

Partitioned Multi-Physics Simulation of Interfering Square Cylinders in a Flow

Bodhinanda Chandra¹, Hayden Liu Weng¹, Simon v. Wenczowski¹

Abstract

Fluid-structure interactions (FSI) are a crucial consideration in the design of various engineering systems, such as breakwaters, wind turbines, or bridges, since the accurate understanding and prediction of impact loads by fluids can provide an important information on structure safety assessment. Such simulation methods are also required to analyze the vibration of tall buildings or other interfering structures under the influence of wind flow. With this motivation, a 2D partitioned FSI simulation considering rigid square cylinders interfering in an air-flow is implemented using the *Kratos Multiphysics* framework. Here, the interfering structures are represented as simple multi-degree of freedom (MDOF) systems with spring and damping connections between the mass points, while the fluid domain is discretized via an unstructured finite element mesh. In order to validate our works, verification tests are done firstly in both the structure and fluid solvers. Following that, different configurations of the simulation cases are presented and analyzed. A good agreement between the numerical results demonstrates the success of the implemented approach.

Keywords: fluid structure interaction, strong staggered coupling, multi-degree of freedom structural dynamic system

1 Introduction

In many engineering disciplines, precise considerations of the reciprocal influence between flow and structural dynamics have become crucial for design decisions. The simulations in the emerging field of fluid structure interaction (FSI) apply coupling mechanisms to link numerical solvers of both disciplines.

The aim of this project is to simulate the behavior of square structures in a two-dimensional turbulent channel flow. The structures are modelled as rigid objects and are given different combinations of degrees-of-freedom (DOF) for their motion. Two individual structures are positioned in the middle of the channel and flow related interactions occur. In a further step, additional internal masses are attached to the structures via springs and dampeners in different configurations. This modification shows an impact on the flow-induced vibrations. Thus, the content of the project is structured in four sections: The second chapter introduces the theoretical background for all components of the fluid-structure-interaction (FSI) simulations. Fluid solvers, structural solvers, and corresponding coupling mechanisms are introduced and discussed separately. Section three is dedicated to the validation of the individual solvers before coupling structure and fluid parts. This includes a cross-check of the functionality of the structural solver and a plausibility study for the fluid simulation. Finally, section four presents the numerical results obtained for the different simulation cases investigated, including a parameter study for the relevant cases. The final chapter contains a summary of the project work and offers an outlook at possible future extensions.

¹ Graduate Student, Computational Mechanics Program, Department of Civil, Geo, and Environmental Engineering, Technische Universität München

2 Underlying Theory

This first chapter provides a summary of the theoretical foundations relevant to the project. Fluid equations and structural equations form a basis on which the numerical solvers are formulated, and are subsequently applied on the respective domains. To combine both components in the FSI simulation cases, coupling techniques are introduced.

2.1 Fluid Simulation

The *Kratos Multiphysics* [5] framework offers a finite element based fluid solver, that works with the non-conservative form of the underlying equations. Following the more detailed explanations in [4], a brief overview is given in this chapter.

2.1.1 Navier-Stokes Equation

For small Mach numbers $Ma < 0.3$, air can be treated as an incompressible fluid [7]. The Navier-Stokes Equations (NSE) for incompressible fluids completely describe the flow field by means of the conservation of momentum and mass. In combination with initial and boundary conditions, the NSE can be solved for the velocity and pressure field:

$$\begin{aligned} \rho \frac{\partial \mathbf{v}}{\partial t} + \rho (\nabla \cdot \mathbf{v}) \mathbf{v} + \nabla p - \mu \Delta \mathbf{v} &= \mathbf{f} \\ \nabla \cdot \mathbf{v} &= 0 \\ \mathbf{v}(t = 0) &= \mathbf{v}_0 \\ \mathbf{v} &= \mathbf{v}_D \text{ (for } \Gamma_D \text{)} \\ \boldsymbol{\sigma} \cdot \mathbf{n} &= \mathbf{f} \text{ (for } \Gamma_N \text{)} \end{aligned} \tag{1}$$

The source of many difficulties lies in the non-linear and non-symmetric convective term, which gains influence with raising Reynolds numbers. Additionally, the pressure must be adjusted to ensure that the flow field satisfies the incompressibility constraint, hence acting as a Lagrange multiplier.

For the fluid solver, the weak formulation of the NSE in the standard Galerkin finite element formulation is used. This requires a local approximation for the velocity ($\mathbf{v} \approx \mathbf{v}^h$) and for the pressure ($p \approx p^h$) by means of shape functions. Besides this (finite dimensional) solution function space, a second space for the test functions has to be defined. Both contain functions of the Sobolev space $\mathcal{H}^1(\Omega)$ with Ω representing the computational domain. The following affiliations can be noted:

- $\mathbf{v} \in \mathcal{H}_D^1 = \{v \in \mathcal{H}^1(\Omega) \mid v \in \mathcal{P}_m \text{ and } v = v_D \text{ on } \Gamma_{Dirichlet}\}$ states, that the velocity solution has to fulfill the Dirichlet boundary conditions.
- $\mathbf{w} \in \mathcal{H}_0^1 = \{w \in \mathcal{H}^1(\Omega) \mid w \in \mathcal{P}_m \text{ and } w = 0 \text{ on } \Gamma_{Dirichlet}\}$ sets the makes the velocity test functions vanish on a Dirichlet boundary.
- $p, q \in \mathcal{L}^2(\Omega)$ states that both the pressure and its test function must be square-integrable functions.

Under these definitions, the Galerkin weak form of the NSE can now be stated using the abbreviating notations $(a, b) = \int_{\Omega} a \cdot b \, d\Omega$:

$$\begin{aligned} \rho \left(\frac{\partial \mathbf{v}}{\partial t}, \mathbf{w} \right) + \rho (\mathbf{v} \cdot \nabla) \mathbf{v} - \mu (\nabla \mathbf{v}, \nabla \mathbf{w}) + (p, \nabla \mathbf{w}) &= \rho (\mathbf{f}, \mathbf{w}) + (\mathbf{w}, \mathbf{t})_{\Gamma_N} \\ (q, \nabla \cdot \mathbf{v}) &= 0 \end{aligned} \quad (2)$$

After the introduction of the spatial discretization, the resulting spatial finite element discretization of the weak form can be written in a matrix form. The convection matrix \mathbf{C} and the load vector \mathbf{F} depend on the unknown \mathbf{v} , which introduces non-linearity:

$$\begin{aligned} \mathbf{M} \frac{\partial \mathbf{v}(t)}{\partial t} + [\mathbf{C}(\mathbf{v}(t)) + \mu \mathbf{L}] \mathbf{v}(t) + \mathbf{G} p(t) &= \mathbf{F}(\mathbf{v}(t)) \\ \mathbf{G}^T \mathbf{v}(t) &= 0 \end{aligned} \quad (3)$$

The involved matrices are calculated as follows:

Mass matrix

$$\mathbf{M} = \rho \int_{\Omega} \mathbf{N} \mathbf{N}^T d\Omega \quad (4)$$

Laplacian matrix

$$\mathbf{L} = \int_{\Omega} \nabla \mathbf{N} \nabla \mathbf{N}^T d\Omega \quad (5)$$

Gradient matrix

$$\mathbf{G} = - \int_{\Omega} \nabla \mathbf{N} \mathbf{N} d\Omega \quad (6)$$

Convection matrix

$$\mathbf{C}(\mathbf{v}) = \rho \int_{\Omega} \mathbf{N} (\mathbf{v} \nabla \mathbf{N}) d\Omega \quad (7)$$

Load vector

$$\mathbf{F}(\mathbf{v}) = \int_{\Omega} \mathbf{N} \mathbf{f} d\Omega + \int_{\Gamma_N} \mathbf{n} \cdot (p \mathbf{I} - \mu \nabla \mathbf{v}) d\Gamma \quad (8)$$

In addition, a time integration scheme is necessary to complete the solution of the problem. Ideally it should be chosen such that it is both stable and accurate. Without taking special care when choosing an element type, an immediate numerical solution of the presented Galerkin finite element formulation can fail because spurious oscillations occur. As a consequence, stabilization techniques must be applied.

2.1.2 Variational Multi Scale (VMS)

The underlying idea of this stabilization technique is to split up the exact solution into two parts: the part which can be resolved by the spatial discretization and the part which cannot. The same is applied to the test functions for pressure and velocity:

- velocity $\mathbf{u} = \mathbf{u}_h + \tilde{\mathbf{u}}$ with test function $\mathbf{v} = \mathbf{v}_h + \tilde{\mathbf{v}}$
- pressure $p = p_h + \tilde{p}$ with test function $q = q_h + \tilde{q}$

The large scales are noted with $(\cdot)_h$ and can be resolved by a mesh with size h . In contrast, the small scales for which the spatial discretization is too coarse are collected in $(\tilde{\cdot})$. Two separate sets of equations can be formulated for large and small scales. Since the small scales also appear in the large scale equation, they need to be approximated to determine the (mainly dissipative) effect of the unresolved scales on the large scales. Different ways are possible to define a small scale solution space. A projection of the original equations allows to derive a model for the effect of the small scales. Introducing the result into the

large scale equation and performing an integration by parts allows one to see the small scale approximation as test functions in the weak formulation for the large scales. A detailed description can be found in [3].

A comparison to Large Eddy Simulations (LES) is possible and shows many parallels:

- Separation of length scales: The spatial filtering in the LES approach is replaced by a scale separation due to the mesh resolution in the VMS.
- Modelling the effect of small scales: Whereas LES introduces a subgrid stress tensor, the VMS adds a stabilization term.
- Foundation: In contrast to the LES, VMS is based on numerical consideration instead of physical considerations.

