Space-mapping in fluid–structure interaction problems

T.P. Scholcz*, A.H. van Zuijlen, H. Bijl

Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands

Abstract

Transient response due to gust loads can lead to structural failure despite the fact that the aero-elastic system is asymptotically stable. Unsteady aeroelastic analysis should therefore be included in the load calculation cycle of the aircraft design process. Especially in the transonic regime, partitioned strong coupling algorithms perform better than loose coupling algorithms and allow larger time-steps in the unsteady simulation. However, the simulation of high fidelity unsteady fluid–structure interaction using strong coupling algorithms is currently too expensive in order to be useful in industry.

To accelerate high fidelity partitioned fluid–structure interaction simulations we apply space-mapping, which is a technique that originates from the field of multi-fidelity optimization. Without loss of generality we assume the availability of a cheap low fidelity fluid solver and an expensive high fidelity fluid solver. The space-mapping approach is used to accelerate the high fidelity computation using black-box input/output information of both the high fidelity fluid solver and low fidelity fluid solver. In order to achieve this, a space-mapping function is defined on the fluid–structure interface which keeps track of the differences between the high fidelity model and the low fidelity model during the coupling iterations. Reformulating the root-finding problem on the fluid–structure interaction interface using the space-mapping function results in the Aggressive Space-Mapping algorithm.

The Aggressive Space-Mapping algorithm is applied to 1-D and 2-D test cases in order to assess the speedup with respect to the Quasi-Newton Inverse Least Squares algorithm. The latter is considered to be the current state of the art in partitioned strong coupling algorithms. The observed speedup depends mainly on the type of FSI problem and the time step size. The maximum observed speedup is about 1.5.

The application of the space-mapping technique to partitioned fluid–structure interaction problems is found to be a promising approach. The framework is non-intrusive and allows the reuse of existing solvers which is especially useful in an industrial environment. It is expected that the space-mapping technique can be combined with higher order time integration schemes that maintain accuracy over a large range of time step sizes.
© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Fluid–Structure Interactions (FSI) play a central role in aerospace engineering and many other fields like civil, mechanical and biomedical engineering [1,2]. Unstable interactions like wing flutter and buffeting can cause structural failure and prediction of their occurrence is of primary importance in the design of aircraft [3]. Asymptotic stability is

---

* Corresponding author. Tel.: +31 0152788690.
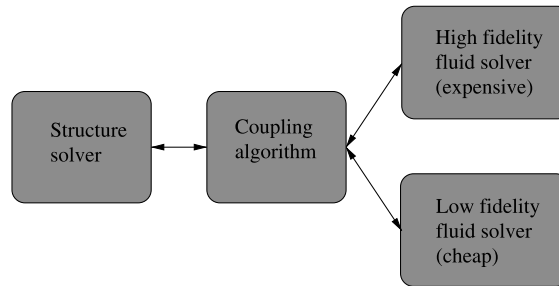 *E-mail address:* t.p.scholcz@tudelft.nl (T.P. Scholcz).

Fig. 1. Schematic of a multi-fidelity coupling algorithm.

a necessary but insufficient condition to guarantee structural integrity. It has been shown that transient growth, induced by sources of external excitation such as gust loads, can lead to structural failure despite the fact that the system is asymptotically stable [4,5]. Unsteady aeroelastic analysis can overcome the shortcomings of asymptotic analysis but it is computationally much more expensive. As an example, the analysis of 50 flight points in the flight envelope for 100 mass cases, 10 control surface configurations, 50 manoeuvres, and 4 control laws would already result in 10,000,000 unsteady aeroelastic simulations to perform a single load calculation cycle. Using engineering experience and lower fidelity models, often corrected with costly wind tunnel data, the current load calculation cycle requires more than 6 weeks, [6].

The replacement of low fidelity simulations with more accurate aeroelastic simulations is attractive because it reduces the number of design cycles, the development risk, the number of flight tests, the cost and time to market and the risk of design modifications in the later design phases [7]. However, the computational effort associated with high fidelity aeroelastic models currently precludes their direct use in industry. Acceleration of time-accurate high fidelity aeroelastic simulation algorithms has therefore become an active area of research, using e.g. multi-level approaches [8–11], multi-solver approaches [12], Interface-GMRES(R) [13,14], Aitken's method and vector extrapolation [15,16] and the Quasi-Newton Inverse Least Squares (QN-ILS) method [12,17–19].

The QN-ILS method has become a popular method due to its combination of efficiency and simplicity, see [12,17,19] and its thorough theoretical basis, see [18]. In [19] it was found that the QN-ILS method outperforms Aitken's method and the Newton–Krylov method from [13] when applied to a (nonlinear) strongly coupled FSI problem. In [18] it was found that the QN-LS method is only slightly slower than GMRes when applied to obtain the solution of several linear systems of equations and in [20] it is shown that the QN-ILS method can be modified to become analytically equivalent to GMRes. A general comparison of various partitioned iterative solution methods for FSI is found in [21,22].

These algorithms are all designed to efficiently solve the coupled problem that results from an implicit time integration scheme applied to the semi-discrete system of equations describing the fluid and solid dynamics, the so called *partitioned* approach. The partitioned approach allows software modularity and reuse of existing field solvers and is therefore more promising in an industrial environment than the *monolithic* approach, which aims at solving the fluid and solid systems simultaneously. In the transonic regime, the flow interacts strongly with the structure since the flow is highly nonlinear and very sensitive to structural motions [23]. Especially for large time steps in the transonic regime, strong coupling procedures are necessary in order to avoid excessive phase-lag errors that would otherwise result from the dominating partitioning error [24]. Strong coupling algorithms are more expensive but unavoidable since loosely coupled algorithms yield unacceptable accuracy in this regime. It is this fact that motivates the development of more efficient partitioned strong coupling algorithms.

In this contribution we investigate the use of low fidelity models to speed up partitioned coupling simulations applied to high fidelity models, the so called *multi-fidelity* approach, see Fig. 1. Without loss of generality we assume that two solvers are available: a cheap low fidelity fluid solver and an expensive high fidelity fluid solver. We define a mapping between the input space of the low fidelity model and the input space of the high fidelity model during the coupling iterations: the space-mapping function. The space-mapping function keeps track of the differences between the high and low fidelity models during the coupling iterations which is subsequently used to speedup the computations. A-priori knowledge of the exact inverse space-mapping function would allow for the direct computation of the high fidelity solution by the inverse mapping of the low fidelity solution to the high fidelity space. However, such a-priori knowledge is not available. This necessitates the iterative approximation of the inverse space-mapping

function during the partitioned coupling iterations. When the space-mapping function is expanded in a first order Taylor series and when the Jacobian is iteratively approximated using input/output information the so called Aggressive Space-Mapping (ASM) algorithm results. Although space-mapping [25] is currently mainly applied in the field of optimization, it can also be used to efficiently solve the coupled problem at each time step of a partitioned FSI simulation since the underlying principles of space-mapping are quite general.

