

The Turek&Hron FSI Benchmarks

Master Seminar Partitioned Fluid-Structure Interaction

Vojtěch Kubáč

Department of Mathematics and Physics

Charles University in Prague

Email: vojta.kubac@gmail.com

Abstract—The purpose of this paper is to discuss a widely used benchmark for Fluid-Structure Interaction (FSI). This benchmark was introduced by Turek and Hron in [1] and will be referred to as Turek&Hron benchmark. There are three families of problems in the paper. The first family serves for validating the fluid solver and is called CFD (Computational Fluid Dynamics). The second one contains three problems from elasticity and is called CSM (Computational Solid Mechanics). Finally, the last family of problems introduces problems for Fluid-Structure Interaction, hence the name is FSI. The setting of these problems is introduced in the first part of this paper.

We present a partitioned solution to the FSI benchmarks. Our solid solver is presented and validated in Section II, for the fluid solver we use an existing code.

In the last section, our results to the coupled FSI problem is discussed.

Index Terms—Fluid-Structure Interaction, benchmark, FEniCS, OpenFOAM

INTRODUCTION

Benchmark problems are important for validation and comparison of different methods, algorithms and implementations of the same problem in computational science. They provide reference values which are believed to be very close to the real solution of given problems so one can check the accuracy and efficiency (if the cost of computation is given too) of their code.

The Turek&Hron benchmark is used for Fluid-Structure Interaction problems. It describes a flow around a cylinder with an elastic flap attached to the back side of it, see Figure 1. The geometry is based on the standard fluid dynamics benchmark flow around a cylinder [2].

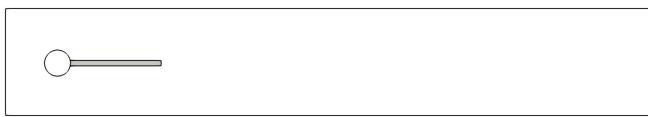


Fig. 1. Geometry for benchmarks, solid part painted in gray.

We use a partitioned approach to solve the Fluid-Structure Interaction benchmarks using the preCICE library [3] for coupling. We have one solver for fluid and one solver for solid (they can be written in different languages), in each time step the forces are mapped from fluid to solid as a Neumann boundary condition and displacement is mapped from solid to fluid and changes its geometry. preCICE considers the mappings and about the convergence in each time step.

As the starting point for our computation, we use code from preCICE tutorials [4] where the Turek&Hron benchmark is solved with OpenFOAM [5] and CalculiX [6]. The Fluid-Structure Interaction code is part of Derek Risseeuw's Master Thesis [7]. preCICE communication with OpenFOAM is done through OpenFOAM adapter [8]. We use the OpenFOAM code to solve the fluid problem, for the validation on CFD problems see [7] (for the specification of CFD problems, see Section I-C of this paper).

For the solid problem we use our own FEniCS [9] code that was written for this seminar. The validation of CSM problems (see Section I-D of this paper) is shown in Section II of this paper.

Section III deals with the FSI benchmarks (see Section I-E of this paper), where the fluid solver and the solid solver are coupled with preCICE. The obtained results are discussed here.

I. TUREK&HRON FSI BENCHMARKS

In this section, we describe in detail the definitions of Turek&Hron benchmarks; the layout more or less follows the original paper [1]. In Subsection I-A we provide the setting of the benchmark, which consists of the governing equations, interaction conditions, boundary and initial conditions and the considered geometry. In Subsection I-B we set the quantities for comparison. Then two parts on the partial tests follow, Subsection I-C about the fluid problem and Subsection I-D about the elasticity problem. Finally, Subsection I-E describes the FSI benchmarks. The last part of this section, Subsection I-F, provides some references connected to Turek&Hron benchmarks.

A. The Benchmark Setting

We consider the flow of an incompressible, Newtonian, homogeneous fluid around a rigid cylinder with an elastic flap attached behind it. We denote the fluid domain as Ω_t^f and the elastic solid domain as Ω_t^s , where the subscript t denotes time since we consider time-dependent geometries. Next, we define the interface between the fluid and elastic solid as $\Gamma_t = \partial\Omega_t^f \cap \partial\Omega_t^s$. Note that the rigid cylinder is not considered as part of the computational domain.

a) Governing Equations: As mentioned above, the fluid is considered to be incompressible, Newtonian and homogeneous. Its state is described by the velocity \mathbf{v}^f and pressure

p^f . The governing equations for such a fluid are Navier-Stokes equations

$$\begin{aligned} \rho^f \partial_t \mathbf{v}^f + \rho^f (\mathbf{v}^f \cdot \nabla) \mathbf{v}^f &= \operatorname{div} \mathbb{T}^f && \text{in } \Omega_t^f, \\ \operatorname{div} \mathbf{v}^f &= 0 \end{aligned}$$

The symbol ρ^f denotes the constant *fluid density* and \mathbb{T}^f denotes the *Cauchy stress tensor* which is given by the constitutive relation

$$\mathbb{T}^f = -p^f \mathbb{I} + \rho^f \nu^f (\nabla \mathbf{v}^f + \nabla \mathbf{v}^{fT}),$$

where ν^f is the constant *viscosity* of the fluid.

The solid structure is assumed to be elastic and compressible. The standard way to write the balance equation for solids is using the Lagrangian description, with respect to some reference (usually initial) configuration $\hat{\Omega}^s$,

$$\hat{\rho}^s \partial_{tt} \hat{\mathbf{u}}^s = \hat{\rho}_s \hat{\mathbf{g}}_s + \widehat{\operatorname{div}} \hat{\mathbb{T}}^s \quad \text{in } \hat{\Omega}^s,$$

where $\hat{\rho}^s$ is the *referential density*, $\hat{\mathbf{g}}_s$ is the *volume force* acting on the body and $\hat{\mathbb{T}}^s = (\det \mathbb{F}) \mathbb{T}^s \mathbb{F}^{-T}$ is the *1st Piola-Kirchhoff stress tensor*. Finally, $\mathbb{F} = \mathbb{I} + \hat{\nabla} \hat{\mathbf{u}}^s$ is the *deformation gradient*. For more detail, see any continuum mechanics book (for example [10]).

