# Numerical benchmarking of fluid-structure interaction: An isogeometric finite element approach

Knut Nordanger[a,*], Adil Rasheed[b], Knut Morten Okstad[b], Arne Morten Kvarving[b], Runar Holdahl[b], Trond Kvamsdal[a,b]

[a] *Department of Mathematical Sciences, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway*
[b] *SINTEF ICT, Department of Applied Mathematics, Postboks 4760 Sluppen, NO-7465 Trondheim, Norway*

**Abstract**

In this paper we describe and evaluate an isogeometric finite element program, IFEM-FSI, for doing coupled fluid-structure interaction simulations. We investigate the role played by employing higher polynomial orders and higher regularity for solving a well known benchmark problem for flow past a circular cylinder with an attached flexible bar at Reynolds number $Re = 100$. Furthermore, we investigate the sensitivity to resolution in the fluid mesh as well as stiffness distribution in the mesh movement algorithm. Mesh quality is also assessed. Our simulations indicate that quadratic and cubic spline elements give better estimation of lift, drag and displacements than linear spline elements.

*Keywords:* Isogeometric analysis, Fluid-structure interaction

## 1. Introduction

Interaction between fluids and marine structures like monopiles and turbine support structures results in vortices which cause problems like scouring ([1]) of the seabed and vortex-induced stresses in the structures. Various techniques ([2, 3, 4, 5]) are being used to prevent the scouring phenomena ranging from the use of different materials to cover the seabed to the use of smart designs of the monopile's base to break the vortices. Most of the time these techniques depend on rigid structures. Optimizing their design is relatively simpler because state-of-the-art Computational Fluid Dynamics (CFD) codes can simulate the flow structures and hence the design effectiveness. Other methods for breaking or weakening the vortices can also be thought of; like the use of flexible structures attached to the monopiles to alter the behavior of the shedded vortices. Modelling the effectiveness of such concepts will involve coupled fluid-structure interaction simulations. The simulations are complicated by the fact that they

[*] Corresponding author
*Email addresses:* knut.nordanger@math.ntnu.no (Knut Nordanger), adil.rasheed@sintef.no (Adil Rasheed), knut.morten.okstad@sintef.no (Knut Morten Okstad), arne.morten.kvarving@sintef.no (Arne Morten Kvarving), runar.holdahl@sintef.no (Runar Holdahl), trond.kvamsdal@math.ntnu.no (Trond Kvamsdal)

require a dynamic mesh and an accurate representation of the deformed geometry under the influence of forces throughout the simulation time. Classical simulation methods based on linear finite elelments do not represent the geometry in an exact sense and introduce errors which can be reduced only by using a very high resolution. Fortunately, a new discretization method based on isogeometric analysis appears promising in this context. Isogeometric analysis (IGA), introduced in [6], has demonstrated that much can be gained in this respect by replacing the traditional low-order finite elements (FE) by volumetric NURBS (Non-Uniform Rational B-Splines). Spline approximations have some desirable properties both with respect to geometrical representation and analysis, since both the order and the smoothness of the basis functions can be easily changed. In particular, numerical results indicate that increased continuity of the finite element basis improve the approximation of both material stresses in structural analysis and sharp boundary layers in CFD analysis [7]. An overview of recent developments in the field of isogeometric analysis can be found in [8].

In this paper we apply the isogeometric finite element approach to the well known FSI benchmark problem presented in [9, 10] as a first step to demonstrating its benefits. The paper starts with a mathematical description of the method in solving fluid structure interaction problems. The description includes the fluid and structure solvers based on mass and momentum conservation equations. The equations are solved on a dynamic mesh to accommodate the geometry changes so a mesh mover is explained in the same section. Finally the set-up is described followed by results and conclusions.

## 2. Theory

Here we present the theory behind our fluid and structural solvers, along with details of how they are coupled and how the mesh movement is done. Mesh generation is also explained. The fluid domain, consisting of an incompressible Newtonian fluid, is denoted $\Omega^f$, while the structural domain, consisting of an elastic solid, is denoted $\Omega^s$.

### *2.1. Fluid solver*

The flow is mathematically described by the incompressible Navier-Stokes equations which read

$$\begin{aligned} \rho\frac{\partial \boldsymbol{u}^f}{\partial t} + \rho\left(\boldsymbol{u}^f\cdot\nabla\right)\boldsymbol{u}^f - \nabla\cdot\boldsymbol{\sigma}\left(\boldsymbol{u}^f,p\right) = \rho\boldsymbol{f} \quad \text{in } \Omega^f \\ \nabla\cdot\boldsymbol{u}^f = 0 \quad \text{in } \Omega^f. \end{aligned} \tag{1}$$

In this setting $\Omega \in \mathbb{R}^d$, $d = 2,3$, is a suitable, sufficiently regular and open domain, $\rho$ is the constant fluid density, $p$ is the pressure, $\boldsymbol{u}^f$ is the fluid velocity vector and $\boldsymbol{f}$ a volumetric body force. The Cauchy stress tensor can be written as

$$\boldsymbol{\sigma}(\boldsymbol{u}^f,p) = -p\boldsymbol{I} + 2\mu\boldsymbol{\epsilon}(\boldsymbol{u}^{\boldsymbol{f}}),$$

where $\boldsymbol{I}$ is the identity tensor, $\mu$ the dynamic viscosity and the strain rate $\boldsymbol{\epsilon}$ is defined as

$$\boldsymbol{\epsilon}(\boldsymbol{u}^f) = \frac{1}{2}\left(\nabla\boldsymbol{u}^f + (\nabla\boldsymbol{u}^f)^T\right).$$

Furthermore we define the boundary to be $\partial\Omega^f = \Gamma^f = \Gamma_D^f \cup \Gamma_N^f \cup \Gamma_M^f$ in order to handle boundaries with Dirichlet, Neumann or mixed boundary conditions. We denote $\Gamma_D^f$ the boundaries with Dirichlet conditions, $\Gamma_N^f$ the boundaries with Neumann conditions and $\Gamma_M^f$ the boundaries with mixed conditions. Mixed boundary conditions are used in situations where the normal velocity component is given, usually zero, together with the tangential stresses, and can model symmetry planes and slip or friction conditions.

The variational formulation can now be expressed as: Find $(\boldsymbol{u}^f, p) \in \boldsymbol{U} \times Q$ such that

$$\left(\rho \frac{\partial \boldsymbol{u}^f}{\partial t}, \boldsymbol{v}\right) + c(\boldsymbol{u}^f; \boldsymbol{u}^f, \boldsymbol{v}) + b(p, \boldsymbol{v}) + a(\boldsymbol{u}^f, \boldsymbol{u}^f) + b(q, \boldsymbol{u}^f) = f(\boldsymbol{v}) \tag{2}$$

for $(\boldsymbol{v}, q) \in \boldsymbol{V} \times Q$. We have defined the spaces

$$\begin{aligned}
\boldsymbol{U} &= \boldsymbol{H}_{\Gamma_D, \Gamma_M^\perp}(\Omega) = \left\{\boldsymbol{v} \in \boldsymbol{H}^1(\Omega) \mid \boldsymbol{v} = \boldsymbol{u}_{\boldsymbol{D}}^f \text{ on } \Gamma_D \text{ and } \boldsymbol{v} \cdot \boldsymbol{n} = u_\perp^f \text{ on } \Gamma_M\right\} \\
\boldsymbol{V} &= \boldsymbol{H}_{\Gamma_D, \Gamma_M^\perp; 0}(\Omega) = \left\{\boldsymbol{v} \in \boldsymbol{H}^1(\Omega) \mid \boldsymbol{v} = 0 \text{ on } \Gamma_D \text{ and } \boldsymbol{v} \cdot \boldsymbol{n} = 0 \text{ on } \Gamma_M\right\} \\
Q &= L^2(\Omega),
\end{aligned}$$

where $\boldsymbol{u}_{\boldsymbol{D}}^f$ and $u_\perp^f$ both are given functions and $\boldsymbol{n}$ is the unit outer normal on $\Gamma$. We have also defined the forms

$$\begin{aligned}
a(\boldsymbol{u}^f, \boldsymbol{v}) &= 2\int_\Omega \mu \boldsymbol{\epsilon}(\boldsymbol{u}^f) \,\boldsymbol{:}\, \boldsymbol{\epsilon}(\boldsymbol{v}) \,\mathrm{d}\boldsymbol{x} \\
b(q, \boldsymbol{v}) &= -\int_\Omega (\nabla \cdot \boldsymbol{v}) q \,\mathrm{d}\boldsymbol{x} \\
c(\boldsymbol{w}; \boldsymbol{u}^f, \boldsymbol{v}) &= \int_\Omega \rho(\boldsymbol{w} \cdot \nabla)\boldsymbol{u}^f \cdot \boldsymbol{v} \,\mathrm{d}\boldsymbol{x} \\
f(\boldsymbol{v}) &= \int_\Omega \rho \boldsymbol{f} \cdot \boldsymbol{v} \,\mathrm{d}\boldsymbol{x} + \int_{\Gamma_N} \boldsymbol{t} \cdot \boldsymbol{v} \,\mathrm{d}\boldsymbol{s}\,,
\end{aligned}$$

where $\boldsymbol{t} = \boldsymbol{\sigma} \cdot \boldsymbol{n}$ is the traction vector on $\Gamma$.

#### *2.1.1. Isogeometric finite element approximation*

In this work we employ an isogeometric finite element method similar to what was introduced in [6] and presented in [11]. The isogeometric finite element method approximates the solution by using a spline basis of polynomial order $p$ and regularity $C^{p-1}$. In traditional finite element formulations $C^0$ Lagrange polynomials of low order (typically $p = 1$ or $p = 2$) are used. Our approach is based on a conforming finite element approximation, i.e.

$$\boldsymbol{U}_h \subset \boldsymbol{U}, \quad \boldsymbol{V}_h \subset \boldsymbol{V}, \quad Q_h \subset Q.$$

The discrete approximation spaces $\boldsymbol{U}_h$, $\boldsymbol{V}_h$, $Q_h$ are chosen as the isogeometric finite element spaces. This gives the semi-discrete formulation of the variational problem stated in Eq. (2): Find $(\boldsymbol{u}_h^f, p_h) \in \boldsymbol{U}_h \times Q_h$ such that

$$\left(\rho \frac{\partial \boldsymbol{u}_h^f}{\partial t}, \boldsymbol{v}_h\right) + c(\boldsymbol{u}_h^f; \boldsymbol{u}_h^f, \boldsymbol{v}_h) + a(\boldsymbol{u}_h^f, \boldsymbol{u}_h^f) + b(p, \boldsymbol{v}_h) + b(q, \boldsymbol{u}_h^f) = f(\boldsymbol{v}_h) \tag{3}$$

for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h \times Q_h$.

As described in [12] we have developed a block-structured B-spline isogeometric finite element approximation of the Navier-Stokes equations described above. A domain $\Omega$ can be subdivided into a number of patches $\Omega_e$ such that $\Omega = \cup_{e=1}^{N} \Omega_e$, where what we call a patch is equivalent to a block. To construct a B-spline basis for $\Omega$ we associate for each patch a knot-vector in each coordinate direction

$$\Xi_k^e = \left\{ \xi_{1,k}^e, \xi_{2,k}^e, \ldots, \xi_{n_k^e + p_k^e + 1,k}^e \right\}$$

for $k = 1, \ldots, d$. Here, $n_k^e$ is the number of B-spline basis functions associated with the knot span. The B-spline basis for patch $\Omega_e$ on the parametric domain $\hat{\Omega} = (0,1)^d$ is written as $\hat{\boldsymbol{\mathcal{S}}}_{\boldsymbol{\alpha}^e}^{\boldsymbol{p}^e}$ where the multi-indices $\boldsymbol{\alpha}^e = (\alpha_1^e, \ldots, \alpha_d^e)$ and $\boldsymbol{p}^e = (p_1^e, \ldots, p_d^e)$ denote the regularity and order for the basis in each coordinate direction, respectively. The corresponding basis for the physical domain $\Omega_e$ can be expressed using the coordinate mapping $\phi_e : \hat{\Omega} \to \Omega_e$ as

$$\boldsymbol{\mathcal{S}}_{\boldsymbol{\alpha}^e}^{\boldsymbol{p}^e} = \left\{ v_h \mid v_h \circ \boldsymbol{\phi}_e \in \hat{\boldsymbol{\mathcal{S}}}_{\boldsymbol{\alpha}^e}^{\boldsymbol{p}^e} \right\}.$$

If the variational formulation allows a discontinuous approximation the spline finite element basis for the domain $\Omega$ can be defined as

$$\boldsymbol{\mathcal{S}}_h = \left\{ v_h \mid v_{h|\Omega_e} \in \boldsymbol{\mathcal{S}}_{\boldsymbol{\alpha}^e}^{\boldsymbol{p}^e} \right\}.$$

If we assume that the knot-vectors and geometrical mapping $\boldsymbol{\phi}_e$ for all the patches are consistent on common edges and faces we can define a continuous basis

$$\boldsymbol{\mathcal{S}}_{\boldsymbol{h}} = \left\{ v_h \in C(\Omega) \mid v_{h|\Omega_e} \in \boldsymbol{\mathcal{S}}_{\boldsymbol{\alpha}^e}^{\boldsymbol{p}^e} \right\}.$$

We use the same basis for the geometry as for the discretization of the velocity and the pressure.

