h\right)$$
$$- (p_h^{n+1,k}, \nabla \cdot \boldsymbol{v}_h) = \rho_f \langle \boldsymbol{f}_s, \boldsymbol{v}_h \rangle \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{26a}$$

$$-\gamma_p \delta t \left(\Pi_h\left(\nabla p_h^{n+1,k+1} - \nabla p_h^{n+1,k}\right), \nabla q_h\right) = \rho_f (\boldsymbol{u}_h^{n+1,k+1}, \nabla q_h) \qquad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}), \tag{26b}$$

$$\boldsymbol{u}_h^{n+1,k+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h + \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}. \tag{26c}$$

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 final system is: given $\boldsymbol{\lambda}_h^k \in \Gamma_h(\Sigma_{t^{n+1}})$, $\boldsymbol{u}_h^{n+1,k} \in$

$\mathcal{V}_h(\Omega^f_{t^{n+1}})$ 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,

$$\rho_f \left(D_k \boldsymbol{u}_h^{n+1,k+1}, \boldsymbol{v}_h \right) + \mu \left(\nabla \boldsymbol{u}_h^{n+1,k+1}, \nabla \boldsymbol{v}_h \right) + \rho_f \left((\boldsymbol{u}_h^{n+1,k} - \boldsymbol{w}^{n+1}) \cdot \nabla \boldsymbol{u}_h^{n+1,k+1}, \boldsymbol{v}_h \right)$$
$$- (p_h^{n+1,k}, \nabla \cdot \boldsymbol{v}_h) = \rho_f \langle \boldsymbol{f}_s, \boldsymbol{v}_h \rangle \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{27a}$$
$$- \gamma_p \delta t \left(\Pi_h \left(\nabla p_h^{n+1,k+1} - \nabla p_h^{n+1,k} \right), \nabla q_h \right)$$
$$= \rho_f \left(\boldsymbol{u}_h^{n+1,k+1}, \nabla q_h \right) \qquad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}), \tag{27b}$$
$$\boldsymbol{u}_h^{n+1,k+1} = \frac{1}{\delta t \gamma_p} \left(\boldsymbol{\lambda}_h + \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i} \right) \qquad \text{on } \Sigma_{t^{n+1}}. \tag{27c}$$

In this case the fluid solver is denoted by $\mathsf{PC}_{\delta t,h}(\boldsymbol{u}_h^{n+1,k}, p_h^{n+1,k}; \boldsymbol{\lambda}^k)$.

Methods (26) and (27) are *predictor-corrector* schemes. These methods have been introduced in [11, 12] without the fluid-structure motivation. In these references the stabilization terms omitted in the present exposition are carefully treated.

Remark 6 *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. How to treat this mesh velocity in the iterative algorithm has been pointed out in Remark 4.*

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

$$\left(\nabla (\boldsymbol{d}_h^f)^{n+1}, \nabla \boldsymbol{v}_h \right) = 0 \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{28a}$$
$$(\boldsymbol{d}_h^f)^{n+1} = \boldsymbol{\lambda}_h. \tag{28b}$$

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 (17) and (18) for the coupling, but these ideas can be easily extended to methods motivated from (19) and (20). Let us start with the pressure-correction method (27). 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

$$\boldsymbol{\lambda}_h^{n+1} = -\mathcal{S}_s^{-1}(\mathcal{S}_f(\widetilde{\boldsymbol{\lambda}}_h^{n+1})) \tag{29}$$

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 $\widetilde{\boldsymbol{\lambda}}_h^{n+1} = \boldsymbol{\lambda}_h^n$. A more accurate second order approximation that reduces the artificial energy introduced to the system is proposed in [36]. However, numerical instabilities occur much earlier with the second order predictor (see the numerical experimentation in [34]). In this work we have adopted as initial condition

$$\widetilde{\boldsymbol{\lambda}}^{n+1} = -\mathcal{S}_s^{-1}((\boldsymbol{\sigma}_n^f)^n), \tag{30}$$

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

$$\widetilde{\boldsymbol{\lambda}}^{n+1} = -\mathcal{S}_s^{-1}(2(\boldsymbol{\sigma}_n^f)^n - (\boldsymbol{\sigma}_n^f)^{n-1}). \tag{31}$$

A stability analysis of an aeroelastic test case (similar to the one in [36]) using (30) and (31) together with explicit procedures has been developed in [39] 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

$$\boldsymbol{\lambda}_h^{n+1,k+1} = -\mathcal{S}_s^{-1}(\widetilde{\mathcal{S}_f}(\boldsymbol{u}_h^{n+1,k}; \boldsymbol{\lambda}_h^{n+1,k})) \tag{32}$$

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 (32) 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 (32) but using $\mathcal{S}_f(\boldsymbol{\lambda}_h^{n+1,k})$ together with the fluid solver (26). 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 (29) and (32) in the numerical experimentation. Alternative versions of (32) can be tested for every application in order to identify which is faster.

Remark 7 *In the algorithm (29) associated to the pressure-correction method, we use the Steklov-Poincaré operator $\mathcal{S}_f(\cdot)$ that involves nonlinear iterations due to the convective term. However, for the predictor-corrector coupling algorithm (32) the semi-linearized version $\mathcal{S}_f(\boldsymbol{u}_h^{n+1,k}; \cdot)$, that does not involve nonlinear iterations, has been used.*

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. We can use explicit and implicit fluid solvers. The first class of solvers is cheaper but the time step allowed is restricted to be smaller than a critical time step size δt_{cr}. Using an implicit procedure the cost per evaluation is more expensive but δt can be larger. It is commonly accepted that to capture well the physics of the flow, a time step size δt of the order of $10 - 100\delta t_{cr}$ should be used.

In order to reduce the computational cost associated to the fluid solver we suggest the use of fractional step methods. Pressure-correction methods (22)-(24) and predictor-corrector methods (26) and (27) 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 (29). This method introduces a splitting error (due to the splitting of the Navier-Stokes equations) and a coupling error (due to the explicit procedure for the coupling). Furthermore, the coupling error affects the stability of the coupling procedure (see e.g. [34]).

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 (32) 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.

The action of a fluid over a structure can induce three different phenomena:

- *Divergence*: It can be considered a *static* instability. It happens when the deformation induced by the fluid to the structure increases the fluid action until failure. It is similar to the *buckling* of a pile loaded by an axial force. This phenomenon is of little importance in bridge design because it happens for very high wind speeds.