The material is specified by the constitutive law for which the *St. Venant-Kirchhoff* model was used in the benchmark. The standard way to write the equation is with the use of the *2nd Piola-Kirchhoff stress tensor* $\hat{\mathbb{S}}^s = \mathbb{F}^{-1} \hat{\mathbb{T}}^s = (\det \mathbb{F}) \mathbb{F}^{-1} \mathbb{T}^s \mathbb{F}^{-T}$. The constitutive law is

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

where $\hat{\mathbb{E}}^s = \frac{1}{2} (\mathbb{F}^T \mathbb{F} - \mathbb{I})$ and λ^s and μ^s are *Lamé constants*.

An alternative pair of constants is the *Young modulus* E^s which describes the response of material to the stress ("the measure of stiffness") and the *Poisson ratio* ν^s which describes the response to deformation of a material in a perpendicular direction ($\nu^s = -\frac{\text{transverse strain}}{\text{axial strain}}$), $\nu^s = 0.5$ for an incompressible material. E^s and ν^s are easier to measure than Lamé constants λ^s and μ^s , but the later ones are more convenient from the point of view of the constitutive equations. Luckily, there is a simple relation between these two sets, it reads

$$\begin{aligned} \nu^s &= \frac{\lambda^s}{2(\lambda^s + \mu^s)} & E^s &= \frac{\mu^s(3\lambda^s + 2\mu^s)}{(\lambda^s + \mu^s)} \\ \mu^s &= \frac{E^s}{2(1 + \nu^s)} & \lambda^s &= \frac{\nu^s E^s}{(1 + \nu^s)(1 - 2\nu^s)}. \end{aligned}$$

b) *Interaction Conditions*: As stated above, we consider a viscous Newtonian fluid, for which usually the *no-slip* boundary condition is used on the boundary when interacting with a solid impermeable structure. This condition means that the fluid sticks to the solid and has the same velocity as the solid structure. Very often the structure does not move (in our case the walls of the channel and surface of rigid circle), this condition is then a zero Dirichlet boundary condition for the fluid velocity \mathbf{v}^f . This is however not the case of the FSI interface. Here the no-slip condition reads

$$\mathbf{v}^f(x, t) = \hat{\mathbf{v}}^s(\chi_s^{-1}(x, t), t) \quad \text{on } \Gamma^t,$$

where $\chi : \hat{\Omega}^s \times I \mapsto \Omega_t^s$, $\chi(X, t) = x$ maps the Lagrangian points to their Eulerian counterparts and is called *deformation*, $\hat{\Omega}^s = \Omega_0^s$ is the *reference configuration* of the solid and $I = (0, T)$ represents the time interval. With the use of the Lagrangian variables we get

$$\mathbf{v}^f(X + \hat{\mathbf{u}}^s(X, t), t) = \hat{\mathbf{v}}^s(X, t) \quad \text{on } \Gamma^t.$$

This condition is called *kinematic condition*. There is one more condition we would like to be satisfied, it is an application of Newton's action-reaction law. More specifically, we require that the forces on the interface are in balance,

$$\mathbb{T}_f \mathbf{n} = \mathbb{T}_s \mathbf{n} \quad \text{on } \Gamma^t,$$

where \mathbf{n} is the outer normal of the fluid ($\mathbf{n} = \mathbf{n}^f$) or solid domain ($\mathbf{n} = \mathbf{n}^s$). Since $\mathbf{n}^f = -\mathbf{n}^s$, the choice is arbitrary. This condition is known as *dynamic condition*.

c) *Boundary and Initial Conditions*: As indicated above, we assume that the fluid sticks on the walls and on the surface of the rigid cylinder, so we imply the zero Dirichlet boundary condition for fluid velocity \mathbf{v}^f there. We also assume that the elastic flap is fixed to the rigid circle, so we again assume the zero Dirichlet boundary condition, this time for the solid displacement $\hat{\mathbf{u}}^s$. On the left boundary we prescribe a parabolic inflow

$$\mathbf{v}^f(0, y) = 1.5 \bar{U} \frac{y(H-y)}{(\frac{H}{2})^2}, \quad (1)$$

where \bar{U} is the mean flow velocity and H is the height of the channel. The maximum velocity of the profile is $1.5\bar{U}$. On the right boundary we set the *do-nothing* boundary condition, $\mathbb{T}_f \mathbf{n} = 0$.

Due to the computational reasons, it is reasonable to start the time dependent tests from zero fluid velocity \mathbf{v}^f and zero solid displacement $\hat{\mathbf{u}}^s$ and then start increasing the inflow boundary condition. The approach suggested in Turek & Hron paper [1] is

$$\mathbf{v}^f(t, 0, y) = \begin{cases} \mathbf{v}^f(0, y) \frac{1-\cos(\frac{\pi}{2}t)}{2} & \text{if } t < 2.0 \\ \mathbf{v}^f(0, y) & \text{otherwise,} \end{cases} \quad (2)$$

where $\mathbf{v}^f(0, y)$ is the velocity profile from (1).

d) *Geometry*: The domain geometry is based on the classical 2D flow around a cylinder benchmark [2]. The only differences are that we consider an elastic flap stick to the rigid cylinder (there is just a rigid cylinder surrounded by fluid in the flow around a cylinder benchmark) and that the channel is a little bit longer (2.5 units instead of 2.2). So the geometry is given by

- The domain dimensions are: length $L = 2.5$, height $H = 0.41$.
- The circle center is positioned at $C = (0.2, 0.2)$ (measured from the left bottom corner of the channel) and the radius is $r = 0.05$.
- The elastic structure bar has length $l = 0.35$ and height $h = 0.02$, the right bottom corner is positioned at

(0.6, 0.19), and the left end is fully attached to the fixed cylinder.