### *2.1.2. Projection method*

In order to solve the mixed variational problem given in Eq. (3) the following inf-sup condition

$$\inf_{q_h \in Q_h, q_h \neq 0} \; \sup_{\boldsymbol{v}_h \in \boldsymbol{V}_h, \boldsymbol{v}_h \neq 0} \frac{b(q_h, \boldsymbol{v}_h)}{\|q_h\|_{L^2(\Omega)} \|\boldsymbol{v}_h\|_{\boldsymbol{H}^1(\Omega)}} \geq C > 0.$$

needs to be satisfied in order to avoid spurious pressure modes [13]. This imposes restrictions on the choices of $\boldsymbol{V}_h$ and $Q_h$.

Traditionally a mixed finite element method with different approximation spaces for pressure and velocity is required. In this work we use a pressure correction projection scheme which allows for equal-order approximation of the velocity and pressure. This is based on the work pioneered by Chorin [14] and Temam [15] in the late 1960s. In order to stabilize the equal-order approximation we employ Minev stabilization as described in [16]. A backward differentiation formula of order 2 (BDF2 scheme) is used for the time integration.

### *2.2. Structural solver*

In a Lagrangian description, the balance equation (conservation of linear momentum) of the structure can be written as

$$\rho^s \frac{\partial^2 \boldsymbol{u}^s}{\partial t^2} = \nabla \cdot \left( J \boldsymbol{F}^{-1} \cdot \boldsymbol{\sigma}(\boldsymbol{u}^s) \right) + \rho^s \boldsymbol{g} \quad \text{in } \Omega^s, \tag{4}$$

where $\boldsymbol{u}^s$ denotes the displacement of the structure, $\rho^s$ is the mass density, and $\boldsymbol{g}$ is the gravitation vector. The deformation gradient tensor, $\boldsymbol{F}$, is given by

$$\boldsymbol{F} = \boldsymbol{I} + \frac{\partial \boldsymbol{u}^s}{\partial \boldsymbol{X}} = \boldsymbol{I} + \nabla \boldsymbol{u}^s,$$

and $J = \det \boldsymbol{F}$.

We use a Total Lagrangian formulation for the structural solver, in which the second Piola-Kirchhoff stress tensor ($\boldsymbol{S}$) is a more convenient stress measure in the constitutive relation. It is related to the Cauchy stress tensor ($\boldsymbol{\sigma}$) through

$$\boldsymbol{\sigma} = \frac{1}{J} \boldsymbol{F} \cdot \boldsymbol{S} \cdot \boldsymbol{F}^T.$$

Assuming isotropic linear-elastic material, the constitutive relation for the structure can be written

$$\boldsymbol{S} = \lambda^s (\operatorname{Tr} \boldsymbol{E}) \boldsymbol{I} + 2\mu^s \boldsymbol{E},$$

where $\boldsymbol{E}$ denotes the Green-Lagrange strain tensor

$$\boldsymbol{E}(\boldsymbol{u}^s) = \frac{1}{2}(\boldsymbol{F}^T \boldsymbol{F} - \boldsymbol{I}).$$

Furthermore, $\lambda^s$ and $\mu^s$ are the Lamé coefficients defined by

$$\lambda^s = \frac{\nu^s E}{(1+\nu^s)(1-2\nu^s)}, \qquad \mu^s = \frac{E}{2(1+\nu^s)}, \tag{5}$$

where $E$ and $\nu^s$ are the Young's modulus the Poisson's ratio, respectively.

The weak form of Equation (4) is obtained by taking the product with a test function $\boldsymbol{v}^s$, and integrating over the undeformed reference configuration, $\Omega_0^s$. This results in

$$\begin{aligned} \int_{\Omega_0^s} \rho^s \frac{\partial^2 \boldsymbol{u}^s}{\partial t^2} \cdot \boldsymbol{v}^s \mathrm{d}\boldsymbol{X} + \int_{\Omega_0^s} \boldsymbol{S}(\boldsymbol{u}^s) : \boldsymbol{E}(\boldsymbol{v}^s) \mathrm{d}\boldsymbol{X} \\ = \int_{\Omega_0^s} \rho^s \boldsymbol{g} \cdot \boldsymbol{v}^s \mathrm{d}\boldsymbol{X} + \int_{\Gamma_0^s} \bar{\boldsymbol{t}} \cdot \boldsymbol{v}^s \mathrm{d}\boldsymbol{X}\,, \end{aligned} \tag{6}$$

where $\bar{\boldsymbol{t}}$ is the prescribed traction vector on the Neumann boundary $\Gamma^s$.

The structural problem is solved by integrating in time the linearized version of Equation (6), using a BDF2 scheme. Alternatively, the Hilber–Hughes–Taylor method [17] may be employed, but has not been used in the current study. As of the fluid solver, an isogeometric FE discretization based on spline basis functions is used in the numerical implementation.

### *2.3. Solver coupling*

We couple the fluid and the structural solver at the interface using a Dirichlet-Neumann coupling as presented in [18] in constrast to the monolithic approach employed in [19, 20] and [21]. An overview of different partitioned methods can be found in [22]. The coupling conditions employed here are

$$\begin{aligned} \boldsymbol{v}_\Gamma^f &= \frac{\mathrm{d}\boldsymbol{u}_\Gamma}{\mathrm{d}t} \\ \boldsymbol{\sigma}_\Gamma^f \cdot \boldsymbol{n} &= \boldsymbol{\sigma}_\Gamma^s \cdot \boldsymbol{n}, \end{aligned}$$

where $\boldsymbol{n}$ is the unit normal vector to the interface $\Gamma$.

For the benchmark case we investigate we have $\rho^f = 0.1\rho^s$, i.e. the fluid density is relatively large compared to the structural density. This known to give added mass effects [23, 24], and such cases can have slow convergence [25]. To stabilize the partitioned approach we use relaxation [26] and define a convergence criteria for the subiterations as

$$c\left(\boldsymbol{u}_{\Gamma,i}^{n+1}, \boldsymbol{u}_{\Gamma,i+1}^{n+1}\right) = \frac{\max_j\left(\left|\boldsymbol{u}_{\Gamma,i+1}^{n+1} - \boldsymbol{u}_{\Gamma,i}^{n+1}\right|\right)_j}{\max_j\left(\left|\boldsymbol{u}_{\Gamma,i+1}^{n+1}\right|\right)_j} < \epsilon_{\mathrm{subit}},$$

where we use a convergence criteria of $\epsilon_{\mathrm{subit}} = 10^{-6}$ for all the simulations in this paper.

Next the partitioned algorithm for solving the FSI problem is presented. Equal time step size $\Delta t$ is applied for both the fluid and structural fields. Furthermore, we define $n_{\max}$ as the maximum number of subiterations. For every time step we use the algorithm

```
while $c\left(\boldsymbol{u}_{\Gamma,i}^{n+1}, \boldsymbol{u}_{\Gamma,i+1}^{n+1}\right) > \epsilon_{\mathrm{subit}}$ and $n+1 <= n_{\max}$ do
    Solve fluid problem and determine fluid forces on the interface $\tilde{\boldsymbol{f}}_{\Gamma,i+1}^{n+1}\left(\boldsymbol{u}_{\Gamma,i}^{n+1}\right)$.
    Transfer relaxed fluid forces to the structural solver
        $\boldsymbol{f}_{\Gamma,i+1}^{n+1} = \omega_i \tilde{\boldsymbol{f}}_{\Gamma,i+1}^{n+1} + (1-\omega_i)\boldsymbol{f}_{\Gamma,i}^{n+1}$
    Solve structural problem for structural displacements $\tilde{\boldsymbol{u}}_{i+1}^{n+1}$
    Check convergence
    Solve grid problem for the new positions
    Compute the new grid velocity
    Derive new fluid velocity along surface to be used as Dirichlet boundary condition
end while
```

For all subiterations the relaxation parameter $\omega$ is kept constant.

### *2.4. Arbitrary Lagrangian-Eulerian description and mesh movement*

We employ the Arbitrary Lagrangian-Eulerian (ALE) concept, as first presented in [27], to handle the movement of the structure within the fluid mesh. Our implementation of the ALE concept herein are based on the work done earlier by our group, see [28], [29] and [30]. For more information about the

ALE concept we refer to [31], [32], [33] and [34]. The ALE description uses a reference domain which we denote $\hat{\Omega}^d$. Coordinates in the reference domain are written $\hat{\boldsymbol{x}}$. Following the notation used in [34] the fluid spatial domain $\Omega_t^f$ is given by

$$\Omega_t^f = \left\{ \boldsymbol{x} \mid \boldsymbol{x} = \boldsymbol{\phi}(\hat{\boldsymbol{x}}, t) \; \forall \hat{\boldsymbol{x}} \in \hat{\Omega}, \, t \in (0, T) \right\},$$

and the mapping used is given by

$$\boldsymbol{\phi}(\hat{\boldsymbol{x}}, t) = \hat{\boldsymbol{x}} + \hat{\boldsymbol{y}}(\hat{\boldsymbol{x}}, t).$$

Here, $\hat{\boldsymbol{y}}$ is the time-dependent displacement of the reference fluid domain. The fluid domain velocity is thus given by

$$\hat{\boldsymbol{u}}^f = \left. \frac{\partial \hat{\boldsymbol{y}}}{\partial t} \right|_{\hat{x}},$$

and is taken while $\hat{\boldsymbol{x}}$ is being held fixed. The ALE description of the incompressible Navier-Stokes equations can now be written as

$$\begin{aligned} \rho \left. \frac{\partial \boldsymbol{u}^f}{\partial t} \right|_{\hat{x}} + \rho \left( \left( \boldsymbol{u}^f - \hat{\boldsymbol{u}}^f \right) \cdot \nabla \right) \boldsymbol{u}^f - \nabla \cdot \boldsymbol{\sigma} \left( \boldsymbol{u}^f, p \right) = \rho \boldsymbol{f} \quad \text{in } \Omega^f \\ \nabla \cdot \boldsymbol{u}^f = 0 \quad \text{in } \Omega^f. \end{aligned} \tag{7}$$

However, for successful application of the ALE description a mesh movement algorithm is needed. Herein, we solve a linear elasticity problem at each time step. Our implementation is based on the developments made by our group first presented in [30] and later extended to handle two moving structures in [35]. A similar approach is used in [36]. Regarding the stability of the chosen mesh movement approach we refer to [31]. The linear elasticity equations can be written as

$$\nabla \cdot \boldsymbol{\sigma}^s + \boldsymbol{f} = \boldsymbol{0} \qquad \text{on } \Omega,$$

where $\boldsymbol{\sigma}$ is the Cauchy stress tensor and $\boldsymbol{f}$ is the external force. The Cauchy stress tensor for an isotropic material and for linear elasticity reads

$$\boldsymbol{\sigma}^s = 2\mu^s \boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda^s \operatorname{Tr}(\boldsymbol{\varepsilon}) \boldsymbol{I},$$

where $\lambda^s$ and $\mu^s$ are the Lamé constants, $\boldsymbol{u}$ the displacement, $\boldsymbol{I}$ the identity tensor and $\boldsymbol{\varepsilon}(\boldsymbol{u})$ the strain tensor. The strain tensor is given by

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2} \left( \nabla \boldsymbol{u} + ((\nabla \boldsymbol{u})^T \right),$$

and the Lamé coefficients are defined in Eq. (5) Here, $\nu^s$ is the Poisson ratio and $E$ the Young modulus. We will specify our mesh problem through these two parameters. Notice, that we might alternatively use a non-linear elasticity solver, similar to the one used to solve the structural displacements above, in the mesh movement algorithm. However, in our numerical tests the linear solver turned out to be more robust.