- *Buffeting*: This dynamic phenomenon is associated to the effect of the fluctuations of the inflow over the structure. For suspended bridges, the inflow is usually very homogeneous. However, in some conditions it can induce low frequency vibrations. Even though these vibrations do not endanger the structure they can induce fatigue effects.

- *Flutter*: 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. Then, the oscillations of the bridge increase until failure. It happens for high velocities ($\sim 60 - 70$ m/s) of the inflow (but much smaller than the divergence velocity).

In the application described now we consider only the flutter phenomenon, which is the most important aeroelastic effect when designing suspension bridges. When this aeroelastic phenomenon was not taken into account by the engineers it caused some historical failures of bridges. The Tacoma bridge is probably the best known case. Suspension bridges have a very low structural damping that make them sensitive to this effect. One of the most important criteria of design is the *flutter limit* velocity (when flutter occurs). An acceptable structure must have a large enough flutter limit velocity. A large gap between the maximum velocity of design and this limit is required.

The flutter analysis has been developed by experimentation in wind tunnels. For instance, the design of the Great Belt bridge (Denmark) involved more than 16 sections (see [31]). 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*. This methodology, that was originated in aeronautics, was extrapolated to bridge aerodynamics by Scanlan and Tomko in [40]. When using this methodology, the assessment of the effect of the fluid over the structure is made with an inflow velocity far from the flutter limit and prescribed deck motions. This experimentation process is very expensive and time consuming. Further, the Reynolds number of the real problem cannot be reproduced in conventional wind tunnels.

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 [41]. For the bridge analyzed herein, the Great Belt bridge, we refer now to some previous works. Jenssen and Kvamsdal in [30] analyze this bridge using the finite volume method and an explicit procedure for the coupling. The aeroelastic derivatives are computed to obtain the flutter limit. Selvam *et al.* in [41] use the finite element method in a moving (non-inertial) frame of reference for a direct simulation of the flutter. The more recent work of Frandsen [22] uses the finite element method and a monolithic approximation of the coupled fluid-structure problem. This reference includes a good review on this topic.

In our numerical simulation no turbulence modeling has been considered. Due to the fact that the bridge deck is a bluff body, the flow is detached and the influence of the turbulence

effects for this case is less important than for the aeroelastic analysis of wings. However, for wide decks, the flow re-attaches at a given point. Nevertheless, we use a stabilized finite element method motivated by a multiscale approach. There is a recent trend among the computational mechanics community to claim that this kind of methods can replace conventional turbulence models (see [25, 26, 27, 4]).

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. First and second order accurate methods (in time) have been considered.

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.

In order to obtain the equations governing the displacement of the bridge section, Newton's law is formulated on the gravity center, and the spring force depending on the displacement of the structure is applied to the elastic center. When the gravity center and the elastic center are in different positions, the resulting governing equations are nonlinear. However, assuming that the rotation angle is *small*, the equations can be easily linearized. 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

$$\boldsymbol{M}\ddot{\boldsymbol{d}}_s + \boldsymbol{C}\dot{\boldsymbol{d}}_s + \boldsymbol{K}\boldsymbol{d}_s = \boldsymbol{f}, \tag{33}$$

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. The displacement vector contains the translation and rotation of the structure, that is,

$$\boldsymbol{d}_s = \begin{pmatrix} d_x \\ d_y \\ d_\theta \end{pmatrix},$$

where d_x and d_y are the displacements along the x and y directions, respectively, and d_θ is the rotation. The *linearized mass* matrix has the following expression:

$$\boldsymbol{M} = \begin{pmatrix} m & 0 & -s_x \\ 0 & m & s_y \\ -s_x & s_y & I_\theta \end{pmatrix}$$

where m, I_θ, s_x and s_y denote the mass, inertial moment and static moments associated to the elastic center (per unit length), respectively. The stiffness matrix is given by

$$\boldsymbol{K} = \begin{pmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_\theta \end{pmatrix}$$

where k_x, k_y and k_θ are the corresponding stiffness coefficients. The damping coefficients of each degree of freedom define the damping matrix, which is taken as

$$\boldsymbol{C} = \begin{pmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & c_\theta \end{pmatrix}.$$

The damping coefficients are usually given as a *percentage logarithmic decrement*. A $l\%$ logarithmic decrement implies a damping coefficient

$$c = \frac{l}{\pi}\sqrt{mk},$$

m and k being the mass and stiffness coefficients, respectively.

The external force vector (including force and moment) over the bridge exerted by the fluid at a given time is

$$\boldsymbol{f} = \begin{pmatrix} f_x \\ f_y \\ m_\theta \end{pmatrix},$$

and can be obtained as

$$[f_x, f_y]^T = \int_{\Sigma_t} \boldsymbol{\sigma}_n^f \, \mathrm{d}\Sigma,$$
$$m_\theta = \int_{\Sigma_t} \boldsymbol{\sigma}_n^f \times \boldsymbol{r} \, \mathrm{d}\Sigma.$$

where, as before, $\boldsymbol{\sigma}_n^f$ is the normal stress on Σ_t exerted by the fluid, and $\boldsymbol{r}$ is the position vector (being the frame of reference centered at the elastic center). We point out that this external force depends on the displacement of the structure, that is, $\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{d}_s)$, and thus the problem is nonlinear.

For the time integration of the ODE (34) we use the unconditional stable *constant-average-acceleration* scheme, also called *trapezoidal rule*, which is described by the following set of equations:

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

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

$$E_s = \frac{1}{2}\dot{\boldsymbol{d}}_s \cdot \boldsymbol{M}\dot{\boldsymbol{d}}_s + \frac{1}{2}\boldsymbol{d}_s \cdot \boldsymbol{K}\boldsymbol{d}_s, \tag{34}$$

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 (see [36] and [34]). 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 in [1] 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 (28).

