


RESEARCH ARTICLE

Open Access



Partitioned simulation strategies for fluid–structure–control interaction problems by Gauss–Seidel formulations

Andreas Winterstein^{1*} , Christopher Lerch², Kai-Uwe Bletzinger¹ and Roland Wüchner¹

*Correspondence:

a.winterstein@tum.de

¹Chair of Structural Analysis,
Technical University of Munich,
Arcisstraße 21, 80333 Munich,
Germany

Full list of author information is
available at the end of the article

Abstract

In this contribution the multi-physics problem of fluid–structure–control interaction (FSCI) is solved by an iterative, partitioned approach utilizing Gauss–Seidel formulations. The aim is to conduct a fully coupled co-simulation of the FSCI problem, where the controller actively influences the dynamics of the structure. The purpose of this manuscript is twofold: In the first part, in order to get a profound idea of the behavior and parametric sensitivity of such systems involving multiple couplings, the simplified model problem introduced for fluid–structure interaction (FSI) by Joosten, Dettmer and Perić is extended by a generic control unit. Since a monolithic solution for this simplified model problem can be found, it is used for first investigations concerning solvability and stability. On this basis, three different variants for coupling the subsystems fluid, structure and controller by a Gauss–Seidel scheme, are derived and systematically investigated. More precisely the FSCI problem is solved without nesting of the subsystems in the first variant and with nesting of two of the respective subsystems in the second and third variant. In the second part, the resulting algorithms are applied to a complex, non-linear, multi-degree of freedom problem, which is a well-known benchmark problem in the FSI community and is therefore extended to FSCI. Applying those algorithms to the multi-degree of freedom problem shows good results and substantiates the applicability to such problems. It follows, actively influencing the dynamics of the structure in the FSCI problem by a controller reduces the structural vibrations induced by the fluid flow significantly.

Keywords: Fluid–structure interaction, Multi-physics, State-feedback control, Partitioned solution procedure, Fluid–structure–control interaction (FSCI), Co-simulation

Introduction

The development in the community of coupled problems tends more and more to dealing with multi-physics problems containing more than two physical fields. One of the first contributions for the partitioned treatment of such problems has been made in [1], which gives a general overview about the treatment of coupled problems by a partitioned approach. More recent developments are for example fluid–structure interaction with electro magnetics [2], fluid–structure–contact interaction [3] or general n-field coupling [4,5].

In this contribution the coupled problem of fluid–structure–control interaction (FSCI) is treated in a partitioned way. This means a closed-loop control unit, which manipulates the dynamics of the structure, is added to the well-known two field problem of fluid–structure interaction (FSI). This kind of problem statement was first mentioned in [1, p. 3262 and 3263], but has not been followed in more detail. Also in [4], FSCI to reduce flow induced structural displacements has been mentioned as a side note in the context of testing the algorithm developed therein. In contrast to the present contribution, a rigid structure with one degree of freedom with small displacements with a very simple control-law is shown. Furthermore [4], utilizes a Jacobi pattern instead of a Gauss–Seidel pattern, which is the basis of the developments in this manuscript.

The objective of applying a control unit to the fluid–structure interaction problem is getting a minimum or at best zero displacement.

In the partitioned approach the subsystems are modeled and solved numerically independently of each other [1,6,7]. The interaction between the subsystems in the overall system is achieved by coupling equations on the interface level.

In the case of the FSCI problem the partitioned approach makes it simpler to add the controller to the problem, which cannot be seen as a physical field, but only as a signal. In contrast to physical fields, which have an affiliation to a certain spatial domain, signals do not have this characteristic.

When using the partitioned approach an important issue is the stability of the overall simulation [2]. When analyzing the stability behavior of multi-physics problems, one may run into troubles due to the superposition of many different effects.

Therefore, in order to get a good insight into the behavior of such complicated problems in the field of computational FSI, it became well established practice to fall back to highly simplified model problems for detailed investigations of different solution schemes. Such simplified models only represent the relevant properties of the actual FSI problem, thus they give more insight and open the opportunity to formulate closed-form formulations. Within this paper such a simplified model problem, used for instance in [8,9, p. 4–6, p. 1365], will be expanded. In [8, Remark I p. 5–6], and conclusion p. 20, 21 as well as in [10] it is shown that this simplified model problem is sufficient for the analysis of a broad spectrum of solution schemes for FSI problems regarding properties like stability, convergence behavior, accuracy and high frequency damping. Thus, the overall behavior of multi-degree of freedom FSI problems is explained quite well [9,11]. The basic findings and algorithms obtained by the simplified model problem can be applied to more complex multi-degree of freedom examples.

This paper is organized as follows: In the section “The simplified model problem” the simplified model problem for FSCI is presented. For this simplified model problem the monolithic equations are derived and a stability analysis is conducted. In the “Alternatives of the Gauss–Seidel pattern for three different physical fields” section, supported by the simplified model problem, three different variants of an iterative Gauss–Seidel scheme for coupling the three subsystems are developed. Those schemes are applied to a numerical multi-degree of freedom example in the “Numerical results for a multi-degree of freedom FSCI problem” section. Finally in the “Conclusion and outlook” section a conclusion and ideas for future work are given.

The simplified model problem

The simplified model problem introduced by [8,9] is extended by a generic control unit. According to [8,9], the system approximating the FSI scenario in the simplest way is the combination of point mass, linear damper and a linear elastic spring. This simplified model problem is illustrated in Fig. 1 in a monolithic version (a) and a partitioned version (b). The decomposition into three subsystems and the explicit realization of interfaces (each creating an interface constraint equation), is visualized in Fig. 1b by the orange separators. Figure 1 is described in more detail during the course of this subsection. The features of the newly proposed simplified FSCI model problem can be summarized as follows: the coupling of a first order ordinary differential equation (ODE) representing the fluid flow and a second order ODE representing the structure reproduces the FSI problem. This is extended to the FSCI problem by adding the algebraic ODE of the controller. In the simplified model problem the combination of viscosity/inertia in one subsystem (fluid flow) and stiffness/inertia in the other (structure) also corresponds to the main characteristics of FSI problems. The physics are still dominated by the FSI subproblem, since inertia is limited to the fluid flow and the structure. The controller is only adjusting the dynamics of the structure. An iterative/strong/implicit coupling is applied, meaning all interface constraints are fulfilled strictly using an interface iteration loop. More precisely, we use a Dirichlet/Neumann [10, p. 4517 et seq.] coupling, with a block Gauss–Seidel procedure [11–13]. The monolithic version in sub-figure (a) results in the well-known single degree of freedom (SDoF) system

$$m\ddot{y}(t) + c\dot{y}(t) + ky(t) = u(t), \tag{1}$$

with its initial conditions

$$\begin{aligned} y(0) &= y_0, \\ \dot{y}(0) &= \dot{y}_0. \end{aligned} \tag{2}$$

Herein the constants m , c , and k represent the mass, the viscous damping and the stiffness. The variables $\ddot{y}(t)$, $\dot{y}(t)$ and $y(t)$ represent the acceleration, velocity and displacement. The variable $u(t)$ on the right hand side represents the control input. Thus, y_0 is the initial displacement and \dot{y}_0 is the initial velocity. Equation (1) is equivalent to [8, p. 5], enhanced by a generic, but representative state-feedback controller. The state-feedback controller equation is defined as

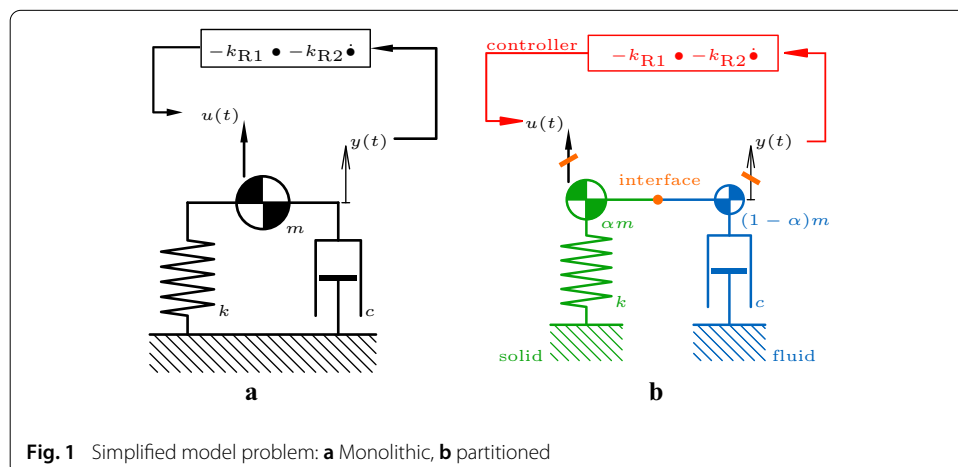


Fig. 1 Simplified model problem: **a** Monolithic, **b** partitioned

$$u(t) = -k_{R1}y(t) - k_{R2}\dot{y}(t). \quad (3)$$

The constant factors k_{R1} for the displacement and k_{R2} for the velocity in the control input are used to tune the controller. Inserting Eq. (3) into (1) one gets

$$m\ddot{y}(t) + (c + k_{R2})\dot{y}(t) + (k + k_{R1})y(t) = 0, \quad (4)$$

with the initial conditions from Eq. (2), which is a controlled SDoF system. In the following this is referred to as simplified model problem. This system stays linear and one-dimensional. The reason for this is the treatment of the control input $u(t)$ as Neumann boundary condition on the SDoF system, which corresponds to a disturbing force $z(t)$. Therefore no additional displacement degree of freedom is added. This results in pure force control for the considerations in this work.

Monolithic approach

For the temporal discretization of the monolithic model problem the implicit Euler time integrator also called “first order backwards differentiation formula” (BDF1) is used. The BDF1 is defined as:

$$\begin{aligned} y^{n+1} &= y^n + \delta t \dot{y}^{n+1}, \\ \dot{y}^{n+1} &= \dot{y}^n + \delta t \ddot{y}^{n+1}. \end{aligned} \quad (5)$$

Herein δt is the discrete time step size and y^{n+1} and y^n are the discrete instances of the displacements, velocities \dot{y}^{n+1} and accelerations \ddot{y}^{n+1} at time steps t^{n+1} and t^n , with n being the time step counter. Rewriting Eq. (5) and applying it to Eq. (4) leads to the time discrete monolithic expression of the coupled system.

$$(m + (c + k_{R2})\delta t + (k + k_{R1})\delta t^2)y^{n+1} - (2m + (c + k_{R2})\delta t)y^n + my^{n-1} = 0 \quad (6)$$

and its discrete initial conditions

$$\begin{aligned} y^{-1} &= y_0 - \delta t \dot{y}_0, \\ y^0 &= y_0. \end{aligned} \quad (7)$$

Thus it is subsequently possible to derive statements, which reflect the choice of the controller parameters k_{R1} and k_{R2} for which the controlled system shows stable dynamics.

Analysis of the time-continuous problem

The stability region Ω_c for the time-continuous, monolithic simplified model problem Eq. (4) is derived using its characteristic polynomial. The characteristic polynomial reads

$$p(s) = ms^2 + (c + k_{R2})s + (k + k_{R1}). \quad (8)$$

This is Eq. (4) transformed to the complex s -plane by a Laplace transform, where s is a complex number. The roots of Eq. (8) are defined as

$$\{s \in \mathbb{C} \mid p(s) = 0\}. \quad (9)$$

In this case asymptotically and bounded input bounded output (BIBO) stability coincide [14, p. 63, et seq.] or [15, p. 45, et seq.]. The time-continuous stability region Ω_c results in

$$\begin{aligned} \Omega_c &= \left\{ k_{R1}, k_{R2} \in \mathbb{R} \mid \max_{i=1,2} \{\operatorname{Re}\{s_i\}\} \leq 0 \right\} \\ &= \{k_{R1}, k_{R2} \in \mathbb{R} \mid k_{R1} \geq -k \wedge k_{R2} \geq -c\}. \end{aligned} \quad (10)$$

Herein s_i denote the two poles of the time-continuous problem, which are specified by its eigenvalues.

Analysis of the time-discrete problem

In a similar way, the stability region Ω_d for the time-discrete, monolithic model problem Eq. (6) is determined. Its characteristic polynomial reads

$$p(z) = (m + (c + k_{R2})\delta t + (k + k_{R1})\delta t^2)z^2 - (2m + (c + k_{R2})\delta t)z + m = 0. \quad (11)$$

This is Eq. (6) transformed to the complex z -plane by a matched z -transform. The roots of Eq. (11) are defined as

$$\{z \in \mathbb{C} \mid p(z) = 0\}. \quad (12)$$

Finally, the map between the time continuous s -plane and the time discrete z -plane is defined as

$$z = e^{\delta t s}. \quad (13)$$

The disturbance force $z(t)$ is not to be mixed up with the z from the matched z -transform.

The two basic stability conditions change for the time discrete case [14]. Consequently, the time-discrete region Ω_d formulated in the z -plane results in

$$\Omega_d = \left\{ k_{R1}, k_{R2} \in \mathbb{R} \mid \max_{i=1,2} \{|z_i|\} \leq 1 \right\}. \quad (14)$$

Mapped back to the s -plane with Eq. (13) this reads

$$\begin{aligned} \Omega_d &= \{k_{R1}, k_{R2} \in \mathbb{R} \mid \frac{1}{\delta t} \ln \left(\max_{i=1,2} \{|z_i|\} \right) \leq 0\} \\ &= \{k_{R1}, k_{R2} \in \mathbb{R} \mid k_{R1} \geq -k \\ &\quad \wedge k_{R2} \geq -c - \delta t(k_{R1} + k)\}. \end{aligned} \quad (15)$$

Herein z_i denote the two poles of the time-discrete problem, which are specified by its eigenvalues.