### *2.5. Mesh generation*

Generation of a high quality block-structured mesh can often be challenging. The computational domain shall be decomposed into 2D quadrilaterals which are not too skewed or distorted. Furthermore, distorted elements and abrupt changes in the element size should be avoided. Such cases can lead to unwanted grid effects. Also, we would like to have smaller elements at parts of the boundary with high curvature and close to solid walls in order to capture boundary layers.

#### *2.5.1. Block-structured mesh generation*

A bottom-up approach is often preferred for constructing a block-structured mesh. For two-dimensional problems the procedure can be described as

1. Define the corner nodes for the blocks.
2. Connect the corners to form the edges.
3. Refine the edges with a suitable grading.
4. Connect the edges to form surfaces.

To define the grading of the mesh a geometrical factor $r$ can be defined as the ratio of the element size of two consecutive elements, i.e. if $\{\boldsymbol{x}_i\}_{i=1}^m$ are the points on the edge or curve and $\Delta s_i = \|\boldsymbol{x}_i - \boldsymbol{x}_{i-1}\|_2$ defines the cell size, then

$$r = \Delta s_i / \Delta s_{i-1},$$

for $i = 2, \ldots, m$. To impose a smooth change in element size, we typically have that $0.8 < r < 1.2$, and for sharp boundary layers we may even use $0.9 < r < 1.1$ to capture the rapid change in the solution.

#### *2.5.2. Spline curves*

In this section we define spline curves, which form the foundation of the mesh generation, as in [11]. Assume that we have a knot-vector

$$\Xi = \{0 = \xi_1, \xi_2, \ldots, \xi_{n+p+1} = 1\}.$$

and a set of control points $\boldsymbol{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n\}$ which defines the spline curve

$$\boldsymbol{c}(\xi) = \sum_{i=1}^{n} \boldsymbol{c}_i B_{i,p}(\xi),$$

where $\{B_i\}_{i=1}^n$ are the basis functions. The parameter $p$ is the polynomial order of the spline curve, and each knot $\xi_i$ may be repeated several times, but the knot-span should be non-decreasing

$$\xi_1 \leq \xi_2 \leq \cdots \leq \xi_{n+p+1}.$$

For $p = 0$ the basis functions are piecewise constants

$$B_{i,0}(\xi) = \begin{cases} 1, & \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise.} \end{cases}$$

The higher order B-spline basis functions are defined as a linear combination of splines of lower order using the *Cox-de Boor recursion formula*

$$B_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} B_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} B_{i+1,p-1}(\xi). \tag{8}$$

We restrict our attention to *open* knot-vectors, i.e. splines that are interpolatory at the end points, and then the first and last knots are repeated $p+1$ times. Furthermore, if the spline is $C^{p-1}$ continuous then all the internal knots have multiplicity one and the knot-vector can be written as

$$\Xi = \{\underbrace{\xi_1, \ldots, \xi_1}_{p+1}, \xi_2, \ldots, \xi_{m-1}, \underbrace{\xi_m, \ldots \xi_m}_{p+1}\},$$

where the number of unique knots is given by $q = n - p + 1$. The corresponding knot-vector without repeated knots is

$$\bar{\Xi} = \{\bar{\xi}_1, \bar{\xi}_2, \ldots, \bar{\xi}_q\}.$$

### *2.5.3. Cubic spline interpolation*

The mesh generation process is dependent on standard cubic spline interpolation [37]. The starting point is a set of $m$ points $\{\boldsymbol{x}_i\}_{i=1}^m$ that we want to approximate by a cubic spline curve $\boldsymbol{c}(\xi)$ such that

- $\boldsymbol{c}(\tilde{\xi}_i) = \boldsymbol{x}_i$ for $\tilde{\xi}_i \in [0,1]$.
- $\boldsymbol{c}(\xi) \in \boldsymbol{C}^2([0,1])$.

The points $\{\tilde{\xi}_i\}_{i=1}^q$ where the spline curve interpolates the data are called the *Greville points*. Two extra conditions are needed to uniquely define the interpolation, and we thus use natural boundary conditions ($\boldsymbol{c}''(0) = \boldsymbol{c}''(1) = 0$). the spline curve at the endpoints. This leads to an $n \times n$ linear system with $n = m + 2$, which can be solved for the unknown control points $\{\boldsymbol{c}_i\}_{i=1}^n$. The interpolation is not uniquely defined since the parametrization can be different.

The entire meshes presented in this paper are made for polynomial order $p = 3$, and only lowered to orders $p = 1$ and $p = 2$ once all patches and refinements have been completed.

### *2.5.4. Surface generation*

For surface generation we employ the concept of Coons patches [38]. Given four boundary curves $\boldsymbol{u}_0(\xi)$, $\boldsymbol{u}_1(\xi)$, $\boldsymbol{w}_0(\eta)$, $\boldsymbol{w}_1(\eta)$ as given in Figure 1. These curves have normalized knot vectors and are connected such that $\boldsymbol{u}_0(\xi_1) = \boldsymbol{w}_0(0)$, $\boldsymbol{u}_0(1) = \boldsymbol{w}_1(0)$, $\boldsymbol{u}_1(1) = \boldsymbol{w}_1(1)$, $\boldsymbol{u}_1(0) = \boldsymbol{w}_0(1)$, thus forming a closed loop. By defining the surfaces

$$\begin{aligned}
S_1(\xi,\eta) &= (1-\eta)\boldsymbol{u}_0(\xi) + \eta\boldsymbol{u}_1(\xi)\\
S_2(\xi,\eta) &= (1-\eta)\boldsymbol{w}_0(\eta) + \xi\boldsymbol{w}_1(\eta)\\
S_3(\xi,\eta) &= (1-\xi)(1-\eta)\boldsymbol{u}_0(0) + \xi(1-\eta)\boldsymbol{u}_0(1) + \eta(1-\xi)\boldsymbol{u}_1(0) + \xi\eta\boldsymbol{u}_1(1)
\end{aligned}$$

the Coons surface paths is given by

$$S_c(\xi,\eta) = S_1(\xi,\eta) + S_2(\xi,\eta) - S_3(\xi,\eta).$$

The Coons surface patch approach is a quick and easy way of building the surfaces. Being able to define the geometry through the boundary curves of each surface or patch is a great advantage.

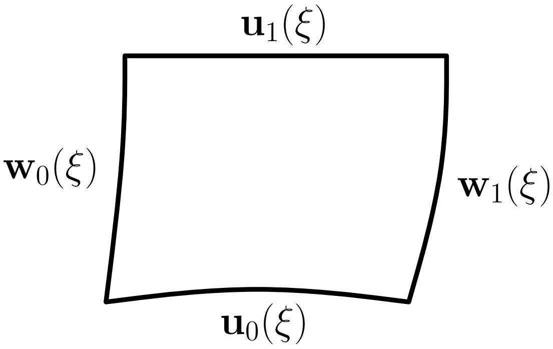


Figure 1: Boundary curves for Coons surface patch.

### 2.6. Calculation of time-dependent quantities

We calculate mean value and amplitude of the time-dependent quantities as in [9]. The mean value of quantity $x$ is denoted $\bar{x}$ and calculated from last period of oscillations as

$$\bar{x} = \frac{1}{2}(x_{\max} + x_{\min}).$$

Similarly, the amplitude of quantity $x$ is denoted $x_{\text{amp}}$ and calculated as

$$x_{\text{amp}} = \frac{1}{2}(x_{\max} - x_{\min}).$$

Frequencies are calculated by the Lomb-Scargle algorithm [39, 40].

## 3. Simulation setup

### 3.1. Problem description

In this paper we aim to simulate the FSI2 benchmark case defined in [9], with updated results given in [10], with our isogeometric code IFEM. The benchmark case is defined for flow past a fixed circular cylinder with a flexible bar attached, see Figure 2. The computational domain is identical to the domain in [9], except that the origin is shifted from the lower left corner to the centre of the cylinder.

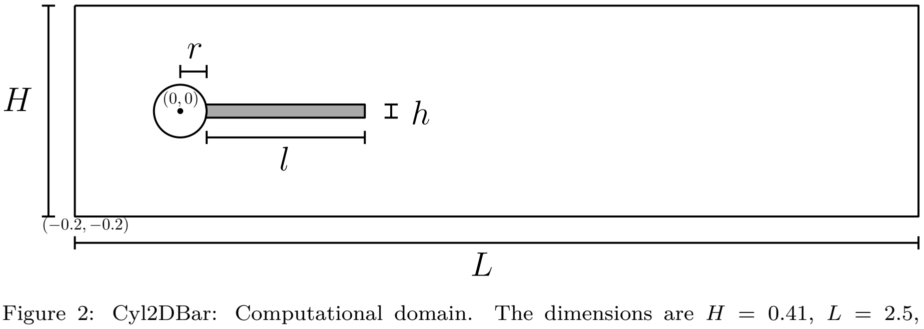


Figure 2: Cyl2DBar: Computational domain. The dimensions are $H = 0.41$, $L = 2.5$, $r = 0.05$, $h = 0.02$ and $l = 0.35$.

Inflow is from the left only and we prescribe a parabolic velocity profile

$$\begin{aligned} v_x^{\text{in}}(0, y) &= 1.5\bar{U}\frac{(y + 0.2)(H - (y + 0.2))}{\left(\frac{H}{2}\right)^2} \\ &= 1.5\bar{U}\frac{4.0}{0.1681}(y + 0.2)\left(0.41 - (y + 0.2)\right), \end{aligned} \tag{9}$$

where $\bar{U}$ is the mean inflow velocity. The top and bottom wall, circle and fluid-structure interface $\Gamma_t^0$ is prescribed the no-slip condition. At the outflow, i.e. the right boundary, the pressure is prescribed to be 0.

The inflow velocity is ramped up through as smooth increase of the velocity profile as suggested in [9] through

$$v_x^{\text{in}}(t, 0, y) = \begin{cases} v_x^{\text{in}}(0, y)\frac{1-\cos(\pi t)}{2} & \text{if } t < 1.0 \\ v_x^{\text{in}}(0, y) & \text{otherwise.} \end{cases}$$

Our main quantities of interest are the vertical position, $y(t)$, of the end of the flexible bar, identified by the letter $A$ in Figure 3, and the lift, $F_L$, and drag forces, $F_D$, on the cylinder and flexible bar considered as one single object. The coordinates of the reference point A at time $t = 0$ is $(0.4, 0)$. The drag and lift forces, $F_D$ and $F_L$ are the horizontal and vertical force components acting on the cylinder and flexible bar respectively. The force components are computed as

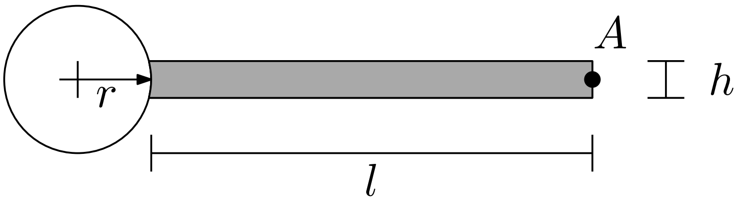


Figure 3: Cyl2DBar: Structural part of the domain

$$F = [F_D, F_L]^T = \int_\Gamma \boldsymbol{\sigma} \cdot \boldsymbol{n} \, \mathrm{d}\boldsymbol{s},$$

where $\Gamma$ is the surface of the cylinder and the flexible bar. Compare with the expression given in Turek.)

The flow and material properties given in Table 1 are used throughout the simulations. As can be incurred from the table, the Reynolds number for all simulations is 100.

### *3.2. Mesh description*

The patch structure along with the refinement edges $e_1 - e_8$ are shown in Figure 4. Detailed refinement information can be found in Table 2.

The resulting number of elements and degrees-of-freedom for the different meshes and polynomial orders are shown in Table 3. The meshes are designed such that the number of fluid and structural elements double for each refinement level.

The mesh for the coarsest grid, G1, for $p = 2$ along with a zoomed view of the mesh close to the cylinder and the flexible are shown in Figure 5. In both figures the fluid domain is light blue, whilst the structural domain is light red.