The paper is organized as follows: A typical FSI model problem is outlined in Section 2, followed by the introduction of structure and fluid operators which are necessary to define a partitioned coupling residual on the FSI interface. The recently introduced Quasi-Newton Inverse Least Squares method is discussed in Section 3 followed by Section 4 on space-mapping in fluid–structure interaction problems. The ASM algorithm is applied to the two-dimensional panel flutter problem and the one-dimensional flexible tube problem in Section 5. Conclusions with respect to the speedup of the ASM algorithms are subsequently given in Section 6.

2. Problem formulation

A typical FSI model consists of a fluid model defined on a deformable domain Ω_f – numerically implemented using the Arbitrary Lagrangian Eulerian (ALE) formulation – which is in interaction with a structure model defined on Ω_s and modelled using the Lagrangian formulation. The fluid domain and structure domain both have a fluid–structure interaction interface Γ_I^f and Γ_I^s which are identical, such that both domains are coupled [1]. A discretization technique is used to obtain the semi-discrete systems of equations describing the dynamics of each physical system. Typically the Finite Volume Method (FVM) is used to obtain the system describing the fluid dynamics whereas the Finite Element Method (FEM) is used to obtain the structural system equations. Let the vector $\mathbf{v}$ denote the discrete state vector of the fluid in Ω_f and vector $\mathbf{u}$ the discrete state vector of the structure in Ω_s at the new time level t^{n+1}. Hiding the dependency on the solution of previous time levels, the coupled problem at time step t^{n+1} is formulated as [12]

$$\mathbf{r}^f(\mathbf{v}; \xi_x(\mathbf{u})) = \mathbf{0} \quad \mathbf{v} \in \mathbb{R}^{N_f} \tag{1}$$

$$\mathbf{r}^s(\mathbf{u}; \xi_y(\mathbf{v})) = \mathbf{0} \quad \mathbf{u} \in \mathbb{R}^{N_s}. \tag{2}$$

Here, $\mathbf{r}^f$ denotes the residual of the discrete fluid equations and $\mathbf{r}^s$ the residual of the discrete structure equations. The function $\xi_x : \mathbb{R}^{N_s} \to \mathbb{R}^{N_I^s}$ maps the structural state vector $\mathbf{u} \in \mathbb{R}^{N_s}$ to the interface displacement vector $\mathbf{x} \in \mathbb{R}^{N_I^s}$, hence $\mathbf{x} = \xi_x(\mathbf{u})$. The function $\xi_y : \mathbb{R}^{N_f} \to \mathbb{R}^{N_I^f}$ maps the fluid state vector $\mathbf{v} \in \mathbb{R}^{N_f}$ to the interface pressure $\mathbf{y} \in \mathbb{R}^{N_I^f}$, hence $\mathbf{y} = \xi_y(\mathbf{v})$. Given a certain interface displacement vector $\mathbf{x}$

$$\mathbf{y} = \xi_y(\mathbf{v}) \quad \text{for } \mathbf{v} = \arg \min_{\mathbf{v} \in \mathbb{R}^{N_f}} \|\mathbf{r}^f(\mathbf{v}; \mathbf{x})\|, \tag{3}$$

defines the *high fidelity fluid operator* $\mathcal{F} : \mathbb{R}^{N_I^s} \to \mathbb{R}^{N_I^f}$

$$\mathbf{y} = \mathcal{F}(\mathbf{x}). \tag{4}$$

Numerical evaluation of the high fidelity fluid operator requires the solution of the minimization problem in Eq. (3) up to a certain tolerance ϵ_f and the evaluation of the map ξ_y to find the interface pressure from the fluid state vector. The minimization of $\mathbf{r}^f$ is in general performed with a Computational Fluid Dynamics solver. Likewise, given an interface pressure $\mathbf{y}$

$$\mathbf{x} = \xi_x(\mathbf{u}) \quad \text{for } \mathbf{u} = \arg \min_{\mathbf{u} \in \mathbb{R}^{N_s}} \|\mathbf{r}^s(\mathbf{u}; \mathbf{y})\|, \tag{5}$$

defines the *structure operator* $\mathcal{S} : \mathbb{R}^{N_I^f} \to \mathbb{R}^{N_I^s}$

$$\mathbf{x} = \mathcal{S}(\mathbf{y}). \tag{6}$$

Numerical evaluation of the structure operator requires the solution of the minimization problem in Eq. (5) up to a certain tolerance ϵ_s and evaluation of the map ξ_x to find the interface displacement from the structure variables. The minimization of $\mathbf{r}^s$ is in general performed with a Computational Structural Dynamics solver. Due to the large range of important scales present in the fluid dynamics we have as a general rule $N_f \gg N_s$. Evaluation of the fluid operator is computationally much more expensive than evaluation of the structure operator as a consequence.

In order to satisfy kinematic and dynamic interface conditions on the fluid–structure interaction interface at every time step of a simulation we require continuity of the interface displacement/velocity and force equilibrium on the fluid–structure interface. These conditions are satisfied when [12,13,17]

$$\mathcal{R}(\mathbf{x}) = \mathbf{0} \quad \text{with } \mathcal{R}(\mathbf{x}) = \mathcal{S} \circ \mathcal{F}(\mathbf{x}) - \mathbf{x}, \tag{7}$$

where $\mathcal{R} : \mathbb{R}^{N_I^s} \to \mathbb{R}^{N_I^s}$ is the *interface residual* function. Strong coupling algorithms aim to minimize the interface residual $\mathcal{R}$ to a certain tolerance ϵ_I using a minimum number of (expensive) fluid operator evaluations:

$$\mathbf{x}^* = \arg \min_{\mathbf{x} \in \mathbb{R}^{N_I^s}} \|\mathcal{R}(\mathbf{x})\|. \tag{8}$$

Note that when $\epsilon_f = \epsilon_s = \epsilon_I = 0$ we find $\mathbf{u} = \mathbf{u}^*$ and $\mathbf{v} = \mathbf{v}^*$ satisfying Eqs. (1) and (2) and the unique interface displacement and pressure are found from $\mathbf{x}^* = \xi_x(\mathbf{u}^*)$ and $\mathbf{y}^* = \xi_y(\mathbf{v}^*)$ respectively. The most basic strong coupling algorithm is the fixed point iteration scheme given in algorithm 1.

Algorithm 1 Fixed point iteration scheme

1: $k = 0$
2: $\mathbf{r}^k = \mathcal{R}(\mathbf{x}^k)$
3: **while** $\|\mathbf{r}^k\| > \epsilon_I$ **do**
4:     $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{r}^k$
5:     $k = k + 1$
6:     $\mathbf{r}^k = \mathcal{R}(\mathbf{x}^k)$
7: **end while**

Algorithm 1 may converge slowly and has a poor robustness, see [13,26]. To obtain better performance, so called Quasi-Newton algorithms were developed, [17–20,26–28]. Quasi-Newton methods are more robust and may obtain superlinear convergence, see [19,28].