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 point out that, when the structure is considered a rigid body, the interface equation (16) has the following *integrated* form,

$$-\mathcal{S}_s^{-1}\left(\int_{\Sigma_t}\mathcal{S}_f(\boldsymbol{\lambda})\,\mathrm{d}\Sigma\right)=\boldsymbol{\lambda}. \tag{35}$$

where $\boldsymbol{\lambda}=\boldsymbol{\lambda}(\boldsymbol{d}_s)$ and $\mathcal{S}_s(\boldsymbol{\lambda})$ gives the forces and moments (not the stresses) that cause a displacement $\boldsymbol{\lambda}$. Likewise, $\mathcal{S}_f(\boldsymbol{\lambda})$ contains not only the components of the normal stress exerted by the fluid, but also the moments per unit of area (length if $d=2$), and therefore the integral of $\mathcal{S}_f(\boldsymbol{\lambda})$ gives the total force and moment exerted by the fluid on Σ_t.

In this case we use a fixed point iterative method to solve the nonlinear interface problem (35). More precisely, the method used here is the integral version of the iteration scheme stated in (32).

Even though this kind of problems have a good convergence, we have used the Aitken acceleration technique for scalar equations. We define the residual of the interface equation as

$$\boldsymbol{r}(\boldsymbol{\lambda}^k)=-\mathcal{S}_s^{-1}\left(\int_{\Sigma_t}\mathcal{S}_f(\boldsymbol{\lambda}^k)\,\mathrm{d}\Sigma\right)-\boldsymbol{\lambda}^k.$$

Exploiting the fact that the structure is considered as a rigid body, the relaxation parameter can be obtained from the expression for scalar equations. In this case, we consider the *diagonal relaxation matrix*

$$\boldsymbol{\omega}^k=\begin{pmatrix}\omega_x^k & 0 & 0\\ 0 & \omega_y^k & 0\\ 0 & 0 & \omega_\theta^k\end{pmatrix},$$

that verifies

$$\boldsymbol{\omega}^k(\boldsymbol{r}(\boldsymbol{\lambda}^k) - \boldsymbol{r}(\boldsymbol{\lambda}^{k-1})) = \boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}.$$

The *relaxed* version of a fixed point iteration applied to (35) is

$$\boldsymbol{\lambda}^{k+1} = -\boldsymbol{\omega}^k \mathcal{S}_s^{-1} \left(\int_{\Sigma_t} \mathcal{S}_f(\boldsymbol{\lambda}^k)\, \mathrm{d}\Sigma \right) + (\boldsymbol{I} - \boldsymbol{\omega}^k)\boldsymbol{\lambda}^k.$$

A deep study of relaxation methods in a fluid-structure framework can be found in [13].

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 summarized in Table 1 and have been extracted from [41]. 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.* in [41]. It has to be taken into account that this assumption affects the obtained results and complicates the comparison to wind tunnel experiments.

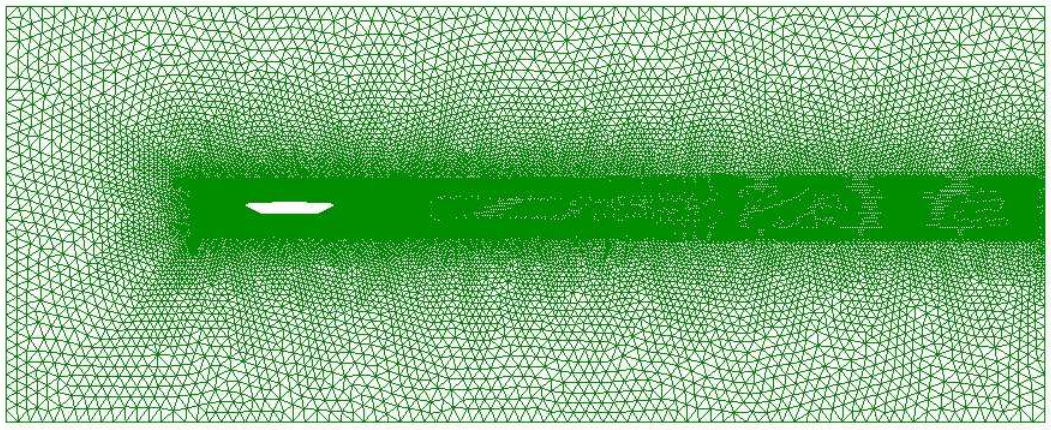


Figure 1: Space domain of analysis and mesh used for the simulation

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

Mass per unit length, $m\ [Kg/m]$	$2.27\ \times 10^3$
Vertical static moment on elastic centre per unit length, $s_y\ [Kg \cdot m/m]$	$1.61\ \times 10^4$
Mass moment of inertia on elastic centre per unit length, $I_\theta\ [Kg \cdot m^2/m]$	$2.47\ \times 10^6$
Vertical spring stiffness, $k_y\ [N/m^2]$	8.78×10^3
Torsional spring stiffness, $k_\theta\ [N \cdot m/m^2]$	7.21×10^6
Vertical logarithmic damping, $l_y\ [\%]$	1
Torsional logarithmic damping, $l_\theta\ [\%]$	0.6

Table 1: Properties of the Great Belt Bridge

acceleration. We plot the results after some time of computation. In figure 3 we plot the energy of the structure, defined in (34). 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 4(a) and 4(b). In both cases a clear dominant frequency governs the movement.

We show contours of the velocity norm and pressure at different time steps in Figures 5 and 6, respectively.

The average number of iterations needed for the convergence of the integral version of method (32) 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 7- 8. We easily see in this case that the flutter instability appears for this velocity. In fact, the instability is translational and torsional (see Figures 7(a) and 7(d)). We plot velocities and accelerations for vertical displacement and rotation angle in Figures 7(b)-7(c) and 7(e)- 7(f). The aerodynamic instability is clearly shown from the increase of the structure energy (Figure 8).