This last point leaves the open question, whether VMS implicitly replaces a physically based turbulence model.

2.1.3 ALE Formulation

In the context of FSI simulations, both the Eulerian and the Lagrangian kinematic description of the flow field show certain shortcomings. Whereas the Eulerian view with fixed meshes is popular for pure fluid applications, problems occur at material interfaces and mobile boundaries. On the other hand, the Lagrangian approach implies a grid following the motion of the continuum but shows deficits when considerable distortions occur in the fluid. Therefore, in more general simulations a different approach is required.

The Arbitrary-Lagrangian-Eulerian approach (ALE) combines advantages of both classical views. While the quality and the topology of the mesh are maintained, a motion of the nodes independent of the velocity of material particles is possible. Thus, the cells function as moving referential domains with a convective flow through their surfaces due to a relative velocity \mathbf{v}_R between the fluid and the mesh. Accordingly, the following parameters must be considered:

- material velocity $\mathbf{v} = \frac{\partial \mathbf{x}}{\partial t}|_{\mathbf{X}}$ of a fluid particle identified by its original position \mathbf{X} .
- mesh velocity $\mathbf{v}_M = \frac{\partial \mathbf{x}}{\partial t}|_{\xi}$ at a spatial position ξ with respect to the mesh.
- relative velocity $\mathbf{v}_R = \mathbf{v} - \mathbf{v}_M$ that appears in the convective part.

Mapping operators establish a formal connection between the different descriptions via their jacobians and allow us to derive the fundamental ALE equation for an abstract flow quantity ψ :

$$\frac{\partial \psi}{\partial t}|_{\mathbf{X}} = \frac{\partial \psi}{\partial t}|_{\xi} + \frac{\partial \psi}{\partial \mathbf{x}}|_{\xi} \cdot (\mathbf{v} - \mathbf{v}_M) = \frac{\partial \psi}{\partial t}|_{\xi} + \nabla \psi \cdot \mathbf{v}_R \quad (9)$$

Taking this equation as a common basis, the conservation equations for mass and momentum can be derived in their differential forms:

$$\frac{\partial \rho}{\partial t}|_{\xi} + \nabla \rho \cdot \mathbf{v}_R = 0 \quad (10)$$

$$\frac{\partial \mathbf{v}}{\partial t}|_{\xi} + \nabla \mathbf{v} \cdot \mathbf{v}_R + \nabla \cdot \boldsymbol{\sigma} = \mathbf{f} \quad (11)$$

For each time step, first the mesh is deformed and \mathbf{v}_M can be computed. Using the relative velocities as shown in (10) and (11), the standard Eulerian form of the NSE can then be maintained.

2.2 Structural Coupling

The structures, as the other component of the simulations, are treated in the simplest form possible to avoid unnecessary complications. That is, as mentioned previously, they are treated as SDOF rigid objects or a set of SDOFs and their corresponding interactions. A solver applying the Generalized- α integration scheme for the SDOF configuration was provided as a starting point, so this section will provide a brief background on the modelling of the system, the integration method, and finally the solution schemes for interacting or multiple degree-of-freedom (MDOF) systems.

2.2.1 Lagrangian description of a Single Degree of Freedom

In general, a dynamic system can be analysed via the balance of internal and external forces acting on it. When reduced to an SDOF system, all of the relevant effects can be attributed to one of three main components: a conservative stiffness value k , a non-conservative dissipation (or dampening) coefficient c , and the inertial response due to the mass m . To further simplify the formulation, only linear effects are considered, such that the previously defined parameters are constant in time. In particular, this assumption indicates that only Rayleigh dampening is considered. Therefore, after defining a displacement variable u , the ensuing equation of motion is simply:

$$m \cdot \ddot{u}(t) + c \cdot \dot{u}(t) + k \cdot u(t) = F(t) \quad (12)$$

where $F(t)$ represents the external force acting on the structure. This equation can be expressed independently for both x and y coordinates when analysing a more general configuration, since no rotation is considered.

As with the fluid part, the equation is then solved numerically by an appropriate scheme which is both stable and accurate. Without careful attention to this point, an initial solution could be distorted or eclipsed by spurious oscillations or divergence from the actual solution due to unresolved higher frequency effects or, even worse, due to resonance effects.

2.2.2 The Generalized- α integration method

As noted previously, one of the key elements that a numerical scheme should provide in solving linear (and non-linear) dynamic problems, comes in the form of numerical dissipation for higher frequencies due to the choice of space and/or time discretization. Some popular schemes which satisfy this are the implicit one step methods such as the Newmark- β and the Hilber-Hughes-Taylor (HHT- α), which offer second order accuracy for linear dynamic problems and are suitable for varying time step sizes. The Generalized- α method takes from these methods and refers to a more general class of methods, thus extending the applicability of these strategies to a broader range of problems. Specifically, by means of varying ρ_∞ , the spectral radius at infinity, the amount of numerical dissipation of the method can be tuned.

In this scheme, the integration is performed with an auxiliary point defined via two parameters α_M and α_F (hence the name of the method) which depend on ρ_∞ , such that the balance equation becomes:

$$m \cdot \ddot{u}_{t+\alpha_M \Delta t} + c \cdot \dot{u}_{t+\alpha_F \Delta t} + k \cdot u_{t+\alpha_F \Delta t} = F_{t+\alpha_F \Delta t} \quad (13)$$

with the definitions:

$$\begin{aligned} u_{t+\alpha_F \Delta t} &= (1 - \alpha_F)u_t + \alpha_F u_{t+\Delta t} \\ \dot{u}_{t+\alpha_F \Delta t} &= (1 - \alpha_F)\dot{u}_t + \alpha_F \dot{u}_{t+\Delta t} \\ \ddot{u}_{t+\alpha_M \Delta t} &= (1 - \alpha_M)\ddot{u}_t + \alpha_M \ddot{u}_{t+\Delta t} \end{aligned} \quad (14)$$

The values at the end of the step, in turn, are calculated by the Newmark approximations:

$$u_{t+\Delta t} = u_t + \Delta t \dot{u}_t + \Delta t^2 \left(\left(\frac{1}{2} - \beta \right) \ddot{u}_t + \beta \ddot{u}_{t+\Delta t} \right) \quad (15)$$

$$\dot{u}_{t+\Delta t} = \dot{u}_t + \Delta t ((1 - \gamma) \ddot{u}_t + \gamma \ddot{u}_{t+\Delta t}) \quad (16)$$

The system is thus solved implicitly for a given initial displacement and velocity, by combining (13), (14), (15), and (16) with the initial acceleration approximated as:

$$\ddot{u}_0 = m^{-1}(F_0 - c \cdot \dot{u}_0 - k \cdot u_0) \quad (17)$$

Further information on this method and its properties can be found in [6] and in [2].

2.2.3 Multiple Degree of Freedom Systems

If more DOF are involved, then it is necessary to describe how these interact among each other. While the treatment itself can be done in a variety of ways, only two specific cases are mentioned here, since further discussion on the matter will be present in the following when referring to the FSI coupling. These two alternatives are a monolithic solution scheme, and a Neumann-Neumann (i. e. force-force) coupling. For simplicity, this section will only cover the two degree-of-freedom (2DOF) case, but the same concept can be extended to more. The system is assumed to be such that only the first DOF is excited externally.

Monolithic solution When the system is solved monolithically, all DOFs are considered simultaneously, and the interactions between them arise as non diagonal entries in the resulting linear system. For instance, for the basic 2DOF system described, the dynamic equation reads:

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{Bmatrix} \ddot{u}_1 \\ \ddot{u}_2 \end{Bmatrix} + \begin{bmatrix} c_1 + c_2 & -c_2 \\ -c_2 & c_2 \end{bmatrix} \begin{Bmatrix} \dot{u}_1 \\ \dot{u}_2 \end{Bmatrix} + \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \begin{Bmatrix} F_f \\ 0 \end{Bmatrix} \quad (18)$$

Coupled solution On the other hand, the system can also be solved in a staggered fashion, meaning only some of the equations are solved at each point in time, and the resulting solutions are fitted against each other until a consistency or convergence criteria is reached. For the 2DOF described, one possible split of the system is:

$$\begin{cases} m_1 \cdot \ddot{u}_1 + c_1 \cdot \dot{u}_1 + k_1 \cdot u_1 = F_f - F_{21} \\ m_2 \cdot \ddot{u}_2 + c_2 \cdot \dot{u}_2 + k_2 \cdot u_2 = F_{21} \end{cases} \quad (19)$$

where $F_{21} = c_2 \cdot (u_1 - u_2) + k_2 \cdot (u_1 - u_2)$. Note that due to the definition of the coupling force, the spring and damper corresponding to the second degree of freedom are already considered and hence don't appear on the left hand side of (19). In this formulation, the equations are solved one after the other consecutively (e. g. via the Generalized- α method) until the resulting coupling force converges.

2.3 Fluid-Structure Interaction

Now that we have described the structural and the fluid solvers, the coupling scheme and algorithm between them will be discussed further in the following subsections.