Table 1: Cyl2DBar: Flow and material properties.

| Parameter | Quantity | Unit |
|---|---|---|
| $\rho^s$ | 10000 | $\mathrm{kg/m^3}$ |
| $\nu^s$ | 0.4 | - |
| $E^s$ | $1.4 \cdot 10^6$ | Pa |
| $\rho^f$ | 1000 | $\mathrm{kg/m^3}$ |
| $\nu^f$ | 0.001 | $\mathrm{m^2/s}$ |
| $\bar{U}$ | 1.0 | m/s |

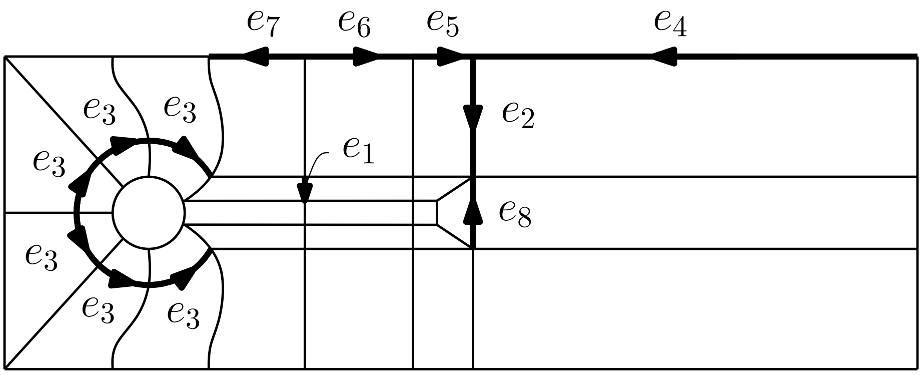
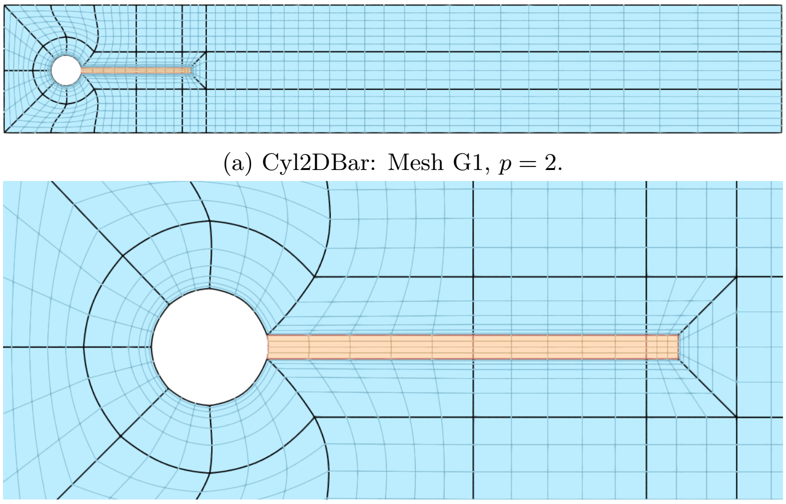


Figure 4: Cyl2DBar: Refinement edges.



(a) Cyl2DBar: Mesh G1, $p = 2$.

(b) Cyl2DBar: Zoomed view of mesh G1 close to the cylinder and the flexible bar, $p = 2$.

Figure 5: Cyl2DBar: Grid G1. The fluid domain is light blue and the structural domain is light. Patch boundaries are drawn with thick black lines.

### 3.3. Time step determination

All simulations use a non-dimensional time step of $\Delta t = 0.025$ and a fixed relaxation parameter $\omega = 0.1$.

Table 2: Cyl2DBar: Detailed refinement information for the simulations meshes. Edge grading factor (geometric stretching factor) is given by $r$ and $n$ is the number of inserted knots along the given edge.

| Edge / Mesh | | G1 | G2 | G3 | G4 | G5 |
|---|---|---|---|---|---|---|
| $e_1$ | $r$ | 0.61 | 0.74 | 0.82 | 0.909 | 0.9605 |
| | $n$ | 5 | 7 | 9 | 13 | 18 |
| $e_2$ | $r$ | 1.0 | 1.0 | 0.97 | 0.97 | 0.97 |
| | $n$ | 5 | 7 | 8 | 13 | 20 |
| $e_3$ | $r$ | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| | $n$ | 2 | 4 | 7 | 10 | 15 |
| $e_4$ | $r$ | 0.94 | 0.97 | 0.99 | 0.985 | 0.99 |
| | $n$ | 25 | 40 | 80 | 98 | 133 |
| $e_5$ | $r$ | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| | $n$ | 2 | 3 | 5 | 7 | 9 |
| $e_6$ | $r$ | 0.95 | 0.96 | 0.97 | 0.96 | 0.96 |
| | $n$ | 4 | 6 | 8 | 15 | 20 |
| $e_7$ | $r$ | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| | $n$ | 4 | 6 | 8 | 12 | 18 |
| $e_8$ | $r$ | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| | $n$ | 3 | 5 | 7 | 10 | 15 |

Table 3: Cyl2DBar: Number of elements and degrees-of-freedom for simulation meshes G1-G5 and polynomial orders $p = 1, 2, 3$.

| | | Structure | | Fluid | |
|---|---|---|---|---|---|
| Mesh | $p$ | $n_{\text{el}}$ | $n_{\text{dof}}$ | $n_{\text{el}}$ | $n_{\text{dof}}$ |
| G1 | 1 | 52 | 140 | 1936 | 3144 |
| G2 | 1 | 108 | 266 | 4012 | 6381 |
| G3 | 1 | 192 | 450 | 8020 | 12612 |
| G4 | 1 | 407 | 912 | 15870 | 24597 |
| G5 | 1 | 800 | 1734 | 31832 | 48849 |
| G1 | 2 | 52 | 204 | 1936 | 4281 |
| G2 | 2 | 108 | 352 | 4012 | 7959 |
| G3 | 2 | 192 | 560 | 8020 | 14862 |
| G4 | 2 | 407 | 1066 | 15870 | 27657 |
| G5 | 2 | 800 | 1944 | 31832 | 53079 |
| G1 | 3 | 52 | 280 | 1936 | 5586 |
| G2 | 3 | 108 | 450 | 4012 | 9705 |
| G3 | 3 | 192 | 682 | 8020 | 17280 |
| G4 | 3 | 407 | 1232 | 15870 | 30885 |
| G5 | 3 | 800 | 2166 | 31832 | 57477 |

### *3.4. Simulation length*

We run all simulations up till non-dimensional time 200, i.e. 8000 time steps. A plot of the cumulative standard deviation is shown in Figure 6 for grid G1. This indicates that the simulations are run long enough.

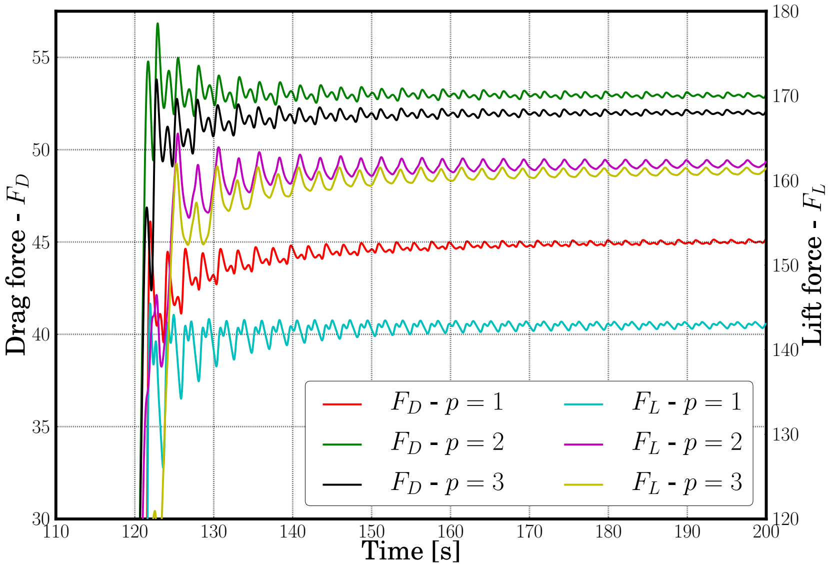


Figure 6: Cyl2DBar: Cumulative standard deviation for grid G1 calculated from $t = 120$.

### *3.5. Mesh stiffness*

Ensuring mesh quality consistent with the model in use is the key to getting accurate results. In the case we are interested in the geometry is expected to undergo deformations and hence the mesh elements can change in shape and size. Fortunately, the quality of the mesh can still be controlled by cleverly choosing the distribution of mesh stiffness throughout the mesh domain. Mesh stiffness is defined as Young's modulus in Eq. (5). When a constant value of mesh stiffness is applied, specified displacements are homogeneously diffused throughout the mesh. On the other hand when the mesh stiffness is specified as varying throughout the domain, nodes in regions of high stiffness move together, i.e. there is little relative motion. Variable mesh stiffness is particularly useful to preserve the mesh distribution (and quality) near fine geometrical features, such as sharp corners, or in boundary layers. In principle, the computed spatio-temporal deflections, drag and lift force should be independent of the choice of mesh stiffness distribution. However, it has been observed that some mesh distributions, like constant mesh stiffness everywhere, resulted in divergence in solution. In this work seven different stiffness distributions were tried. In all the cases the mesh stiffness distribution was maximum close to the wall so that the initial mesh quality in the vicinity is preserved over the full course of simulation and then it decreased as a function of the distance from the wall $R$

to zero.Expressions given below give a mathematical description of the mesh stiffness as a function of $R$.

1. Base case: $E_{\text{bc}} = 1 + 200e^{-100R}$
2. $E_1 = 1 + 200e^{-50R}$
3. $E_2 = 1 + 1000e^{-100R}$
4. $E_3 = \begin{cases} 201 & \text{if } R < 0.025 \\ 1 + 200e^{-100(R-0.025)} & \text{otherwise} \end{cases}$
5. $E_4 = \begin{cases} 201 & \text{if } R < 0.015 \\ 1 + 200e^{-100(R-0.015)} & \text{otherwise} \end{cases}$
6. $E_5 = \begin{cases} 201 & \text{if } R < 0.035 \\ 1 + 200e^{-100(R-0.035)} & \text{otherwise} \end{cases}$
7. $E_6 = \begin{cases} 1001 & \text{if } R < 0.035 \\ 1 + 1000e^{-100(R-0.035)} & \text{otherwise} \end{cases}$

Stiffness contours are presented in Figure 7 and a more quantitative profile of the stiffness as a function of the distance $R$ is given by the Figure 8 for grid G3 for easier comparison. $E_{bc}$ is the least stiff mesh while $E_6$ is the stiffest mesh of all the cases simulated here. The basecase mesh stiffness has been used for all other simulations in this paper, and it was chosen based on experience of stability and required solution time. More specifically, it was found that in order to maintain a low element distortion close to the solid, the mesh stiffness should have a constant level close to the solid and then decay exponentially away from there.

### *3.6. Definition of test cases*

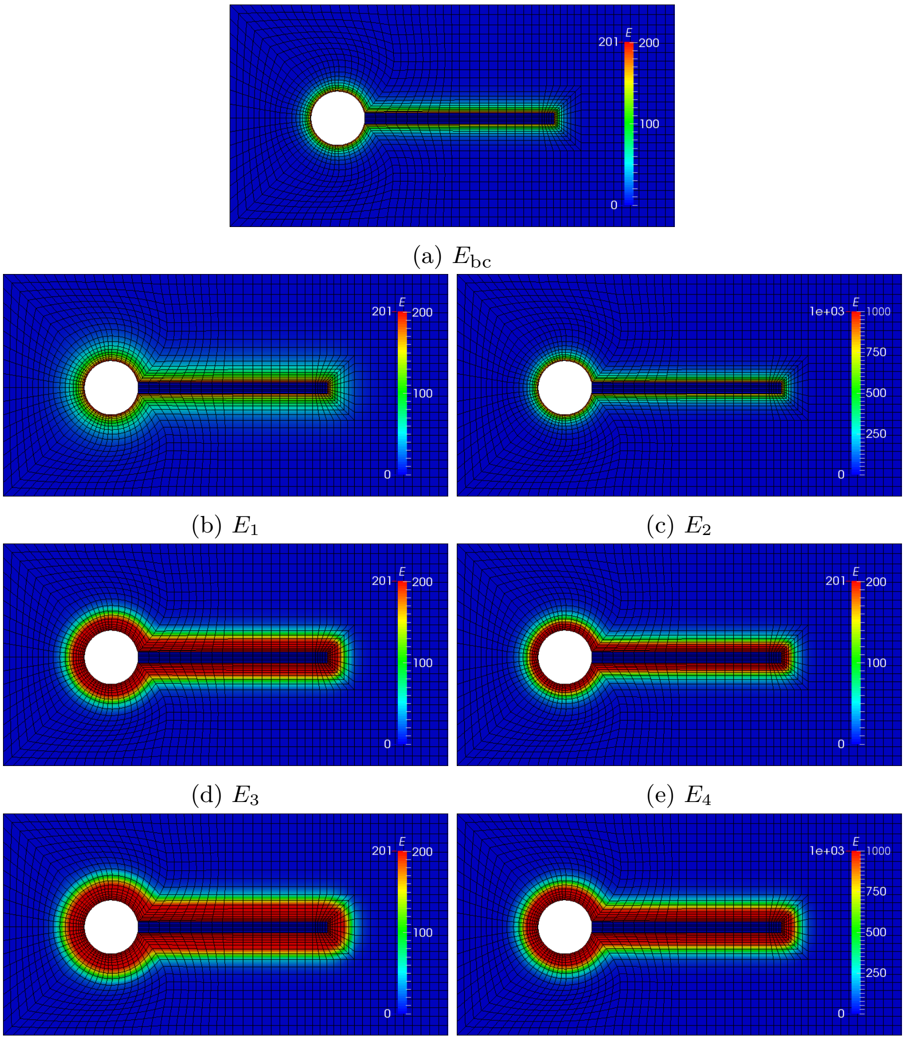
Effects of changing the order of the elements (linear, quadratic and cubic) was investigated for five different mesh resolutions resulting in a total of fifteen simulations. Six additional simulations were conducted to understand the effect of changing the mesh stiffness.