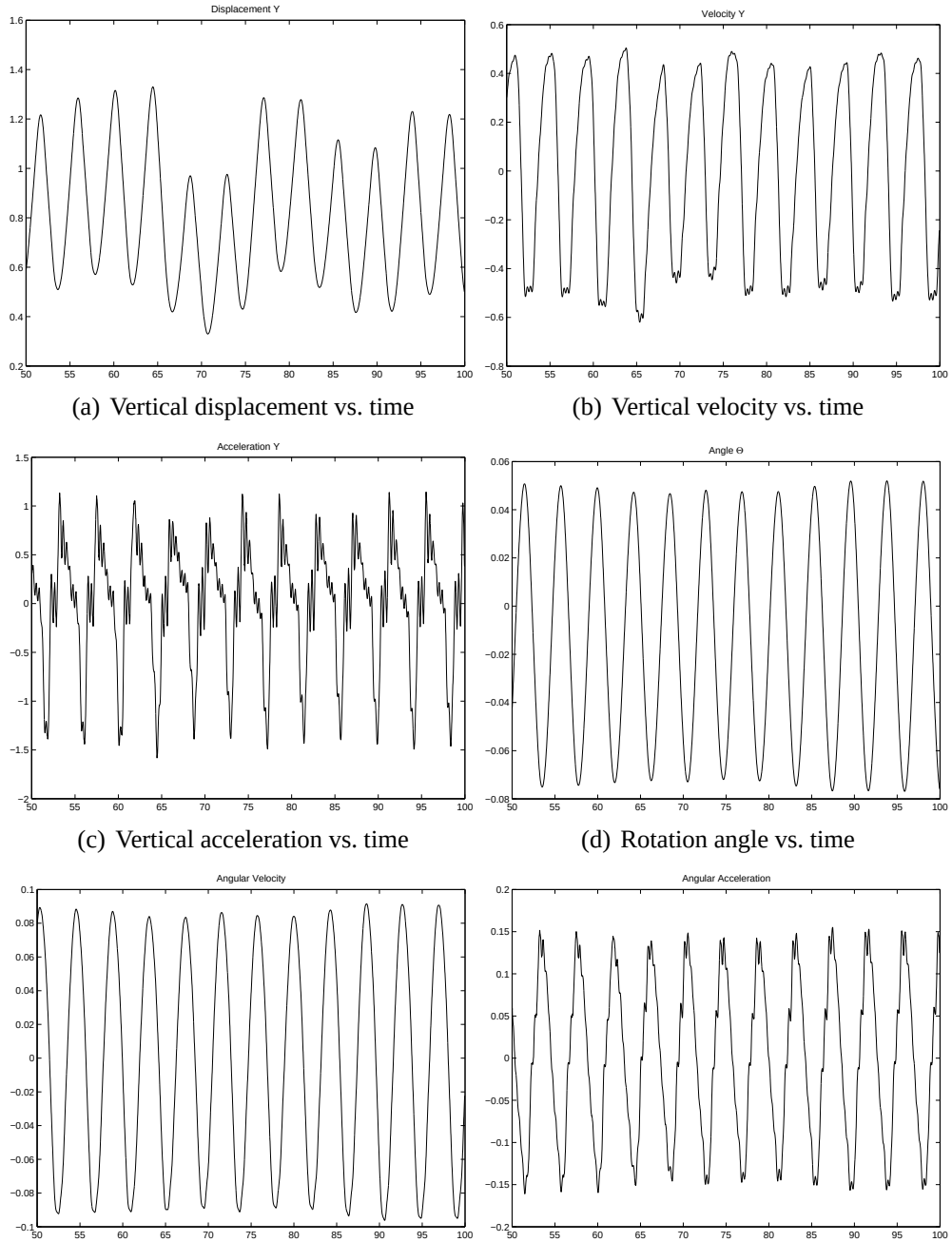
Obviously, the number of iterations needed for the inflow velocity of 55 m/s increases with the structure energy. We end this section with the plots of the velocity norm and pressure at different time steps in Figures 9 and 10, respectively.

6.5 Aeroelastic derivatives using numerical experimentation

A different approach to the direct flutter simulation is the calculation of the aeroelastic derivatives. This is the usual procedure when using wind tunnel tests. We refer to [40] for an introduction to this methodology.

In [39] Rossi has used the coupling method (29) for the assessment of aeroelastic derivatives. In fact, our fluid software has been used for the assessment of the flutter derivatives. This method involves the use of a pressure-correction method together with an explicit coupling procedure. The bridge model is identical to the one presented herein.

In this case, the results obtained are in agreement with the wind tunnel experiments. The key difference, compared to the direct flutter simulation presented here, is the fact that in the



(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

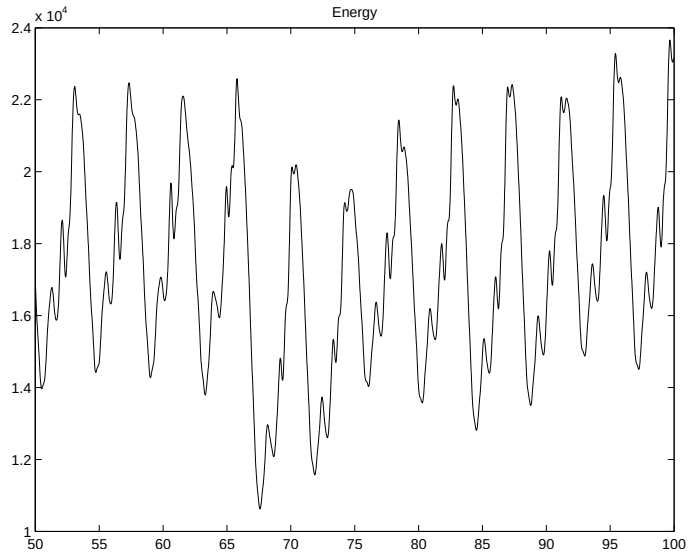
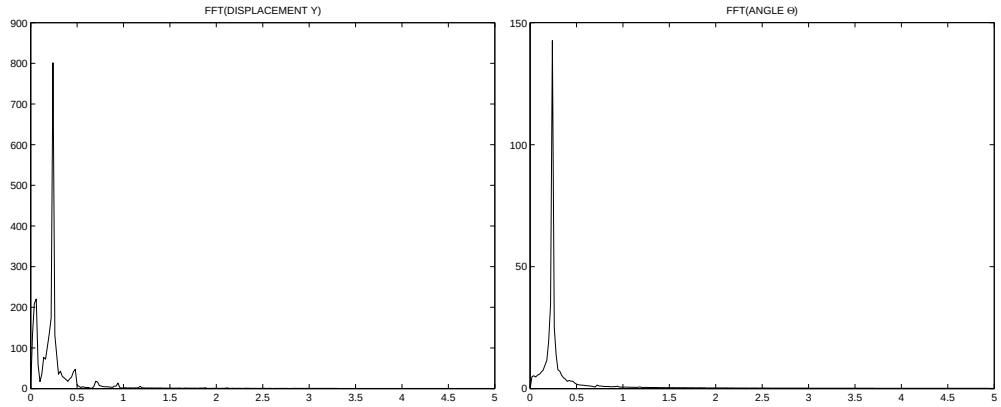