2.3.1 Basic Ideas and Types of FSI Simulations

The basic idea of a FSI scheme is to connect the mechanical behavior between two completely different systems, the structure and the fluid, which have different mathematical models and physical behavior. In addition to that, the FSI scheme also acts as an interface between them, which generally provides a way to connect the two distinct solvers or rules the flow of information from one system to another. Generally, two different approaches can be used in FSI to obtain the solution of coupled problems. The *monolithic* approach solves the governing equations for both the flow and the structure simultaneously, within a single solver (just as was the case when coupling structural DOFs). This approach involves a larger and more complex tangent matrix which is typically non-symmetric and normally requires suitable preconditioners to invert. As a result, the monolithic FSI problem will converge as fine as the approximation of the tangent matrix allows; resulting in a very robust solver if it can be done effectively. Meanwhile, another possibility is to keep the fluid and the structure solvers separated as two distinct blocks; this approach is known as a *staggered* or *partitioned* approach, and is widely used in FSI since it can preserve each software's modularity by coupling existing, different, and possibly more efficient flow and structural solvers that have been developed specifically for each governing equation. However, the development of stable and accurate coupling algorithms is generally required in the partitioned simulation as the coupling condition used will strongly influence the stability of the results.

Furthermore, the partitioned methods are divided into *one-way* and *two-way* coupling. The one-way coupling methods provide a possibility to reduce the computational effort of FSI by only passing the fluid pressure force acting at the structure to the structure solver. On the other hand, the more intuitive two-way coupling methods transfer the displacement of the structure back to the fluid solver as an additional step in comparison to the one-way coupling scheme. Two-way coupling schemes are further divided into *weakly* and *strongly* coupled methods where the convergence at the boundary between structure and fluid is, respectively, neglected or considered. The differences and comparisons of the mentioned coupling methods in several FSI examples can be found in more details at the following sources: [11], [10], and [1].

2.3.2 FSI Coupling with Relaxation Scheme

In the current project, the staggered two-way strongly coupled FSI scheme is chosen considering a relaxation scheme. The FSI interface is implemented to firstly transfer the fluid pressure force acting at the structure to the structure solver. The force transferred will further cause deformation on the structure and this deformation is then sent back to the fluid solver as a response of the given force. This process of transferring and receiving information, namely force and displacement, is then continued in an iterative loop until a certain convergence criteria is satisfied. In particular, the convergence criteria must preserve the continuity of velocity and position (or the Dirichlet conditions) as well as the equilibrium of forces (the Neumann condition) at the interface between the structure and the fluid domain. Ideally, one can write,

$$\begin{aligned} \boldsymbol{x}_{s\Gamma} &= \boldsymbol{x}_{f\Gamma}, \\ \boldsymbol{v}_{s\Gamma} &= \boldsymbol{v}_{f\Gamma}, \\ \boldsymbol{t}_{s\Gamma} &= \boldsymbol{t}_{f\Gamma}. \end{aligned} \tag{20}$$

where, \mathbf{x}_Γ , \mathbf{v}_Γ , and \mathbf{t}_Γ are the position, velocity, and the traction vectors at the boundary, respectively, and the subscripts s and f indicate the partitioned structure and fluid parts. Therefore, the coupling loop between the structure and the fluid parts will always continue until convergence, up to certain given tolerance, of displacements \mathbf{u} and forces \mathbf{F} is reached, or until the maximum number of iteration is reached.

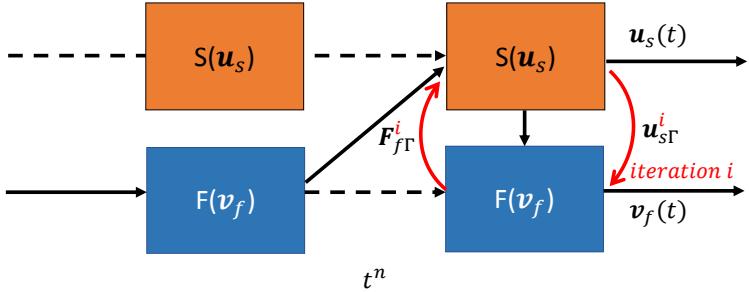


Figure 1: Solution algorithm for two-way coupling at every time step

In order to improve the convergence of the FSI iteration, relaxation of the displacement passed from the structure to the fluid is commonly applied. The relaxation of the FSI cycle at each sub-iteration i at a certain time step is defined as:

$$\mathbf{u}_{f\Gamma}^{i+1} = \hat{\mathbf{u}}_{s\Gamma}^{i+1} = \omega^i \mathbf{u}_{s\Gamma}^{i+1} + (1 - \omega^i) \mathbf{u}_{s\Gamma}^i, \quad (21)$$

where \mathbf{u}_{Γ}^i is the displacement at iteration i and ω^i is the relaxation parameter. Choosing a fixed value of ω^i for all iterations leads to fixed point methods. In this family of methods, the relaxation ω has to be set empirically at the beginning of the convergence iteration. Notice that if $\omega = 1$, the system is not relaxed, since it passes directly $\mathbf{u}_{f\Gamma}^{i+1} = \mathbf{u}_{s\Gamma}^{i+1}$. For under-relaxation cases, ω is chosen in the range of $0 < \omega < 1$. This leads to more robust iterations but is typically less efficient. In the current project, the relaxation parameter is set fixed as $\omega = 1$ and it works well for our specific numerical examples. It should be noted, however, that there is no optimal value of the relaxation factor that can be determined as it is generally problem specific. Moreover, a dynamically changing relaxation factor can also be implemented using, for instance, the Aitken method [8], which takes into account a residual function, to further improve the convergence of the scheme and reduce the number of iterations required.

2.3.3 Fluid-Structure-Structure Coupling

As we introduce a structure-structure coupling on top of the FSI coupling, the sequence call implementation become slightly more complicated. In this scheme, at every FSI iteration, a structure-structure interaction scheme is performed. As what have been explained earlier, the staggered Neumann-Neumann coupling scheme is chosen for the interaction between one structure and another. There, a similar type of strongly coupled iteration is performed and will be stopped only until the convergence of transferred forces \mathbf{F} is reached, i.e. $|F_{s12} - F_{s21}| \leq \varepsilon_s$, where ε_s is the convergence tolerance of the structure coupling. The solution algorithm of the fluid-structure-structure coupling is illustrated by Figure 2 below. Here, only structure $S1$ is interacting with the fluid, and thus, the hydrodynamic force is only applied on the boundary Γ between $s1$ and f . The force exerted by the fluid on $S1$ will then move the system of multi-degree of freedom structures consisting of both $S1$ and the inner structure $S2$. Notice that the FSI iteration here will only proceed if the structure-structure coupling converges.

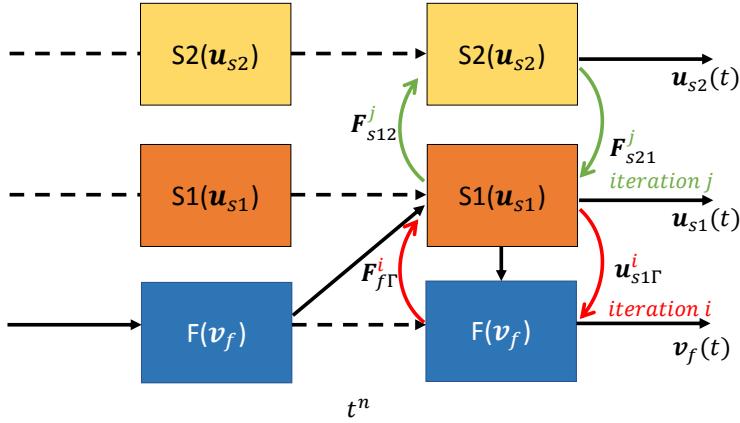


Figure 2: Solution algorithm for FSI coupling including a "Jacobi" type iteration

As for cases where more than one (active) structure is contacting the fluid, a "Jacobi" type iteration is implemented within our FSI scheme. That is, at every FSI iteration, the hydrodynamic force from the fluid will be sent to all fluid-structure interfaces Γ_k independently, where the subscript k indicates the number of the structure interacting with the fluid domain. Then, the MDOF structural systems will do their own independent Neumann-Neumann iteration process until convergence is reached. Finally, the response displacement and velocity of the interface will be sent back to the fluid meshes and the FSI iteration will proceed.

3 Verification of solvers

This section attempts at quantifying the quality of the numerical solutions achieved with the numerical methods and FSI formulations introduced explained in the previous section. First, the structural coupling scheme is tested and compared with the monolithic solver. Then, the plausibility of the fluid solver implemented in Kratos is checked by looking at a specific dimensionless number describing oscillating flow mechanisms.

3.1 Verification of the structural solver

The staggered and the monolithic approach are compared to verify the functionality of the structural MDOF solver. External forces from the fluid are not considered and the system is excited by an initial displacement. The MDOF structural model and the mechanical parameters are given as,

- $c_1 = c_2 = 1.0 \text{ Nm/s}$
- $k_1 = k_2 = 100.0 \text{ N/m}$
- $m_1 = 1.0; m_2 = 0.1 \text{ kg}$
- $u_1(t = 0) = -u_2(t = 0) = 1.0 \text{ m}$

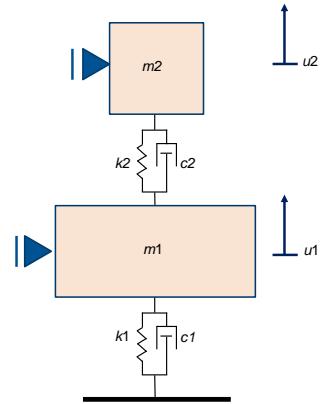


Figure 3: Structural solver verification model

For a time step of $dt = 0,0001 \text{ s}$, the results for the displacements $u_1(t)$ and $u_2(t)$ show a very good agreement and are visualized in the top part of Figure 4. The error plot of the two structural displacement is, furthermore, shown in the bottom part of the same figure. Here, notice that the absolute-normalized error is considerably small ($< 1\%$) over the entire simulation time. A noticeable difference is the slightly lower amplitudes in the monolithic approach, which can be also seen as the oscillating pattern in the error plot. This difference in the amplitude appears due to the induced numerical damping resulting from the implicit algorithm used in the monolithic approach. The choice of a smaller time step dt helps to avoid this dampening effect of the implicit time integration scheme, e.g. a 10 times smaller time step will result in a proportionally smaller error (i. e. 10% of the original). Seeing these results and assessments we can deduce that our Neumann-Neumann coupling scheme works well for MDoF structural systems.