3. The Quasi-Newton Inverse Least Squares algorithm

The purpose of this section is two-fold. It will introduce the Quasi-Newton Inverse Least Squares (QN-ILS) method as an accelerator of the fixed point iteration scheme given in algorithm 1 and provide some results which are necessary for the development of the Aggressive Space Mapping algorithm in Section 4.2.

Introducing $\mathcal{H} = \mathcal{S} \circ \mathcal{F}$, the fixed point iteration update on line 4 of algorithm 1 can be written as

$$\mathbf{x}^{k+1} = \mathcal{H}(\mathbf{x}^k). \tag{9}$$

To improve convergence of the fixed point iteration scheme $\mathbf{x}^k$ is replaced by a better candidate $\mathbf{x}^{new}$ such that

$$\mathbf{x}^{k+1} = \mathcal{H}(\mathbf{x}^{new}). \tag{10}$$

It is assumed that the new candidate can be written as a linear combination of the previous iterates $\mathbf{x}^0 \dots \mathbf{x}^k$

$$\mathbf{x}^{new} \in \mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k). \tag{11}$$

The new candidate is substituted into the expression for the residual and subsequently linearized

$$\mathcal{R}(\mathbf{x}^{new}) = \mathcal{R}\left(\mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k)\right) \tag{12}$$

$$\approx \mathcal{R}(\mathbf{x}^k) + \left(\frac{\partial \mathcal{R}}{\partial \mathbf{x}}\right) \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k) \tag{13}$$

$$\approx \mathbf{r}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{r}^i - \mathbf{r}^k), \tag{14}$$

where $\mathbf{r}^i = \mathcal{R}(\mathbf{x}^i)$ for $i = 0 \ldots k$. The coefficients c_i^k are then found from a minimization of the linearized residual $\mathcal{R}(\mathbf{x}^{new})$

$$\mathbf{c}^k = \arg \min_{\mathbf{c}^k \in \mathbb{R}^k} \left\| \mathbf{r}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{r}^i - \mathbf{r}^k) \right\|, \tag{15}$$

such that the better candidate is found from

$$\mathbf{x}^{new} = \mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k). \tag{16}$$

Substituting $\mathbf{x}^{new}$ from Eq. (16) in the expression for the update in Eq. (10) it is found that

$$\mathbf{x}^{k+1} = \mathcal{H}\left(\mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k) \right) \tag{17}$$

$$\approx \mathcal{H}(\mathbf{x}^k) + \left(\frac{\partial \mathcal{H}}{\partial \mathbf{x}} \right) \sum_{i=0}^{k-1} c_i^k (\mathbf{x}^i - \mathbf{x}^k) \tag{18}$$

$$\approx \mathbf{x}_{\mathcal{H}}^k + \sum_{i=0}^{k-1} c_i^k (\mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k), \tag{19}$$

where $\mathbf{x}_{\mathcal{H}}^i = \mathcal{H}(\mathbf{x}^i)$ for $i = 0 \ldots k$.

The linearization in Eq. (19) avoids a true evaluation of $\mathcal{H}(\mathbf{x}^{new})$ since all the displacement vectors in Eq. (19) are readily available when they are stored in previous iterations.

Practical implementation details of the QN-ILS method are found in [17]. The differences in Eqs. (14) and (19) are calculated as

$$\Delta \mathbf{r}^i = \mathbf{r}^i - \mathbf{r}^k \tag{20}$$

$$\Delta \mathbf{x}_{\mathcal{H}} = \mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k, \tag{21}$$

for $i = 0 \ldots k - 1$. Subsequently the differences are stored in separate matrices $\mathbf{V}^k$ and $\mathbf{W}^k$

$$\mathbf{V}^k = [\Delta \mathbf{r}^{k-1} \ \Delta \mathbf{r}^{k-2} \ldots \Delta \mathbf{r}^0] \tag{22}$$

$$\mathbf{W}^k = [\Delta \mathbf{x}_{\mathcal{H}}^{k-1} \ \Delta \mathbf{x}_{\mathcal{H}}^{k-2} \ldots \Delta \mathbf{x}_{\mathcal{H}}^0]. \tag{23}$$

Since $\mathbf{x}_{\mathcal{H}}^k = \mathbf{x}^k + \mathbf{r}^k$ the update in (19) can be written

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{W}^k \mathbf{c}^k + \mathbf{r}^k, \tag{24}$$

which is interpreted as a Quasi-Newton method

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\frac{\partial \mathcal{R}^k}{\partial \mathbf{x}} \right)^{-1} \mathbf{r}^k, \tag{25}$$

with the inverse of the Jacobian approximated by

$$\left(\frac{\partial \mathcal{R}^k}{\partial \mathbf{x}} \right)^{-1} \approx \mathbf{W}^k \mathbf{R}^{k^{-1}} \mathbf{Q}^{kT} - \mathbf{I}, \tag{26}$$

and $\mathbf{Q}^k$ and $\mathbf{R}^k$ the matrices that result from the economy-size QR-decomposition of $\mathbf{V}^k$. The QR-decomposition is used to minimize (15) in a least squares sense. The QN-ILS algorithm is summarized in Algorithm 2.

Algorithm 2 QN-ILS

1: $k = 0$
2: $\mathbf{x}_{\mathcal{H}}^k = \mathcal{H}(\mathbf{x}^k)$
3: $\mathbf{r}^k = \mathbf{x}_{\mathcal{H}}^k - \mathbf{x}^k$
4: **while** $\|\mathbf{r}^k\| > \epsilon_I$ **do**
5:     **if** $k = 0$ **then**
6:         $\mathbf{x}^{k+1} = \mathbf{x}^k - (\nabla_{\mathbf{x}}^k \mathcal{R})^{-1} \mathbf{r}^k$
7:     **else**
8:         **for** $i = 0$ to $k - 1$ **do**
9:             $\Delta \mathbf{x}_{\mathcal{H}}^i = \mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k$
10:            $\Delta \mathbf{r}^i = \mathbf{r}^i - \mathbf{r}^k$
11:         **end for**
12:         $\mathbf{V}^k = \begin{bmatrix} \Delta \mathbf{r}^{k-1} \ \Delta \mathbf{r}^{k-2} \ \dots \ \Delta \mathbf{r}^0 \end{bmatrix}$
13:         $\mathbf{W}^k = \begin{bmatrix} \Delta \mathbf{x}_{\mathcal{H}}^{k-1} \ \Delta \mathbf{x}_{\mathcal{H}}^{k-2} \ \dots \ \Delta \mathbf{x}_{\mathcal{H}}^0 \end{bmatrix}$
14:         Calculate $\mathbf{V}^k = \mathbf{Q}^k \mathbf{R}^k$
15:         Calculate $\mathbf{R}^k \mathbf{c}^k = -\mathbf{Q}^{kT} \mathbf{r}^k$
16:         $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{W}^k \mathbf{c}^k + \mathbf{r}^k$
17:     **end if**
18:     $k = k + 1$
19:     $\mathbf{x}_{\mathcal{H}}^k = S \circ F(\mathbf{x}^k)$
20:     $\mathbf{r}^k = \mathbf{x}_{\mathcal{H}}^k - \mathbf{x}^k$
21: **end while**