Figure 3: Bridge energy vs. time for inflow velocity $\boldsymbol{u}_{in} = (50, 0)$ m/s



(a) Vertical displacement: amplitude vs. frequency

(b) Rotation angle: amplitude vs. frequency

Figure 4: Fourier transform of vertical displacement and rotation angle of the bridge for inflow velocity $\boldsymbol{u}_{in} = (50, 0)$ m/s

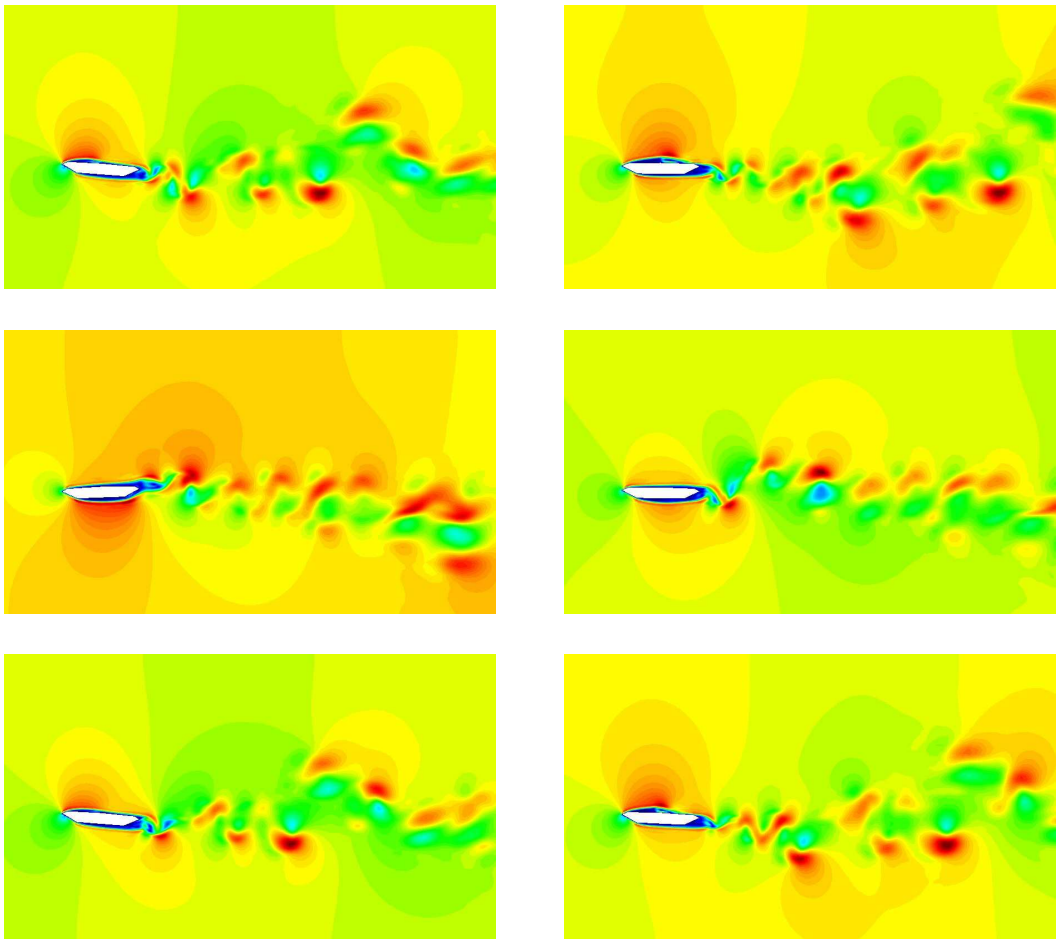


Figure 5: Contours of the velocity norm at different time steps (increasing time from left to right and from top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (50, 0)$ m/s