- The control points are the end of the elastic flap $A(t)$, with $A(0) = (0.6, 0.2)$, and the leftmost point of rigid cylinder $B = (0.15, 0.2)$.

B. Quantities for Comparison

For the comparison of results three different quantities are used

- 1) The displacement at the end of the elastic beam, at the point $A(t)$.
- 2) Forces exerted by the fluid on the whole submerged body, i.e. lift and drag forces acting on the cylinder and the beam structure together

$$(F_D, F_L) = \int_S \mathbf{T} \mathbf{n} d\sigma,$$

where S denotes the part of the circle being in contact with the fluid plus part of the boundary of the beam structure being in contact with the fluid and \mathbf{n} is the outer unit normal vector to the fluid domain.

- 3) Pressure difference between points $A(t)$ and B

$$\Delta p(t) = p(B) - p(A(t)).$$

C. The Fluid Problem

The first validation problem is for the fluid problem. These tests are referred to as CFD tests (the abbreviation stands for Computational Fluid Dynamics). Here the geometry is the same as in the FSI case described above in the subsection *Geometry*, with the difference that the flap behind cylinder is completely rigid. There are two possibilities how to make the flap rigid, the first, more straightforward one, is to cut the flap out of the geometry and consider just the fluid domain Ω_0^f (which is now independent of time) and prescribe standard no-slip condition on the interaction between the fluid and the flap. The other possibility is to use the full FSI setting (with elastic solid) and make the solid almost rigid by setting large structural parameters ($\rho^s \approx 10^6 \frac{\text{kg}}{\text{m}^3}$, $\mu^s \approx 10^{12} \frac{\text{kg}}{\text{ms}^2}$). The first choice is more than natural for partitioned approach, while for the monolithic approach it requires generating a new mesh and significant changes in the code.

Three different choices for the problem-specific constants are considered (see Table I). The first two lead to stationary solutions, whereas the third results in a turbulent flow. The values for comparison are the forces exerted by the fluid (the lift and drag).

D. The Solid Problem

The elasticity validation problem considers an elastic flap fixed on the left. On this flap acts a volume force (for example gravitational) $\mathbf{g} = (0, g)[\frac{\text{m}}{\text{s}^2}]$. The geometry is the same as the geometry of the flap in FSI benchmarks as described in the section *Geometry*. The test problems are abbreviated as CSM problems (Computational Structural Mechanics). The set of the test problems again consists of different choices of problem specific parameters (prescribed in Table II). Another

TABLE I
PARAMETER SETTINGS FOR THE CFD TESTS

| dimensional parameter | CFD1 | CFD2 | CFD3 |
|--|------|------|------|
| $\rho^f [10^3 \frac{\text{kg}}{\text{m}^3}]$ | 1 | 1 | 1 |
| $\nu^f [10^{-3} \frac{\text{m}^2}{\text{ms}}]$ | 1 | 1 | 1 |
| U | 0.2 | 1 | 2 |
| non-dimensional parameter | CFD1 | CFD2 | CFD3 |
| $\text{Re} = \frac{Ud}{\nu^f}$ | 20 | 100 | 200 |
| \bar{U} | 0.2 | 1 | 2 |

difference is that the first two problems, CSM1 and CSM2, are computed as steady-state solutions, while CSM3 problem is time-dependent. To be more specific, in the first two cases we seek for an equilibrium solution; the volume force \mathbf{g} is in balance with the elasticity and the time derivative of displacement is set to zero, $\partial_{tt} \hat{\mathbf{u}}^s = \mathbf{0}$. In the CSM3 setting we start from undeformed configuration and at the time $t = 0$ the volume force starts to act. This will deform the elastic body and at some point the elasticity forces starts to move the flap back up to the initial state. This is happening again and again so we obtain an oscillatory movement of the flap. We assume no dissipation or resistance here so one should obtain a perfectly elastic behavior.

E. The Fluid-Structure Interaction

Now, we finally get to the Fluid-Structure Interaction problems, the abbreviation of these final tests is FSI. Three cases are considered again, the first one (FSI1) results in a steady solution, while the other two (FSI2, FSI3) result in turbulent flows with oscillating flap. FSI3 has faster oscillations compared to FSI2, while FSI2 possesses a larger deformation of the elastic beam. For the problem-specific values see Table III. Note that the fluid parameters for FSI are the same as in some CFD problems, the same holds for the structural parameters and CSM problems.

F. Further Publications

The test problems described above are computed by several researcher groups using different approaches, the results of their computations are collected in Trek et al. [11]. There is also a web-page [12] with the benchmark setting and the reference values to download.