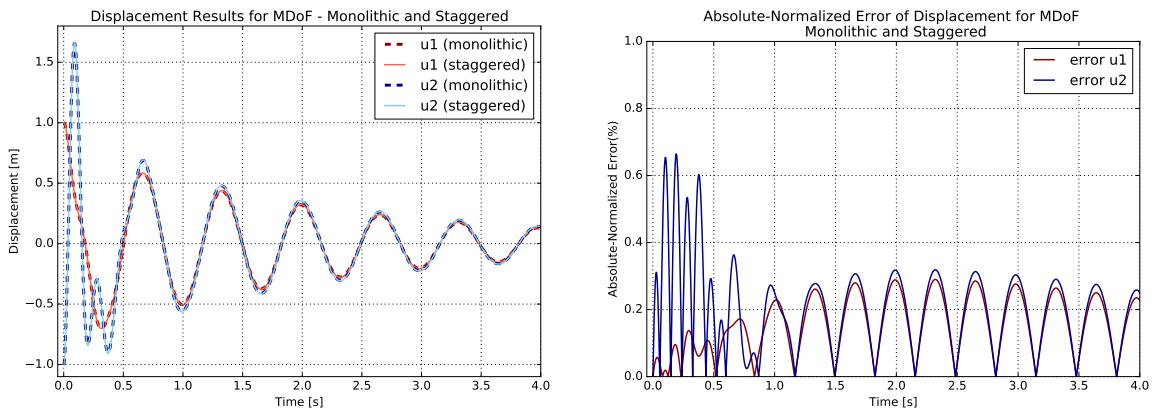


Figure 4: Displacements (left) and absolute-normalized error (right) of both DOFs with monolithic and staggered solution

3.2 Plausibility of the fluid simulation

Although a detailed validation of the fluid simulation would exceed the scope of this project, the plausibility of the solution is investigated. The Kármán vortex street develops behind the square cylinder and can be considered as the key feature of the flow. Figure 5 shows the detaching eddies by means of the x-velocity distribution.

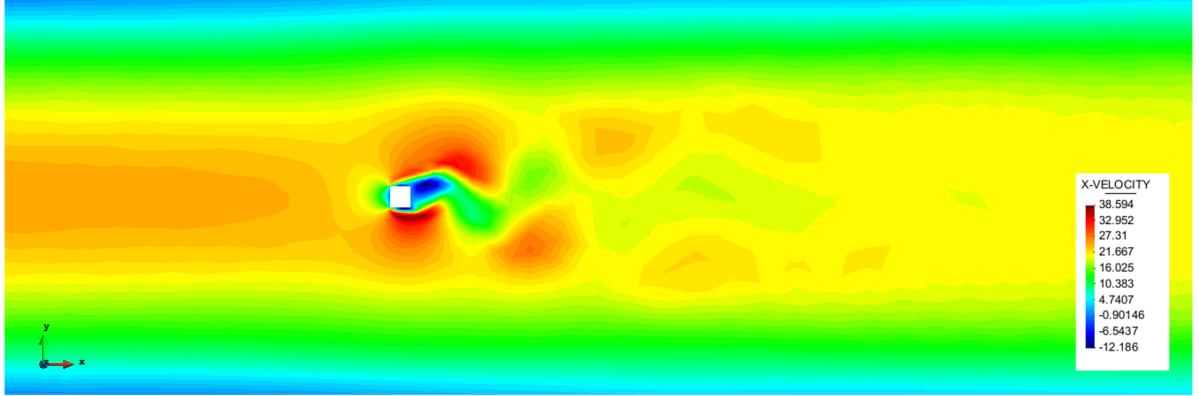


Figure 5: Fluid simulation with vortex street behind a single cylinder

After approximately 100 s simulation time, the periodic vortex shedding behind the cylinders leads to approximately harmonically oscillating forces on the bluff bodies. A plot of the forces in y-direction is given in Figure 6.

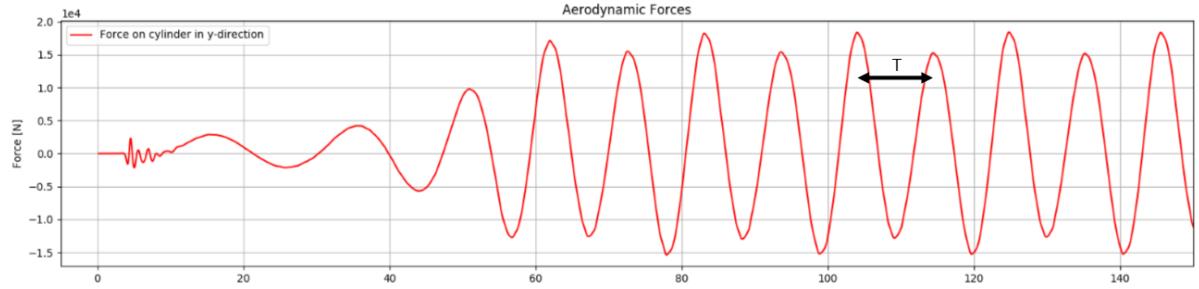


Figure 6: Force on square cylinder in the transversal y-direction

Knowing the period T of the vortex detachment allows us to calculate the dimensionless Strouhal number as a descriptive parameter. Here, the characteristic length scale is given by the side length of the cylinder.

$$St = \frac{f \cdot L}{U} = \frac{L}{U \cdot T} \approx \frac{30 \text{ m}}{25 \frac{\text{m}}{\text{s}} \cdot 10 \text{ s}} = 0.12 \quad (22)$$

Since the value lies in the range of expectation (around 0.1~0.2), the fluid simulation results can be considered plausible.

4 Numerical results

This chapter describes the setup of the presented simulation cases including the specification of numerical parameters. Since no MDOF solver was available in the *Kratos Multiphysics* framework, its implementation is briefly described.

4.1 Geometry

Preprocessing the two-dimensional case requires the definition of the geometries and the assignment of corresponding boundary conditions. The length of the channel is 1800 m and the first cylinder is positioned at one third of this length. Apart from one simulation, all other cases include a second square-cylinder that

is shifted by 150 m in stream-wise direction. The width of the channel is 600 m and the square-cylinders have a side length of 30 m. The entire configuration is visualized in Figure 7.

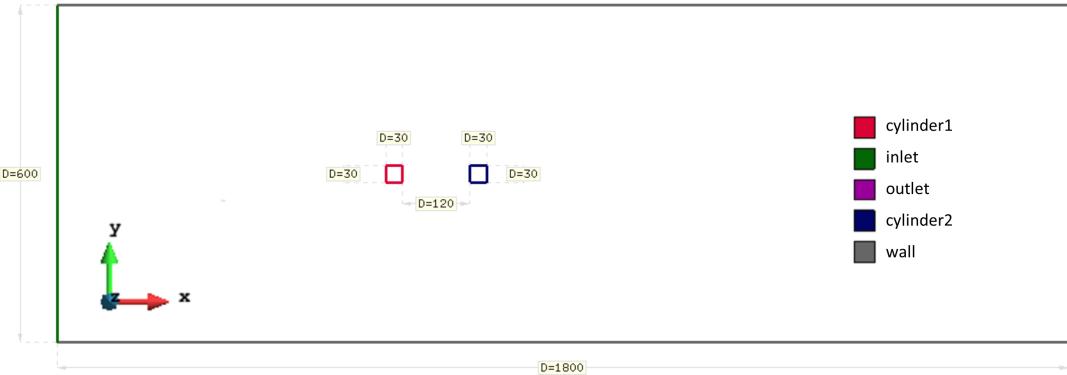


Figure 7: Geometry of the channel with two square-cylinders and boundary definitions

Based in the described geometry, a mesh has to be generated and locally refined in the critical regions around the cylinders. The triangles around the cylinders have a side-length of 3 m, the minimum throughout the entire mesh, whereas further cells are up to 20 times as large. In total, the mesh consists out of roughly 3000 triangle elements. A average computation time of 10 min results from this configuration.

4.2 Boundary and initial conditions

The fluid enters the domain through the *inlet* boundary on the left. A slightly oscillating parabolic velocity profile was chosen to enhance the vortex shedding from the bluff bodies. In the middle of the channel the flow reaches its maximum velocity of $25.0 \pm 0.5 \text{ m/s}$. The oscillation has a frequency of 0.5 Hz and accelerates the vortex shedding. The side walls of the channel are defined as slip boundaries. In contrast, the square-cylinders are assigned a no-slip boundary condition and wall shear stresses can appear at their surface. Finally, the *outlet* on the right has a zero pressure boundary condition.

The initial velocity field is constant 0 m/s in the entire fluid domain. The inlet velocity profile is given a ramp-up time of 3 s. Also the structures are at rest at the beginning of the simulation.