Clearly recognizable in Eq. (15) is the fact that the time-continuous stability region Ω_c representing real physics gets extended to an apparently larger, time-discrete stability region Ω_d . This has to be taken into account when conducting a simulation based controller design.

Partitioned approach

The initial step of a partitioned approach is the decomposition of the multi-physics problem into single-physics subproblems and appropriate interface constraints covering the interactions. This is referred to as partitioning [6] and is shown in Fig. 1b.

A preparatory step for reaching a suitable partitioning of the simplified model problem is the reformulation of the ODE Eq. (1) as

$$(\alpha m)\ddot{y}(t) + ((1 - \alpha)m)\ddot{y}(t) + c\dot{y}(t) + ky(t) = u(t) + z(t). \quad (16)$$

The disturbance force on the right hand side $z(t)$ has to be split up into $z_F(t)$ and $z_S(t)$, since Eqs. (17) and (18), which are the partitioned equations, need a disturbance force each. In combination with Eqs. (2) and (3) this leads finally to the partitioned, simplified model problem:

$$((1 - \alpha)m)\ddot{y}_F(t) + c\dot{y}_F(t) = z_F(t), \quad (17)$$

is referred to as the fluid subsystem (index F),

$$(\alpha m)\ddot{y}_S(t) + ky_S(t) = u_S(t) + z_S(t), \quad (18)$$

as the structural subsystem (index S) and

$$u_C(t) = -k_{R1}y_C(t) - k_{R2}\dot{y}_C(t), \quad (19)$$

as controller subsystem (index C). The physical interaction is shifted to the interface constraints (index I)

$$\begin{aligned} y_F(t) - y_S(t) &= 0, \\ z_F(t) + z_S(t) &= 0, \\ y_S(t) - y_C(t) &= 0, \\ u_S(t) - u_C(t) &= 0. \end{aligned} \quad (20)$$

The initial conditions for the structure are given with

$$\begin{aligned} y_S(0) &= y_0, \\ \dot{y}_S(0) &= \dot{y}_0. \end{aligned} \quad (21)$$

Thus, the structural domain is represented by the elastic spring k and the point mass share αm , the fluid domain by the linear damper c and the point mass share $(1 - \alpha)m$. The interface constraints cover the interactions between these two domains (FSI) and between structure and controller (SCI). $y(t)$ describes the displacement, which corresponds to the measured output. $z(t)$ is the disturbance (force) originating from the partitioning and $u(t)$ the control input. The interface constraint equations are formulated in Eq. (20).

The parameter $\alpha \in [0, 1)$ describes the mass distribution between fluid and structural subsystem, i.e.

$$\frac{m_S}{m} = \alpha \quad \text{and} \quad \frac{m_F}{m} = 1 - \alpha, \quad (22)$$

and allows to precisely quantify the added-mass effect [8,10,16], which also applies to FSCI problems. Also other “ α ”-parameters regarding the damping c and the stiffness k would be feasible [8, p. 5]. At this stage only one parameter α associated with the mass m is considered. In the dominating FSI subproblem the convergence properties of the relaxed iteration factor βA in the limit case $\delta t \rightarrow 0$ depend only on this one parameter [11, Sect. 3, p. 763].

The temporal discretization of the partitioned simplified model problem Eqs. (17), (18), (19), (20) and (21), with the BDF1 scheme leads to the discrete, partitioned, simplified model problem. It consists of the discrete fluid Eq. (23), structural Eq. (24) and controller subsystem Eq. (25):

$$\begin{aligned} z_F^{n+1} &= \frac{(1 - \alpha)m + c\delta t}{\delta t^2} y_F^{n+1} \\ &\quad - \frac{(1 - \alpha)m + c\delta t}{\delta t^2} y_F^n - \frac{(1 - \alpha)m}{\delta t} \dot{y}_F^n, \end{aligned} \quad (23)$$

$$\begin{aligned} z_F^{n+1} &= \mathcal{G}_F \left(y_F^{n+1} \right), \\ y_S^{n+1} &= \frac{\delta t^2}{\alpha m + k\delta t^2} z_S^{n+1} + \frac{\delta t^2}{\alpha m + k\delta t^2} u_S^{n+1} \\ &\quad + \frac{\alpha m}{\alpha m + k\delta t^2} y_S^n + \frac{\alpha m \delta t}{\alpha m + k\delta t^2} \dot{y}_S^n, \end{aligned} \quad (24)$$

$$\begin{aligned} y_S^{n+1} &= \mathcal{G}_S \left(z_S^{n+1}, u_S^{n+1} \right), \\ u_C^{n+1} &= -\frac{k_{R1}\delta t + k_{R2}}{\delta t} y_C^{n+1} + \frac{k_{R2}}{\delta t} \dot{y}_C^n, \\ u_C^{n+1} &= \mathcal{G}_C \left(y_C^{n+1} \right) \end{aligned} \quad (25)$$

and the discrete interface equations

$$\begin{aligned} y_F^{n+1} - y_S^{n+1} &= 0, \\ \text{i.e. } \mathcal{I}_{FS,y} \left(y_F^{n+1}, y_S^{n+1} \right) &= 0, \end{aligned} \quad (26)$$

$$\begin{aligned} z_F^{n+1} + z_S^{n+1} &= 0, \\ \text{i.e. } \mathcal{I}_{FS,z} \left(z_F^{n+1}, z_S^{n+1} \right) &= 0, \end{aligned} \quad (27)$$

$$\begin{aligned} y_S^{n+1} - y_C^{n+1} &= 0, \\ \text{i.e. } \mathcal{I}_{SC,y} \left(y_S^{n+1}, y_C^{n+1} \right) &= 0, \end{aligned} \quad (28)$$

$$\begin{aligned} u_S^{n+1} - u_C^{n+1} &= 0, \\ \text{i.e. } \mathcal{I}_{SC,u} \left(u_S^{n+1}, u_C^{n+1} \right) &= 0. \end{aligned} \quad (29)$$

The operators \mathcal{G} and \mathcal{I} describe the input-output relation for the specific subsystem and the interface constraint for the specific coupling, respectively.

The FSI subproblem, i.e. the coupling between fluid and structure, converges to the solution of Eqs. (23), (24), (26) and (27). The emerging system is the “fluid–structure (FS) subsystem”, $\mathcal{G}_{FS} \left(u_S^{n+1} \right)$. Accordingly, the converged solution of the SCI subproblem, i.e. the coupling between structure and controller fulfills Eqs. (24), (25), (28) and (29). This leads to a “structure–controller (SC) subsystem”, $\mathcal{G}_{SC} \left(z_S^{n+1} \right)$.

Alternatives of the Gauss–Seidel pattern for three different physical fields

Three different alternatives for the serial Gauss–Seidel pattern for the fluid–structure–control interaction problem are described and applied to the partitioned simplified model problem consisting of Eqs. (23)–(29). The resulting nonlinear interface equation system can be solved in different ways. Since in this contribution we want to concentrate on the algorithmic aspects of the overall problem, the simplest iterative technique applying fixed point iterations Eq. (30), accelerated by relaxation Eq. (31) is used [17, p. 652–659]. Applying a Gauss–Seidel pattern, this implies the subsequent solution of all involved single-physics subsystems in each iteration step.

$${}^{k+1}y_S^{n+1} = {}_1A {}^k y_S^{n+1} + b^n. \quad (30)$$

$${}^{k+1}y_S^{n+1} = \beta \left({}_1A {}^k y_S^{n+1} + b^n \right) + (1 - \beta) {}^k y_S^{n+1},$$

$${}^{k+1}y_S^{n+1} = \beta A {}^k y_S^{n+1} + \beta b^n, \quad (31)$$

Herein the index k indicates the iteration counter for the interface iterations, the index n the counter for the time integration, β denotes the user-defined relaxation parameter, ${}_1A$ and βA denote the unrelaxed and relaxed iteration factors.

For each of those variants the limits of the unrelaxed ${}_1A$ and the relaxed βA iteration factors are derived and the optimal relaxation parameter β^* is calculated. The algorithms in pseudocode notation for the different alternatives can be found in the appendix.

No nesting (FSCI)

In the context of this subsection, the acronym FSCI also stands for the more specific iterative coupling scheme illustrated in Fig. 2, where the Gauss–Seidel communication pattern is realized without nesting of any subproblems, i.e. the coupled problem is solved with a single fixed-point iteration loop. This means only one interface equation system

$\mathcal{I}_{\text{FSCI}}$ has to be solved. In the graphical representation as a block diagram, each of the physical fields and the interface equations are outlined by one of the blocks. The arrows describe the input and output quantities, which are passed between the blocks. Applying Eq. (30) to the partitioned, simplified model problem the equations of the algorithm condense down to

$$\begin{aligned}
 {}^{k+1}y_S^{n+1} &\stackrel{(24)}{=} \mathcal{G}_S \left({}^k z_S^{n+1}, {}^k u_S^{n+1} \right) \\
 &\stackrel{(27),(29)}{=} \mathcal{G}_S \left(-{}^k z_F^{n+1}, {}^k u_C^{n+1} \right) \\
 &\stackrel{(23),(25)}{=} \mathcal{G}_S \left(-\mathcal{G}_F \left({}^k y_F^{n+1} \right), \mathcal{G}_C \left({}^k y_C^{n+1} \right) \right) \\
 &\stackrel{(26),(28)}{=} \mathcal{G}_S \left(-\mathcal{G}_F \left({}^k y_S^{n+1} \right), \mathcal{G}_C \left({}^k y_S^{n+1} \right) \right), \\
 \text{i.e. } {}^{k+1}y_S^{n+1} &= {}_1A_{\text{FSCI}} {}^k y_S^{n+1} + b^n.
 \end{aligned} \tag{32}$$

Herein b^n is the part remaining constant during the iteration process.

The limit of the iteration factor

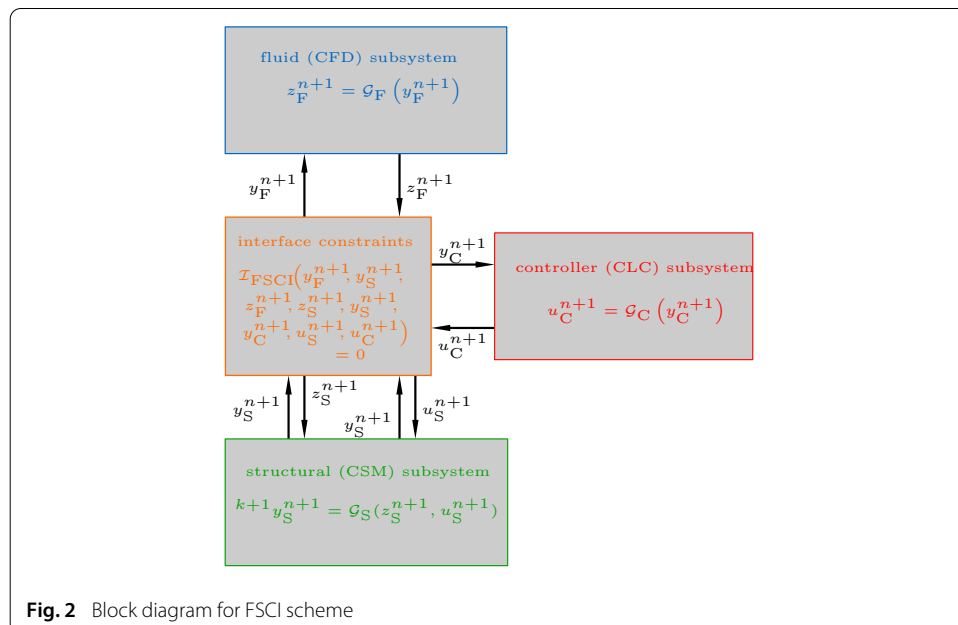
$$\lim_{\delta t \rightarrow 0} \{ {}_1A_{\text{FSCI}} \} = \frac{\alpha - 1}{\alpha}, \tag{33}$$

shows pure dependency on the mass distribution α .

Supplemented by relaxation, the FSCI scheme shown in Eq. (32) is extended to

$$\begin{aligned}
 {}^{k+1}y_S^{n+1} &= \beta \mathcal{G}_S \left(-\mathcal{G}_F \left({}^k y_S^{n+1} \right), \mathcal{G}_C \left({}^k y_S^{n+1} \right) \right) + (1 - \beta) {}^k y_S^{n+1}, \\
 \text{i.e. } {}^{k+1}y_S^{n+1} &= \beta A_{\text{FSCI}} {}^k y_S^{n+1} + \beta b^n.
 \end{aligned} \tag{34}$$

The limit of the iteration factor



$$\lim_{\delta t \rightarrow 0} \{\beta A_{\text{FS}CI}\} = \frac{\alpha - \beta}{\alpha}, \tag{35}$$

is now clearly determined by the mass distribution α and the relaxation parameter β .

The optimal relaxation parameter becomes

$$\beta_{\text{FS}CI}^* = \frac{\alpha m + k \delta t^2}{m + (c + k_{R2}) \delta t + (k + k_{R1}) \delta t^2}. \tag{36}$$

Each term in the denominator is positive, non-zero for physically relevant parameters and stable controller settings according to Eq. (10). Thus, it can always be found.