TABLE II
PARAMETER SETTINGS FOR THE CSM TESTS

| dimensional parameter | CSM1 | CSM2 | CSM3 |
|--|-------------------|-------------------|-------------------|
| $\rho^s [10^3 \frac{\text{kg}}{\text{m}^3}]$ | 1 | 1 | 1 |
| ν^s | 0.4 | 0.4 | 0.4 |
| $\mu^s [10^6 \frac{\text{kg}}{\text{ms}^2}]$ | 0.5 | 2.0 | 0.5 |
| g | 2 | 2 | 2 |
| non-dimensional parameter | CSM1 | CSM2 | CSM3 |
| ν^s | 0.4 | 0.4 | 0.4 |
| E^s | 1.4×10^6 | 5.6×10^6 | 1.4×10^6 |
| g | 2 | 2 | 2 |

TABLE III
PARAMETER SETTINGS FOR THE FSI TESTS

| dimensional parameter | FSI1 | FSI2 | FSI3 |
|--|-------------------|-------------------|-------------------|
| $\rho^s [10^3 \frac{\text{kg}}{\text{m}^3}]$ | 1 | 10 | 1 |
| ν^s | 0.4 | 0.4 | 0.4 |
| $\mu^s [10^6 \frac{\text{kg}}{\text{ms}^2}]$ | 0.5 | 0.5 | 2.0 |
| $\rho^f [10^3 \frac{\text{kg}}{\text{m}^3}]$ | 1 | 1 | 1 |
| $\nu^f [10^{-3} \frac{\text{m}^2}{\text{ms}}]$ | 1 | 1 | 1 |
| U | 0.2 | 1 | 2 |
| non-dimensional parameter | FSI1 | FSI2 | FSI3 |
| $\beta = \frac{\rho^s}{\rho^f}$ | 1 | 10 | 1 |
| ν^s | 0.4 | 0.4 | 0.4 |
| $Ae = \frac{E^s}{\rho^f U^2}$ | 3.5×10^4 | 1.3×10^3 | 1.4×10^3 |
| $Re = \frac{Ud}{\nu^f}$ | 20 | 100 | 200 |
| \bar{U} | 0.2 | 1 | 2 |

II. CSM-TESTS RESULTS

As mentioned in the introduction to this paper, in this section, we present the results of our computations to CSM validation tests. The code used for this computations can be found on my GitHub page [13]. The spatial discretization uses Finite Element Method and for the time dependent cases we choose Crank-Nicolson discretization in time. The computational framework we used is FEniCS [9].

A. Stationary Problems

The first two cases are stationary problems and the material constants differ only in the value of Young modulus E^s . In CSM2 case the Young modulus is higher which means the structure possesses higher stiffness, therefore we expect smaller deformation than in CSM1 case. Indeed, the results confirm this statement, in Figure 2 the difference is shown. For the comparison of computed values see Tables IV and V. The results are computed on iteratively refined meshes (mesh level L0 means no refinement, L1 means once refined, L2 twice, L3 three times and L4 four times refined). The refinement from a coarser mesh to a finer one is done by dividing each element of the coarser mesh into four smaller elements, the finer mesh then consists of those smaller elements. For the mesh generation and refinement *gmsh* [14] meshing tools were used. The column *dofs* in Table V

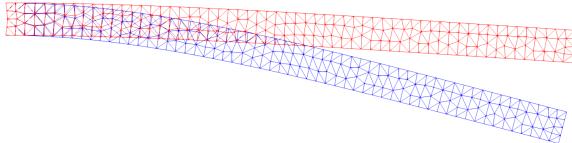


Fig. 2. Displacement to CSM1 (blue) and CSM2 (red) benchmarks. For the values see Tables IV and V.

TABLE IV
DISPLACEMENT VALUES FOR CSM1 TESTS.

| mesh level | dofs | ux of A [$\times 10^{-3}$] | uy of A [$\times 10^{-3}$] |
|------------|-------|------------------------------|------------------------------|
| L0 | 290 | -7.050196 | -65.38392 |
| L1 | 970 | -7.151004 | -65.90946 |
| L2 | 4026 | -7.176712 | -66.04488 |
| L3 | 15282 | -7.183877 | -66.08239 |
| L4 | 59490 | -7.186371 | -66.09550 |
| ref. | | -7.187 | -66.10 |

TABLE V
DISPLACEMENT VALUES FOR CSM2 TESTS.

| mesh level | dofs | ux of A [$\times 10^{-3}$] | uy of A [$\times 10^{-3}$] |
|------------|-------|------------------------------|------------------------------|
| L0 | 290 | -0.4596192 | -16.78146 |
| L1 | 970 | -0.4665459 | -16.92244 |
| L2 | 4026 | -0.4682577 | -16.95861 |
| L3 | 15282 | -0.4687436 | -16.96857 |
| L4 | 59490 | -0.4689118 | -16.97205 |
| ref. | | -0.4690 | -16.97 |

represents the number of degrees of freedom for the respective refinement level. The row *ref.* represents the reference values downloaded from [12].

We see that we were able to obtain reference values with our computations (on the refinement level L4) and that we were quite close even for level L1 with 970 degrees of freedom.

B. CSM3 Benchmark

For the time dependent case CSM3 (the same structural properties as CSM1) we provide again a case study for mesh coarseness, with the same iterative refinement as in the previous section. We also show dependency on the size of the time step. Our approach uses fixed step size and the time discretization scheme is Crank-Nicolson scheme (for the reasoning of this choice see the discussion in Remark 1).

a) *Mesh coarseness case study:* These results are computed with fixed time step size $\Delta t = 0.005$. We plot our computed results in multiple figures. Figures 3 and 4 are plots over the whole time interval for the displacement in the direction x and y , respectively. From these plots we see a good overall agreement with the reference values, but we don't see how close the values actually are. For this reason, we also plot Figures 5 and 6 where the displacements are plotted in detail in the first oscillation. We see a convergence to the reference values, and that the results for L4 refinement copy the reference values.

b) *Time step size case study:* Now, we choose mesh L4 and compute the benchmark for different time step sizes $\Delta t = 0.02, 0.01, 0.005$. We again show the plots over the whole time interval in Figures 7 and 8 for the displacement in the direction x and y , respectively. And also the detailed plots in Figures 9 and 10 as before. We again see that all choices behave nice from the not so detailed point of view and a good convergence to the reference values is obtained. Note that the case with the smallest time step is the same case as the case with the finest mesh from the mesh coarseness study.