## 4. Results and discussion

A comparison of lift, drag and displacements with respect to grid resolution and element order is presented here. Although the computational efficiency was not the main focus of attention in this work we include this information for one particular grid. Finally, a subsection demonstrates the effect of changing the mesh stiffness.

### *4.1. Mesh stiffness and quality*

The accuracy of simulation involving stationary bodies depend on the quality of the rigid mesh. However, for an FSI simulation involving deformations of solid bodies it is necessary to ensure that all the intermediate mesh configurations resulting during such movements are of high quality. To this effect, during the mesh generation step we applied four different criteria to obtain high quality meshes. It is worth mentioning that we have used only quadrilateral elements in all the simulations. Therefore, all subsequently mentioned mesh quality metrics are for quadrilaterals. In this paper we use four different mesh metrics for



(a) $E_{\mathrm{bc}}$

(b) $E_1$

(c) $E_2$

(d) $E_3$

(e) $E_4$

(f) $E_5$

(g) $E_6$

Figure 7: Cyl2DBar: Mesh stiffness distributions for grid G3.

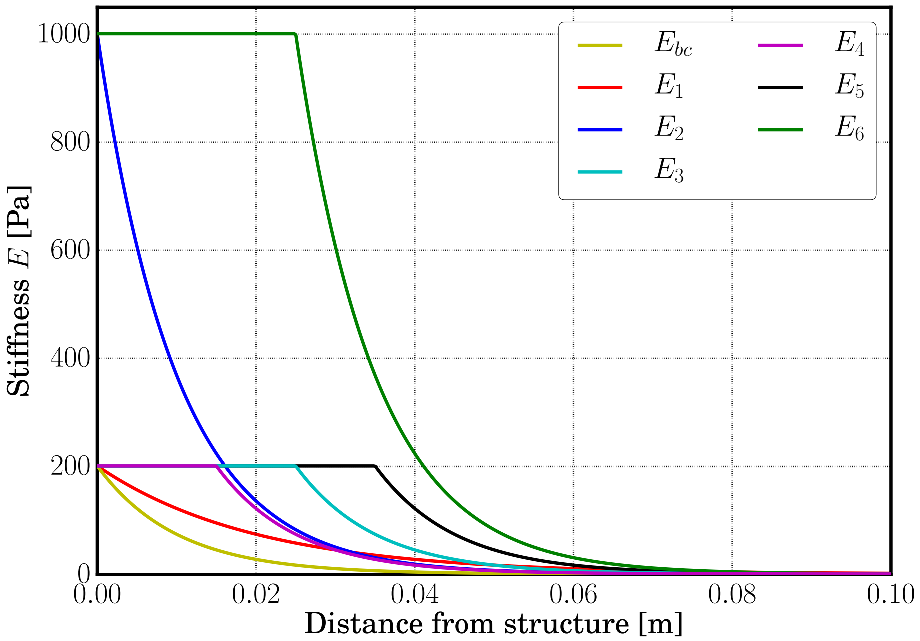
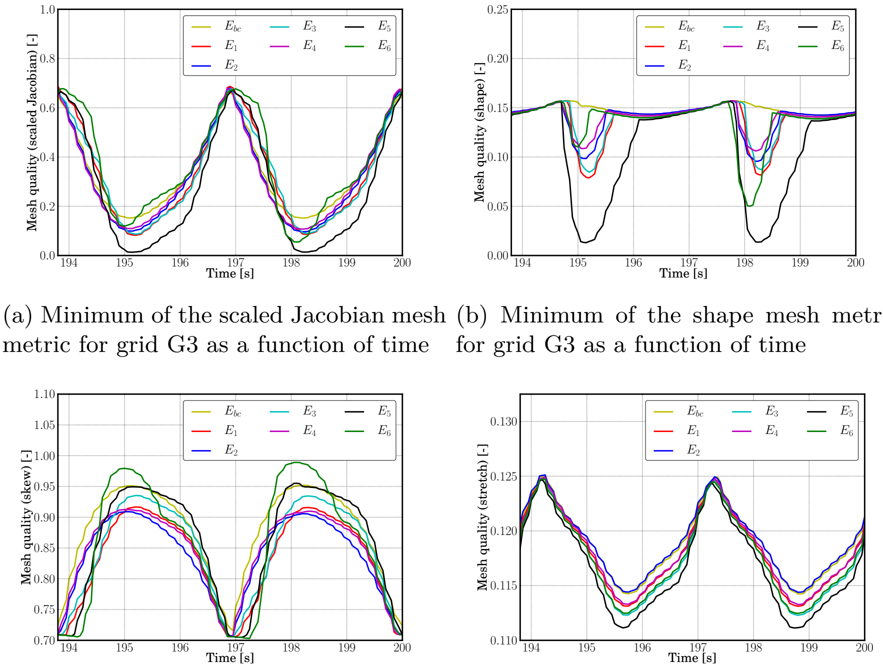


Figure 8: Cyl2DBar: Different mesh stiffnesses investigated.

assessing the quality of the finite element meshes. Firstly, the scaled Jacobian can vary from $-1$ to 1 [41]. A positive value is necessary for the mesh to have the minimum quality. Typically, the acceptable range is $[0.3, 1]$ [42]. A negative value signals an invalid element. Secondly, the shape quality metric is defined in [43]. This metric attains the value 1 if the quadrilateral is a square and 0 if it is degenerate. Thirdly, the skew quality metric is also defined in [43]. This metric aims to detect element distortions arising from large or small angles. It attains the value 1 if the quadrilateral is a rectangle and 0 if it is degenerate. Lastly, the stretch quality metric is a measure of the aspect ratio. This also attains values between 0 and 1.

Seven different mesh stiffness distribution were investigated and ensured that the mesh quality is maintained throughout the whole cycle considered. Although such investigations were conducted for all the grids and orders, in Figure 9 we just present the case for G3 grid and quadratic elements. The scaled Jacobian mesh metric for grid G3 is shown for the minimum quality element in Figure 9a. It is clear from the figure that the scaled Jacobian is always positive and therefore there are no intersecting grid lines which could result in unphysical results. The shape mesh metric for grid G3 is shown for the minimum quality element in Figure 9b. Since the boundary layer close to the solid structure was to be resolved, very fine resolution in the direction normal to the wall was required. It is therefore natural to expect that some cells close to the junction between cylinder and bar will have some degree of degeneracy. Similarly, the skew mesh metric for the worst quality element is shown in Figure 9c. It is clear that even for the worst element the skewness value never gets below 0.7 throughout the cycle. Finally the stretch mesh metric for the worst quality element is shown in the Figure 9d. It is worth mentioning that the aspect ratio can easily be in of



(a) Minimum of the scaled Jacobian mesh metric for grid G3 as a function of time

(b) Minimum of the shape mesh metric for grid G3 as a function of time

(c) Maximum of the skew mesh metric for grid G3 as a function of time

(d) Maximum of the stretch mesh metric for grid G3 as a function of time.

Figure 9: Cyl2DBar: Mesh quality

the order of $10^5$ and the solvers used in this work can easily handle such aspect ratios, see [12]. In light of that the stretch mesh metric is highly adorable. Satisfied with the quality of intermediate mesh configurations, seven different simulations were conducted to quantify the effects of different mesh stiffness distributions ($E_{bc}$,$E_1$,$E_2$,$E_3$,$E_4$,$E_5$,$E_6$). The results are presented in the Table 4. As expected all the different mesh stiffness distributions predicted the drag, lift and deformations characteristics within acceptable limits when compared to the reference data. Investigations of total CPU and number of subiterations required at each time step for the simulations show no major differences for the various mesh stiffnesses.

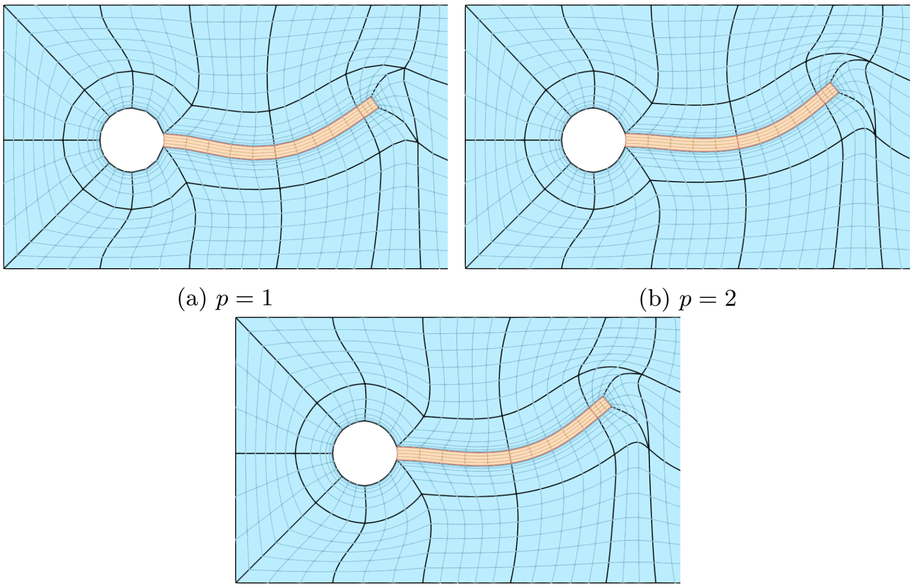
The qualitative differences between spline elements of order $p = 1$, $p = 2$ and $p = 3$ is shown in Figure 10 for grid G1 in a deformed state. It is clear from the figures that the use of higher order spline elements ($p = 2, 3$) gives a smoother representation of the interface between the fluid and structure leading to smaller error when the forces are transferred at the interface.