Nesting of the FSI sub-problem ([FS]CI)

The acronym [FS]CI denotes the specific iterative coupling scheme illustrated in Fig. 3, where the Gauss–Seidel communication pattern is depicted with nesting of the FSI sub-problem, which is indicated by bracketing [FS]. The nesting of sub-problems corresponds to the inclusion of bi-coupling schemes mentioned in [18]. As it can be seen in Fig. 3, two interface constraint equations are to be set up. One for the FS subproblem (inner interface constraints) \mathcal{I}_{FSI} and one for the overall coupling between the FS subsystem with the control subsystem (outer interface constraints) $\mathcal{I}_{[\text{FS}]CI}$. At first the FS loop is iterated with constant control input until convergence. The converged values are used as information for the iterations of the outer loop. If the outer loop does not converge, the algorithm has to return to the inner loop. Before proceeding to the next time step, inner and outer loop have to be converged. The scheme is again applied to the partitioned simplified model problem. Since the pure FSI is solved in its own iteration loop, it is possible to calculate the best relaxation factor once for the FSI problem, involving two coupled fields and for the complete FSCI problem involving three coupled fields.

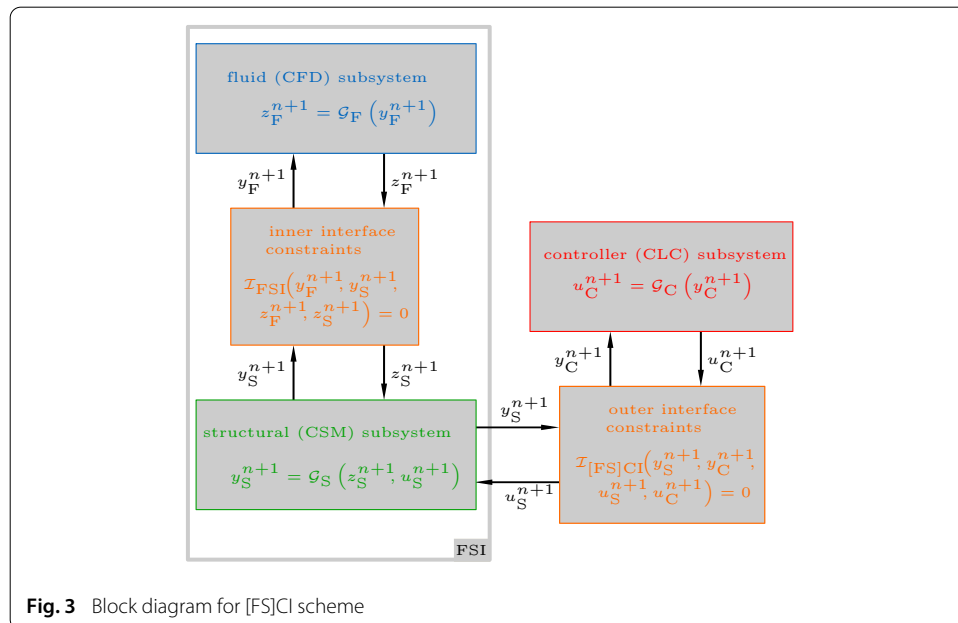


Fig. 3 Block diagram for [FS]CI scheme

The inner FSI fixed-point iteration of the algorithm condenses down to

$$\begin{aligned}
 {}_{l+1}^k y_S^{n+1} &\stackrel{(24)}{=} \mathcal{G}_S \left({}_l^k z_S^{n+1}, {}^k u_S^{n+1} = \text{const.} \right) \\
 &\stackrel{(27)}{=} \mathcal{G}_S \left(-{}_l^k z_F^{n+1}, {}^k u_S^{n+1} = \text{const.} \right) \\
 &\stackrel{(23)}{=} \mathcal{G}_S \left(-\mathcal{G}_F \left({}_l^k y_F^{n+1} \right), {}^k u_S^{n+1} = \text{const.} \right) \\
 &\stackrel{(26)}{=} \mathcal{G}_S \left(-\mathcal{G}_F \left({}_l^k y_S^{n+1} \right), {}^k u_S^{n+1} = \text{const.} \right), \\
 \text{i.e.} \quad {}_{l+1}^k y_S^{n+1} &= {}_1 A_{\text{FSI}} {}_l^k y_S^{n+1} + {}^k b^n.
 \end{aligned} \tag{37}$$

Herein the iteration counter l is used for the inner iteration loop and the iteration counter k for the outer iteration loop. For the inner FSI fixed point iteration the constant part is ${}^k b^n$.

The limit of the inner iteration factor

$$\lim_{\delta t \rightarrow 0} \{ {}_1 A_{\text{FSI}} \} = \frac{\alpha - 1}{\alpha}, \tag{38}$$

shows pure dependency on the mass distribution α .

Supplemented by relaxation the inner FSI part of the scheme reads

$$\begin{aligned}
 {}_{l+1}^k y_S^{n+1} &= \beta \mathcal{G}_S \left(-\mathcal{G}_F({}_l^k y_S^{n+1}), {}^k u_S^{n+1} = \text{const.} \right) + (1 - \beta) {}_l^k y_S^{n+1}, \\
 \text{i.e.} \quad {}_{l+1}^k y_S^{n+1} &= \beta A_{\text{FSI}} {}_l^k y_S^{n+1} + \beta^k b^n.
 \end{aligned} \tag{39}$$

The limit of the inner iteration factor

$$\lim_{\delta t \rightarrow 0} \{ \beta A_{\text{FSI}} \} = \frac{\alpha - \beta}{\alpha}, \tag{40}$$

now is obviously determined by the mass distribution α and the relaxation parameter β .

The optimal relaxation parameter becomes

$$\beta_{\text{FSI}}^* = \frac{\alpha m + k \delta t^2}{m + c \delta t + k \delta t^2}. \tag{41}$$

It can always be found since each term in the denominator is positive non-equal to zero for physically relevant parameters independent of the controller settings.

Assuming convergence, the inner FSI fixed-point iteration can be substituted by the equivalent FS subsystem $\mathcal{G}_{\text{FS}} \left({}^k u_S^{n+1} \right)$ for analyzing the outer [FS]CI fixed-point iteration. Consequently, this outer [FS]CI fixed-point iteration of the algorithm condenses down to

$$\begin{aligned}
 {}^{k+1} y_S^{n+1} &= \mathcal{G}_{\text{FS}} \left({}^k u_S^{n+1} \right) \\
 &\stackrel{(29)}{=} \mathcal{G}_{\text{FS}} \left({}^k u_C^{n+1} \right) \\
 &\stackrel{(25)}{=} \mathcal{G}_{\text{FS}} \left(\mathcal{G}_C \left({}^k y_C^{n+1} \right) \right) \\
 &\stackrel{(28)}{=} \mathcal{G}_{\text{FS}} \left(\mathcal{G}_C \left({}^k y_S^{n+1} \right) \right), \\
 \text{i.e.} \quad {}^{k+1} y_S^{n+1} &= {}_1 A_{[\text{FS}] \text{CI}} {}^k y_S^{n+1} + b^n.
 \end{aligned} \tag{42}$$

Herein the factor b^n remains constant during all iterations. The limit of the outer iteration factor

$$\lim_{\delta t \rightarrow 0} \{ {}_1 A_{[\text{FS}] \text{CI}} \} = 0, \tag{43}$$

is always zero, independently of the parameter setting.

Supplemented by relaxation the outer [FS]CI part of the scheme reads

$$\begin{aligned}
 & k^{+1}y_S^{n+1} = \beta \mathcal{G}_{FS}(\mathcal{G}_C(ky_S^{n+1})) + (1 - \beta)ky_S^{n+1}, \\
 \text{i.e. } & k^{+1}y_S^{n+1} = \beta A_{[FS]CI} k y_S^{n+1} + \beta b^n.
 \end{aligned}
 \tag{44}$$

The limit of the outer iteration factor

$$\lim_{\delta t \rightarrow 0} \{ \beta A_{[FS]CI} \} = 1 - \beta,
 \tag{45}$$

shows pure dependency on the relaxation parameter β .

The optimal relaxation parameter becomes

$$\beta_{[FS]CI}^* = \frac{m + c\delta t + k\delta t^2}{m + (c + k_{R2})\delta t + (k + k_{R1})\delta t^2}.
 \tag{46}$$

Each summand in the denominator is positive and non-equal to zero for physically relevant parameters and stable controller settings according to Eq. (10). Thus, it can always be found.

Nesting of the SCI sub-problem (F[SC]I)

The acronym F[SC]I denotes the specific iterative coupling scheme illustrated in Fig. 4, where the Gauss–Seidel communication pattern is depicted with a nesting of the SCI subproblem, which is made clear by bracketing [SC]. Comparable to the [FS]CI problem, for the F[SC]I problem two interface equations are also to be set up. \mathcal{I}_{SCI} for the inner and $\mathcal{I}_{F[SC]I}$ for the outer iteration loop. As already indicated, the solution procedure for the F[SC]I problem is done just the other way around like in the [FS]CI problem.

Accordingly, first the SC loop is iterated applying a constant disturbance force until convergence. The converged values are used as information for the iterations of the outer loop. If the outer loop does not converge, the algorithm has to return to the inner loop. Before proceeding to the next time step, inner and outer loop have to be converged. The scheme is again applied to the partitioned simplified model problem. The inner SCI fixed-point iteration of the algorithm condenses down to

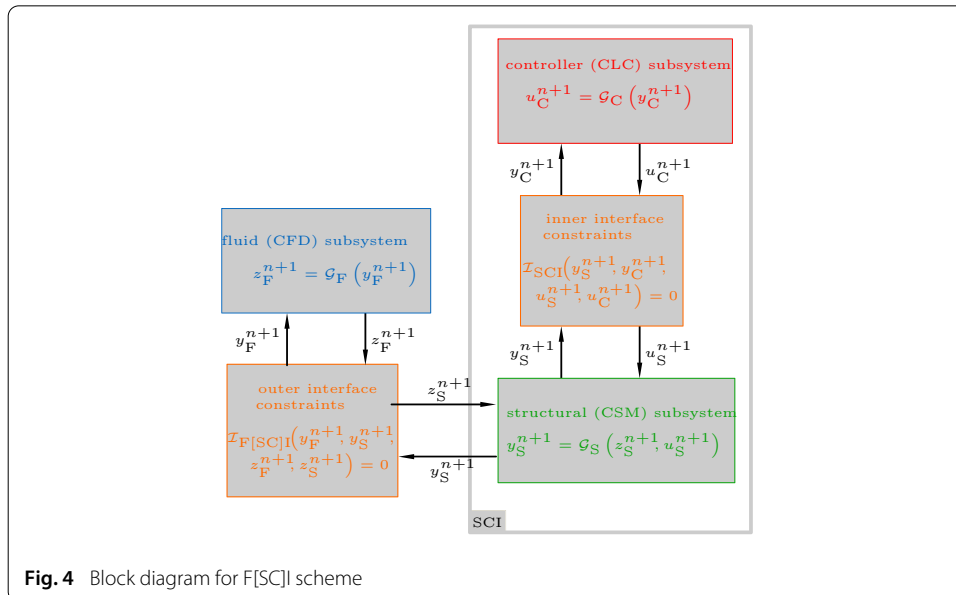


Fig. 4 Block diagram for F[SC]I scheme

$$\begin{aligned}
{}_{l+1}^k y_S^{n+1} &\stackrel{(24)}{=} \mathcal{G}_S \left({}^k z_S^{n+1} = \text{const.}, {}_l^k u_S^{n+1} \right) \\
&\stackrel{(29)}{=} \mathcal{G}_S \left({}^k z_S^{n+1} = \text{const.}, {}_l^k u_C^{n+1} \right) \\
&\stackrel{(25)}{=} \mathcal{G}_S \left({}^k z_S^{n+1} = \text{const.}, \mathcal{G}_C \left({}_l^k y_C^{n+1} \right) \right) \\
&\stackrel{(28)}{=} \mathcal{G}_S \left({}^k z_S^{n+1} = \text{const.}, \mathcal{G}_C \left({}_l^k y_S^{n+1} \right) \right),
\end{aligned} \tag{47}$$

$$\text{i.e. } {}_{l+1}^k y_S^{n+1} = {}_1 A_{\text{SCI}} {}_l^k y_S^{n+1} + {}^k b^n.$$

Again the iteration counter l is used for the inner iteration loop and the iteration counter k for the outer iteration loop. As defined for the FSCI and the [FS]CI problem, ${}^k b^n$ is the constant part of the inner iteration loop.

The limit of the inner iteration factor

$$\lim_{\delta t \rightarrow 0} \{ {}_1 A_{\text{SCI}} \} = 0, \tag{48}$$

is always zero independently of the parameter setting.

Supplemented by relaxation the inner SCI part of the scheme reads

$$\begin{aligned}
{}_{l+1}^k y_S^{n+1} &= \beta \mathcal{G}_S \left({}^k z_S^{n+1} = \text{const.}, \mathcal{G}_C \left({}_l^k y_S^{n+1} \right) \right) + (1 - \beta) {}_l^k y_S^{n+1}, \\
\text{i.e. } {}_{l+1}^k y_S^{n+1} &= \beta A_{\text{SCI}} {}_l^k y_S^{n+1} + \beta^k b^n.
\end{aligned} \tag{49}$$

The limit of the inner iteration factor

$$\lim_{\delta t \rightarrow 0} \{ \beta A_{\text{SCI}} \} = 1 - \beta, \tag{50}$$

shows pure dependency on the relaxation parameter β .