4.3 Mechanical properties

The fluid is given approximately the physical properties of air, which are listed in table 1. Due to the low Mach number, the air can be treated as an incompressible Newtonian fluid.

density	$\rho = 1.2 \text{ kg/m}^3$
viscosity	$\nu = 15.11 \cdot 10^{-6} \text{ m}^2/\text{s}$

Table 1: Definition of properties for the fluid

The structures are composed out of discrete elements, i. e. masses, dampers, and springs, as illustrated in Figure 8. Only the external DOFs, the "outer" boxes, are exposed to the fluid-dynamic forces. In cases with two degrees-of-freedom (MDOF cases), a second internal mass is attached to the first mass via a spring and a damper. This "inner" box is not affected by the fluid and also not restricted in its movement by collisions with the "outer" box.

DOF 1:

$$\begin{aligned} m_1 &= 100\,000 \text{ kg} \\ c_1 &= 3162.3 \text{ Ns/m}^2 \\ k_1 &= 10\,000 \text{ N/m} \end{aligned}$$

This DOF is also affected by forces resulting from the fluid.

DOF 2 :

$$\begin{aligned} m_2 &= 100\,000 \text{ kg} \\ c_2 &= 3500.0 \text{ Ns/m}^2 \\ k_2 &= 10\,000 \text{ N/m} \end{aligned}$$

This DOF is decoupled from the fluid. It appears in the MDOF cases.

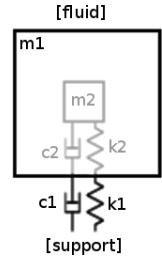


Figure 8: Definition of properties for the structural elements

The specified parameters are kept constant and used for the cases discussed in the following. A separate chapter is dedicated to a parameter study.

4.4 Case configuration

In the scope of this project, six simulation cases are investigated. The sequence is chosen such, that an increasing number of degrees of freedom and structural solvers is used. Finally, the last case includes eight structural solvers, i. e. four MDOF solvers, besides the fluid solver. An overview is given in table 2 and, more detailed, in Appendix A.

<i>Case 1:</i>			<i>Case 2:</i>		
1 SDOF			2 SDOF		
0 MDOF			0 MDOF		
<i>Case 3:</i>			<i>Case 4:</i>		
2 SDOF			0 SDOF		
0 MDOF			1 MDOF		
<i>Case 5:</i>			<i>Case 6:</i>		
0 SDOF			0 SDOF		
2 MDOF			4 MDOF		

Table 2: Definition of simulation cases with different numbers of structural solvers.

4.5 Solving routines

To simulate the described cases, many functionalities provided by *Kratos Multiphysics* can be used. However, for the treatment of MDOF systems additional solving routines are necessary. Their implementation is briefly discussed with focus on the applied coupling strategies.

4.5.1 Algorithms implemented in Kratos

The fluid solver is offered by the *Kratos* framework and uses a stabilized finite element strategy. Its properties can be specified in a *.json-file. Here, a Navier-Stokes solver, that uses the *fractional step method* for time discretization, is chosen. The underlying idea of this solver type is to decompose the problem into two sub-steps [4]. For the velocity field, the *bi-conjugate gradient stabilized* method (BiCGSTAB) is

applied as an efficient solver for the non-symmetric linear system. An *algebraic multi grid* solver (AMGCL) is used for the linear equation system of the pressure equation. Both are iterative solvers and accelerate the computation compared to direct solvers like the *SuperLU* solver.

To compute movements in two directions, several changes are necessary in the given SDOF class definition. The structural solver is based on the *Generalized- α* method treated before. It is implemented as a member function of the structure class and takes the external force as an additional argument. In preparation for further steps, this and other routines are duplicated and adapted for the y-direction.

```

1 def solveStructureX ( self , m1x ) :
...
3     self . u1x = RHS / LHS
4     self . v1x = self . a1v * ( self . u1x - self . u0x ) + self . a2v * self . v0x + self . a3v * self . a0x
5     self . a1x = self . a1a * ( self . u1x - self . u0x ) + self . a2a * self . v0x + self . a3a * self . a0x
7 def solveStructureY ( self , m1y ) :
...

```

The external forces resulting from the fluid are computed by means of a function defined in the main file (excerpt copied here for convenience). Additional branches are introduced for the cases with bidirectional movement or no movement. For the first, the force needs to be considered as a vector with two entries:

```

def ExtractForce ( nodes , dof_type ) :
    fx = 0.0
    fy = 0.0
    if dof_type == "DISPLACEMENT_X" :
        ...
    if dof_type == "DISPLACEMENT_Y" :
        ...
    if dof_type == "DISPLACEMENT_XY" :
        for node in nodes :
            reaction = node . GetSolutionStepValue ( REACTION , 0 )
            fx += -reaction [ 0 ]
            fy += -reaction [ 1 ]
        return [ fx , fy ]
    if dof_type == "NONE" :
        return 0

```

After the solution of the structural part, the calculated displacement of the structure is given back to the fluid solver so the mesh can be deformed accordingly. Due to this, the coupling between the structure and the flow follows a staggered Dirichlet-Neumann coupling.

4.5.2 Extension for MDOF systems

With this knowledge about the existing implementations, the next step is to achieve a MDOF system with a staggered coupling. Therefore, following the idea described previously, a second SDOF system object without spring stiffness or damping coefficient is introduced:

```

1 structure_dispX_solver1 = StructureMDOF_K ( Dt , mass , zeta , stiffness , ...
    rho_inf , initial_disp , initial_vel , initial_acc , output_filename )
3 structure_dispX_solver2 = StructureMDOF_K ( Dt , mass2 , 0 , 0 , rho_inf , ...
    [ 0 , 0 ] , [ 0 , 0 ] , [ 0 , 0 ] , output_filename2 )
5

```

Instead, these elements are the connections to the first SDOF. That is, the two are coupled via the equivalent forces transferred. Said forces are computed in a separate function:

```

1 def SpringAndDamperForce(displacement1, velocity1, displacement2, velocity2):
2     u1 = displacement1
3     u2 = displacement2
4     v1 = velocity1
5     v2 = velocity2
6     F = (u1 - u2)*kLink + (v1-v2)*cLink
7     return F

```

The strong coupling between both DOFs is implemented as a strong Neumann-Neumann coupling and, thus, is based on the exchange of forces in both directions. The forces acting on the DOFs have different signs (orientations). The iteration loop is left when the dynamic equilibrium is reached i. e. the magnitude of the exchanged force is the same in both exchange directions up to a specified tolerance. As always, it is important to highlight that for the external DOF, the force from the fluid also has to be considered.

```

1 structure_dispX_solver1.solveStructureX(reactionx - force2to1x)
2 structure_dispX_solver1.solveStructureY(reactiony - force2to1y)
3 force1to2x = SpringAndDamperForce(...)
4 force1to2y = SpringAndDamperForce(...)
5
6 structure_dispX_solver2.solveStructureX(force1to2x)
7 structure_dispX_solver2.solveStructureY(force1to2y)
8 force2to1x = SpringAndDamperForce(...)
9 force2to1y = SpringAndDamperForce(...)

```

4.6 Simulation cases

In the following, the simulation results of all cases defined in table 2 and more detailed described in appendix A) are presented. The displacements of the DOFs are visualized and the most prominent aspects of each case are discussed.

4.6.1 Case 1

Whereas the first square cylinder is fixed, the second cylinder is able to move in lateral direction (y-direction). One structural solver is needed for the SDOF system. Figure 9 shows a plot of the displacement over time.

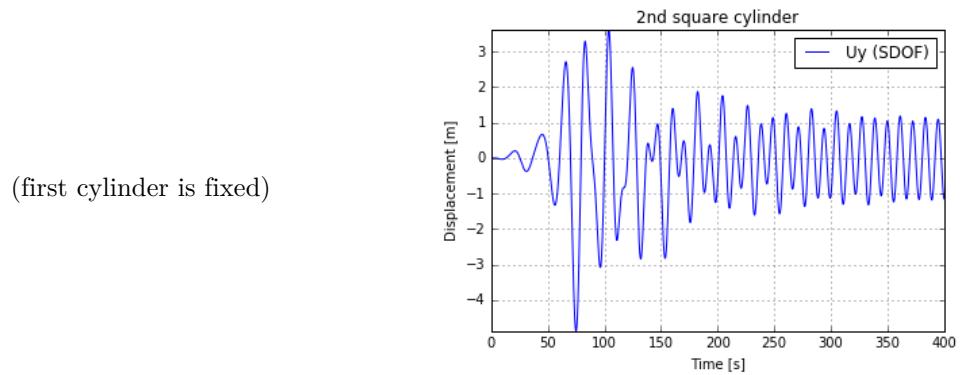


Figure 9: Displacement plot case 1

The lateral motion is caused by the vortex shedding and, therefore, does not start immediately. The first vortices that detach from the first cylinder are very powerful and lead to maximal displacements. A steady state is reached after roughly 250 seconds of simulation time.

4.6.2 Case 2

The case configuration allows both the first and the second cylinder to move in lateral direction. Again, a plot of the displacements is given below in Figure 10

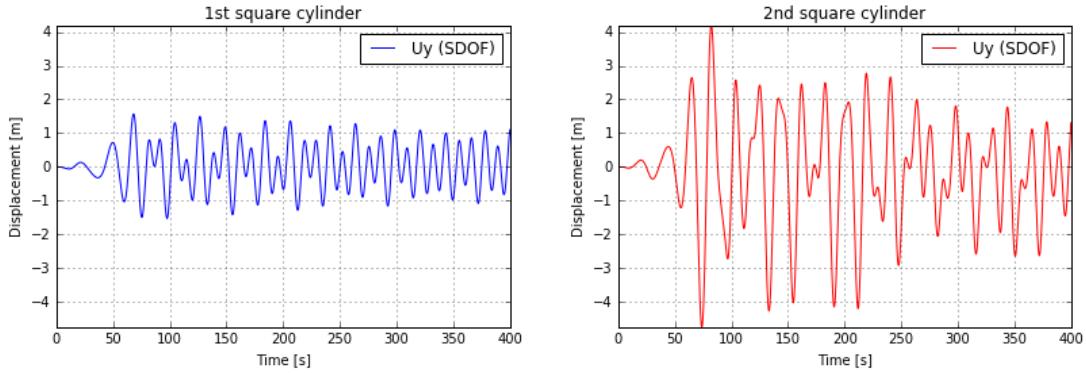


Figure 10: Displacement plot case 2

The displacements of the second cylinder are larger than those at the first cylinder. This can be explained by noting that the first cylinder generates the vortex street that affects the second cylinder.