4. The space-mapping algorithm

In order to enhance the convergence and robustness properties of the QN-ILS algorithm we include information of simplified fluid operators that are much cheaper to evaluate. Simplified fluid operators were used to approximate the Jacobian in the Newton-GMRES method, see [19,27]. This requires two evaluations of the simplified fluid operator in each Krylov iteration to compute a finite difference approximation of a Jacobian vector product. The simplified fluid operators should be chosen in such a way that the relevant physics of the problem is maintained, neglecting for example certain nonlinearities and/or the deformation of the fluid domain. Once the simplified fluid operator is available it is used as a black-box, in the same fashion as the original fluid operator. In this contribution we use a simplified fluid operator to find a preconditioned rootfinding problem, to which an *outer* Quasi-Newton method is applied. A straightforward way to obtain a preconditioned root-finding problem is to apply a transformation of the original rootfinding problem [29]

$$\mathcal{G}(\mathcal{R}(\mathbf{x})) = \mathbf{0}, \tag{27}$$

such that

$$\frac{\partial \mathcal{G}}{\partial \mathbf{v}} \frac{\partial \mathcal{R}}{\partial \mathbf{x}} \approx \mathbf{I}, \tag{28}$$

if $\mathcal{G}$ is close to $\mathcal{R}^{-1}$. In that case only a few iterations are required to achieve convergence. When the functions $\mathcal{G}$ and $\mathcal{R}$ are not explicitly known and the Jacobians $\frac{\partial \mathcal{G}}{\partial \mathbf{v}}$ and $\frac{\partial \mathcal{R}}{\partial \mathbf{x}}$ are either unavailable or very expensive to compute space-mapping algorithms [25] can be used to find a preconditioned rootfinding problem indirectly. Space-mapping algorithms exploit the combination of less sophisticated models with the accuracy of more complex models, using input/output information only.

Let $\tilde{\mathbf{r}}^f(\tilde{\mathbf{v}}, \mathbf{z})$ denote the residual of the discrete fluid equations describing the simplified fluid model. The simplified fluid model has the interface displacement $\mathbf{z} \in \mathbb{R}^{\tilde{N}_I^s}$ as an argument. Given an interface displacement $\mathbf{z}$

$$\tilde{\mathbf{y}} = \xi_p(\tilde{\mathbf{v}}) \quad \text{for } \tilde{\mathbf{v}} = \arg \min_{\tilde{\mathbf{v}} \in \mathbb{R}^{\tilde{N}_f}} \|\tilde{\mathbf{r}}^f(\tilde{\mathbf{v}}; \mathbf{z})\| \tag{29}$$

defines the *low fidelity fluid operator* $\tilde{\mathbf{y}} = \tilde{F}(\mathbf{z})$, such that the low fidelity fluid–structure interaction problem becomes

$$\tilde{\mathcal{R}}(\mathbf{z}) = \mathbf{0} \quad \text{with } \tilde{\mathcal{R}}(\mathbf{z}) = \mathcal{S} \circ \tilde{\mathcal{F}}(\mathbf{z}) - \mathbf{z}. \tag{30}$$

Note that the state variables of the high fidelity and low fidelity fluid models may differ and that the low fidelity fluid state vector $\tilde{\mathbf{v}}$ is not necessary an element of the same discrete space as the high fidelity fluid state vector $\mathbf{v}$: $\tilde{N}_f \neq N_f$. Also, on the interface we may have $\tilde{N}_I^f \neq N_I^f$ and $\tilde{N}_I^s \neq N_I^s$. In that case interface restriction and prolongation operators can be used in the space-mapping function in order to make the mappings between the two discrete spaces. The solution of the low fidelity fluid–structure interaction problem is given by

$$\mathbf{z}^* = \arg \min_{\mathbf{z} \in \mathbb{R}^{\tilde{N}_I^s}} \|\tilde{\mathcal{R}}(\mathbf{z})\|. \tag{31}$$

The minimization in (31) is up to a tolerance $\tilde{\epsilon}_I$. The "arg min" notation in (31) is used to point out that a partitioned fluid–structure interaction problem is solved without specifying which coupling algorithm is used for the solve since this is not relevant for the understanding of the space mapping method. This notation is used throughout the paper and when numerical experiments are discussed we mention the choices made for the coupling algorithms. The space-mapping approach requires the definition of a space-mapping function. This is the topic of Section 4.1.

4.1. Space-mapping function

A misalignment function between the high fidelity interface residual and the low fidelity interface residual on the fluid–structure interface is defined as

$$r(\mathbf{z}, \mathbf{x}) = \|\tilde{\mathcal{R}}(\mathbf{z}) - \mathcal{R}(\mathbf{x})\|. \tag{32}$$

For a given high fidelity model interface displacement $\mathbf{x} \in X$ it is useful to know which low fidelity model interface displacement $\mathbf{z} \in Z$ yields the best approximation to the interface residual $\mathcal{R}$, hence with the smallest misalignment r. Finding the best $\mathbf{z}$ for a given $\mathbf{x}$ defines the space-mapping function $\mathcal{P} : \mathbb{R}^{N_I^s} \to \mathbb{R}^{N_I^s}$

$$\mathbf{p} = \mathcal{P}(\mathbf{x}) = \arg \min_{\mathbf{z} \in \mathbb{R}^{N_I}} r(\mathbf{z}, \mathbf{x}). \tag{33}$$

To evaluate the space-mapping function numerically a second "auxiliary" fluid–structure interaction problem needs to be solved with the low fidelity fluid operator. This problem can be solved with any coupling algorithm as long as it results in a stable and convergent iterative process. The choice for this coupling method defines the *inner* method. An example of the numerical evaluation of the space-mapping function is summarized in Algorithm 3 which uses basic fixed-point iterations as an inner method.