The optimal relaxation parameter becomes

$$\beta_{\text{SCI}}^* = \frac{\alpha m + k \delta t^2}{\alpha m + k_{\text{R2}} \delta t + (k + k_{\text{R1}}) \delta t^2}. \tag{51}$$

αm and $(k + k_{\text{R1}}) \delta t^2$ in the denominator are positive and non-equal to zero for physically relevant parameters and stable controller settings according to Eq. (10).

Thus, the optimal relaxation factor can always be found by additionally requiring

$$k_{\text{R2}} \delta t \neq -(\alpha m + (k + k_{\text{R1}}) \delta t^2).$$

Assuming convergence, the inner SCI fixed-point iteration can accordingly be substituted by the equivalent SC subsystem $\mathcal{G}_{\text{SC}} \left({}^k z_S^{n+1} \right)$ for analyzing the outer F[SC]I fixed-point iteration. Consequently, this outer F[SC]I fixed-point iteration of the algorithm condenses down to

$$\begin{aligned}
{}^{k+1} y_S^{n+1} &= \mathcal{G}_{\text{SC}} \left({}^k z_S^{n+1} \right) \\
&\stackrel{(27)}{=} \mathcal{G}_{\text{SC}} \left(-{}^k z_{\text{F}}^{n+1} \right) \\
&\stackrel{(23)}{=} \mathcal{G}_{\text{SC}} \left(-\mathcal{G}_S \left({}^k y_{\text{F}}^{n+1} \right) \right) \\
&\stackrel{(26)}{=} \mathcal{G}_{\text{SC}} \left(-\mathcal{G}_{\text{F}} \left({}^k y_S^{n+1} \right) \right),
\end{aligned} \tag{52}$$

$$\text{i.e. } {}^{k+1} y_S^{n+1} = {}_1 A_{\text{F[SC]I}} {}^k y_S^{n+1} + b^n.$$

The limit of the outer iteration factor

$$\lim_{\delta t \rightarrow 0} \{ {}_1 A_{\text{F[SC]I}} \} = \frac{\alpha - 1}{\alpha}, \tag{53}$$

shows pure dependency on the mass distribution α .

Supplemented by relaxation the outer F[SC]I part of the scheme reads

$${}^{k+1}y_S^{n+1} = \beta \mathcal{G}_{SC} \left(-\mathcal{G}_F \left({}^k y_S^{n+1} \right) \right) + (1 - \beta) {}^k y_S^{n+1}, \tag{54}$$

i.e. ${}^{k+1}y_S^{n+1} = \beta A_{F[SC]I} {}^k y_S^{n+1} + \beta b^n$.

And the limit of the outer iteration factor

$$\lim_{\delta t \rightarrow 0} \{ \beta A_{F[SC]I} \} = \frac{\alpha - \beta}{\alpha}, \tag{55}$$

now is obviously determined by the mass distribution α and the relaxation parameter β .

The optimal relaxation parameter becomes

$$\beta_{F[SC]I}^* = \frac{\alpha m + k_{R2} \delta t + (k + k_{R1}) \delta t^2}{m + (c + k_{R2}) \delta t + (k + k_{R1}) \delta t^2}. \tag{56}$$

It always exists since the denominator is positive and non-equal to zero for physically relevant parameters and stable controller settings according to Eq. (10).

From the simplified model problem it can be concluded that all three types of the Gauss–Seidel schemes show unconditional stability for reasonable physical parameters and stable controller settings. Furthermore, it was possible to derive optimal relaxation parameters β^* . Thus, the schemes can be subsequently applied to a multi-degree of freedom problem for further investigations, which have not been possible with the simplified model problem.

Numerical results for a multi-degree of freedom FSCI problem

The inspiration for the multi-degree of freedom FSCI experiment were the numerical benchmarks proposed in [19,20, p. 195–197]. Since the investigations should go beyond the study of the pure FSI effects, the experimental setup had to be slightly modified. All in all, the principle arrangement remains the same and can be seen in Fig. 5.

In contrast to the rigid cylinder in [19], a square, as suggested in [20], is placed in the channel. To this rigid square an elastic flag (characters *R* to *E*) is attached. The square and flag are placed asymmetric in the channel in order to stimulate a fast onset of the excitation mechanism depicted in Fig. 7. The phenomenology of the problem is described in [19,20].

In the following we are actively trying to influence the dynamics of the structure by a controller, extending the FSI to the FSCI problem. The main objective of this is to reduce

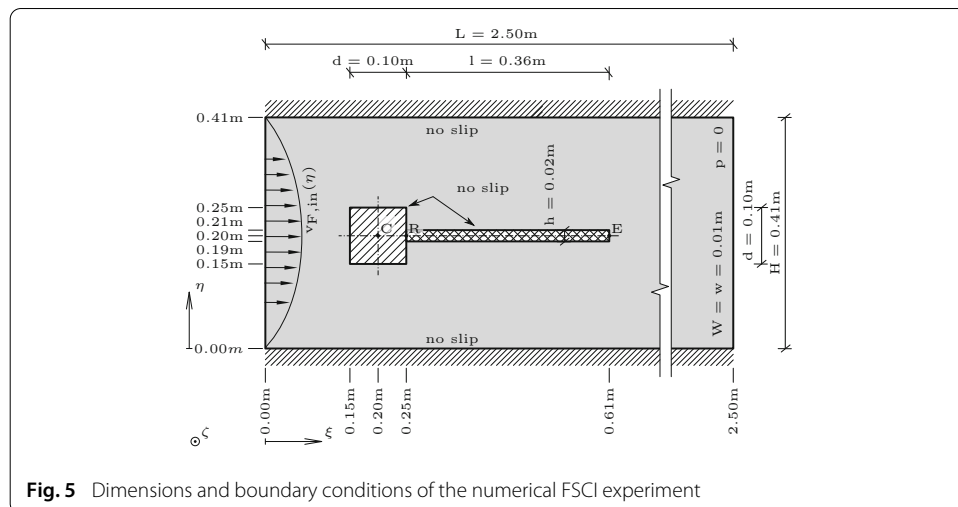


Fig. 5 Dimensions and boundary conditions of the numerical FSCI experiment

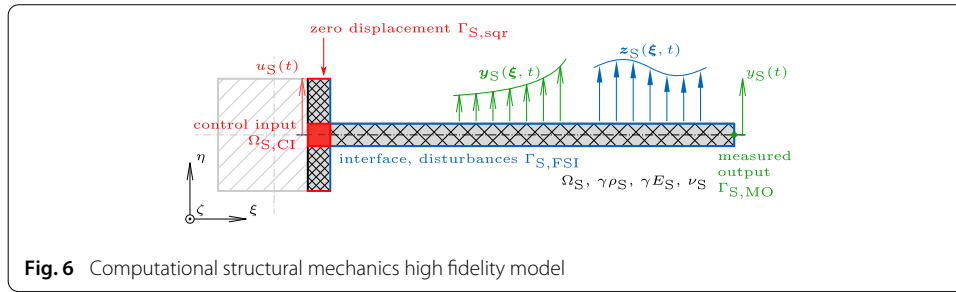


Fig. 6 Computational structural mechanics high fidelity model

or in the best case entirely suppress the amplitude of the end-point displacement at point E . Similar to the simplified model problem, the control input is a Neumann boundary condition, which is applied at the root point of the flag R . For the design of the controller, a reduced order model is necessary [15,21], which is exemplarily shown in Fig. 8. This low fidelity structural model was only used for the controller. In the coupled FSCI simulation the structure was simulated like in classical FSI by the high fidelity model depicted in Fig. 6.

For all simulations of the multi-degree of freedom FSCI problem the open-source software Kratos Multi-Physics [22,23] was used.

Description of the subsystems involved

Just as mentioned in the introduction, the FSCI problem involves three subsystems, namely a fluid flow, a structural mechanics part and a controller.

At first the computational fluid dynamics (CFD) subsystem is introduced. The main dimensions and the boundary conditions of the problem can be seen in Fig. 5. The time constant inlet velocity is described by the function

$$v_{F,in}(\eta) = v_{max}4 \frac{\eta}{H} \left(1 - \frac{\eta}{H}\right). \quad (57)$$

This is a quadratic parabola with v_{max} at its peak value. Herein η is the coordinate running from the bottom of the channel to its width H . The material parameters for the fluid flow are chosen in accordance to the CFD3 specifications in [19], leading to a strongly unsteady flow with vortex shedding. This vortex shedding is additionally supported by the aforementioned eccentric placement of the square in the channel. Thus the following specifications are chosen: $\rho_F = 1000 \text{ kg/m}^3$, $\nu_F = 0.001 \text{ m}^2/\text{s}$ and $v_{in} = 2 \text{ m/s}$, which leads to a Reynolds number $Re = 200$. The fluid flow is discretized by a monolithic finite element formulation with triangular elements developed in [24], using a variational multi-scale (VMS) method for stabilization. In this case, the integration in time is done by a second order backwards differentiation formula (BDF2). The BDF2 scheme approximates the velocity as

$$j^{n+1} = \frac{1}{\delta t} \left(\frac{3}{2}y^{n+1} - 2y^n + \frac{1}{2}y^{n-1} \right). \quad (58)$$

Again j^{n+1} , y^{n+1} , y^n and y^{n-1} denote the velocity and displacements at the discrete time instances t^{n+1} , t^n and t^{n-1} . The time step is set to $\delta t = 0.01 \text{ s}$ for all subsystems. The fluid problem is formulated in arbitrary Lagrangian-Eulerian (ALE) kinematics [25] with a mesh motion algorithm based on [26].

Next, the computational structural mechanics subsystem (CSM) is presented. The CSM subsystem is represented by a high fidelity multi-degree of freedom model which is the initially suggested CSM system as proposed in [19]. The specifications of the high fidelity model can be seen in Fig. 6.

Herein, $z_s(\xi, t)$ is the disturbance force from the fluid flow and $y_S(\xi, t)$ is the displacement of the structure at the interface. Again one can see that the control input $u_S(t)$ is applied only at the root point of the elastic flag and the displacement $y_S(t)$ is measured solely at its tip. The special aspect of the high fidelity model is the back part of the square (finely crosshatched), which is originally assumed to be rigid in [19], but is considered elastic in the current investigation. It is used to linearly distribute the root point excitation along the back side of the square in order to match the ALE boundary conditions of the fluid domain. Therefore a pseudo material with $\nu = 0$ and $\rho = 0$ is set, to avoid artificially introduced deformations and inertia effects at the back of the square. The high fidelity CSM model itself is discretized by a structured mesh of 4-node (2D) non-linear, fully integrated plane stress elements formulated in Total Lagrangian kinematics. In this case the temporal discretization is performed by Newmark's method. The material used for the simulations is a linear St. Venant Kirchhoff material with the parameters $\rho_s = 1000 \text{ kg/m}^3$, $E_S = 5.6 \cdot 10^6 \text{ N/m}^2$ and $\nu_S = 0.4$. The values used for the simulations match the CSM2 benchmark of [19] scaled by the factor of γ . The gravity constant is set to $g = 2 \text{ m/s}^2$ and is acting in ξ direction. Figure 7 shows an extract of the simulation results for one second by using the parameters for fluid flow and structural model described in this subsection ($\gamma = 10^2$) in order to show the deformation mechanism. The figure shows the deformed mesh and the velocity contours. The controller is not activated yet. The displacements at point E with and without activated controller are plotted in Figs. 10 and 11.

The meshes on the interface of fluid and structure subsystem coincide, thus no additional mapping operation is necessary.