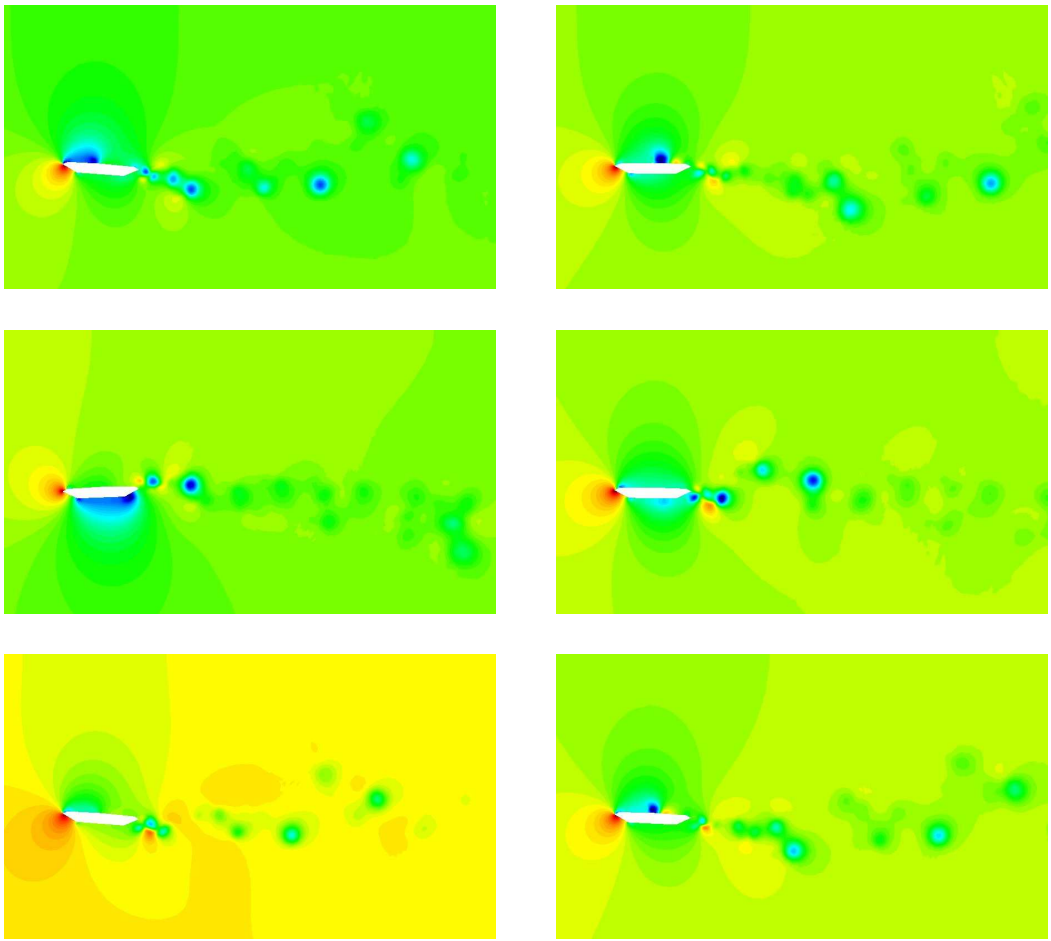
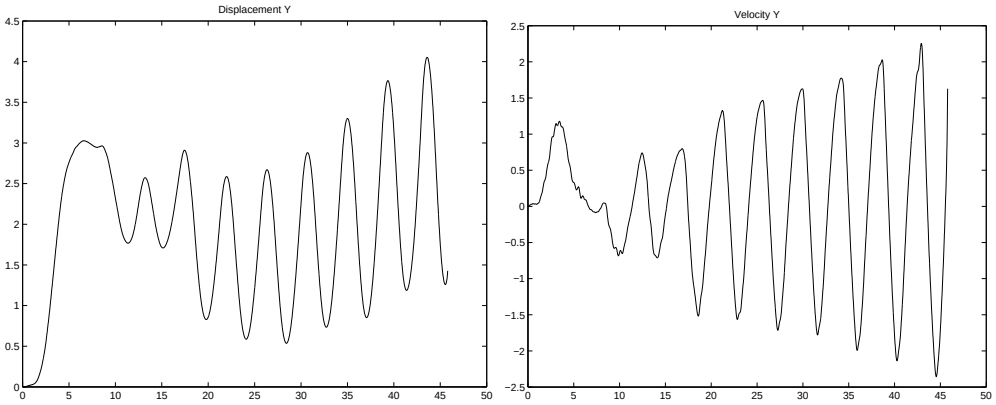
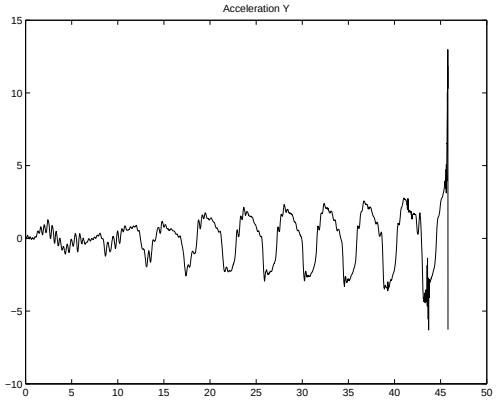


Figure 6: Contours of the pressure at different time steps (increasing time from left to right and from top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (50, 0)$ m/s

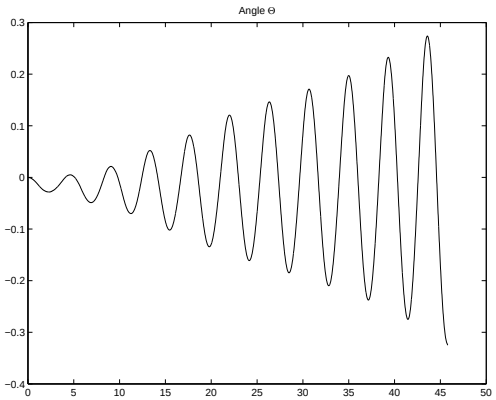


(a) Vertical displacement vs. time

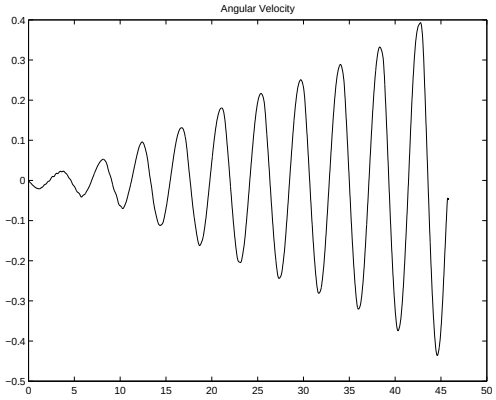
(b) Vertical velocity vs. time



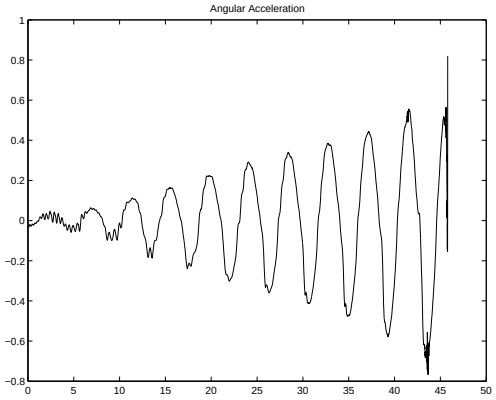
(c) Vertical acceleration vs. time



(d) Rotation angle vs. time



(e) Angular velocity vs. time



(f) Angular acceleration vs. time

Figure 7: Movement of the bridge for inflow velocity $\boldsymbol{u}_{in} = (55, 0)$ m/s

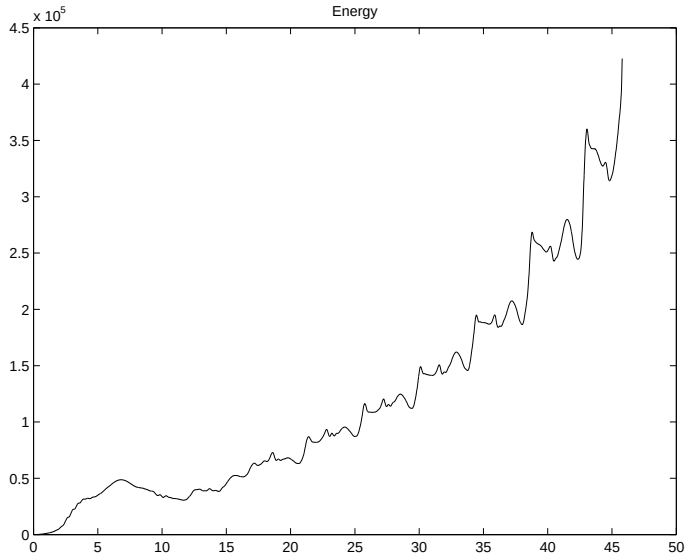


Figure 8: Bridge energy vs. time for inflow velocity $\boldsymbol{u}_{in} = (55, 0)$ m/s