Remark 1: CSM3 benchmark is sensitive to the choice of

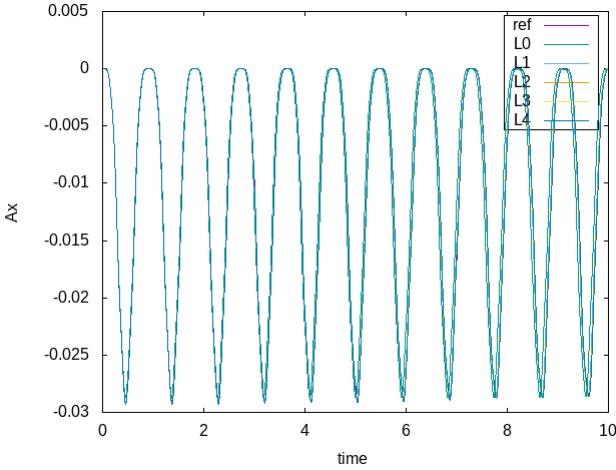


Fig. 3. Displacement in the direction x for CSM3 with various mesh coarseness.

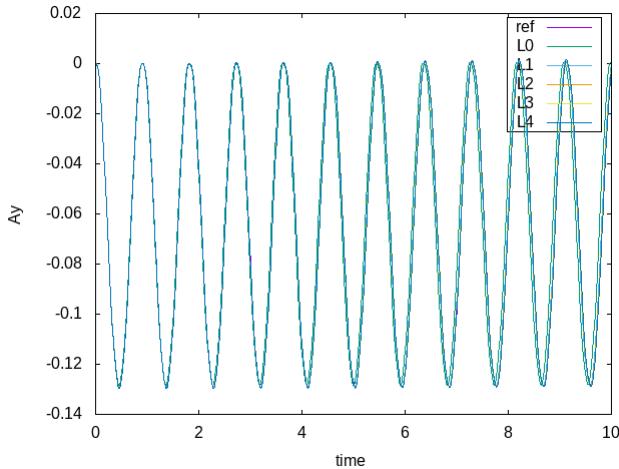


Fig. 4. Displacement in the direction y for CSM3 with various mesh coarseness.

the time discretization scheme. Since no damping is considered (friction, resistance, etc. is not taken into account), a choice of a discretization scheme with damping property would lead to undesired results. This is shown in Figures 11 and 12. The considered discretization schemes are Crank-Nicolson scheme (CN), backward Euler scheme (BE) -also known as implicit Euler- and backward difference formula of the second order (BDF). How this discretization schemes discretize a general differential equation $y'(x) = f(x, y)$ is listed in Table VI.

It is a known fact that BE has a strong damping property (which can be wanted or even needed in some cases, but undesired in ours) of the first order and CN is a non-damping scheme of the second order. On the other hand BE is more stable. BDF then combines advantages of both, it is of the second order and possesses a good stability, it has also damping properties but much smaller than BE. These theoretical statements are confirmed in Figures 11 and 12.

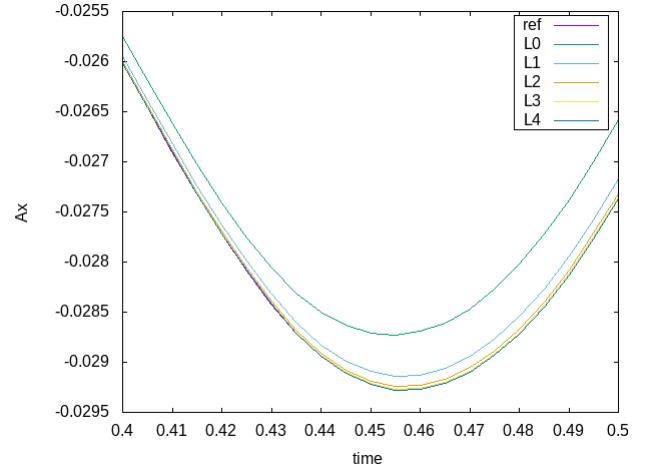


Fig. 5. Displacement in the direction x for CSM3 with various mesh coarseness, in detail.

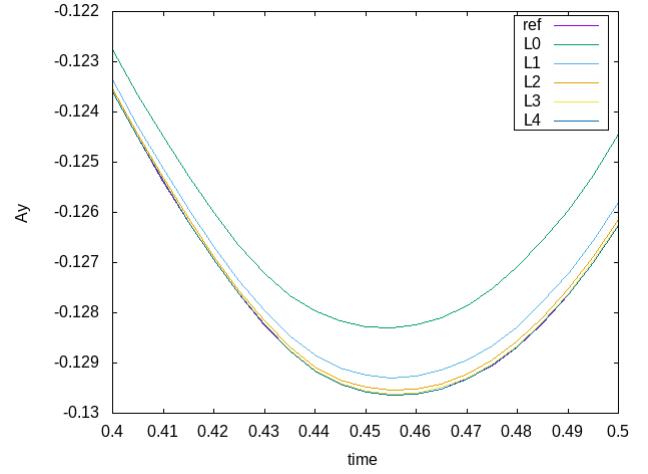


Fig. 6. Displacement in the direction y for CSM3 with various mesh coarseness, in detail.

To summarize, for our case we choose CN scheme as the one that is non-damping. This is the key property of the time stepping scheme for us.

C. Conclusion

In all cases, we were able to obtain values close to the reference values with a reasonable price. In the sense, that we don't need the time steps to be too small or the mesh resolution to be too high. This gives us a feeling that our solid solver works well and we can use it in our FSI validation benchmarks.