The third subsystem consists of a low fidelity CSM model, which is implemented in the controller and the control law itself. In the low fidelity CSM model the overall structural dynamics are condensed to a single degree of freedom system. The low fidelity model can be seen in Fig. 8. It has been derived from the high fidelity multi-degree of freedom model. The structural model itself is approximated by a simple second-order ODE, which matches the boundary conditions of the high fidelity model and is used by the controller to calculate $u_S(t)$. The distributed displacements $y_S(\xi, t)$ between points R and E are approximated by quadratic shape functions, which should be a good enough assumption for the dominant mode shape of the investigated problem (see Fig. 7). They are defined as $y_u(\xi) = 1 - \left(\frac{\xi - \xi_R/l}{l}\right)^2$ for the control input $u_S(t)$ and $y_x(\xi) = \left(\frac{\xi - \xi_R/l}{l}\right)^2$ for the state variable $x_S(t)$. Thus, the real physics of the high fidelity model reduces to

$$(\gamma m)\ddot{x}_S(t) + (\gamma k)x_S(t) = (\gamma b_0)u_S(t) + z_S(t). \quad (59)$$

In the latter equation the single state $x_S(t)$ directly corresponds to the measured output $y_S(t)$ at the end point E resulting in $y_S(t) = x_S(t)$. The parameter b_0 , which is associated with the control input at the root point R , $u_S(t)$, is used to replace the root point excitation, which avoids introducing an additional degree of freedom. Applying the principle of virtual

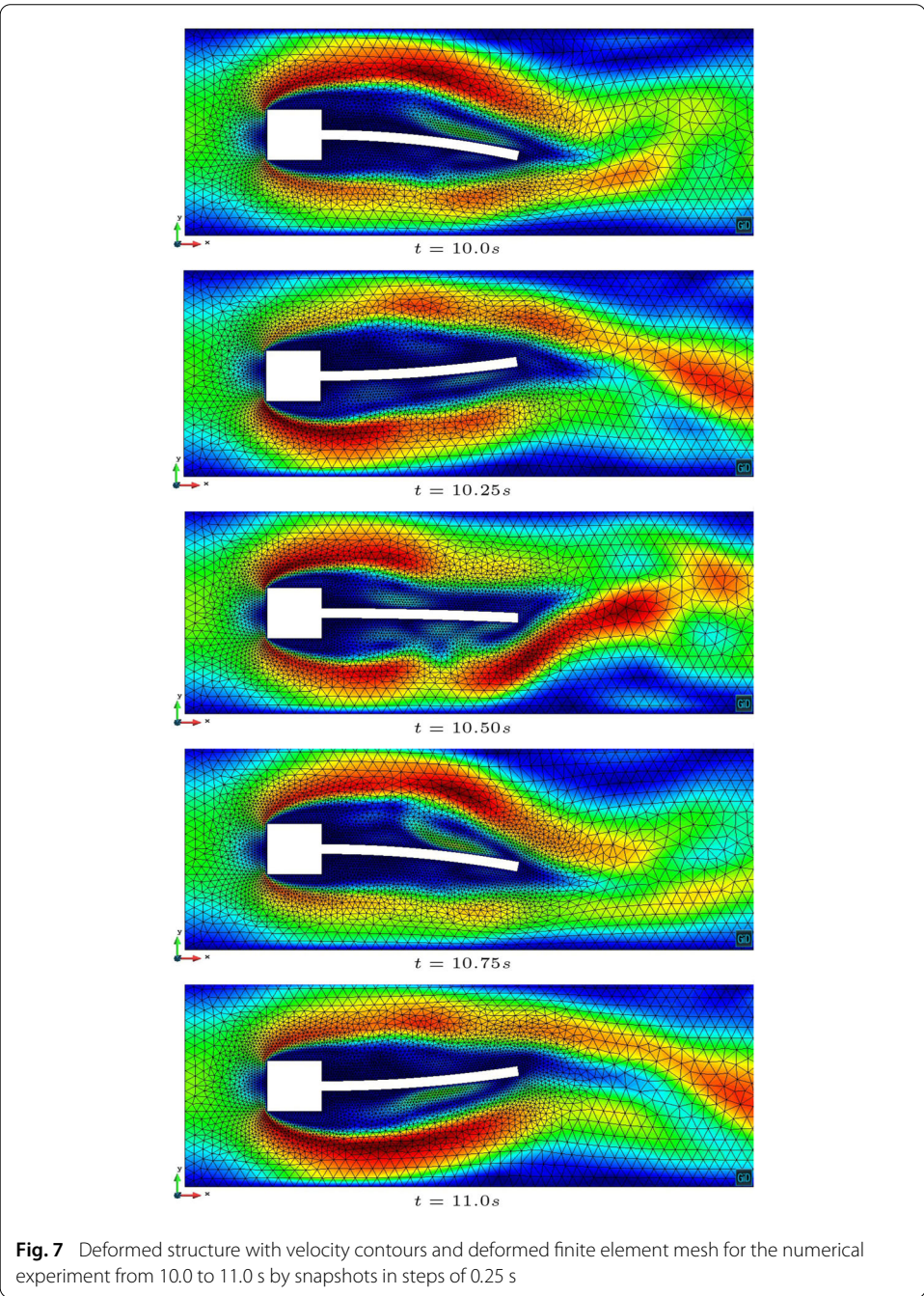


Fig. 7 Deformed structure with velocity contours and deformed finite element mesh for the numerical experiment from 10.0 to 11.0 s by snapshots in steps of 0.25 s

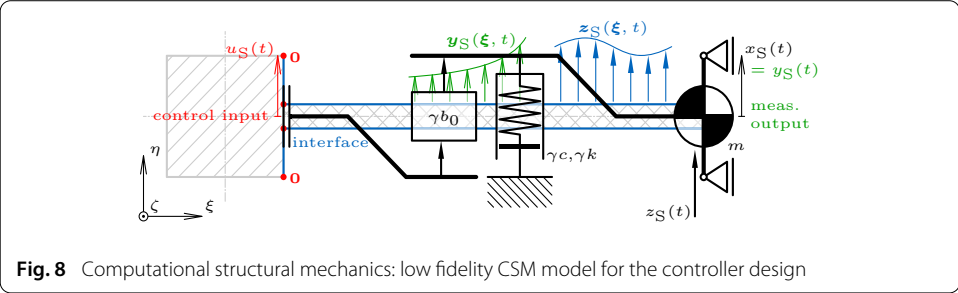


Fig. 8 Computational structural mechanics: low fidelity CSM model for the controller design

Table 1 Values selected for controller parameters

Component	Parameter	Value	Unit
Controlled system	m	0.0144	kg
	c	0	N s/m
	k	2.400549	N/m
	b_0	-1.600366	N/m
	e	0.01	1
State-feedback	Q	I	-
	r	1	-
	k_{R1}	-0.3028	1
	k_{R2}	-1.0027	s
State observer	λ_B	-2000	rad/s
	k_B	2000	rad/s

work (PvW) with a distributed cross sectional mass $\mu = \rho wh$, where w is the width, h the height of the cross section and with \tilde{EI} is the distributed sectional stiffness, leads to

$$\begin{aligned} & \mu \int_{\xi_R}^{\xi_{R+1}} y_x^2(\xi) d\xi \ddot{x}_S(t) + \tilde{EI} \int_{\xi_R}^{\xi_{R+1}} y_x^2(\xi) d\xi x_S(t) \\ & = \mu \int_{\xi_R}^{\xi_{R+1}} y_x(\xi) y_u(\xi) d\xi \ddot{u}_S(t) \\ & \quad + \tilde{EI} \int_{\xi_R}^{\xi_{R+1}} y_x(\xi) y_u(\xi) d\xi u_S(t) + \sum_i y_x(\xi_i) z_{\eta S}(\xi_i, t). \end{aligned} \tag{60}$$

The last term of Eq. (60) denotes the discrete disturbance forces coming from the nodes i on the interface mesh of the fluid domain which are to be summed up here. The open parameters in Eq. (59) can be obtained as:

$$\begin{aligned} m &= \frac{\mu l}{5} = \frac{\rho whl}{5}, k = \frac{\tilde{EI} l}{5} = \frac{E_S wh^3}{4l^3}, \\ b_0 &= -\tilde{EI} \frac{2l}{15} = \frac{E_S wh^3}{6l^3}, z_S(t) = \sum_i \left(\frac{\xi_i - \xi_R}{l} \right)^2 z_{\eta S}(\xi_i, t). \end{aligned} \tag{61}$$

Those approximations are applied to the centerline of the structure and have to be projected to the surfaces of the flag by appropriate projection operations. The time discrete low fidelity CSM model is finally given with the adapted time discretization Eq. (24) from the simplified model problem. The equivalent values to match the multi-degree of freedom model for m , k and b_0 can be found in Table 1. Those parameters can be scaled by the parameter γ accordingly.

With the low fidelity CSM model, the controller can be designed. In this case, a state-feedback control following state observer is implemented, which is state of the art for modern methods for controller design and is also used in the context of many applications in control theory. Herein, the controller state feedback matrix is specified via a linear-quadratic regulator approach (LQR) and the observer output-feedback matrix is set via pole placement as generally described in [15,21]. The block diagram can be seen in Fig. 9. The controlled system, which is seen by the controller, is represented by the equation

$$(\gamma m)\ddot{y}(t) + (\gamma c)\dot{y}(t) + (\gamma k)y(t) = (\gamma b_0)u(t) + ez(t), \tag{62}$$

including the measured output $y(t)$, the control input $u(t)$, the disturbance force $z(t)$ and the scaling factor γ . For a parameter exploration to design the controller, the disturbance

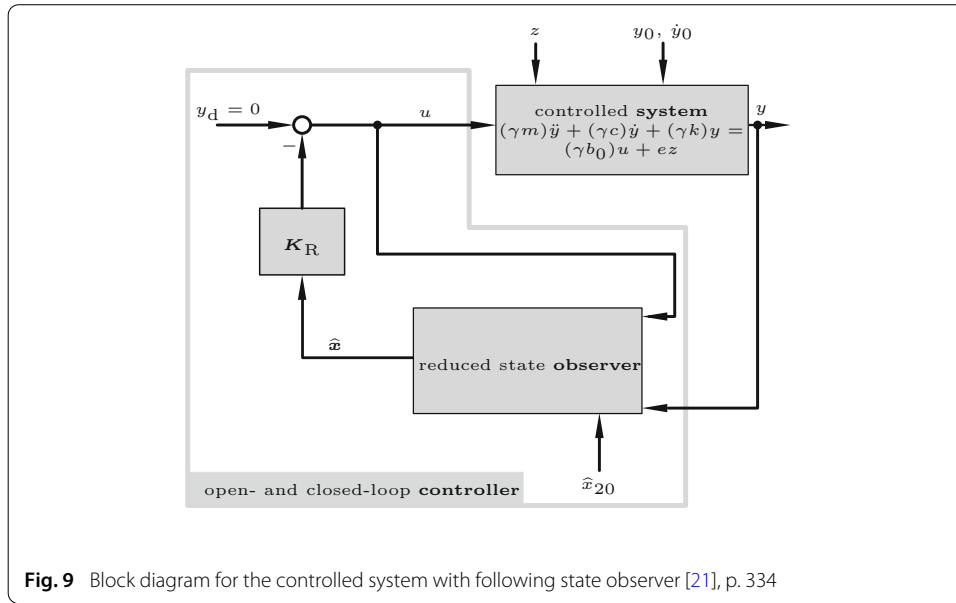


Fig. 9 Block diagram for the controlled system with following state observer [21], p. 334

force $z(t)$ is applied from recorded FSI simulations of the system. The system is rewritten in state-space representation, with $x_1 := y(t)$ and $x_2 := \dot{y}(t)$ being the entries in the state vector $\mathbf{x} = [x_1, x_2]^T$, reading

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{b_0}{m} \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ \frac{e}{m} \end{bmatrix} z(t), \tag{63}$$

i.e. $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u(t) + \mathbf{E}z(t)$

and the output equation

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \tag{64}$$

i.e. $y(t) = \mathbf{C}\mathbf{x}$.

where \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{E} are constant matrices. Since the system is fully controllable and fully observable, state-feedback control and state observer are possible. Thus a control law similar to the one presented for the model problem can be used. This is given by

$$u(t) = -\mathbf{K}_R\mathbf{x}, \tag{65}$$

with $\mathbf{K}_R = [k_{R1} k_{R2}]$ being the constant state feedback matrix. Herein the constants in \mathbf{K}_R are determined with the LQR approach described in detail in [21, chapter 7]. This involves user-definable weights $\mathbf{Q} \in \mathbb{R}^{2,2}$ related to state the \mathbf{x} and $r \in \mathbb{R}^{1,1}$ for control input $u(t)$. With an appropriate choice of $\mathbf{Q} \neq \mathbf{Q}(\gamma)$ and $r \neq r(\gamma)$ the state-feedback matrix becomes independent of γ , because also $\mathbf{A} \neq \mathbf{A}(\gamma)$, $\mathbf{B} \neq \mathbf{B}(\gamma)$ and $\mathbf{C} \neq \mathbf{C}(\gamma)$.

Since the displacement $y(t)$, being state one, should be measured during the simulations, it is directly accessible ($x_1 = y(t)$). The second state should not be measured directly and thus needs an approximation. This approximation $x_2 \approx \hat{x}_2$ is done from measurements of state one by a reduced state observer. The state space representation therefore can be split up into one part for the measurement $y(t)$ and a second one for the estimation $\hat{\mathbf{x}}$ in the following way:

$$\begin{aligned} \hat{\mathbf{x}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}u, \\ \text{i.e. } \begin{bmatrix} \dot{y}(t) \\ \hat{\mathbf{x}}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} y(t) \\ \hat{\mathbf{x}}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{b_0}{m} \end{bmatrix} u(t). \end{aligned} \quad (66)$$

Conducting the observer design and applying the BDF2 scheme results in

$$\begin{aligned} x_1^{n+1} &= y^{n+1}, \\ [\text{LHS}] \tilde{x}_2^{n+1} &= -\frac{1}{\delta t} \left(-2\tilde{x}_2^n + \frac{1}{2}\tilde{x}_2^{n-1} \right) + [\text{RHS}] y^{n+1}, \\ \hat{x}_2^{n+1} &= \tilde{x}_2^{n+1} + k_B y^{n+1}, \\ u^{n+1} &= -k_{R1} x_1^{n+1} - k_{R2} \hat{x}_2^{n+1}, \end{aligned} \quad (67)$$

where

$$\begin{aligned} [\text{LHS}] &= \frac{3}{2\delta t} - a_B + b_B k_{R2} = \frac{3}{2\delta t} + \frac{c}{m} + k_B + \frac{b_0}{m} k_{R2}, \\ [\text{RHS}] &= e_B - b_B (k_{R1} + k_{R2} k_B), \\ &= -\left(\frac{c}{m} + k_B \right) k_B - \frac{k}{m} - \frac{b_0}{m} (k_{R1} + k_{R2} k_B) \end{aligned} \quad (68)$$

and \tilde{x}_2 results from the design of the reduced state observer.