4.6.3 Case 3

The second cylinder is given the possibility to move in x- and y-direction. For the two DOFs, two decoupled structural solvers are applied. The histories of the displacements are given in Figure 11.

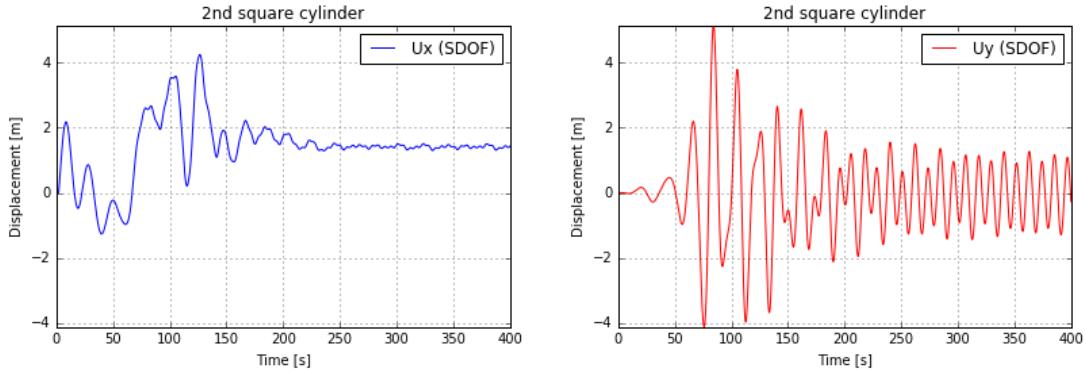


Figure 11: Displacement plot case 3

Again, the displacement in lateral direction takes a certain time to gain amplitude and reaches its maximum afterwards. In contrast, the displacement in x-direction can be observed right after the start of the simulation as the fluid drags the structure. During the period of the intensive vortex shedding, maximal values are reached. After roughly 250 seconds, the steady state seems to be reached and the x-displacement takes an approximately constant value of $\approx 1.5 \text{ m}$.

4.6.4 Case 4

Similar to case 1, the second cylinder is allowed to move in lateral direction. In this case, the second cylinder contains an MDOF system with two masses. The displacements of both DOFs visualized in Figure 12.

(first cylinder is fixed)

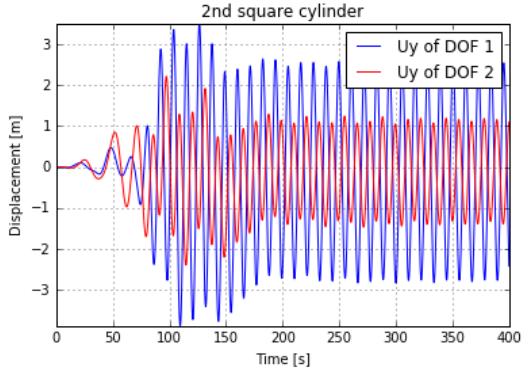


Figure 12: Displacement plot case 4

The movement of DOF 1 is caused by external aerodynamics, while a spring and a damper transfer forces to the second DOF. After a period with high amplitudes corresponding to the initial vortices is over, the steady state is reached. The system seems to oscillate in one of its eigenmodes and the displacement of both DOFs take opposite signs.

4.6.5 Case 5

Building up on case 4, the second cylinder is given an additional MDOF system with two masses moving in x-direction. As indicated in Figure 13, the first cylinder is fixed.

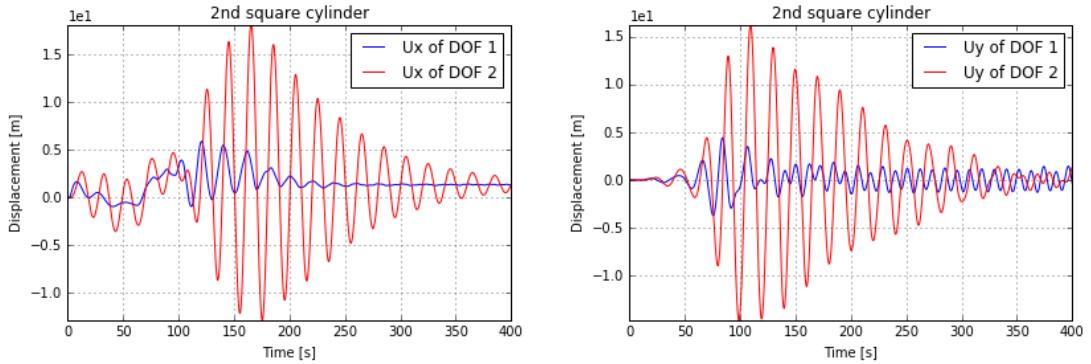


Figure 13: Displacement plot case 5

The expected onset behavior of the movements can be observed. The second DOF shows larger displacements. After a period of extremely large amplitudes is passed, the steady state is reached. The second DOF seems to have a stabilizing effect here.

4.6.6 Case 6

Both cylinders are defined as MDOF systems and are allowed to move in both directions. In total, eight structural solvers were applied. The results are shown below in Figure 14.

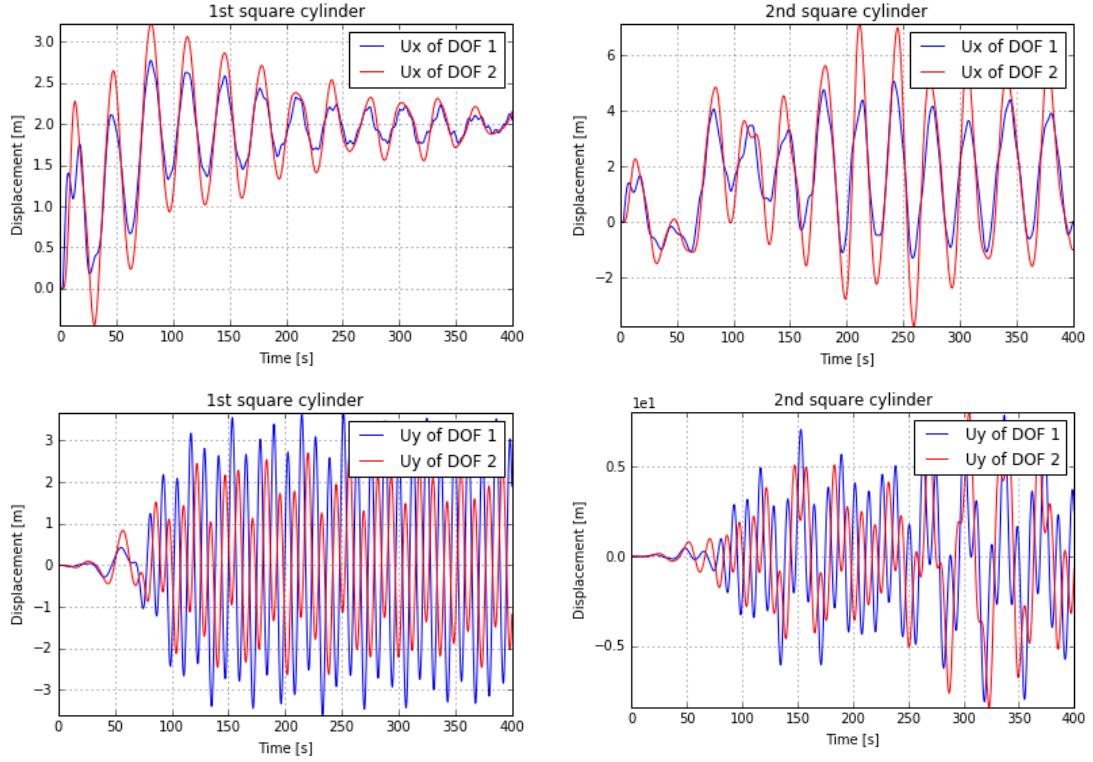


Figure 14: Displacement plot case 6

The displacement of the second cylinder in lateral direction has the highest amplitude and, in general, the results deviate from harmonic oscillations. However, the systems appear to oscillate in their eigenmodes. Specifically, in x-direction the two DOFs move in a synchronized fashion, whilst the y-direction shows an opposing behavior at particular points in time. Additionally, the frequency of the movement in y-direction is higher given the excitation from the vortices.

4.7 Effect of the attached DOF

Since a direct visualization of the displacement plots for the degrees of freedom in cases 2 vs. 4 and vs. 5 doesn't provide a clear picture of the effect of the second (internal) DOF on the system, this is further investigated by means of the frequency spectra. A Fast Fourier Transform (FFT) is performed on the time dependent displacement of the second cylinder on each direction (i. e. x - and y - directions are treated independently) since the DOFs themselves were uncoupled.