### *4.2. Velocity and pressure contours*

Figures 11 and 12 give the velocity and pressure distribution, respectively, in the domain for a complete oscillation period $T$. One can notice the existence of a consistently high pressure on the leading side of the cylinder. The high pressure zone is created due to the impingement of flow on the surface of the cylinder. After the impingement the flow bifurcates and flows around the cylinder. In the

Table 4: Cyl2DBar: Results for mesh stiffness investigations

| | $F_D$ | $F_{D,\text{amp}}$ | $F_L$ | $F_{L,\text{amp}}$ | $u_x$ | $u_{x,\text{amp}}$ | $u_y$ | $u_{y,\text{amp}}$ |
|---|---|---|---|---|---|---|---|---|
| $E_{\text{bc}}$ | 214.94 | 78.97 | 1.20 | 230.5 | $-0.01486$ | 0.0128 | 0.0013 | 0.0813 |
| $E_1$ | 214.60 | 78.69 | 1.09 | 229.6 | $-0.01476$ | 0.0128 | 0.0013 | 0.0811 |
| $E_2$ | 214.80 | 79.03 | 1.21 | 230.6 | $-0.01485$ | 0.0128 | 0.0013 | 0.0813 |
| $E_3$ | 214.60 | 78.73 | 1.24 | 229.2 | $-0.01475$ | 0.0128 | 0.0013 | 0.0810 |
| $E_4$ | 214.79 | 78.89 | 1.23 | 229.9 | $-0.01481$ | 0.0128 | 0.0013 | 0.0812 |
| $E_5$ | 213.98 | 77.96 | 1.69 | 232.1 | $-0.01444$ | 0.0124 | 0.0013 | 0.0803 |
| $E_6$ | 214.49 | 78.67 | 1.19 | 229.7 | $-0.01475$ | 0.0128 | 0.0013 | 0.0811 |
| Ref. | 215.06 | 77.65 | 0.61 | 237.8 | $-0.01485$ | 0.0127 | 0.0013 | 0.0817 |



(a) $p = 1$

(b) $p = 2$

(c) $p = 3$

Figure 10: Cyl2DBar: Deformed G1 grid for spline element order $p = 1$, $p = 2$, $p = 3$.

absence of the bar on the trailing side, the flow would have reached a statistically steady state characterized by a repeating pattern of swirling vortices caused by the unsteady flow separation. The vortices shed on the upper side of the cylinder interact with those shed on the lower side giving rise to a von Karman vortex street. When the bar is attached to the cylinder such an interaction between the vortices is delayed. However, because of the elastic nature of the material and inherent instability in the flow the equilibrium of the bar is disturbed and it starts oscillating in an up-and-down motion. In Figure 12, one can see that for $t = 0T$ when the bar is in the uppermost extremity, the bar begins to obstruct the flow resulting in a retardation of flow and the development of a high pressure zone. A corresponding low pressure zone exists on the bottom side of the bar. The imbalance in the pressure on the two sides of the bar pushes the bar downward resulting in the neutralization of the pressure. However, as the bar starts approaching the horizontal position it, owing to inertia, continues to move downward. This results in the development of a high pressure zone on the lower side of the bar and the downward motion is stopped once the pressure value increases to a level where it can prevent further downward motion. At this point the motion once again sees a complete reversal in direction. In Figure 11 one can notice that the flow accelerates in the region close to the lateral surface of the bar when it attains a convex shape while the flow accelerates in the region close to the tip in the concave side. The flow on the either side of the bar thus accelerates on either side of the bar in a periodic fashion and have no interactions till they have passed a distance close the length of the bar.

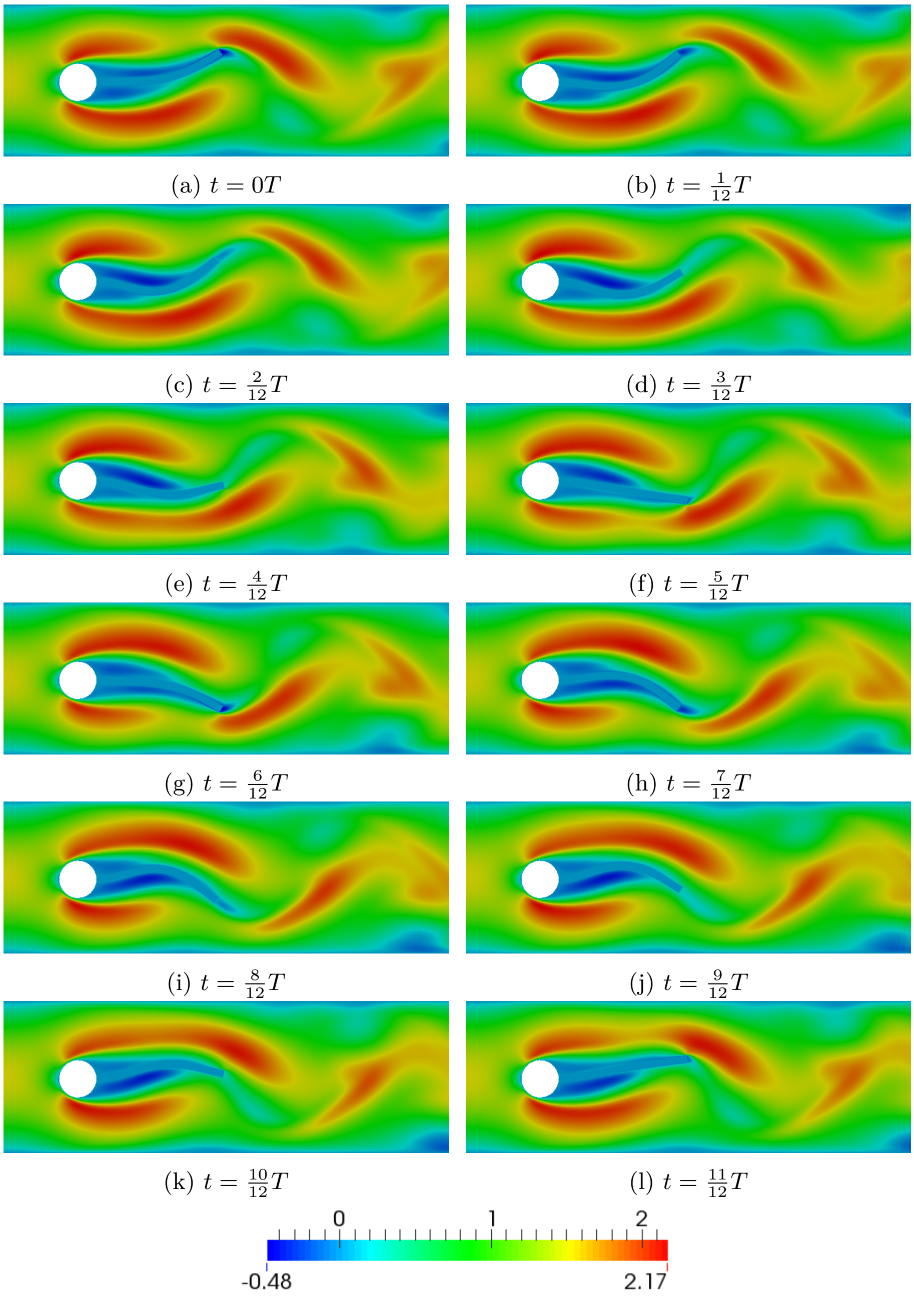
### *4.3. Drag*

Figure 13 gives a detailed comparison of the drag force for different grids and order of elements. Figure 13a shows the drag forces averaged over the last two cycles along with the variations over the cycles. It is clear that for quadratic and cubic elements ($p = 2, 3$) the predicted average drag forces are in excellent agreement with the reference data. For linear elements ($p = 1$), the grid resolution has a relatively bigger impact on the predicted drag. It is not entirely clear if grid independence was in fact realized or not. However, even with the coarsest grids, a switch to quadratic elements improves the prediction remarkably. It can also be inferred from the figure that quadratic elements are sufficient for producing the results in good agreement with the reference data.

Figure 13b, 13c and 13d give a better insight into the evolution of the drag forces over time. It appears that for linear elements different grids predict very different evolutions of the drag force. However, a promotion to quadratic elements diminishes the differences. Except for the coarsest grid G1, all the grids predict similar evolution. Further promotion to cubic elements results in the collapsing of all the evolution profiles of the drag force to a single curve. When one looks at Table 5, where all the frequency results are gathered, one finds that the predictions of drag frequency for quadratic and cubic elements are within 5% while the error goes up to 15% if linear elements are used with coarsest grid G1.

### *4.4. Lift*

A detailed comparison of the lift forces for different grids and orders of elements with the reference data is presented in Figure 14. Figure 14a shows the lift forces averaged over the last two cycles along with the variations over the cycles. Because of the statistical symmetry in the case under symmetry the profile of lift over the upstroke should be exactly the same but opposite in direction during the downstroke. Therefore the lift force averaged over a full cycle will get cancelled resulting in zero net lift force. This can be seen in Figure 14a. Although the simulations conducted with linear elements did not result an exact cancellation of lift forces over a full cycle it is still very small. To get better insight into the evolution of lift forces as a function of time, the temporal profiles of lift, for different grids, for linear, quadratic and cubic elements are presented in Figures 14b, 14c and 14d respectively. It can be seen that for linear elements, grids G1, G2 and G3 show oscillations which vanish when the grid resolution is increased (G4 and G5). In fact the profiles of lift force for G4 and G5 are very close to the reference profile. A promotion from linear to quadratic and then subsequently to cubic elements improves the predication remarkably



(a) $t = 0T$

(b) $t = \frac{1}{12}T$

(c) $t = \frac{2}{12}T$

(d) $t = \frac{3}{12}T$

(e) $t = \frac{4}{12}T$

(f) $t = \frac{5}{12}T$

(g) $t = \frac{6}{12}T$

(h) $t = \frac{7}{12}T$

(i) $t = \frac{8}{12}T$

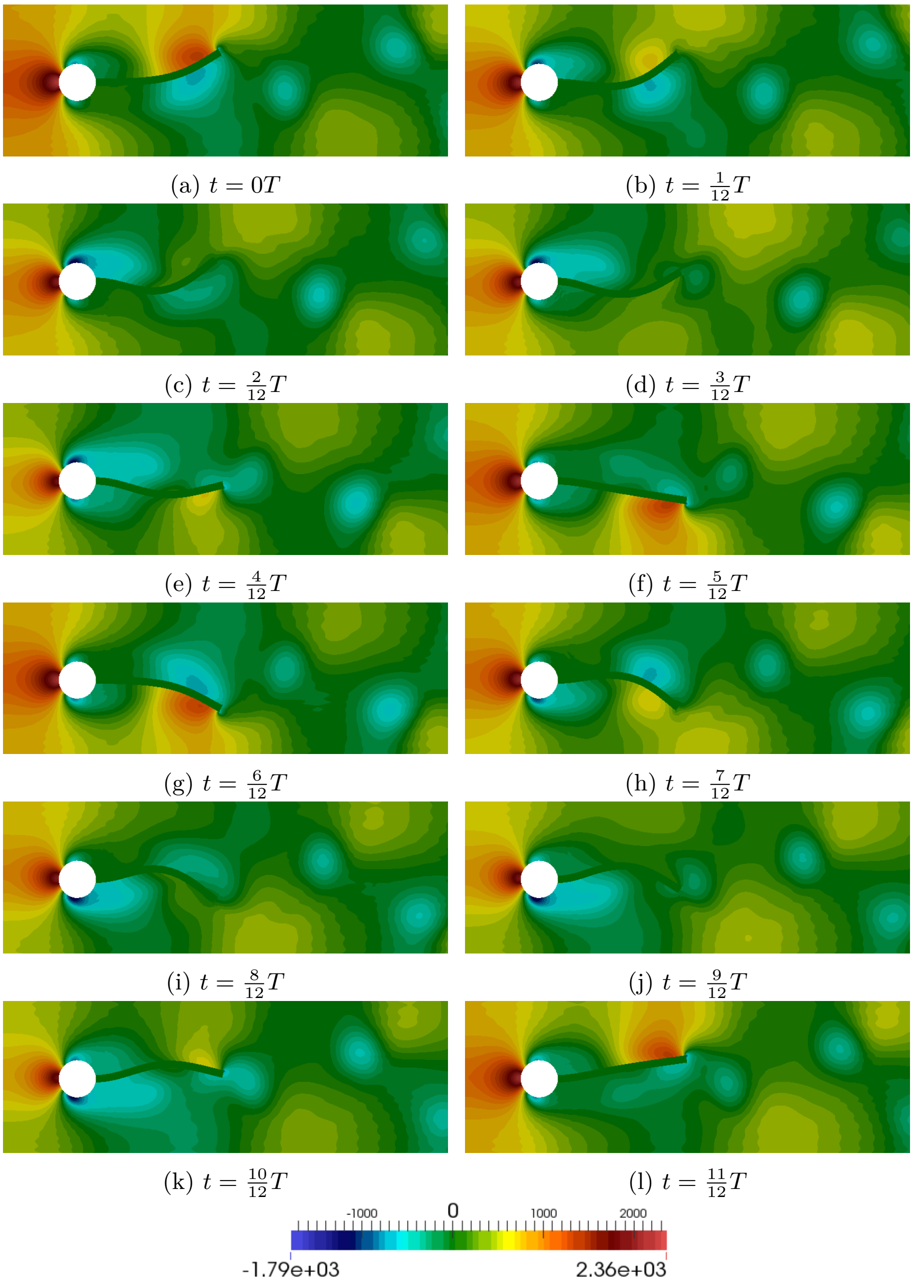
(j) $t = \frac{9}{12}T$

(k) $t = \frac{10}{12}T$

(l) $t = \frac{11}{12}T$

(m) Common legend for $u_x$.

Figure 11: Cyl2DBar: $x$-component, $u_x$ of velocity for twelve timeshots in an oscillation cycle. The oscillation period is denoted $T$.