For multi-degree of freedom problems, like the one presented in this section, it is not possible to derive an optimal relaxation parameter β^* [11, p. 769] as was done for the simplified model problem. Thus in this paper it is computed by Aitken acceleration as proposed for FSI problems in [8, 11].

Residual calculation and numerical accuracy

The overall coupled partitioned FSCI problem as well as the fluid subsystem were solved by an iterative approach. For the structural and the controller subsystem a direct solver was used. For iterative solution procedures the residual calculation and the accuracy of the solution play an important role in order to gain correct results [2, 27, p. 201]. In the following a closer look is taken to the iterative solution of the interface equation system. For FSI [28] shows that in order to achieve the desired accuracy for the coupled problem using an iterative approach, the numerical accuracy of the solution of the subsystems has to be at least two orders of magnitude higher than the desired numerical accuracy of the coupled system. Thus, it makes sense to use the outcome of those investigations also for the FSCI problem.

Another crucial part is the calculation of the residual of the interface equation system. Since we are dealing with Dirichlet–Neumann coupling, it is obvious to calculate the residual vector \mathcal{R}_y from the interface displacements, which correspond to the structural displacements y_S of the high fidelity model. This means

$${}^k \mathcal{R}_y = {}^k y_S - {}^{k-1} y_S. \quad (69)$$

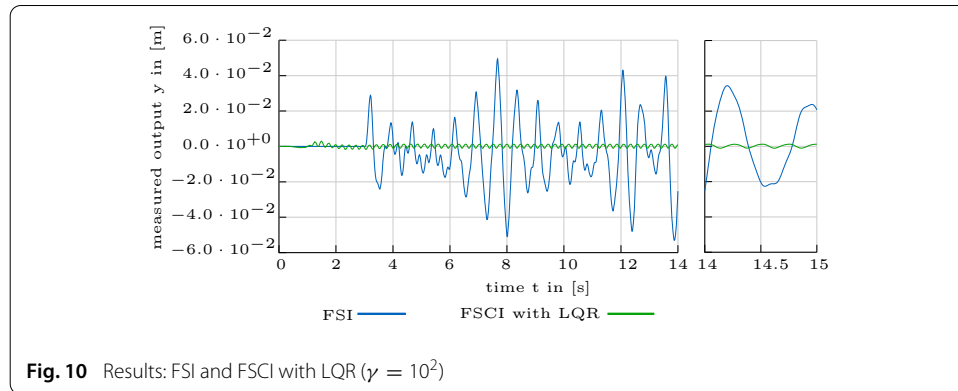
The convergence at the interface is achieved if

$$\left\| {}^k \mathcal{R}_y \right\| / \sqrt{n_{dof}} \leq \varepsilon_{\mathcal{I}}. \quad (70)$$

where $\|\dots\|$ denotes the L^2 norm of the residual vector and $\varepsilon_{\mathcal{I}}$ is the desired accuracy on the interface. The index k denotes the iteration counter. The residual is normalized by the square root of the number of degrees of freedom on the interface n_{dof} [28]. For the results in the following simulations the numerical accuracy criteria can be found in Table 2. In this

Table 2 Recommendation for the numerical accuracy of solvers and interface iterations on basis of [28]

	FSI	FSCI	[FS]CI	F[SC]I
ε_F	$10^{-(p+2)}$	$10^{-(p+2)}$	$10^{-(p+4)}$	$10^{-(p+4)}$
ε_S	$10^{-(p+2)}$	$10^{-(p+2)}$	$10^{-(p+4)}$	$10^{-(p+4)}$
$\varepsilon_{\mathcal{I}}^{\text{inner}}$	10^{-p}	10^{-p}	$10^{-(p+2)}$	$10^{-(p+2)}$
$\varepsilon_{\mathcal{I}}^{\text{outer}}$	–	–	10^{-p}	10^{-p}

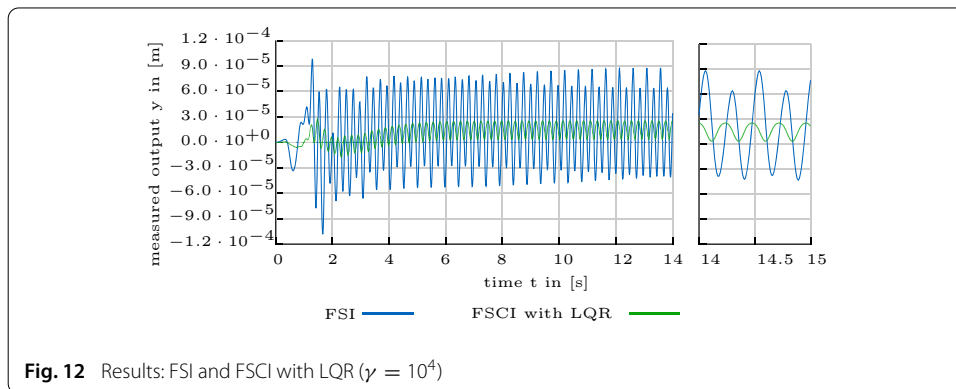
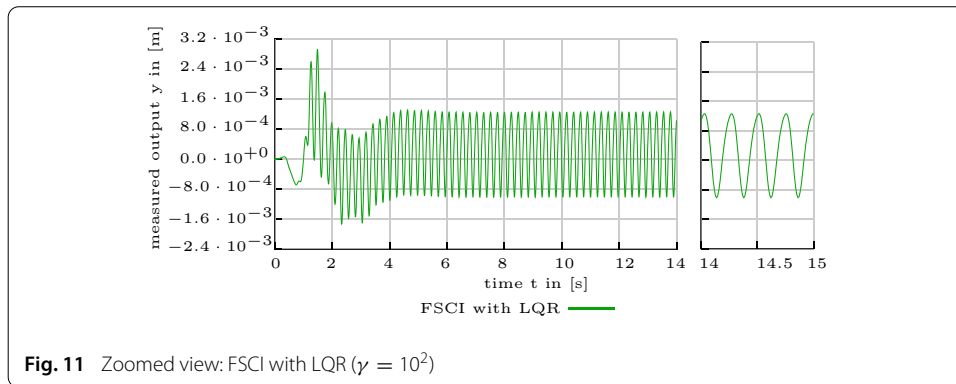


table, the proportions of values for the stopping criteria of the coupled simulations for the different variants of partitioned simulation patterns are listed. At first the overall desired numerical accuracy, which finally is to be achieved for the overall coupled simulation, was selected in this case to be $p = 7$, resulting in a value of $\varepsilon_{\mathcal{I}} = 10^{-7}$ for the interface iterations. Afterwards the values of ε_F for the fluid solver and ε_S for the structural solver as well as for the inner interface iteration loop $\varepsilon_{\mathcal{I}}^{\text{inner}}$ and the outer interface iteration loop $\varepsilon_{\mathcal{I}}^{\text{outer}}$ were adopted according to the criteria described above.

Presentation and interpretation of the results

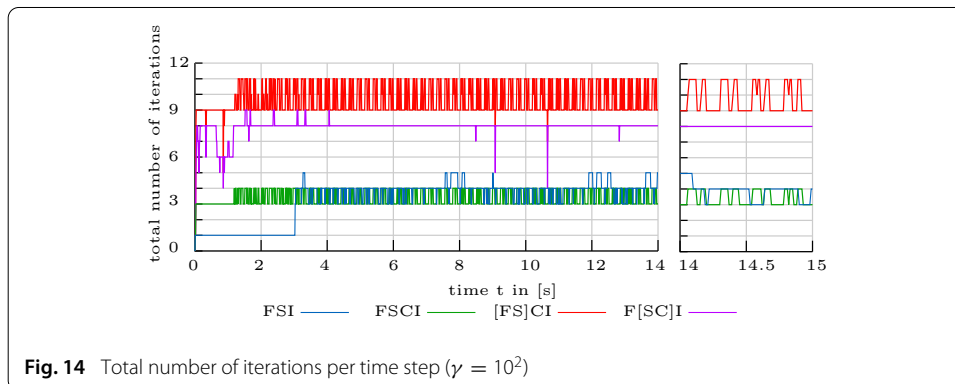
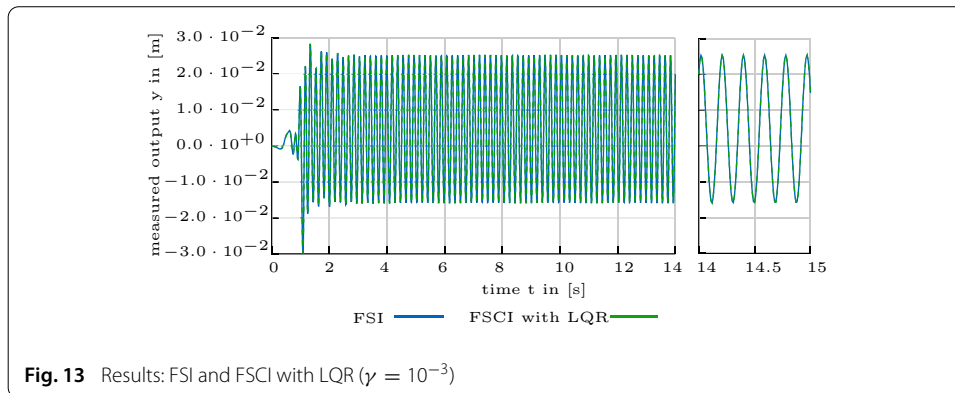
The simulations were conducted for 15 s and the measured output, i.e. the tip displacement (point *E*) of the elastic flag, has been plotted. The result for a pure FSI and a FSCI simulation with no nesting for a scaling factor $\gamma = 10^2$ and $\gamma = 10^4$ can be seen in Figs. 10 and 12. Additionally the results for the controlled system can be seen in an amplified version for $\gamma = 10^2$ in Fig. 11. One can see that the controller applied to the root point *R* of the flag is able to reduce the magnitude of the tip displacement at point *E* in the order of magnitude of almost 10^2 . In Figs. 10, 11, 12 and 13 the horizontal axis represents evolution in time and is subdivided into divisions of two seconds for the time interval from 0 to 14 s and is stretched for the time interval from 14 to 15 s. The vertical axis represents the measured output corresponding to the tip displacement of the flag. The vertical axes in Figs. 10 and 11 have a different scaling, but both figures show the same results for FSCI with LQR. Furthermore, the remaining oscillation in Fig. 11 is more regular than the one measured from the pure FSI simulation.

Although, the parameter setting is chosen to be optimal for the $\gamma = 10^2$, the controller still works for the value of $\gamma = 10^4$ since the structure behaves more inertial and is stiffer. Regarding the eigenvalues of the coupled system this means that the eigenfrequencies of the structure are dominating the overall behavior. It also follows that the number of interface iterations will decrease. By reducing the factor of γ , the added mass effect comes



more into play and the system becomes softer. This means the eigenfrequencies of the fluid flow start to dominate the system behavior and the nonlinear behavior of the coupled system becomes stronger. It follows that a new controller design has to be conducted. For a value of $\gamma \leq 10^{-2}$ the structure is too soft and light to be able to control its behavior by a force applied at its rootpoint R . The results for such an example can be seen in Fig. 13.

In Fig. 14 the overall number of interface iterations per time step for the different schemes applied to the numerical test example are plotted for the factor $\gamma = 10^2$ for the different coupling variants are plotted. For example in the case of F[SC]I, we count the number of interface iterations for the [SC] loop (inner loop) first and afterwards add the number of interface iterations between [SC] and F (outer loop). It can be seen that the FSCI scheme needs a similar number of iterations as the pure FSI. One reason for this can be for example seen in Table 2. For the FSCI problem the number of coupling loops is the same as for the pure FSI problem and thus also the total number of interface iterations is almost the equal. Looking closer at the number of interface iterations one can see, that the FSCI even needs less iterations at certain points than the pure FSI. This is, due to the fact that the controller stabilizes the system leading to a more homogeneous oscillation with a smaller amplitude. Using the F[SC]I scheme increases the number of iterations almost by a factor of three and using the [FS]CI by almost a factor of four. According to Table 2 an additional loop and also a higher numerical accuracy in the solvers of the subsystems is needed and the number of overall interface iterations increases. The reason that the F[SC]I requires fewer interface iterations than the [FS]CI version is the stabilizing behavior of the controller, which influences the outer FS loop in a positive way. Conversely, the contrary effect occurs for the [FS]CI scheme.



Conclusion and outlook

Within this paper, the algorithmic treatment and solution approaches of fluid–structure–control interaction (FSCI) with iterative Gauss–Seidel schemes is discussed. The aim was to conduct a fully coupled co-simulation of the FSCI problem, with the controller actively influencing the dynamics of the system.

The simplified model problem utilized in a first step is representative of the convergence behavior and stability for structural force control. Thus, all three developed variants of the Gauss–Seidel scheme prove unconditional stability for the simplified model problem in case of physically relevant parameters and stable controller settings. Furthermore, an optimal relaxation factor β^* could be determined for the simplified model problem. Hence, the simplified model problem is qualitatively capable of constituting the basic properties of the FSCI problem concerning stability and convergence. For the non-linear multi-degree of freedom problem this means one can conclude from the simplified model problem that it should be possible to reduce the displacement significantly by applying a controller with an appropriate set of controller parameters. Since Aitken acceleration is utilized instead of a constant relaxation factor, the schemes developed are supposed to converge in the multi-degree of freedom case. However, since the simplified model problem only covers the main effects of the FSCI problem (e.g. the added mass effect), no detailed and quantitative conclusions concerning convergence patterns and stability issues can be drawn for the multi-degree of freedom problem.