For cases 3 and 5, presented here, the data contains $N = 8000$ samples, which leads to a sampling frequency of $f_s = 20 \text{ Hz}$. Here, the DOFs can move in two directions and the displacement of the first (external) DOF is considered. Due to the decreased stiffness, both DOFs move slowly and low frequencies of $< 0.2 \text{ Hz}$ result. Figure 15 shows the frequency spectrum for each direction.

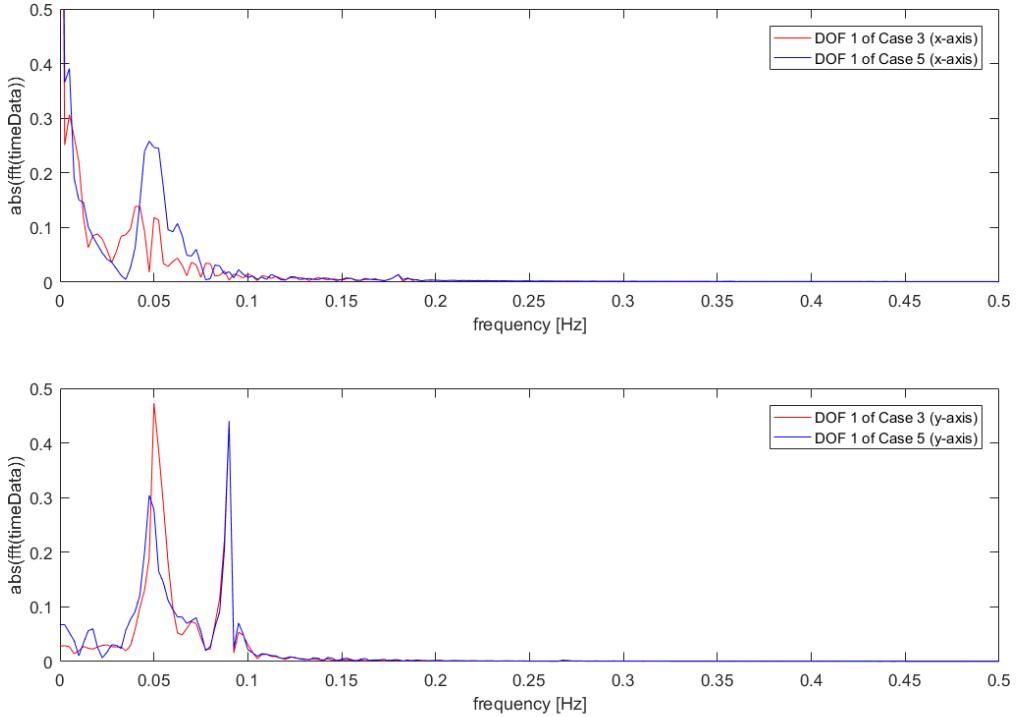


Figure 15: Frequency spectra of the displacement in x -direction (top) and y -direction (bottom)

The two spectra differ only slightly, because no special care was taken of the tuning of the system. More so, by using the same values for all properties of this second DOF, the effects of the internal degree of freedom end up being detrimental in one of the directions. The spectrum for the x -axis of case 3 (without a second DOF) shows minor peaks around $f = 0.05 \text{ Hz}$. Rather than vanishing, these peaks, along with the surrounding frequencies, are collected into a single major peak centered at $f = 0.05 \text{ Hz}$. Therefore, the corresponding frequency is now more critical than before in terms of potential resonance effects.

In the meantime, the y -axis shows a pair of peaks at $f = 0.05 \text{ Hz}$ and $f = 0.08 \text{ Hz}$ approximately for case 3. The addition of the new DOF in case 5 corresponds with a decrease of the peak at $f = 0.05 \text{ Hz}$. While the peak does not entirely vanish, its value decreases to around 60% its original value.

Thus, it would appear that the MDOF system, with its current parameter values, is transferring the energy from the y -direction into the x direction, which is possible only due to the fluid interactions (since the structural solvers are decoupled).

4.8 Parameter study

Finally, the parameter study conducted here investigates the stability of the simulation under variation of the chosen dimensionless parameters. This is first done mainly on the structural solver, since some parameter ratios are particularly critical to specific coupling strategies - such as the mass ratio in a Dirichlet-Neumann coupling. While this is not the case for the Neumann-Neumann coupling (i. e. the mass-ratio), it is still interesting to determine the limits of the simulation. An additional incursion revolving around the fluid simulation is performed to verify the limits of the setup defined.

In all cases the simulations are run until a simulation time of 400 s or until the simulation stopped due to divergence or otherwise crashed. These tests were run initially with case 4 (where only y -axis displacement is possible) to determine if the qualitative behavior of the coupled system was sensible, with ratios of $\varphi_2/\varphi_1 = 0.02$ and 500, where φ represents the three relevant parameters: mass, stiffness, and dampening. Once this was accomplished, the limiting cases were approximated with case 5 (allowing also x -axis displacement) since now either axis could diverge first. The obtained results are presented in the following.

4.8.1 Variation of the second mass m_2

In this case mass ratio $R_m = \frac{m_2}{m_1}$ of the two DOFs is changed by choosing different values for m_2 . If it is too low, then the second degree of freedom is unable generate inertial forces, so the displacement is simply following the external DOF. Conversely, if the mass is much higher, the DOF is essentially fixed, and displacement is practically zero. This is precisely the behavior observed from Figure 16.

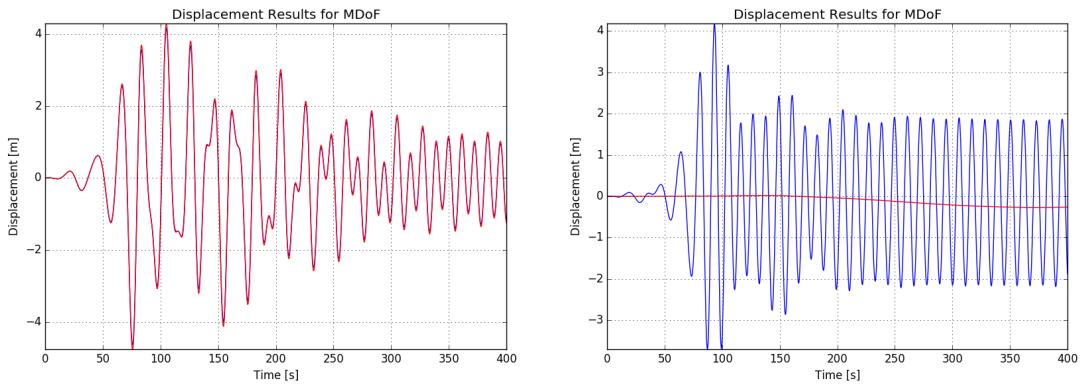


Figure 16: Displacement for both DOF in case 4 with much lower (left) and higher (right) mass 2. Note: Red symbolizes the internal degree of freedom.

If we now simulate case 5 with more extreme values of mass 2, the simulation starts requiring more iterations (due to the diminished inertial response), and ends up crashing for $R_m < 0.002$. That is, the simulation was still stable for $R_m = 0.002$, but crashed at $R_m = 0.001$. On the other hand, higher values didn't lead to instability since the DOF was essentially fixed.

4.8.2 Variation of the second spring stiffness k_2

For the stiffness ratio $R_k = \frac{k_2}{k_1}$, again varying values of k_2 were chosen. In this case, the opposite effects were perceived, which is expected since the mass and the stiffness are related via the natural frequency of the system: $\omega_n = \sqrt{\frac{k}{m}}$. Therefore, lowering the stiffness essentially leaves the second DOF unattached (since dampening effects alone are less able to excite it) and raising it couples the displacements of both via a rigid link. Again, this is precisely the behavior observed from Figure 17.

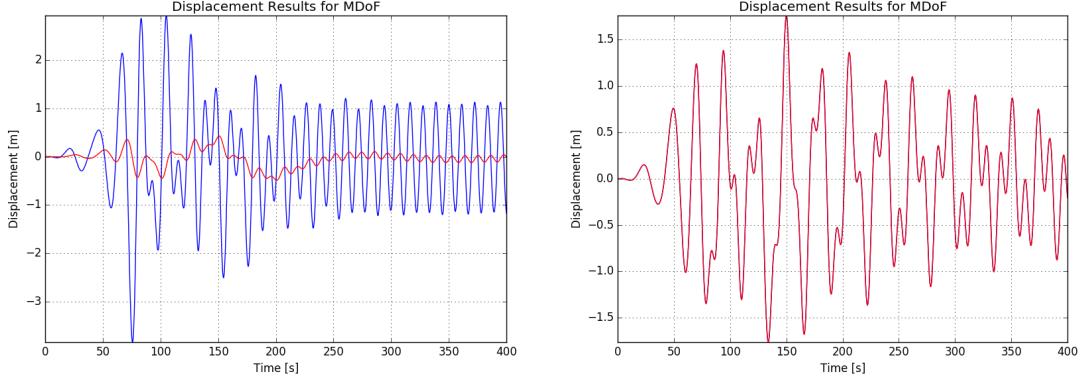


Figure 17: Displacement for both DOF in case 4 with much lower (left) and higher (right) stiffness 2. Note: Red symbolizes the internal degree of freedom.

If we now simulate case 5 with more extreme values, the simulation remains stable for lower values of stiffness, but ends up crashing for $R_k > 5000$. The simulation was still stable here, but crashed for $R_k = 8000$.