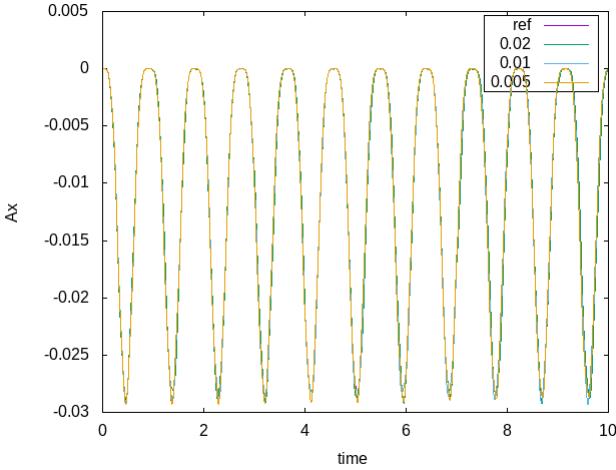


Fig. 7. Displacement in direction x for CSM3 with various time step sizes.

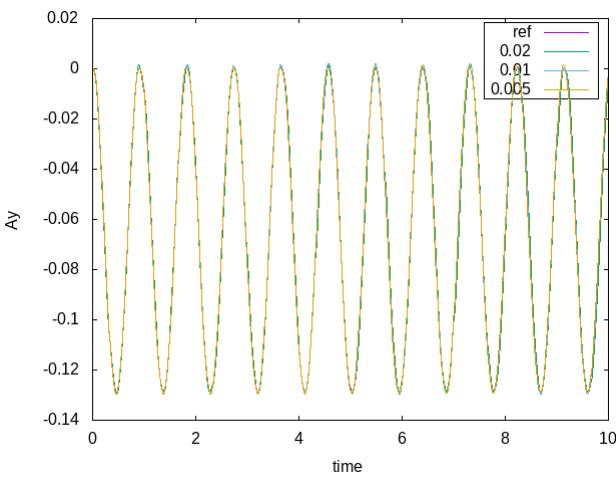


Fig. 8. Displacement in direction y for CSM3 with various time step sizes.

III. FSI-TESTS RESULTS

Now we are in the position where we have two solvers, one solver for fluid flow (OpenFOAM, validated in [7]) and one for the elasticity problem (FEniCS, validated in Section II). Both succeeded in the partial tests so this gives us a hope for working good together in the coupled Fluid-Structure-Interaction problem.

a) OpenFOAM implementation: There is one difficulty right in the beginning. OpenFOAM works only in 3D and our benchmark is 2D. The usual workaround is to set the width of the domain in the third z dimension to some small number (corresponding to the width of one element) and do some minor changes in the code. This is actually what we do in the OpenFOAM solver. But then a question arises, what to do for 2D problems when we need to couple with another (solid) solver? One possible solution is to move to 3D even in the solid part, this is the approach used in the Cylinder with a flap tutorial [4]. Since FEniCS works in 2D without any difficulties

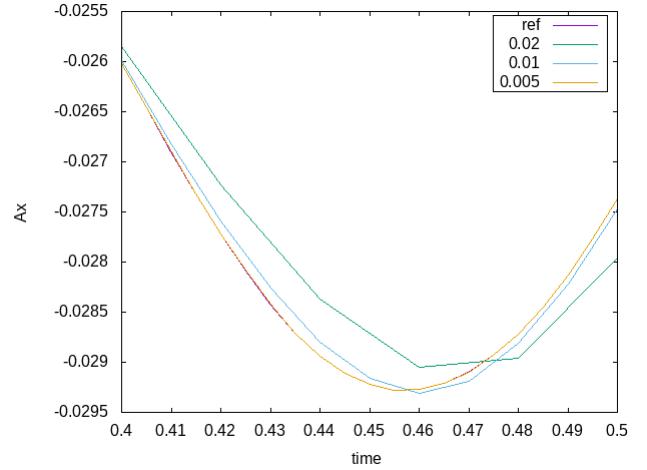


Fig. 9. Displacement in direction x for CSM3 with various time step sizes, in detail.

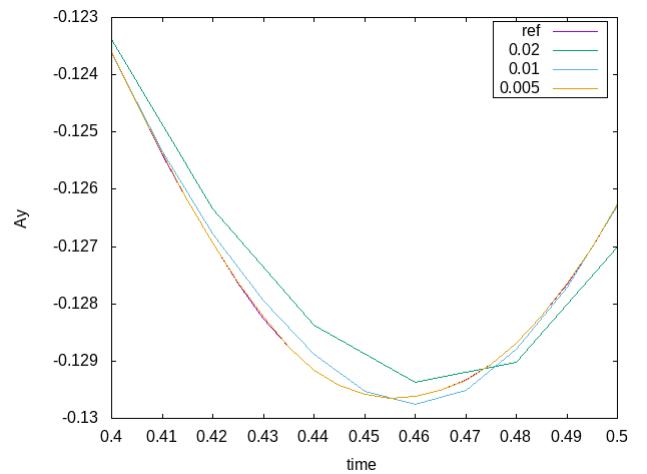


Fig. 10. Displacement in direction y for CSM3 with various time step sizes, in detail.

and moving to 3D would increase the number of degrees of freedom, and hence the computational cost, we decided for another approach.

The other way is to adjust the mapping between the solvers in such a way, that it handles the moving between 2D and 3D. In our case, the 3D (OpenFOAM, fluid) domain has width just one element in the z direction. So for mapping of the displacement from 2D (FEniCS, solid) domain to the 3D domain we need to map the values to the front side as well as to the back side of the fluid domain. This is because the displacement is supposed to move the 3D domain independently of the z direction. The picture we can imagine is extending the 2D domain to 3D with values constant along the z axis before the mapping.