Applying the variants of the Gauss–Seidel scheme, developed by the simplified model problem, to a non-linear multi-degree of freedom problem in a second step showed good results. The flow induced vibrations could be reduced significantly. Simulations for differ-

ent material parameter settings of the CSM subsystem also have shown the limits of the chosen controller type and design. Furthermore they show the limits for controllability of this kind of system by applying a force at the root point of the flag. Investigating the total number of iterations per time step illustrates, that the FSCI scheme with no nesting is the best variant for controlling the selected multi-degree of freedom problem presented in this contribution.

As an outlook, more advanced control laws could be applied to the multi-degree of freedom problem in order to reduce the oscillation even more. Furthermore, the number of iterations could be reduced by using more sophisticated coupling schemes like for example presented in [4,29,30].

Abbreviations

FSI: fluid–structure interaction; FSCI: fluid–structure–control interaction; ODE: ordinary differential equation; SDoF: single degree of freedom system; BDFN: backwards differentiation formula of order N; BIBO: bounded input bounded output; SCI: structure–control interaction; VMS: variational multi-scale; ALE: arbitrary Lagrangian Eulerian; CFD: computational fluid dynamics; CSM: computational structural mechanics; PvW: principle of virtual work; LQR: linear quadratic regulator .

Authors' contributions

AW, CL, K-UB and RW have prepared the manuscript. All authors read and approved the final manuscript.

Author details

¹Chair of Structural Analysis, Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany, ²Chair of Automatic Control Technical University of Munich, Boltzmannstr. 15, 85748 Garching/Munich, Germany.

Acknowledgements

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

The software used is [23].

Consent for publication

Non-exclusive rights of use.

Ethics approval and consent to participate

Not applicable.

Funding

Not applicable.

Appendix

The appendix shows the algorithms developed for FSCI, [FS]CI and F[SC]I throughout this paper and are written here in pseudocode notation, in order to facilitate an own implementation by the reader.

Algorithm 1: Pseudocode for the partitioned FSCI scheme (no nesting).

```

// initialize states, i.e. set ICs...
 ${}^{k_{\text{end}}}\mathbf{x}_F^0 \leftarrow \mathbf{x}_F^{\text{init}}, {}^{k_{\text{end}}}\mathbf{x}_S^0 \leftarrow \mathbf{x}_S^{\text{init}}, {}^{k_{\text{end}}}\mathbf{x}_C^0 \leftarrow \mathbf{x}_C^{\text{init}}$ 

// initialize displacements and measured output...
 ${}^0\mathbf{y}_S^0 \leftarrow \mathbf{y}_S^{\text{init}}, {}^{k_{\text{end}}}\mathbf{y}_S^0 \leftarrow \mathbf{y}_S^{\text{init}}$ 
// time loop...
for  $n \leftarrow 0$  to  $n \leftarrow n_{\text{end}} - 1$  do
  // predict displacements and measured output...
   ${}^0\mathbf{y}_S^{n+1} \leftarrow {}^{k_{\text{end}}}\mathbf{y}_S^n, {}^0\mathbf{y}_S^{n+1} \leftarrow {}^{k_{\text{end}}}\mathbf{y}_S^n$ 

  // interface iteration loop, i.e. FSCI loop...
  for  $k \leftarrow 0$  to  $k \leftarrow k_{\text{max}}$  do
    // map displacements from solid to fluid...
    // and copy measured output from solid to controller...
     ${}^k\mathbf{y}_F^{n+1} \leftarrow \mathcal{M}_y({}^k\mathbf{y}_S^{n+1})$ 
     ${}^k\mathbf{y}_C^{n+1} \leftarrow {}^k\mathbf{y}_S^{n+1}$ 

    // solve fluid and controller in parallel...
     ${}^k\mathbf{z}_F^{n+1} \leftarrow \mathcal{G}_F^{[k\mathbf{x}_F^{n+1}]}({}^k\mathbf{y}_F^{n+1})$ 
     ${}^k\mathbf{u}_C^{n+1} \leftarrow \mathcal{G}_C^{[k\mathbf{x}_C^{n+1}]}({}^k\mathbf{y}_C^{n+1})$ 

    // map forces from fluid to solid...
    // and copy control input from controller to solid...
     ${}^k\mathbf{z}_S^{n+1} \leftarrow \mathcal{M}_z({}^k\mathbf{z}_F^{n+1})$ 
     ${}^k\mathbf{u}_S^{n+1} \leftarrow {}^k\mathbf{u}_C^{n+1}$ 

    // solve solid...
     $\begin{bmatrix} {}^k\mathbf{y}_S^{n+1} \\ {}^k\mathbf{y}_S^{n+1} \end{bmatrix} \leftarrow \mathcal{G}_S^{[k\mathbf{x}_S^{n+1}]}({}^k\mathbf{z}_S^{n+1}, {}^k\mathbf{u}_S^{n+1})$ 

    // calculate residuum of displacements and measured output...
     ${}^k\mathcal{R}_y^{n+1} \leftarrow {}^k\mathbf{y}_S^{n+1} - {}^{k-1}\mathbf{y}_S^{n+1}$ 
     ${}^k\mathcal{R}_y^{n+1} \leftarrow {}^k\mathbf{y}_S^{n+1} - {}^{k-1}\mathbf{y}_S^{n+1}$ 
     ${}^k\mathcal{R}_{y,y}^{n+1} := [{}^k\mathcal{R}_y^{n+1} \quad {}^k\mathcal{R}_y^{n+1}]$ 

    // check for convergence ...
     ${}^k\epsilon^{n+1} \leftarrow \|\mathcal{R}_{y,y}^{n+1}\| / \sqrt{n_{\text{dof}}}$ 
    if  ${}^k\epsilon^{n+1} < \max_{\epsilon}$  then
      | break
    end
    // update Aitken factor
    if  $k = 0$  then
      |  ${}^0\beta^{n+1} \leftarrow \text{init}\beta$ 
    else
      |  ${}^k\beta^{n+1} \leftarrow {}^{k-1}\beta^{n+1} \frac{{}^{k-1}\mathcal{R}_{y,y}^{n+1\text{T}} ({}^{k-1}\mathcal{R}_{y,y}^{n+1} - {}^k\mathcal{R}_{y,y}^{n+1})}{\|{}^{k-1}\mathcal{R}_{y,y}^{n+1} - {}^k\mathcal{R}_{y,y}^{n+1}\|^2}$ 
    end

    // update displacements and measured output...
     ${}^{k+1}\mathbf{y}_S^{n+1} \leftarrow {}^k\mathbf{y}_S^{n+1} + {}^k\beta^{n+1} {}^k\mathcal{R}_y^{n+1}$ 
     ${}^{k+1}\mathbf{y}_S^{n+1} \leftarrow {}^k\mathbf{y}_S^{n+1} + {}^k\beta^{n+1} {}^k\mathcal{R}_y^{n+1}$ 
  end
end

```

Algorithm 2: Pseudocode for the partitioned [FS]CI scheme (nesting of [FS] subsystem).

```

// initialize states, i.e. set ICs...
 $m_{\text{end}}^{k_{\text{end}}} \mathbf{x}_F^0 \leftarrow \mathbf{x}_F^{\text{init}}, m_{\text{end}}^{k_{\text{end}}} \mathbf{x}_S^0 \leftarrow \mathbf{x}_S^{\text{init}}, m_{\text{end}}^{k_{\text{end}}} \mathbf{x}_C^0 \leftarrow \mathbf{x}_C^{\text{init}}$ 

// initialize displacements and measured output...
 $m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^0 \leftarrow \mathbf{y}_S^{\text{init}}, m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^0 \leftarrow \mathbf{y}_S^{\text{init}}$ 
// time loop...
for  $n \leftarrow 0$  to  $n \leftarrow n_{\text{end}} - 1$ 
  // predict displacements and measured output...
   $m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^{n+1} \leftarrow m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^n, m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^{n+1} \leftarrow m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^n$ 

  // outer interface iteration loop, i.e. [FS]CI loop...
  for  $k \leftarrow 0$  to  $k \leftarrow k_{\text{max}}$ 
    // copy measured output from solid to controller...
     $k \mathbf{y}_C^{n+1} \leftarrow m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^{n+1}$ 

    // solve controller...
     $k \mathbf{u}_C^{n+1} \leftarrow \mathcal{G}_C^{[k \mathbf{x}_C^{n+1}]}(k \mathbf{y}_C^{n+1})$ 

    // copy control input from controller to solid...
     $k \mathbf{u}_S^{n+1} \leftarrow k \mathbf{u}_C^{n+1}$ 

    // predict displacements...
     $k \mathbf{y}_S^{n+1} \leftarrow m_{\text{end}}^{k_{\text{end}}} \mathbf{y}_S^{n+1}$ 

    // inner interface iteration loop, i.e. FSI loop...
    for  $m \leftarrow 0$  to  $m \leftarrow m_{\text{max}}$ 
      // map displacements from solid to fluid...
       $m \mathbf{y}_F^{n+1} \leftarrow \mathcal{M}_y(k \mathbf{y}_S^{n+1})$ 

      // solve fluid...
       $m \mathbf{z}_F^{n+1} \leftarrow \mathcal{G}_F^{[m \mathbf{x}_F^{n+1}]}(m \mathbf{y}_F^{n+1})$ 

      // map forces from fluid to solid...
       $m \mathbf{z}_S^{n+1} \leftarrow \mathcal{M}_z(m \mathbf{z}_F^{n+1})$ 

      // solve solid...
       $\begin{bmatrix} k \mathbf{y}_S^{n+1} \\ k \mathbf{y}_S^{n+1} \\ m \mathbf{y}_S^{n+1} \end{bmatrix} \leftarrow \mathcal{G}_S^{[m \mathbf{x}_S^{n+1}]}(m \mathbf{z}_S^{n+1}, k \mathbf{u}_S^{n+1})$ 

      // calculate residuum of displacements...
       $m \mathcal{R}_y^{n+1} \leftarrow k \mathbf{y}_S^{n+1} - m_{-1} \mathbf{y}_S^{n+1}$ 
      // check for inner convergence...
       $m \varepsilon^{n+1} \leftarrow \left\| \frac{k \mathcal{R}_y^{n+1}}{m \mathcal{R}_y^{n+1}} \right\| / \sqrt{m_{\text{dof}}}$ 
      if  $m \varepsilon^{n+1} < \max \varepsilon$  then
        break
      end

      // update inner Aitken factor...
      if  $m = 0$  then
         $m \beta^{n+1} \leftarrow \text{init} \beta$ 
      else
        if  $\dim \{m \mathcal{R}_y^{n+1}\} = 1$  then
           $m \beta^{n+1} \leftarrow m_{-1} \beta^{n+1} \frac{k \mathcal{R}_y^{n+1}}{m_{-1} \mathcal{R}_y^{n+1} - k \mathcal{R}_y^{n+1}}$ 
        else
          end
        end
      end
    end
  end
end

```

Pseudocode for the partitioned [FS]CI scheme.(continued).

```

|
|   
$${}^k_m\beta^{n+1} \leftarrow {}^{k-1}_m\beta^{n+1} \frac{{}^{k-1}_m\mathcal{R}_y^{n+1\text{T}} ({}^{k-1}_m\mathcal{R}_y^{n+1} - {}^k_m\mathcal{R}_y^{n+1})}{\|{}^{k-1}_m\mathcal{R}_y^{n+1} - {}^k_m\mathcal{R}_y^{n+1}\|^2}$$

|   // update displacements...
|   
$${}^{k+1}_m\mathbf{y}_S^{n+1} \leftarrow {}^k_m\mathbf{y}_S^{n+1} + {}^k_m\beta^{n+1} {}^k_m\mathcal{R}_y^{n+1}$$

|   end
|
|   // calculate residuum of measured output...
|   
$${}^k_m\mathcal{R}_y^{n+1} \leftarrow {}^{k-1}_m\mathbf{y}_S^{n+1} - {}^{k-1}_m\mathbf{y}_S^{n+1}$$

|   // check for outer convergence...
|   
$${}^k_m\varepsilon^{n+1} \leftarrow \|{}^k_m\mathcal{R}_y^{n+1}\| / \sqrt{n_{dof}}$$

|   if  ${}^k_m\varepsilon^{n+1} < \max \varepsilon$  then
|   | break
|   end
|
|   // update outer Aitken factor...
|   if  $k = 0$  then
|   |  ${}^0_m\beta^{n+1} \leftarrow \text{init } \beta$ 
|   else
|   | 
$${}^k_m\beta^{n+1} \leftarrow {}^{k-1}_m\beta^{n+1} \frac{{}^{k-1}_m\mathcal{R}_y^{n+1}}{{}^{k-1}_m\mathcal{R}_y^{n+1} - {}^k_m\mathcal{R}_y^{n+1}}$$

|   end
|
|   // update displacements and measured output...
|   
$${}^{k+1}_m\mathbf{y}_S^{n+1} \leftarrow {}^k_m\mathbf{y}_S^{n+1}$$

|   
$${}^{k+1}_m\mathbf{y}_S^{n+1} \leftarrow {}^k_m\mathbf{y}_S^{n+1} + {}^k_m\beta^{n+1} {}^k_m\mathcal{R}_y^{n+1}$$

|   end
end