Algorithm 3 Numerical evaluation of $\mathbf{p}^k = \mathcal{P}(\mathbf{x}^k)$

Require: $\mathbf{x}^k, \mathbf{z}^0, \epsilon_s$
 1: $i = 0$
 2: $\mathbf{r}^k = \mathcal{R}(\mathbf{x}^k)$
 3: $\tilde{\mathbf{r}}^i = \tilde{\mathcal{R}}(\mathbf{z}^i)$
 4: **while** $\|\tilde{\mathbf{r}}^i - \mathbf{r}^k\| > \epsilon_s$ **do**
 5:     $\mathbf{z}^{i+1} = \mathcal{S} \circ \tilde{\mathcal{F}}(\mathbf{z}^i) - \mathbf{r}^k$
 6:     $i = i + 1$
 7:     $\tilde{\mathbf{r}}^i = \mathcal{S} \circ \tilde{\mathcal{F}}(\mathbf{z}^i) - \mathbf{z}^i$
 8: **end while**
 9: $\mathbf{p}^k = \mathbf{z}^i$
10: **return** $\mathbf{p}^k, \mathbf{r}^k$

Evaluation of the space-mapping function requires a single expensive evaluation of the high fidelity fluid operator and several cheap evaluations of the low fidelity fluid operator to perform the mapping.

4.2. The aggressive space-mapping algorithm

The following definition is cited from [25]

Definition 1. A space-mapping function $\mathcal{P}$ is called a perfect mapping iff $\mathbf{z}^* = \mathcal{P}(\mathbf{x}^*)$.

Substituting $\mathbf{x}^*$ into the space-mapping function defined by Eq. (33) and using the definition in Eqs. (32), (8) and (31) it follows that $\mathcal{P}$ as defined in (33) is always a perfect mapping. It is now possible to apply a Quasi-Newton method to the new rootfinding problem

$$\mathcal{K}(\mathbf{x}) = \mathbf{0} \quad \text{with } \mathcal{K}(\mathbf{x}) = \mathcal{P}(\mathbf{x}) - \mathbf{z}^*, \tag{34}$$

with $\mathcal{K} : \mathbb{R}^{N_i^s} \to \mathbb{R}^{N_i^s}$, which is the *outer* method. This results in the Aggressive Space-Mapping (ASM) algorithm as defined in [25,30]. If $\tilde{\mathcal{R}}$ is close to $\mathcal{R}$ we now have

$$\frac{\partial \mathcal{K}}{\partial \mathbf{x}} = \frac{\partial \mathcal{P}}{\partial \mathbf{x}} \approx \mathbf{I}, \tag{35}$$

and it is likely that the Quasi-Newton algorithm converges faster when applied to the new rootfinding problem in Eq. (34). The ASM algorithm consists of two steps:

1. Solve for the low fidelity fluid–structure interaction solution $\mathbf{z}^*$.
2. Apply a Quasi-Newton algorithm to the new rootfinding problem in Eq. (34).

Although the space mapping function in (33) is always a perfect mapping, convergence is not guaranteed. To understand why this is case we use to concept of model flexibility. The definition of model flexibility from [25] states

Definition 2. A model is called more flexible than another if the set of its reachable aims contains the set of reachable aims of the other. Two models are equally flexible if their sets of reachable aims coincide.

Hence, a low fidelity model is more flexible than a high fidelity model if $\tilde{\mathcal{R}}(Z) \supset \mathcal{R}(X)$ and less flexible if $\tilde{\mathcal{R}}(Z) \subset \mathcal{R}(X)$. From the lemma's in [25] we also find that

1. If the low fidelity model is more flexible than the high fidelity model then $\mathcal{P} : X \to Z$ is injective if $\mathcal{R} : X \to R$ is injective.
2. If the low fidelity model and the high fidelity model are equally flexible and if $\mathcal{R} : X \to R$ is injective then $\mathcal{P}$ is a bijection.
3. If the high fidelity model is more flexible than the low fidelity model then $\mathcal{P} : X \to Z$ is surjective.

Convergence problems may for example occur if the space mapping function is surjective in the region of the high fidelity model solution $\mathbf{x}^*$. Perfect mapping is a property that concerns only a *point* while model flexibility concerns a *region*. It can therefore happen that the mapping is perfect since $\mathbf{x}^*$ maps perfectly to $\mathbf{z}^*$ while $\mathbf{x}^*$ is not the only solution that maps to $\mathbf{z}^*$ which means that it is non-unique. It can still be useful to apply the ASM algorithm when this happens if it is combined with conventional algorithms as will become clear in the numerical examples.

To approximate the space-mapping Jacobian $\frac{\partial \mathcal{P}}{\partial \mathbf{x}}$ in Step 2 of the ASM algorithm mainly *Broyden's method* is used in the space-mapping community [25,30]. Although Broyden's method has been used in the FSI community as well [26,31] and in the first work on space-mapping accelerated algorithms for FSI [32], we choose the QN-ILS method from Section 3 as the outer iterative method in this contribution.

Broyden's method uses information of only two recent iterates to approximate the Jacobian while the recently developed QN-ILS method uses information from several previous iterates, therefore belonging to the class of multi-iterate methods [33]. This is a heuristic explanation of the reported success of the QN-ILS method in the FSI community [12,17,19] when compared to other Quasi-Newton methods.

To apply the QN-ILS method from Section 3 to the new rootfinding problem, substitute $\mathbf{x}^{new}$ from Eq. (11) into Eq. (34) and linearize

$$\mathcal{K}(\mathbf{x}^{new}) = \mathcal{P}\left(\mathbf{x}^k + \sum_{i=0}^{k-1} c_i^k(\mathbf{x}^i - \mathbf{x}^k)\right) - \mathbf{z}^* \tag{36}$$

$$\approx \mathcal{P}(\mathbf{x}^k) - \mathbf{z}^* + \left(\frac{\partial \mathcal{P}}{\partial \mathbf{x}}\right)\sum_{i=0}^{k-1} c_i^k(\mathbf{x}^i - \mathbf{x}^k)$$

$$\approx \mathbf{p}^k - \mathbf{z}^* + \sum_{i=0}^{k-1} c_i^k(\mathbf{p}^i - \mathbf{p}^k). \tag{37}$$

The coefficients c_i^k are subsequently found from minimization of the linearized residual $\mathcal{K}(\mathbf{x}^{new})$

$$\mathbf{c}^k = \arg\min_{\mathbf{c}^k \in \mathbb{R}^k}\left\| \mathbf{p}^k - \mathbf{z}^* + \sum_{i=0}^{k-1} c_i^k(\mathbf{p}^i - \mathbf{p}^k)\right\|, \tag{38}$$

such that $\mathbf{x}^{new}$ can be substituted in Eq. (10) and subsequently linearized to find the update

$$\mathbf{x}^{k+1} \approx \mathbf{x}_{\mathcal{H}}^k + \sum_{i=0}^{k-1} c_i^k(\mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k). \tag{39}$$

The ASM-ILS algorithm is summarized in Algorithm 4. The numerical evaluation of the space-mapping function on line 2 and line 18 in Algorithm 4 is performed using for example Algorithm 3.