The computation of the forces exerted by the fluid on the solid is in OpenFOAM performed on elements (not in vertices as the displacement). Therefore we have just one value along

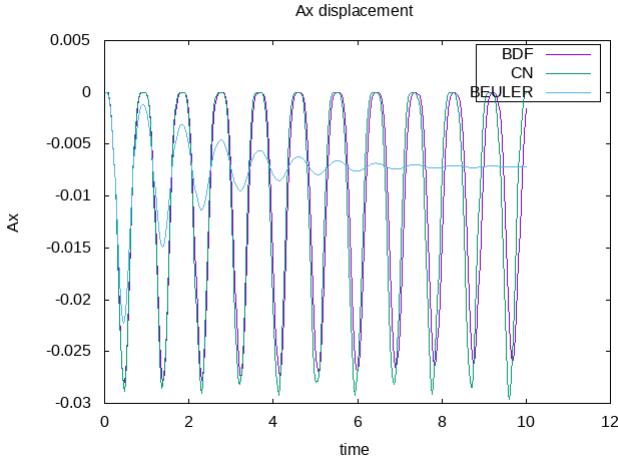


Fig. 11. Displacement in the direction x for CSM3 with various time discretization schemes.

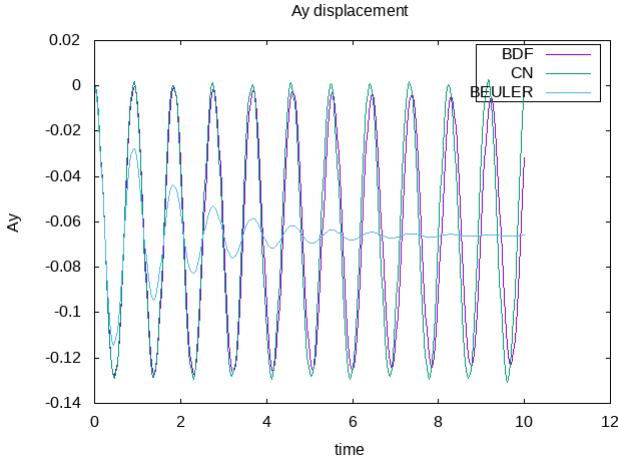


Fig. 12. Displacement in the direction y for CSM3 with various time discretization schemes.

the z axis. In FEniCS the forces are incorporated as a surface function, so the values in the nodes lying on the coupling boundary are needed. We see that the mapping of forces is actually done without changing the dimensionality. It is a mapping between two 1-dimensional boundary meshes. But there is another problem, the units of forces need to match. The forces in FEniCS are presented as specific surface forces, with the unit $[Nm^{-1}]$ (after integration over the surface we end up with $[N]$). But OpenFOAM returns forces with the unit $[N]$ (they are integrals over the surface element). This discrepancy will be discussed later.

We attempted to compute only the time dependent cases FSI2 and FSI3, starting with the FSI3 case that has smaller displacements but faster oscillations. This one is also modeled in the OpenFOAM-CalculiX tutorial [4], with a few differences. The inflow does not have a parabolic profile but is constant in

TABLE VI
DISCRETIZATION SCHEMES FOR TIME DERIVATIVE, WHERE Δx IS THE FIXED STEP SIZE.

| name | scheme | order |
|------|---|-------|
| BE | $\frac{y_{n+1} - y_n}{\Delta x} = f(x_{n+1}, y_{n+1})$ | 1 |
| CN | $\frac{y_{n+1} - y_n}{\Delta x} = \frac{1}{2}[f(x_{n+1}, y_{n+1}) + f(x_n, y_n)]$ | 2 |
| BDF | $\frac{1}{\Delta x} (\frac{3}{2}y_{n+2} - 2y_{n+1} + \frac{1}{2}y_n) = f(x_{n+2}, y_{n+2})$ | 2 |

the y direction. This means, we prescribe

$$\mathbf{v}^f(0, y) = \bar{U}$$

instead of (1). This is because the parabolic profile requires an additional package to load into OpenFOAM and the tutorial example should be easy to run, hence without extra packages. The results from OpenFOAM-CalculiX coupling show that this will preserve the behavior we expect, although we don't expect the results to really match the reference values. In case of working code, we would, of course, rewrite the inflow condition to the parabolic profile. But we will show that we were not able to obtain good results, so there is no need to strive for exactly the same setting. Switching to the parabolic profile would not help us with our computations.

The second difference to the original setting of Turek&Hron FSI benchmarks is the increase of the inflow velocity $\mathbf{v}^f(t, 0, y)$. In the original paper [1], a smooth growth with the use of cosine function (2) was suggested. In the tutorial, a simple linear growth is used. This shouldn't make any difference for the developed flow, as long as the fluid solver is able to solve the equations.

b) *Computed results:* Our computations however haven't been successful. The code used for the computations can be found here [15]. All the obtained results have very small displacement of the elastic solid, so the results were more like flow around a rigid body. We weren't able to correct the codes in such a way that it works fine. In the following we will try to point out some possible reasons.

The solid solver was checked for more problems, like prescribing some other surface force, i.e. pulling down the right end of the beam, and the results were in agreement with expectations.

So it seems like both solvers really solve what they are supposed to solve. This means that the problem is, most likely, somewhere in the mapping between the solvers or in the usage of preCICE. Moreover, FEniCS adapter is still in an experimental mode, which is another potential source of errors.

The obtained results are shown in the plots below. In Figure 13 are the results over the whole time interval for the OpenFOAM-CalculiX code and author's OpenFOAM-FEniCS codes. In Figure 14 are OpenFOAM-FEniCS attempts without the OpenFOAM-CalculiX results, to show the differences between the results of our codes in more detail.

I did several attempts to compute the results, with the nonlinear model (the one described in the section I-B) as

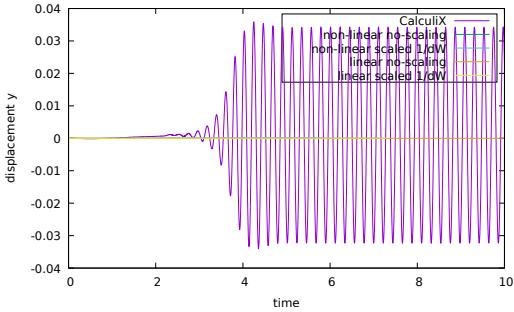


Fig. 13. Displacement of the elastic flap in the y direction for several solid elasticity codes.