```

Algorithm 3: Pseudocode for the partitioned F[SC]I scheme(nested [SC] subsystem).

```

// initialize states, i.e. set ICs...
 $k_{\text{end}} \mathbf{x}_F^0 \leftarrow \mathbf{x}_F^{\text{init}}, k_{\text{end}} \mathbf{x}_S^0 \leftarrow \mathbf{x}_S^{\text{init}}, k_{\text{end}} \mathbf{x}_C^0 \leftarrow \mathbf{x}_C^{\text{init}}$ 

// initialize displacements and measured output...
 $k_{\text{end}} \mathcal{Y}_S^0 \leftarrow \mathcal{Y}_S^{\text{init}}, m_{\text{end}} \mathcal{Y}_S^0 \leftarrow \mathcal{Y}_S^{\text{init}}$ 
// time loop...
for  $n \leftarrow 0$  to  $n \leftarrow n_{\text{end}} - 1$ 
  // predict displacements and measured output...
   $m_{\text{end}} \mathcal{Y}_S^{n+1} \leftarrow m_{\text{end}} \mathcal{Y}_S^n, m_{\text{end}} \mathcal{Y}_S^{n+1} \leftarrow m_{\text{end}} \mathcal{Y}_S^n$ 

  // outer interface iteration loop, i.e. F[SC]I loop...
  for  $k \leftarrow 0$  to  $k \leftarrow k_{\text{max}}$ 
    // map displacements from solid to fluid...
     $k \mathbf{y}_F^{n+1} \leftarrow \mathcal{M}_y \left( m_{\text{end}} \mathcal{Y}_S^{n+1} \right)$ 

    // solve fluid...
     $k \mathbf{z}_F^{n+1} \leftarrow \mathcal{G}_F \left[ k \mathbf{x}_F^{n+1} \right] \left( k \mathbf{y}_F^{n+1} \right)$ 

    // map forces from fluid to solid...
     $k \mathbf{z}_S^{n+1} \leftarrow \mathcal{M}_z \left( k \mathbf{z}_F^{n+1} \right)$ 

    // predict measured output...
     $k \mathcal{Y}_S^{n+1} \leftarrow m_{\text{end}} \mathcal{Y}_S^{n+1}$ 

    // inner interface iteration loop, i.e. SCI loop...
    for  $m \leftarrow 0$  to  $m \leftarrow m_{\text{max}}$ 
      // copy measured output from solid to controller...
       $m \mathcal{Y}_C^{n+1} \leftarrow m \mathcal{Y}_S^{n+1}$ 

      // solve controller...
       $m \mathbf{u}_C^{n+1} \leftarrow \mathcal{G}_C \left[ m \mathbf{x}_C^{n+1} \right] \left( m \mathcal{Y}_C^{n+1} \right)$ 

      // copy control input from controller to solid...
       $k \mathbf{u}_S^{n+1} \leftarrow m \mathbf{u}_C^{n+1}$ 

      // solve solid...
       $\begin{bmatrix} k \mathcal{Y}_S^{n+1} \\ k \mathbf{y}_S^{n+1} \\ m \mathcal{Y}_S^{n+1} \end{bmatrix} \leftarrow \mathcal{G}_S \left[ m \mathbf{x}_S^{n+1} \right] \left( k \mathbf{z}_S^{n+1}, k \mathbf{u}_S^{n+1} \right)$ 

      // calculate residuum of measured output...
       $k \mathcal{R}_y^{n+1} \leftarrow k \mathcal{Y}_S^{n+1} - m \mathcal{Y}_S^{n+1}$ 
      // check for inner convergence...
       $m \mathcal{E}^{n+1} \leftarrow \left\| \frac{k \mathcal{R}_y^{n+1}}{m \mathcal{R}_y^{n+1}} \right\| / \sqrt{n_{\text{dof}}}$ 
      if  $m \mathcal{E}^{n+1} < \max \varepsilon$  then
        break
      end

      // update inner Aitken factor...
      if  $m = 0$  then
         $k \beta^{n+1} \leftarrow \text{init} \beta$ 
      else
         $k \beta^{n+1} \leftarrow m \beta^{n+1} \frac{k \mathcal{R}_y^{n+1}}{m \mathcal{R}_y^{n+1} - k \mathcal{R}_y^{n+1}}$ 

        // update measured output...
         $m+1 \mathcal{Y}_S^{n+1} \leftarrow m \mathcal{Y}_S^{n+1} + k \beta^{n+1} \frac{k \mathcal{R}_y^{n+1}}{m \mathcal{R}_y^{n+1}}$ 
      end
    end
  end
end

```

Pseudocode for the partitioned F[SC]I scheme (continued).

```

// calculate residuum of displacements...
 ${}^k\mathcal{R}_y^{n+1} \leftarrow m_{\text{end}}^k y_S^{n+1} - m_{\text{end}}^{k-1} y_S^{n+1}$ 

// check for outer convergence...
 $k \varepsilon^{n+1} \leftarrow \| {}^k\mathcal{R}_y^{n+1} \| / \sqrt{n_{\text{dof}}}$ 
if  $k \varepsilon^{n+1} < \max_{\varepsilon}$  then
| break
end

// update outer Aitken factor...
if  $k = 0$  then
|  $\beta^{n+1} \leftarrow \text{init}\beta$ 
else
| if  $\dim \{ {}^k\mathcal{R}_y^{n+1} \} = 1$  then
| |  $\beta^{n+1} \leftarrow \beta^{n+1} \frac{{}^{k-1}\mathcal{R}_y^{n+1}}{{}^{k-1}\mathcal{R}_y^{n+1} - {}^k\mathcal{R}_y^{n+1}}$ 
| else
| |  $\beta^{n+1} \leftarrow \beta^{n+1} \frac{{}^{k-1}\mathcal{R}_y^{n+1\text{T}} ({}^{k-1}\mathcal{R}_y^{n+1} - {}^k\mathcal{R}_y^{n+1})}{\| {}^{k-1}\mathcal{R}_y^{n+1} - {}^k\mathcal{R}_y^{n+1} \|^2}$ 
| end
end

// update displacements and measured output...
 $m_{\text{end}}^{k+1} y_S^{n+1} \leftarrow m_{\text{end}}^k y_S^{n+1} + \beta^{n+1} {}^k\mathcal{R}_y^{n+1}$ 
 $m_{\text{end}}^{k+1} y_S^{n+1} \leftarrow m_{\text{end}}^k y_S^{n+1}$ 
end
end

```

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 23 May 2018 Accepted: 29 November 2018

Published online: 13 December 2018

References

1. Felippa CA, Park KC, Farhat C. Partitioned analysis of coupled mechanical systems. *Comput Methods Appl Mech Eng.* 2001;190:3247–70.
2. Bathe K-J, Zhang H, Yan Y. The solution of Maxwell's equations in multiphysics. *Comput Struct.* 2014;132:99–112. <https://doi.org/10.1016/j.compstruc.2013.09.006>.
3. Mayer UM, Popp A, Gerstenberger A, Wall WA. 3d fluid–structure–contact interaction based on a combined XFEM FSI and dual mortar contact approach. *Computat Mech.* 2010;46(1):53–67. <https://doi.org/10.1007/s00466-010-0486-0>.
4. Sicklinger S, Belsky V, Engelmann B, Elmqvist H, Olsson H, Wüchner R, Bletzinger K-U. Interface Jacobian-based co-simulation. *Int J Num Methods Eng.* 2014;98(6):418–44. <https://doi.org/10.1002/nme.4637>.
5. Bungartz H-J, Lindner F, Gatzhammer B, Mehl M, Scheufele K, Shukaev A, Uekermann B. PreCICE—a fully parallel library for multi-physics surface coupling. *Comput Fluids.* 2016;141:250–8. <https://doi.org/10.1016/j.compfluid.2016.04.003>.
6. Bathe K-J, Zhang H. Finite element developments for general fluid flows with structural interactions. *Int J Num Methods Eng.* 2004;60(1):213–32. <https://doi.org/10.1002/nme.959>.
7. Bathe K-J, Zhang H, Ji S. Finite element analysis of fluid flows fully coupled with structural interactions. *Comput Struct.* 1999;72(1–3):1–16. [https://doi.org/10.1016/S0045-7949\(99\)00042-5](https://doi.org/10.1016/S0045-7949(99)00042-5).
8. Dettmer WG, Perić D. A new staggered scheme for fluid–structure interaction. *Int J Num Methods Eng.* 2013;93(1):1–22. <https://doi.org/10.1002/nme.4370>.
9. Joosten MM, Dettmer WG, Perić D. On the temporal stability and accuracy of coupled problems with reference to fluid–structure interaction. *Int J Num Methods Fluids.* 2010;64(10–12):1363–78. <https://doi.org/10.1002/flid.2333>.
10. Causin P, Gerbeau JF, Nobile F. Added-mass effect in the design of partitioned algorithms for fluid–structure problems. *Comput Methods Appl Mech Eng.* 2005;194(42–44):4506–27. <https://doi.org/10.1016/j.cma.2004.12.005>.
11. Joosten MM, Dettmer WG, Perić D. Analysis of the block Gauss–Seidel solution procedure for a strongly coupled model problem with reference to fluid–structure interaction. *Int J Num Methods Eng.* 2009;78(7):757–78. <https://doi.org/10.1002/nme.2503>.
12. Matthies HG, Steindorf J. Partitioned strong coupling algorithms for fluid–structure interaction. *Comput Struct.* 2003;81(8–11):805–12. [https://doi.org/10.1016/S0045-7949\(02\)00409-1](https://doi.org/10.1016/S0045-7949(02)00409-1).

13. Wood C, Gil AJ, Hassan O, Bonet J. Partitioned block-Gauss-Seidel coupling for dynamic fluid–structure interaction. *Comput Struct*. 2010;88(23–24):1367–82. <https://doi.org/10.1016/j.compstruc.2008.08.005>.
14. Föllinger O, Konigorski U, Lohmann B, Roppenecker G, Trächtler A. *Regelungstechnik: Einführung in die Methoden und Ihre Anwendung*; [aktualisierter Lehrbuch-Klassiker], 11, völlig neu bearb. Aufl. ed. Berlin: VDE-Verl; 2013.
15. Unbehauen H. *Regelungstechnik II: Zustandsregelungen, Digitale und Nichtlineare Regelsysteme*, 8., vollständig überarbeitete und erweiterte Auflage edn. Studium Technik. Vieweg+Teubner Verlag, Wiesbaden; 2000. <https://doi.org/10.1007/978-3-322-94390-3>.
16. van Brummelen EH. Added mass effects of compressible and incompressible flows in fluid–structure interaction. *J Appl Mech*. 2009;76(2):021206. <https://doi.org/10.1115/1.3059565>.
17. Strang G. *Computational Science and Engineering*. 2nd ed. Wellesley: Wellesley-Cambridge Press; 2012.
18. Bungartz H-J, Lindner F, Mehl M, Uekermann B. A plug-and-play coupling approach for parallel multi-field simulations. *Comput Mech*. 2015;55(6):1119–29. <https://doi.org/10.1007/s00466-014-1113-2>.
19. Turek S, Hron J. Proposal for numerical benchmarking of fluid–structure interaction between an elastic object and laminar incompressible flow. In: Bungartz H-J, Schäfer M, editors. *Fluid–structure interaction*, vol. 53. Lecture notes in computational science and engineering. Berlin: Springer; 2006. p. 371–85.
20. Wall WA. *Fluid-struktur-interaktion mit stabilisierten finiten elementen*. Ph.D. thesis, Universität Stuttgart. 1999.
21. Lunze J. *Regelungstechnik 2: Mehrgrößensysteme, digitale Regelung*, 6., neu bearbeitete Aufl. edn. Springer-Lehrbuch. Berlin: Springer; 2010. <https://doi.org/10.1007/978-3-642-10198-4>.
22. Dadvand P, Rossi R, Oñate E. An object-oriented environment for developing finite element codes for multi-disciplinary applications. *Arch Comput Methods Eng*. 2010;17(3):253–97. <https://doi.org/10.1007/s11831-010-9045-2>.
23. KRATOSMultiphysics. <https://github.com/KratosMultiphysics/Kratos/wiki>.
24. Cotela Dalmau J, Oñate E, Rossi R. *Applications for Turbulence Modeling in Civil Engineering*, (2016).
25. Donéa J, Huerta A. *Finite element methods for flow problems*. New Jersey: Wiley, Chichester and Hoboken; 2003. <https://doi.org/10.1002/0470013826>.
26. Stein K, Tezduyar T, Benney R. Mesh moving techniques for fluid–structure interactions with large displacements. *J Appl Mech*. 2003;70(1):58. <https://doi.org/10.1115/1.1530635>.
27. Dettmer WG. *Finite element modelling of fluid flow with moving free surfaces and interfaces including fluid–solid interaction*. Ph.D. thesis, University of Wales Swansea, School of Engineering. 2004.
28. Küttler U, Wall WA. Fixed-point fluid–structure interaction solvers with dynamic relaxation. *Comput Mech*. 2008;43(1):61–72. <https://doi.org/10.1007/s00466-008-0255-5>.
29. Degroote J, Bathe K-J, Vierendeels J. Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. *Comput Struct*. 2009;87(11–12):793–801. <https://doi.org/10.1016/j.compstruc.2008.11.013>.
30. Degroote J. *Development of algorithms for the partitioned simulation of strongly coupled fluid–structure interaction problems*. Ph.D. thesis, Ghent University. 2010.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