Algorithm 4 ASM - ILS

Require: $\mathbf{x}^0, \mathbf{z}^*, \epsilon_I$
 1: $k = 0$
 2: $\mathbf{p}^k = \mathcal{P}(\mathbf{x}^k)$
 3: **while** $\|\mathbf{r}^k\| > \epsilon_I$ **do**
 4:     **if** $k = 0$ **then**
 5:         $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{z}^* - \mathbf{p}^k$
 6:     **else**
 7:         **for** $i = 0$ to $k - 1$ **do**
 8:             $\Delta\mathbf{x}_{\mathcal{H}}^i = \mathbf{x}_{\mathcal{H}}^i - \mathbf{x}_{\mathcal{H}}^k$
 9:             $\Delta\mathbf{p}^i = \mathbf{p}^i - \mathbf{p}^k$
10:         **end for**
11:         $\mathbf{V}^k = \begin{bmatrix} \Delta\mathbf{p}^{k-1} & \Delta\mathbf{p}^{k-2} & \ldots & \Delta\mathbf{p}^0 \end{bmatrix}$
12:         $\mathbf{W}^k = \begin{bmatrix} \Delta\mathbf{x}_{\mathcal{H}}^{k-1} & \Delta\mathbf{x}_{\mathcal{H}}^{k-2} & \ldots & \Delta\mathbf{x}_{\mathcal{H}}^0 \end{bmatrix}$
13:         Calculate $\mathbf{V}^k = \mathbf{Q}^k\mathbf{R}^k$
14:         Calculate $\mathbf{R}^k\mathbf{c}^k = \mathbf{Q}^{kT}(\mathbf{z}^* - \mathbf{p}^k)$
15:         $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{W}^k\mathbf{c}^k + \mathbf{r}^k$
16:     **end if**
17:     $k = k + 1$
18:     $\mathbf{p}^k = \mathcal{P}(\mathbf{x}^k)$
19: **end while**
20: **return** $\mathbf{x}^* = \mathbf{x}^k$

Other inner methods could be used for the numerical evaluation of the space-mapping function such as the QN-ILS method itself, Broyden's method or Aitken's method. As will become clear in Section 4.3, the choice for the inner iterative method depends on how expensive the evaluation of the low fidelity fluid operator is. When the cost of the

low fidelity fluid operator is very low when compared to the high fidelity fluid operator this choice will not affect the total speedup of the ASM-ILS algorithm significantly as long as the inner method result in a stable algorithm. When the cost of the low fidelity fluid operator is low but not several orders lower than the cost of the high fidelity model it is wise to choose an efficient coupling algorithm for the inner method since it can have a large impact on the total speedup of the ASM-ILS algorithm in this case.

4.3. Speedup

The speedup of the ASM-ILS method with respect to the QN-ILS method is determined by the decrease of computational effort per time step to obtain the high fidelity model solution $\mathbf{x}^*$ up to a specified tolerance ϵ_I. Let w^i_f and w^j_c be a measure of the cost (flops or CPU time) necessary to evaluate $\mathcal{R}(\mathbf{x}^i)$ and $\tilde{\mathcal{R}}(\mathbf{z}^j)$ respectively. The average cost per time step of a high fidelity and low fidelity model residual evaluation is then found from

$$\bar{w}_f = \frac{1}{n_f} \sum_{i=0}^{i=n_f} w^i_f \quad \text{and} \quad \bar{w}_c = \frac{1}{n_c} \sum_{j=0}^{j=n_c} w^j_c, \tag{40}$$

where n_f and n_c are the total number of high fidelity (fine) and low fidelity (coarse) iterations respectively (including the iterations necessary to find $\mathbf{z}^*$) per time step. The total cost per time step of the ASM-ILS method is subsequently estimated by

$$W^A \approx \bar{w}^A_f n^A_f + \bar{w}^A_c n^A_c, \tag{41}$$

whereas the total cost of the QN-ILS method is estimated by

$$W^Q \approx \bar{w}^Q_f n^Q_f. \tag{42}$$

In (41) and (42), the superscript A refers to a quantity associated with the Aggressive Space Mapping algorithm and the superscript Q refers to a quantity associated with a conventional Quasi-Newton algorithm.

The estimates in Eqs. (41) and (42) are based on the premise that the largest part of the computational effort is spent in order to evaluate the low fidelity and high fidelity residuals in the computation, neglecting all other (overhead) costs. Numerical experiments justify this premise.

The speedup of the ASM-ILS algorithm relative to the QN-ILS algorithm is subsequently found from the ratio of work per time step

$$S_p \approx \frac{W^Q}{W^A} = \frac{n^Q_f}{n^A_f + \frac{\bar{w}^A_c}{\bar{w}^Q_f} n^A_c}, \tag{43}$$

which is valid when $\bar{w}^A_f \approx \bar{w}^Q_f$. The ASM-ILS method is more efficient than the QN-ILS method if $S_p > 1$. The speedup becomes insensitive to the number of low fidelity residual evaluations n^A_c if the ratio $\frac{\bar{w}^A_c}{\bar{w}^Q_f}$ is sufficiently small. The choice for the inner iterative method does not affect the speedup in this case as long as it results in a stable algorithm for the evaluation of the space mapping function. The expression of speedup in Eq. (43) reflects the principle of space-mapping: If the low fidelity residual function is cheap to evaluate and sufficiently accurate we have $n^A_f < n^Q_f$ and $\frac{\bar{w}^A_c}{\bar{w}^Q_f} n^A_c \ll 1$ resulting in $S_p > 1$. To obtain a metric of the total speedup of a simulation we define

$$\hat{S}_p \approx \frac{\sum W^Q}{\sum W^A}, \tag{44}$$

where the sum is taken over all the time steps in the numerical simulation. The total estimated speedup $\hat{S}_p$ is used to compare the performance of the coupling algorithms while the speedup per time step S_p can be printed after completion of each time-step which gives an indication of the success of the space-mapping algorithm during the simulation.

5. Numerical test cases

In order to assess the performance of the Aggressive Space-Mapping method we apply it to two test cases of increasing complexity. Algorithm 4 is applied to a 2-D test case in Section 5.1: the 2-D supersonic panel problem. In this test case the high and low fidelity models are both linear and the fluid–structure interaction occurs on a line. Several inner iterative algorithms are compared: the Gauss–Seidel, Broyden's and the QN-ILS algorithm. Finally, Algorithm 4 is applied to the 1-D flexible tube problem in Section 5.2. Both the high and low fidelity models use the same set of nonlinear equations describing an incompressible Newtonian fluid in a flexible tube while the fluid–structure interaction occurs on a line. The low fidelity model is constructed using a coarser discretization than the high fidelity model. The QN-ILS algorithm is used for the inner and outer iterative algorithm in this case.