(a) $t = 0T$

(b) $t = \frac{1}{12}T$

(c) $t = \frac{2}{12}T$

(d) $t = \frac{3}{12}T$

(e) $t = \frac{4}{12}T$

(f) $t = \frac{5}{12}T$

(g) $t = \frac{6}{12}T$

(h) $t = \frac{7}{12}T$

(i) $t = \frac{8}{12}T$

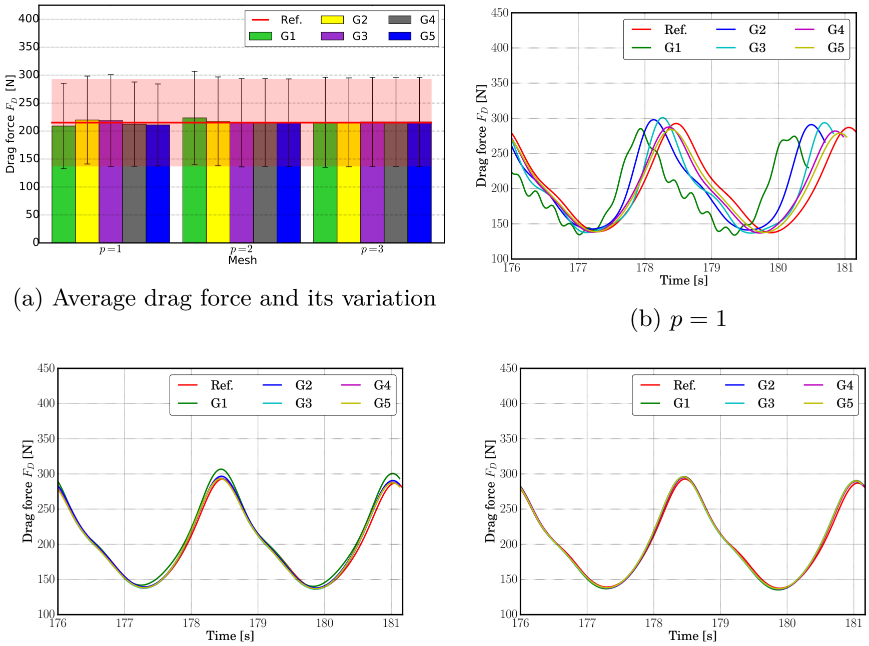
(j) $t = \frac{9}{12}T$

(k) $t = \frac{10}{12}T$

(l) $t = \frac{11}{12}T$

(m) Common legend for $u_x$.

Figure 12: Cyl2DBar: Pressure for twelve timeshots in an oscillation cycle. The oscillation period is denoted $T$.



(a) Average drag force and its variation

(b) $p = 1$

(c) $p = 2$

(d) $p = 3$

Figure 13: Cyl2DBar: Drag force $F_D$ over the last two cycles for $p = 1, 2, 3$ compared against reference results from [44].

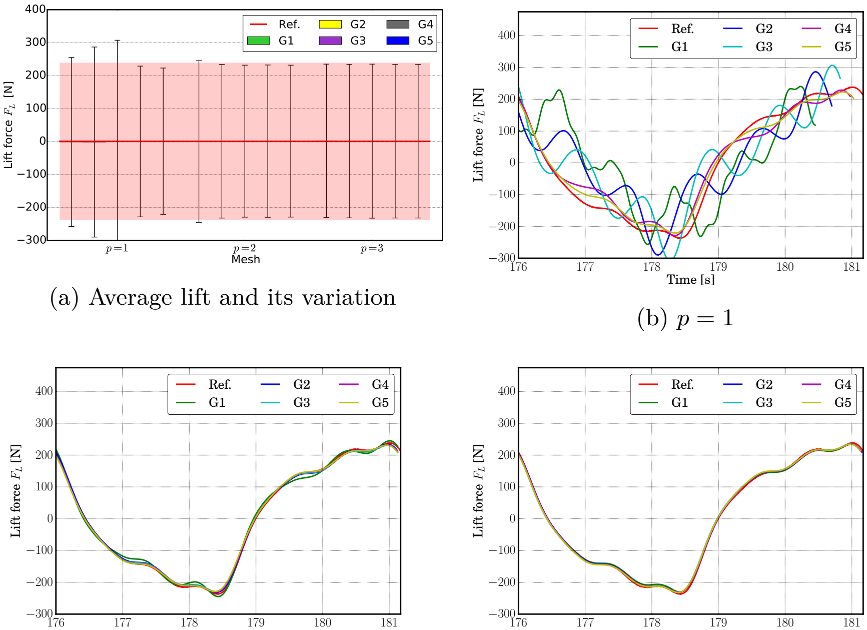
and the effect is more pronounced for the coarser grids G1 and G2. For the cubic elements, once again, like the drag force, the lift force profiles for different grids collapse on each other. When one looks at Table 5 one finds that the predictions of lift frequency for quadratic and cubic elements are within 3% while the error goes up to 11% if linear elements are used with coarsest grid G1.

### *4.5. Lateral and transverse displacements*

Figures 15 and 16 present the comparison of lateral and transverse displacements of the center of the tip against the reference values. Once again the comparison is first made of the displacements averaged over a full cycle and then for their evolution as a function of time. A similar trend as above is observed where the linear elements fail to give precise estimates even with very high resolution while the higher order elements give excellent estimates even with the coarsest grid. Frequency results for the displacements are also listed in Table 5.

### *4.6. Computational cost for linear, quadratic and cubic elements*

All the simulations were conducted using our in-house CFD code IFEM. The linear solvers are based on the PETSc package [45] version 3.4.2 and are compiled with the Intel C++ compiler version 13.0.1, using the SGI MPT MPI implementation, all running on SUSE Linux Enterprise Server 11. The simulations were run on the "Vilje" supercomputer at the Norwegian University of Science and Technology which is currently ranked as number 99 on the top 500



(a) Average lift and its variation

(b) $p = 1$

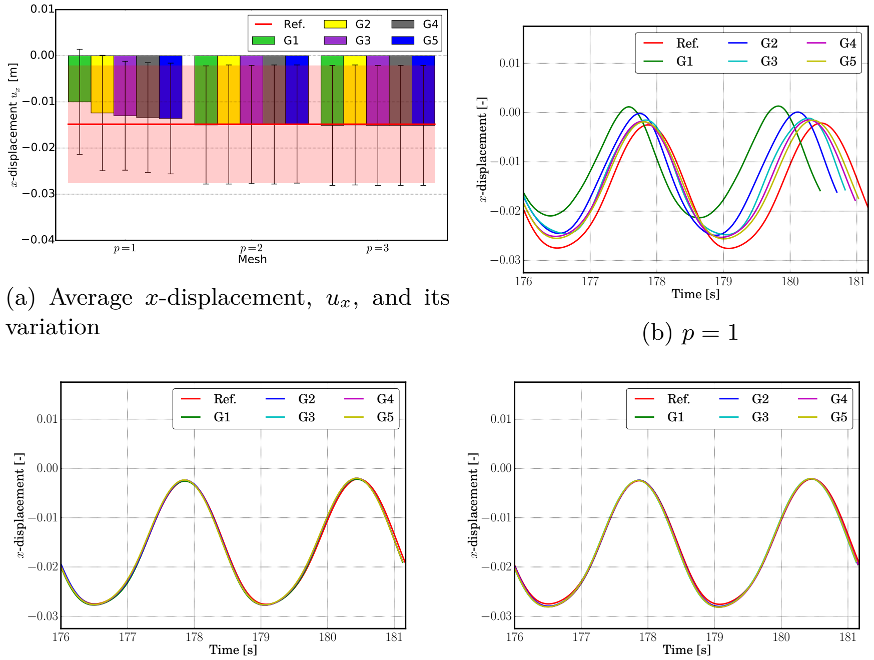
(c) $p = 2$

(d) $p = 3$

Figure 14: Cyl2DBar: Lift force $F_L$ over the last two cycles for $p = 1, 2, 3$ compared with reference results from [44].

Table 5: Cyl2DBar: Frequency results

| | $p$ | $f_{F_D}$ | $f_{F_L}$ | $f_{u_x}$ | $f_{u_y}$ |
|---|---|---|---|---|---|
| G1 | 1 | 4.52 | 2.24 | 4.50 | 2.24 |
| | 2 | 3.92 | 1.94 | 3.87 | 1.94 |
| | 3 | 3.90 | 1.94 | 3.88 | 1.94 |
| G2 | 1 | 4.23 | 2.11 | 4.22 | 2.11 |
| | 2 | 3.86 | 1.95 | 3.92 | 1.95 |
| | 3 | 3.88 | 1.94 | 3.89 | 1.94 |
| G3 | 1 | 4.09 | 2.06 | 4.11 | 2.06 |
| | 2 | 3.87 | 1.94 | 3.89 | 1.94 |
| | 3 | 3.90 | 1.94 | 3.87 | 1.94 |
| G4 | 1 | 3.95 | 2.01 | 3.99 | 2.00 |
| | 2 | 3.87 | 1.94 | 3.89 | 1.94 |
| | 3 | 3.90 | 1.94 | 3.87 | 1.94 |
| G5 | 1 | 3.91 | 1.98 | 3.99 | 1.98 |
| | 2 | 3.88 | 1.94 | 3.89 | 1.94 |
| | 3 | 3.91 | 1.94 | 3.87 | 1.94 |
| Ref. | | 3.86 | 1.93 | 3.86 | 1.93 |



(a) Average $x$-displacement, $u_x$, and its variation

(b) $p = 1$

(c) $p = 2$

(d) $p = 3$

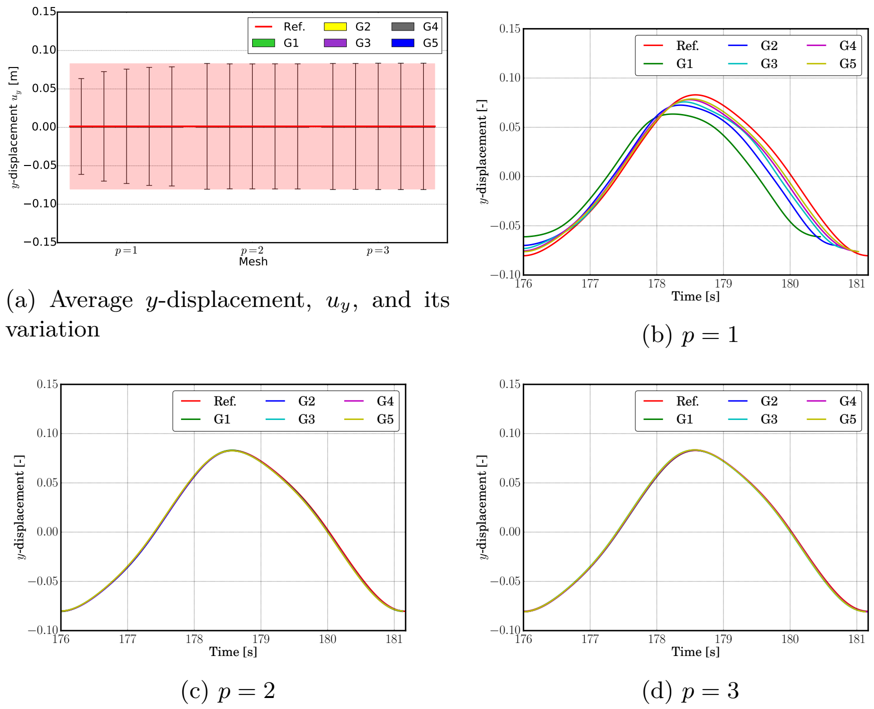
Figure 15: Cyl2DBar: $x$-displacement $u_x$ over the last two cycles for $p = 1, 2, 3$ compared with reference results from [44].

list (June 2014). This is an SGI Altix system with Intel Xeon E5-2670 (Sandy Bridge) processors. The 1404 computational nodes in the system consists of 2 octa-core processors in SMP, with 20MB L3 cache per processor. The nodes are connected using a high-speed infiniband network. A plot of the total CPU time required for running the simulations for all grids is shown in Figure 17.

## 5. Conclusions

In this work we have presented a validation of our isogeometric finite element based FSI code against the benchmark FSI case proposed by Turek in [9] and [10]. Effects of different mesh resolutions, stiffness distributions and orders of elements on the estimation of drag, lift and displacements were investigated. Furthermore, four different criteria for mesh generation were imposed resulting in the simulations providing good agreement with the reference data. Most important conclusions from the work can be enumerated as follows:

1. Different mesh stiffness distributions were tested out of which seven were presented in this paper. It turned out that if for a particular mesh stiffness distribution value, the solution converged then they gave the same estimate of quantities of interest, i.e. drag, lift and displacements in the current case. Furthermore, the different mesh stiffness distribution resulted in similar computational cost associated with them.
2. Linear elements are not good even with reasonably fine grid. This can be attributed to the errors associated with interpolation of forces when trans-



(a) Average $y$-displacement, $u_y$, and its variation

(b) $p = 1$

(c) $p = 2$

(d) $p = 3$

Figure 16: Cyl2DBar: $y$-displacement $u_y$ over th last two cycles for $p = 1, 2, 3$ compared with reference results from [44].