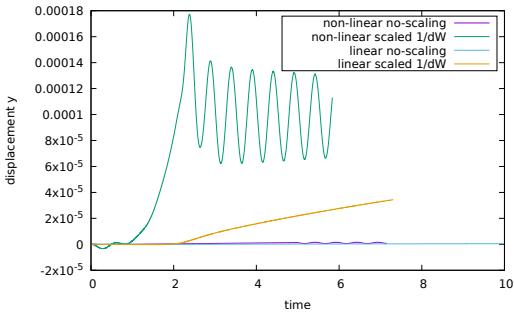


Fig. 14. Displacement of the elastic flap in the y direction for several solid elasticity codes.

well as with the simplified linear model (which is faster to solve). I also tried some scaling of the forces, because we need our force function to have unit [Nm^{-1}]. This is because of the reasons described in the Subsection *OpenFOAM implementation*. The scaling is done by dividing the width of the OpenFOAM domain in the dead z direction. Another scaling that makes sense is the length of the boundary elements in the direction perpendicular to axis z , the length of real 2D boundary. This hasn't been done so far.

The third possibility is the scaling that uses the length of the FEniCS element along the boundary. This is the one the author believes to be the right one. Unfortunately, he doesn't have results with this scaling, nor has he implemented it.

A detailed study of the mapping and transformation of the forces is needed but this is beyond the scope of this seminar paper (in the beginning we thought it wouldn't be necessary).

CONCLUSION

In this paper we discussed the Turek&Hron FSI benchmark [1] and described a partitioned approach to solve the FSI problems. Our partitioned approach used OpenFOAM [7] for the fluid, FEniCS [15] for the solid and preCICE [3] for the communication between the solvers.

We provided the validation for the partial CSM (Computational Solid Mechanics) problems in the Section II and for the validation of the CFD (Computational Fluid Dynamics) we refereed to Derek Risseeuw's Master Thesis [7].

We showed that our elasticity solver works good for the pure elasticity problems. But the results for FSI problems are bad, even though the fluid solver we are using is well established and is getting a good results for other FSI implementations.

The possible reasons could be the mapping, some issues inside FEniCS adapter, or simply its improper usage. FEniCS adapter is still being developed so, obtaining good results could be seen as a validation or approving of the correctness of FEniCS adapter.

What are we observing is small displacement that is probably caused by having small forces in the FEniCS code. Those forces are mapped by preCICE from OpenFOAM and author's personal opinion is that the problem is somewhere here.

Of course, it is also possible that there are some bugs or mistakes in the solid solver or in the usage of the preCICE library in the FEniCS code.

REFERENCES

- [1] S. Turek and J. Hron, "Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow," in *Fluid-Structure Interaction - Modelling, Simulation, Optimization*, ser. Lecture Notes in Computational Science and Engineering, H. Bungartz and M. Schäfer, Eds., no. 53. Springer, 2006, pp. 371–385, ISBN 3-540-34595-7.
- [2] S. Turek and M. Schäfer, "Benchmark computations of laminar flow around cylinder," in *Flow Simulation with High-Performance Computers II*, ser. Notes on Numerical Fluid Mechanics, E. Hirschel, Ed., vol. 52. Vieweg, Jan. 1996, pp. 547–566.
- [3] H.-J. Bungartz, F. Lindner, B. Gatzhammer, M. Mehl, K. Scheufele, A. Shukaev, and B. Uekermann, "preCICE – a fully parallel library for multi-physics surface coupling," *Computers and Fluids*, vol. 141, pp. 250–258, 2016, advances in Fluid-Structure Interaction. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0045793016300974>
- [4] preCICE - Cylinder with a flap tutorial. <https://github.com/precice/tutorials/tree/master/FSI/cylinderFlap/OpenFOAM-CalcuX>.
- [5] OpenFOAM. <https://openfoam.org/>.
- [6] Calculix. <http://www.calculix.de/>.
- [7] D. Risseeuw, "Fluid structure interaction modelling of flapping wings," thesis, Delft University of Technology, Jan. 2019. [Online]. Available: <https://repository.tudelft.nl/islandora/object/uuid:70bedde-e870-4c62-9a2f-8758b4e49123?collection=education>
- [8] OpenFOAM Adapter. <https://github.com/precice/openfoam-adapter>.
- [9] M. S. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. E. Rognes, and G. N. Wells, "The fenics project version 1.5," *Archive of Numerical Software*, vol. 3, no. 100, 2015.
- [10] M. E. Gurtin, E. Fried, and L. Anand, *The Mechanics and Thermodynamics of Continua*. Cambridge University Press, 2010.
- [11] S. Turek, J. Hron, M. Razzaq, and M. Schäfer, "Numerical benchmarking of fluid-structure interaction: A comparison of different discretization and solution approaches," in *Fluid-Structure Interaction II: Modelling, Simulation, Optimisation*, H. Bungartz, M. Mehl, and M. Schäfer, Eds. Springer, 2010, doi 10.1007/978-3-642-14206-2.
- [12] Featflow. <http://www.featflow.de/en/benchmarks/cfdbenchmarking/fsi.benchmark.html>.
- [13] Kubáč - CSM code. https://github.com/VojtechKubac/FSI_seminar/tree/master/CSM_benchmark.
- [14] C. Geuzaine and J.-F. Remacle, "Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities," *International Journal for Numerical Methods in Engineering*, vol. 79, pp. 1309 – 1331, 2009.
- [15] Kubáč - FSI code. https://github.com/VojtechKubac/FSI_seminar/tree/master/FSI.