5.1. 2-D supersonic panel problem

In this section the ASM-ILS algorithm is applied to a simple academic test problem – the supersonic panel flutter problem – in order to assess the speedup as defined in Section 4.3. To this end, a structure model, high fidelity fluid model and low fidelity fluid model are defined in Sections 5.1.1–5.1.3 respectively. Finally, numerical experiments are performed in Section 5.1.4 in order to investigate the influence of physical parameters and time step sizes on the speedup of the ASM-ILS algorithm.

5.1.1. Structure model
The flexible panel is governed by the Euler–Bernoulli beam equation

$$
\begin{cases}
\rho_p t_p \dfrac{\partial^2 w}{\partial t^2} + \dfrac{\partial^2}{\partial x^2}\left[\dfrac{EI}{1-\nu^2}\dfrac{\partial^2 w}{\partial x^2}\right] = -\Delta p(x) & \text{on } \Gamma_p, \\
w\left(\pm\dfrac{h}{2}\right) = \dfrac{\partial w}{\partial x}\left(\pm\dfrac{h}{2}\right) = 0.
\end{cases}
\tag{45}
$$

In Eq. (45), ρ_p denotes the density of the panel, t_p the thickness of the panel, E the Young's modulus, ν Poisson's ratio, $I = \frac{1}{12}t_p^3$ the moment of inertia and w the vertical panel displacement. The forcing term is given by the pressure difference over the panel

$$
\Delta p = p_I(x) - p_\infty \quad \text{on } \Gamma_p,
\tag{46}
$$

where p_I denotes the pressure of the fluid on the fluid–structure interface Γ_p and p_∞ the freestream pressure, see Fig. 2. The boundary value problem (45) is discretized using the finite element method with Hermitian shape functions such that the nodal unknowns are displacements *and* rotations.

5.1.2. High fidelity fluid model
The high fidelity fluid is governed by the two-dimensional unsteady linearized potential equation

$$
\begin{cases}
\nabla^2\phi - \dfrac{1}{a_\infty^2}\dfrac{D}{Dt}\left(\dfrac{D\phi}{Dt}\right) = 0 & \text{in } \Omega_f, \\
\phi(x,y) = 0 & \text{on } \Gamma_f, \\
v(x,y) = 0 & \text{on } \Gamma_w, \\
v(x,y) = \dfrac{Dw}{Dt} & \text{on } \Gamma_p,
\end{cases}
\tag{47}
$$

where ϕ denotes the perturbed fluid potential and M_∞ and a_∞ are the freestream Mach number and freestream fluid speed of sound. The substantial derivative in Eq. (47) is given by

$$
\frac{D}{Dt} = \frac{\partial}{\partial t} + M_\infty a_\infty \frac{\partial}{\partial x}.
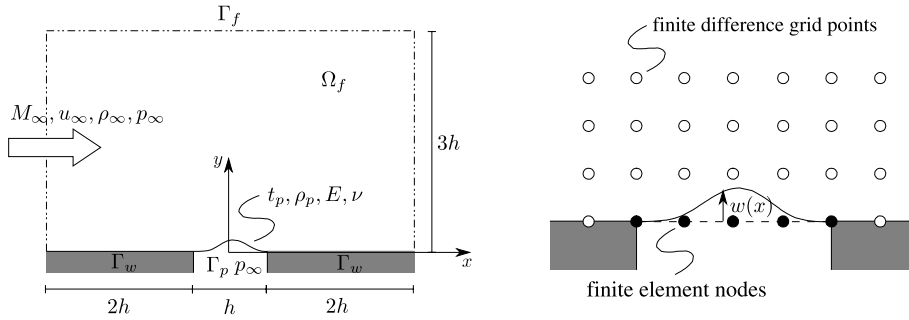\tag{48}
$$

Fig. 2. Schematic representation of the panel flutter problem.

The linearized potential equation is valid for subsonic flow and supersonic flow but is not applicable in the transonic regime. The horizontal velocity component u and vertical component v of the fluid are recovered from the potential according to

$$u = u_\infty + \frac{\partial \phi}{\partial x} \quad \text{and} \quad v = \frac{\partial \phi}{\partial y}. \tag{49}$$

The interface pressure on top of the panel is given by Bernoulli's equation [11]

$$p_I(x) = p_\infty - \rho_\infty \frac{D\phi}{Dt} \quad \text{on } \Gamma_p. \tag{50}$$

The fluid boundary value problem (47) is discretized using the Finite Difference method. The finite difference grid conforms with the finite element mesh of the panel.

5.1.3. Low fidelity fluid model

Using the piston analogy model (see [34]), the interface pressure is approximated by

$$p_I(x) = p_\infty + \rho_\infty M_\infty a_\infty \left(\frac{M_\infty^2 - 2}{\sqrt{(M_\infty^2 - 1)^3}} \frac{\partial w}{\partial t} + M_\infty a_\infty \frac{\partial w}{\partial x} \right), \tag{51}$$

with (x) on Γ_p. The piston analogy is valid for $M_\infty > 1.6$. The pressure in Eq. (51) directly depends on the deflection of the panel. The computational effort is therefore negligible compared to the use of the high fidelity fluid model.

5.1.4. Numerical experiments

The similarity parameters are the Mach number M_∞, the fluid-to-structure mass ratio ζ and the ratio of characteristic time-scales λ defined by

$$\zeta = \frac{\rho_\infty L}{\rho_p t_p} \quad \text{and} \quad \lambda = \frac{L a_\infty^{-1}}{(\rho_p t_p)^{1/2} L^2 (EI)^{-1/2}}. \tag{52}$$

The values of these parameters – for each test case under consideration – are collected in table Table 1. Linear stability analysis is used to obtain the critical Mach number, i.e. the Mach number which separates the stable from the unstable regime. The critical Mach number $M_{cr} = 2.27$ and circular frequency $\omega_{cr} = 460$ rad/s of test case FSI-weak agree with the values reported in [34].

The Newmark-β time integration scheme is used to integrate the structure and fluid equations of motion. This scheme is known to be second order accurate and unconditionally stable for $\gamma = 1/2$ and $\beta = 1/4$. We use the a-form implementation, see [35].

The panel is released from an initial displacement equal to the flutter mode $w^0 = 0.1\xi$, see Fig. 3. The corresponding steady fluid potential ϕ is depicted in Fig. 4.

The simulations are performed at the critical Mach numbers M_{cr}. The observed frequency ω_{cr} in the simulation is then equal to the critical frequency obtained by linear stability analysis. The nondimensional coupled period
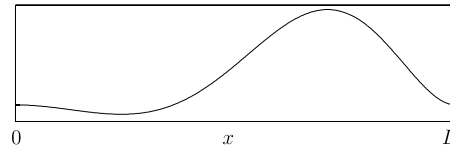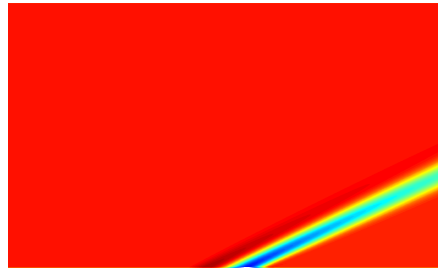
Fig. 3. Initial panel deflection $w^0 = 0.1\xi$.