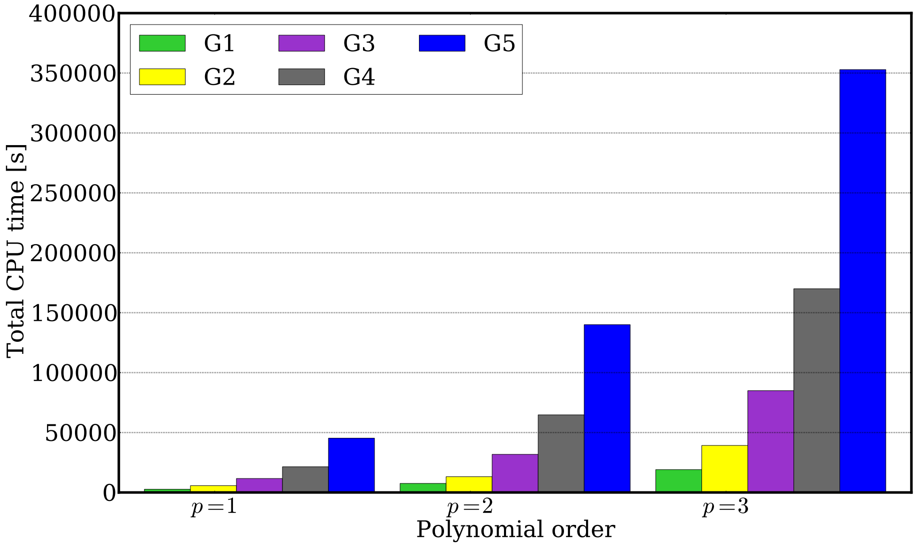


Figure 17: Cyl2DBar: Total CPU time for grids G1-G5 and polynomial orders $p = 1, 2, 3$.

ferred from fluid to solid domain because of the fact that the geometry can not be represented in an exact form using linear elements. Switching to quadratic or cubic elements led to an exact representation of the geometry in combination with better numerical accuracy resulting in a better estimation of drag, lift and displacements.

3. A switch to quadratic from linear elements resulted in a three-fold increase in the computational time and a seven-fold in the case of cubics. However, from the results it is apparent that the associated accuracy with quadratics was sufficient and gave estimates of the relevant quantities within 5% accuracy.

However, we want to underline the fact that the cases investigated herein were for low Reynolds number ($Re = 100$). Since the ultimate goal of this development work is to understand the fluid-structure interaction of monopile flap under operational meteorological conditions, one will always encounter a higher Reynolds number flow which can be characterized by turbulence. Work is underway to implement a turbulence model similar to Large Eddy Simulation (LES) known as Variational Multiscale Models (VMS) in an isogeometric framework.

## 6. Acknowledgment

The authors acknowledge the financial support from the Norwegian Research Council and the industrial partners of *NOWITECH: Norwegian Research Centre for Offshore Wind Technology* (grant no: 193823/S60) (http://www.nowitech.no) and FSI-WT (grant no: 216465/E20) (http://www.fsi-wt.no). Furthermore, the authors greatly acknowledge the *Norwegian metacenter for computational science* (NOTUR-reference number: NN9322K/1589) (www.notur.no) for giving us access to the *Vilje* high performance computer at the Norwegian University of Science and Technology (NTNU).

## References

[1] H. Breusers, A. Raudkivi, Scouring: Hydraulic Structures Design Manual Series, IAHR Design Manual, Taylor & Francis, 1991.

[2] L. De Vos, J. De Rouck, P. Troch, P. Frigaard, Empirical design of scour protections around monopile foundations. part 1: Static approach., Coastal Engineering 58 (6) (2011) 540–553, cited By 0.

[3] L. De Vos, J. De Rouck, P. Troch, P. Frigaard, Empirical design of scour protections around monopile foundations. part 2: Dynamic approach, Coastal Engineering 60 (1) (2012) 286–298.

[4] A. Nielsen, B. Sumer, J. Fredsoe, E. Christensen, Scour protection around offshore wind turbines: Monopiles, no. 210 GSP, 2010, pp. 440–449.

[5] B. De Sonneville, D. Rudolph, T. Raaijmakers, Scour reduction by collars around offshore monopiles, no. 210 GSP, 2010, pp. 460–470.

[6] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering 194 (39-41) (2005) 4135–4195.

[7] I. Akkerman, Y. Bazilevs, V. Calo, T. Hughes, S. Hulshoff, The role of continuity in residual-based variational multiscale modeling of turbulence, Computational Mechanics 41 (3) (2008) 371–378.

[8] V. Nguyen, C. Anitescu, S. Bordas, T. Rabczuk, Isogeometric analysis: An overview and computer implementation aspects, Mathematics and Computers in Simulation 117 (2015) 89–116.

[9] S. Turek, J. Hron, Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow, Tech. rep., University of Dortmund (2006).

[10] S. Turek, J. Hron, M. Madlik, M. Razzaq, H. Wobker, J. Acker, Numerical simulation and benchmarking of a monolithic multigrid solver for fluid-structure interaction problems with application to hemodynamics, Lecture Notes in Computational Science and Engineering 73 LNCSE (2010) 193–220.

[11] J. A. Cottrell, T. J. Hughes, Y. Bazilevs, Isogeometric analysis: toward integration of CAD and FEA, Wiley, Chichester, 2009.

[12] K. Nordanger, R. Holdahl, A. M. Kvarving, A. Rasheed, T. Kvamsdal, Implementation and comparison of three isogeometric Navier-Stokes solvers applied to simulation of flow past a fixed 2D NACA0012 airfoil at high Reynolds number, Computer Methods in Applied Mechanics and Engineering 284 (2015) 664–688.

[13] F. Brezzi, On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers, Rev. Francaise Automat. Informat. Recherche Operationelle Ser Rouge 8.

[14] A. J. Chorin, Numerical solution of the Navier-Stokes equations, Mathematics of Computation 22 (1968) 745–762.

[15] R. Temam, Sur l'approximation de la solution des equations de navier-stokes par la methode des pas fractionnaires (ii), Archive for Rational Mechanics and Analysis 33 (5) (1969) 377–385.

[16] P. Minev, A stabilized incremental projection scheme for the incompressible Navier-Stokes equations, International Journal for Numerical Methods in Fluids 36 (4) (2001) 441–464.

[17] H. M. Hilber, T. J. R. Hughes, R. L. Taylor, Improved numerical dissipation for time integration algorithms in structural dynamics, Earthquake Engineering & Structural Dynamics 5 (3) (1977) 283–292.

[18] S. Badia, A. Quaini, A. Quarteroni, Modular vs. non-modular preconditioners for fluid-structure systems with large added-mass effect, Computer Methods in Applied Mechanics and Engineering 197 (49-50) (2008) 4216–4232.

[19] Y. Bazilevs, V. Calo, T. Hughes, Y. Zhang, Isogeometric fluid-structure interaction: Theory, algorithms, and computations, Computational Mechanics 43 (1) (2008) 3–37.

[20] Y. Bazilevs, J. Gohean, T. Hughes, R. Moser, Y. Zhang, Patient-specific isogeometric fluid-structure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device,

Computer Methods in Applied Mechanics and Engineering 198 (45-46) (2009) 3534–3550.

[21] U. Langer, H. Yang, Robust and efficient monolithic fluid-structure-interaction solvers, International Journal for Numerical Methods in Engineering.

[22] E. van Brummelen, Partitioned iterative solution methods for fluid-structure interaction, International Journal for Numerical Methods in Fluids 65 (1-3) (2011) 3–27.

[23] H. Morand, R. Ohayon, Fluid-Structure Interaction: Applied Numerical Methods, Wiley-Masson Series Research in Applied Mathematics, Wiley, 1995.

[24] E. van Brummelen, Added mass effects of compressible and incompressible flows in fluid-structure interaction, Journal of Applied Mechanics, Transactions ASME 76 (2) (2009) 1–7.

[25] P. Le Tallec, J. Mouro, Fluid structure interaction with large structural displacements, Computer Methods in Applied Mechanics and Engineering 190 (24-25) (2001) 3039–3067.

[26] P. Causin, J. Gerbeau, 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) (2005) 4506–4527.

[27] J. Donea, S. Giuliani, J. Halleux, An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions, Computer Methods in Applied Mechanics and Engineering 33 (1-3) (1982) 689–723.

[28] C. B. Jenssen, T. Kvamsdal, K. M. Okstad, J. Amundsen, Parallel methods for fluid-structure interaction, in: B. Kågström, J. Dongarra, E. Elmroth, J. Waśniewski (Eds.), Applied Parallel Computing Large Scale Scientific and Industrial Problems, Vol. 1541 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 1998, pp. 263–274.

[29] T. Kvamsdal, C. B. Jenssen, K. M. Okstad, J. Amundsen, Fluid-structure interactions for structural design, in: T. Kvamsdal (Ed.), Computational Methods for Fluid-Structure Interactions, Tapir Publisher, Trondheim, 1999, pp. 211–238.

[30] T. Kvamsdal, K. M. Okstad, K. Sørli, P. Pegon, Two-level adaptive mesh movement algorithms for FSI-computations, in: T. Kvamsdal (Ed.), Computational Methods for Fluid-Structure Interactions, Tapir Publisher, Trondheim, 1999, pp. 121–132.

[31] L. Formaggia, F. Nobile, Stability analysis for the arbitrary lagrangian eulerian formulation with finite elements, East-West Journal of Numerical Mathematics 7 (2) (1999) 105–131.

[32] J. Donea, A. Huerta, J.-P. Ponthot, A. Rodriguez-Ferran, Encyclopedia of Computational Mechanics, John Wiley, Chichester, West Sussex, 2004, Ch. Arbitrary Lagrangian-Eulerian Methods.

[33] M. Souli, D. Benson, Arbitrary Lagrangian Eulerian and Fluid-Structure Interaction: Numerical Simulation, ISTE, Wiley, 2010.

[34] Y. Bazilevs, K. Takizawa, T. Tezduyar, Computational Fluid-Structure Interaction: Methods and Applications, Wiley Series in Computational Mechanics, Wiley, 2013.

[35] K. Herfjord, G. Olsen, T. Kvamsdal, K. M. Okstad, Fluid–Structure Interaction in the case of Two Independent Cylinders in Vortex Induced Vibrations (VIV), International Journal of Applied Mechanics and Engineering 7 (2002) 65–83.

[36] K. Stein, T. Tezduyar, R. Benney, Automatic mesh update with the solid-extension mesh moving technique, Computer Methods in Applied Mechanics and Engineering 193 (21-22) (2004) 2019–2032.

[37] R. Bartels, J. Beatty, B. Barsky, An Introduction to Splines for Use in Computer Graphics and Geometric Modeling, Morgan Kaufmann Series in Computer Graphics and Geometric Modeling, Morgan Kaufmann, 1987.

[38] S. A. Coons, Surfaces for computer-aided design of space forms, Tech. Rep. MAC-TR-41, Cambridge, MA, USA (1967).

[39] N. Lomb, Least-squares frequency analysis of unequally spaced data, Astrophysics and Space Science 39 (2) (1976) 447–462.

[40] J. D. Scargle, Studies in astronomical time series analysis. II - Statistical aspects of spectral analysis of unevenly spaced data, The Astrophysical Journal 263 (1982) 835–853.

[41] P. Knupp, Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part II - A framework for volume mesh optimization and the condition number of the Jacobian matrix, International Journal for Numerical Methods in Engineering 48 (8) (2000) 1165–1185.

[42] C. J. Stimpson, C. D. Ernst, P. Knupp, P. P. Pebay, D. Thompson, The verdict geometric quality library, Tech. Rep. 2007-1751, Sandia National Laboratories (2007).

[43] P. Knupp, Algebraic mesh quality metrics for unstructured initial meshes, Finite Elements in Analysis and Design 39 (3) (2003) 217–241.

[44] S. Turek, FEATFLOW Web page, http://www.featflow.de/en/benchmarks/cfdbenchmarking/fsi_benchmark.html (2014).

[45] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, H. Zhang, PETSc Web page, http://www.mcs.anl.gov/petsc (2014).