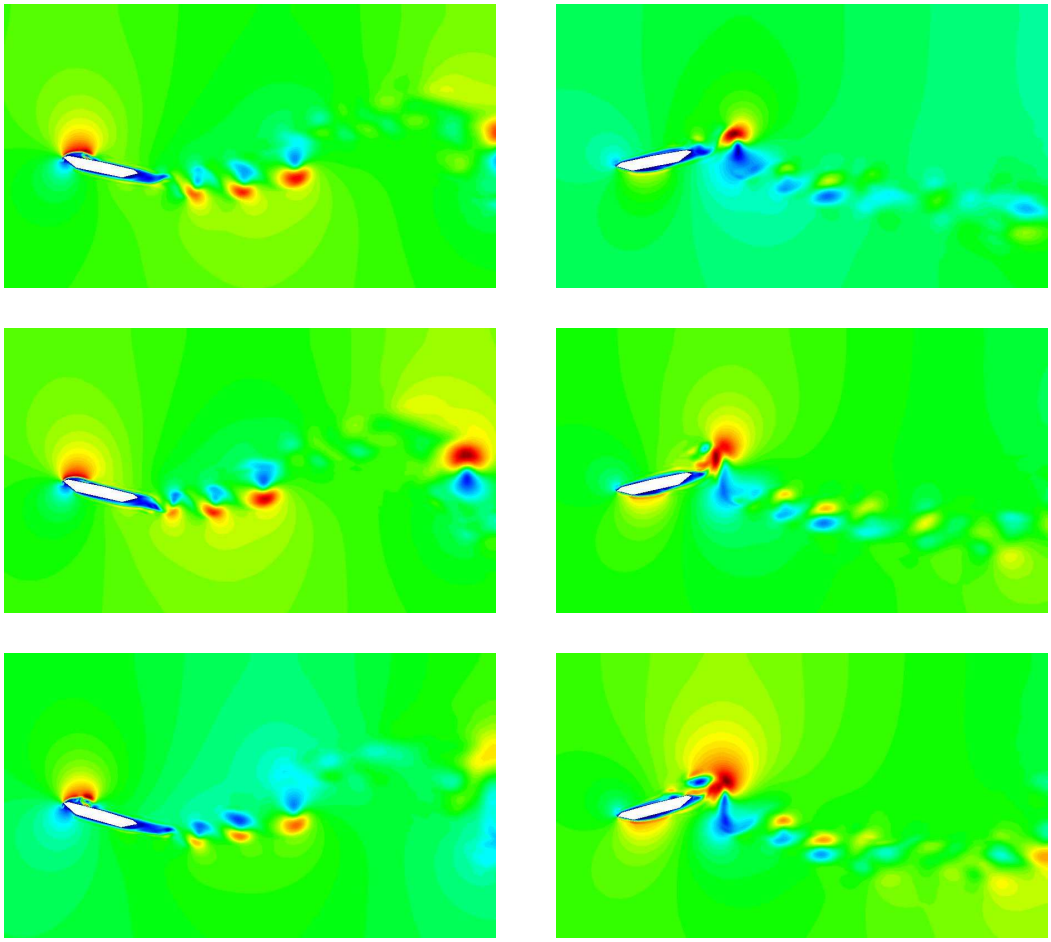


Figure 9: Contours of the velocity norm at different time steps (increasing time from left to right and from top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (55, 0)$ m/s

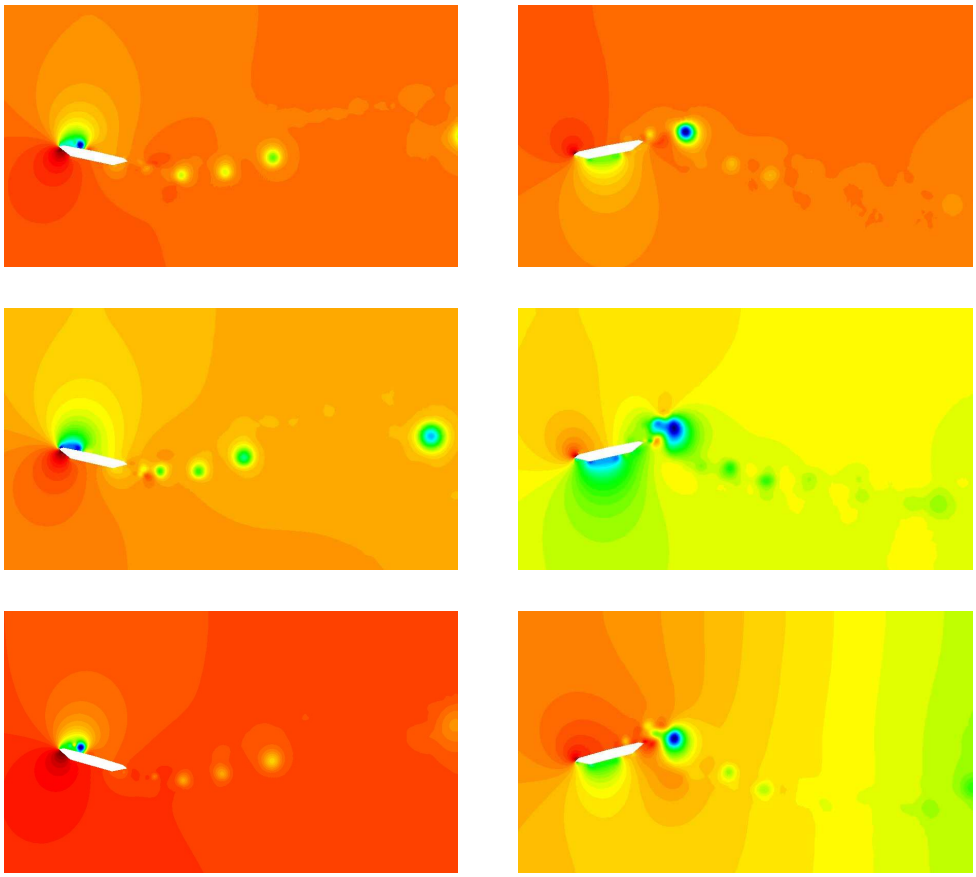


Figure 10: Contours of the pressure at different time steps (increasing time from left to right and from top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (55, 0)$ m/s

reference mentioned the problem considered is the same as in the wind tunnel, and thus the results can be fairly compared.