4.8.3 Variation of the second dampening coefficient c_2

For the dampening ratio $R_c = \frac{c_2}{c_1}$, c_2 was varied. In this case, the resulting displacements are initially harder to interpret, but still fit within expectations. First, lowering the dampening results in a practically undamped system, therefore behaving like the external DOF with a different amplitude and a phase lag of half a period. On the other hand, if the dampening is too strong, the dynamic response simply follows the external DOF since the velocities of both DOFs are coupled. This behavior observed in Figure 18.

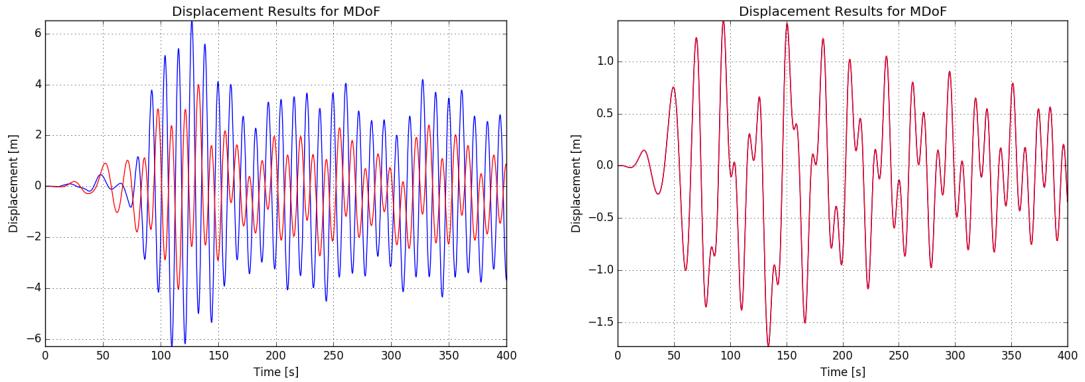


Figure 18: Displacement for both DOF in case 4 with much lower (left) and higher (right) dampening 2. Note: Red symbolizes the internal degree of freedom.

It is worth mentioning that for this parameter, even case 4 was already requiring more iterations, so instability would have occurred soon after, without the need of simulating case 5.

4.8.4 Variation of the fluid simulation parameters

To conclude the parameter study, an alternative approach focuses on the fluid simulation. In this case, two sub-cases were considered: varying the time step, and the velocity. In the case of the time step, it is known that a fine enough time resolution is required to yield accurate results, since otherwise the information transfer occurs too slowly to be updated correctly. The crash here occurred after $dt = 3.0\text{ s}$

(but was still stable at $dt = 2.5$ s).

One way to analyze this crash is via the CFL number, which is related to stability of the numerical scheme. Using the mean velocity and the smallest elements, this case yields a CFL number of

$$CFL = \frac{u \Delta t}{\Delta x} = \frac{25.0 \frac{m}{s} \cdot 3.0 s}{3.0 m} = 25.0$$

which acts as a stability limit for the fluid problem. As a comparison, the original CFL number was $CFL = 0.416$ (for $dt = 0.05$ s).

By changing the velocity, however, we would obtain the equivalent CFL number when raising the value 60 times. This value is, however, much higher than necessary to reach a limit in the simulation: with a velocity less than 10 times higher the simulation already collapses. In fact, a more careful study shows the simulation is still stable for $v = 170 \frac{m}{s}$ but not for $v = 180 \frac{m}{s}$.

To better understand this, the simulation is repeated with $v = 250 \frac{m}{s}$ but only until the last converged value, which is within the 40.0 s time frame (or 0.1 the original simulation time). Figure 19 presents the deformed configuration for the final time step as well as the displacement results obtained.

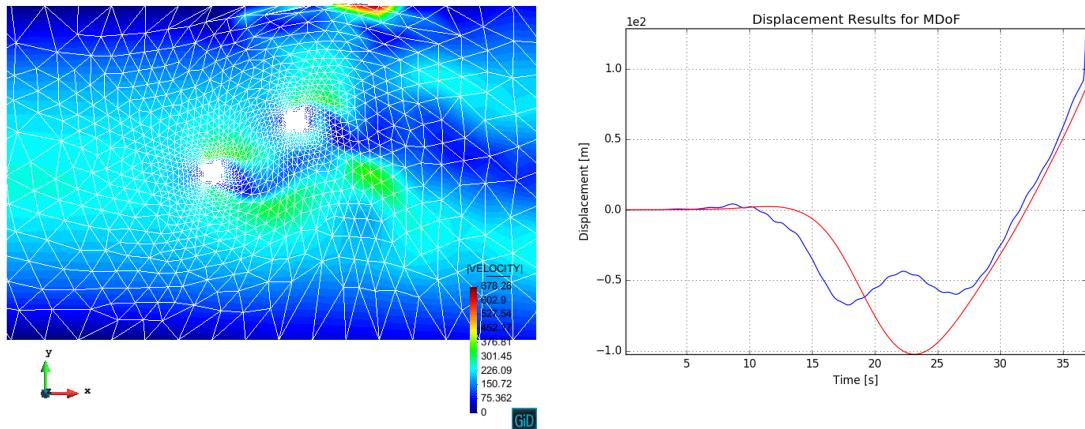


Figure 19: Final simulated time step (left) and displacement curve for both DOF under a high velocity simulation (right). Note: Red symbolizes the internal degree of freedom.

The limiting factor is, therefore, identified to be the mesh deformation from the ALE approach, since one of the nodes actually falls outside the domain of the problem. This leads to a conflict with the defined parametrization (i. e. the Jacobian of the mapping function would be negative) and ultimately leads to the divergence in the solution. Indeed, the CFL number here is only $CFL = 4.16$, much lower than the limiting value obtained earlier.

5 Conclusion and Future Work

Using the *Kratos Multiphysics* framework, a two-dimensional FSI simulation was set up to analyze the behavior of two square cylinders in a channel air flow. The given routines were extended by an additional MDOF solver allowing further variants of the configuration with two DOFs in one direction. Neumann-Neumann coupling was used to implement the structural solver and it was verified against a monolithic solver based on the Implicit Euler method. The obtained results and the stability of the simulation were

investigated under different aspects.

Possible further steps could include introducing a rotational degree of freedom for the cylinder, hence coupling the x- and y- directions of the rigid boxes. Also, an additional structural coupling between some DOFs belonging to different structures seems reachable and could yield new interesting results. Different coupling schemes (e.g. Aitken) and other stabilization techniques for the fluid simulation (e. g. OSS or ASGS) could also be tested. These improvements would allow further cases of technical relevance in different engineering disciplines to be simulated.

REFERENCES

- [1] Friedrich-Karl Benra et al.
“A comparison of one-way and two-way coupling methods for numerical analysis of fluid-structure interactions”.
In: *Journal of applied mathematics* 2011 (2011).
- [2] A Bonelli et al.
“Analyses of the Generalized- α method for linear and non-linear forced excited systems”.
In: *Structural Dynamics-EURODYN*, vol. 2, pp. 1523-1528.
2002.
- [3] J. Cotela, E.O.I. de Navarra, and R. Rossi.
Applications of Turbulence Modeling in Civil Engineering.
International Center for Numerical Methods in Engineering, 2016.
- [4] J. Donea and A. Huerta.
Finite Element Methods for Flow Problems.
John Wiley & Sons, 2003.
- [5] International Center for Numerical Methods in Engineering.
Kratos Multiphysics.
URL: <http://www.cimne.com/kratos/>.
- [6] Silvano Erlicher, Luca Bonaventura, and Oreste S Bursi.
“The analysis of the generalized- α method for non-linear dynamic problems”.
In: *Computational Mechanics* 28.2 (2002), pp. 83–104.
- [7] J.H. Ferziger and M. Peric.
Computational Methods for Fluid Dynamics.
Springer Berlin Heidelberg, 2012.
- [8] Ulrich Küttler and Wolfgang A Wall.
“Fixed-point fluid–structure interaction solvers with dynamic relaxation”.
In: *Computational Mechanics* 43.1 (2008), pp. 61–72.
- [9] Miriam Mehl.
Efficient Coupling and Flow Simulations in Partitioned Fluid-Structure Interaction Simulations.
Orlando, USA, 2009.
- [10] Artur K Pozarlik and Jim BW Kok.
“Numerical investigation of one-and two-way fluid-structure interaction in combustion systems”.
In: (2007).

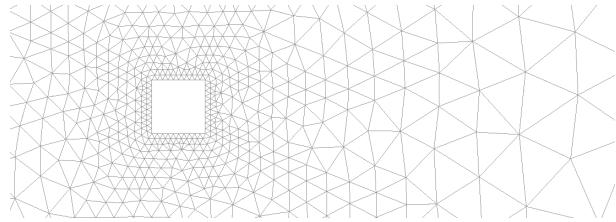
- [11] Jean-Marc Vassen et al.
“Strong coupling algorithm to solve fluid-structure-interaction problems with a staggered approach”.
In: *7th European Symposium on Aerothermodynamics*, Vol. 692, p. 128.
2011.

A Case Configuration

Case 0:

0 SDOF solvers
0 MDOF solvers

The fixed cylinder acts as an immobile boundary.



Case 1:

1 SDOF solvers
0 MDOF solvers

The second cylinder can move in lateral direction.



Case 2:

2 SDOF solvers
0 MDOF solvers

Both cylinders can move in lateral direction.



Case 3:

2 SDOF solvers
0 MDOF solvers

The second cylinder can move in both directions.



Case 4:

0 SDOF solvers
1 MDOF solvers

The second cylinder has two DOFS in lateral direction.



Case 5:

0 SDOF solvers
2 MDOF solvers

The second cylinder has two DOFS in each direction.



Case 6:

0 SDOF solvers
4 MDOF solvers

Both cylinders have two DOFS in each direction.