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 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, as reported in [39]. 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. 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.

[2] F.J. Blom. A monolithic fluid-structure interaction algorithm applied to the piston problem. *Computer Methods in Applied Mechanics and Engineering*, 167:369–391, 1998.

[3] D. Boffi and L. Gastaldi. Stability and geometric conservation laws for ALE formulation. *Computer Methods in Applied Mechanics and Engineering*, 193:4717–4739, 2004.

[4] M. Braack and E. Burman. A multiscale method towards turbulent flow based on local projection stabilization. *SIAM Journal on Numerical Analysis*, submitted.

[5] 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.

[6] M. Cervera, R. Codina, and M. Galindo. On the computational efficiency and implementation of block-iterative algorithms for nonlinear coupled problems. *Engineering Computations*, 13(6):4–30, 1996.

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

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

[9] R. Codina and S. Badia. Second order fractional step schemes for the incompressible Navier-Stokes equations. Inherent pressure stability and pressure stabilization. In *Proceedings of WCCM VI*, Beijing, China, 2004.

[10] R. Codina and S. Badia. On some pressure segregation methods of fractional-step type for the finite element approximation of incompressible flow problems. *Computer Methods in Applied Mechanics and Engineering*, in press.

[11] R. Codina and A. Folch. A stabilized finite element predictor–corrector scheme for the incompressible Navier–Stokes equations using a nodal based implementation. *International Journal for Numerical Methods in Fluids*, 44:483–503, 2004.

[12] 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.

[13] 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.

[14] S. Deparis, M. Discacciati, and A. Quarteroni. A domain decomposition framework for fluid-structure interaction problems. In *Proceedings of ICCFD3*, Toronto, 2004.

[15] S. Deparis, M.A. Fernández, and L. Formaggia. Acceleration of a fixed point algorithm for fluid-structure interaction using transpiration conditions. *Mathematical Modelling and Numerical Analysis*, 37(4):601–616, 2003.

[16] C. Farhat, P. Geuzaine, and G. Brown. Application of a three-field nonlinear fluid-structure formulation to the prediction of the aeroelastic parameters of an F-16 fighter. *Computers & Fluids*, 32:3–29, 2003.

[17] C. Farhat, P. Geuzaine, and C. Grandmont. The discrete geometric conservation law and the non-linear stability of ALE schemes for the solution of flow problems on moving grids. *Journal of Computational Physics*, 174:669–692, 2001.

[18] C. Farhat, M. Lesoinne, and N. Maman. Mixed explicit/implicit time integration of coupled aerolastic problems: three-field formulation, geometric conservation and distributed solution. *International Journal for Numerical Methods in Fluids*, 21:807–835, 1995.

[19] C. Farhat, M. Lesoinne, and P. Le Tallec. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 157:95–114, 1998.

[20] 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.

[21] 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.

[22] J.B. Frandsen. Numerical bridge deck studies using finite elements. Part I: Flutter. *Journal of Fluids and Structures*, 19(2):171–191, 2003.

[23] 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.

[24] P. Geuzaine, C. Grandmont, and C. Farhat. Design and analysis of ALE schemes with provable second-order time accuracy for inviscid and viscous flow simulations. *Journal of Computational Physics*, 191(1):206–227, 2003.

[25] T.J.R. Hughes, L. Mazzei, and K.E. Jansen. Large eddy simulation and the variational multiscale method. *Computing and Visualization in Science*, 3:47–59, 2000.

[26] T.J.R. Hughes, L. Mazzei, and A.A. Oberai. The multiscale formulation of large eddy simulation: decay of homogeneous isotropic turbulence. *Physics of Fluids*, 13–2, 2001.

[27] T.J.R. Hughes, A.A. Oberai, and L. Mazzei. Large eddy simulation of turbulent channel flows by the variational multiscale method. *Physics of Fluids*, 13–6, 2001.

[28] W. Hundsdorfer and J.G. Verwer. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*. Springer, 2003.

[29] B. Irons and R.C.Tuck. A version of the Aitken accelerator for computer implementation. *International Journal for Numerical Methods in Engineering*, 1:275–277, 1969.

[30] C.B. Jensen and T. Kvamsdal. Computational methods for FSI-simulation of slender bridges on high performance computers. In T. Kvamsdal, editor, *Computational Methods for Fluid-Structure Interaction*, pages 31–40, Tapir Forlag, Trondheim, Norway, 1999.

[31] A. Larsen and A.S. Jacobsen. Aerodynamic design of the great belt east bridge. In A. Larsen, editor, *Aerodynamics of Large Bridges*. Balkema, Rotterdam, Netherlands.

[32] H.G. Matthies and J. Stein. Partitioned but strongly coupled iteration schemes for nonlinear fluid-structure interaction. *Computers & Structures*, 80:1991–1999, 2002.

[33] H.G. Matthies and J. Stein. Partitioned strong coupled algorithms for fluid-structure interaction. *Computers & Structures*, 81:805–812, 2003.

[34] 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.

[35] F. Nobile. *Numerical Approximation of Fluid-Structure Interaction problems with application to Haemodynamics*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2001.

[36] 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.

[37] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford Science Publications, 1999.

[38] P. Raback, J. Ruokolainen, M. Lyly, and E. Jarvinen. Fluid-structure interaction boundary conditions by artificial compressibility. In *ECCOMAS Computational Fluid Dynamics Conference*, Swansea, Wales, UK, 2001.

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

[40] R.H. Scanlan and J.J. Tomko. Airfoil and bridge deck flutter derivatives. *Journal of Engineering Mechanics division ASCE 97 (EM6)*, pages 1717–1737, 1971.

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

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

[43] R. Temam. Sur l'approximation de la solution des équations de Navier–Stokes par la méthode des pas fractionaires (I). *Archives for Rational Mechanics and Analysis*, 32:135–153, 1969.