Fig. 4. Steady initial fluid field.

Table 1
Similarity parameters of the 2-D FSI test cases.

Test case	M_{cr}	ζ	λ
FSI-weak	2.27	5.47e^{-2}	1.47e^{-2}
FSI-medium	2.28	7.41e^{-2}	1.47e^{-2}
FSI-strong	2.33	3.00e^{-1}	1.47e^{-2}

Table 2
Numerical parameters.

Description	Symbol	Value
Fluid grid size: small	$N_x \times N_y$	161×97
Fluid grid size: medium	$N_x \times N_y$	321×193
Fluid grid size: large	$N_x \times N_y$	641×385
Number of finite elements	N_e	$32 \cup 64 \cup 128$
Number of time steps	N_t	$20 \cup 60 \cup 140$
Time step	Δt	$2 P_{cr}/N_t$
Outer tolerance	ϵ_I	$1 \cdot 10^{-6}$
Inner tolerance	ϵ_s	$1 \cdot 10^{-7}$

$P_{cr} = 2\pi a_\infty/\omega_{cr} L$ is divided into 10, 30 and 70 time steps for each test case to study the influence of the time step on the speed up. To study the effect of the fluid grid size on the speedup we do all computations on a small, medium and large fluid grid. The coupled periods of test case FSI-weak, FSI-medium and FSI-strong are $P_{cr}^1 = 8.2$, $P_{cr}^2 = 7.0$ and $P_{cr}^3 = 3.5$ respectively. The numerical parameters used in the simulations are assembled in Table 2. The medium fluid grid size $N_x \times N_y = 321 \times 193$ and number of finite elements $N_e = 64$ in Table 2 correspond to a test case used in [5].

An inner product of the panel displacement with the flutter mode is defined by

$$a_g(t) = \frac{1}{\int_0^L \xi^2(x)dx} \int_0^L \xi(x)w(x,t)dx. \tag{53}$$

The inner product is used to plot the time history of the panel after it is released from its initial deflection. An example is given in Fig. 5a which shows the high fidelity and low fidelity model responses. Both responses demonstrate that the panel oscillates in the flutter mode at the predicted frequency $\omega_{cr} = \frac{2\pi a_\infty}{L P_{cr}} = 539$ rad/s.
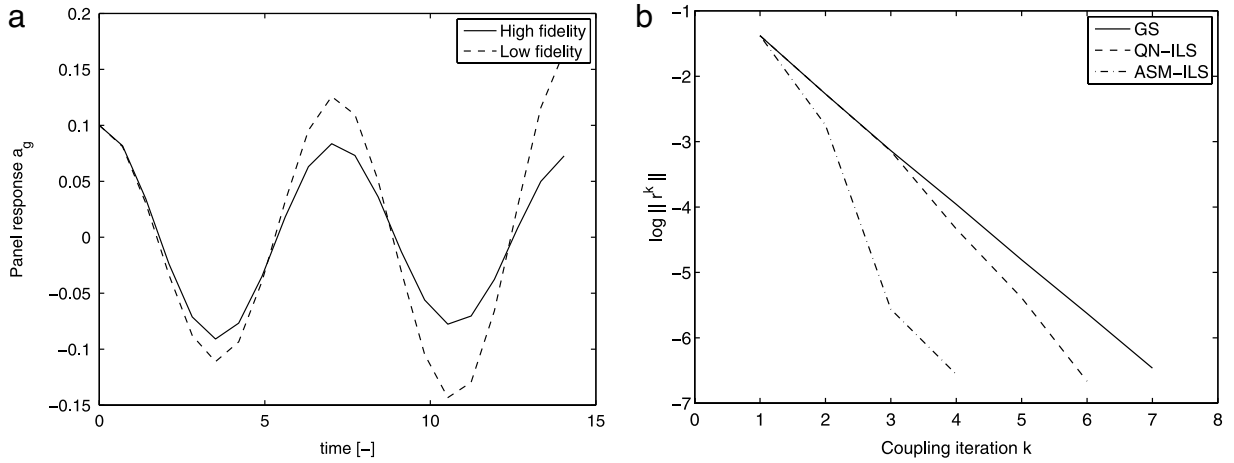
Fig. 5. (a) Panel response a_g for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Interface residual convergence during a representative time step of test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$.
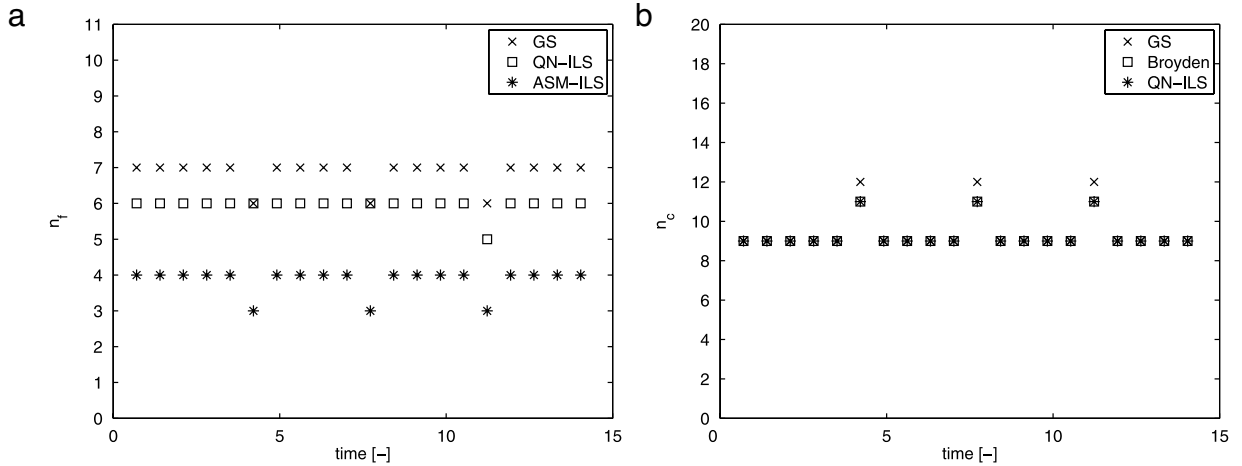

Fig. 6. (a) Number of high fidelity fluid operator evaluations for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Number of low fidelity fluid operator evaluations in the ASM-ILS algorithm for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$.

Fig. 5b shows the interface residual convergence of a representative time step. It can be seen that the ASM-ILS algorithm outperforms the QN-ILS algorithm. An improvement is already visible after the first iteration and becomes better with the adaption of the space mapping Jacobian. The number of high fidelity fluid operator evaluations as a function of time are plotted in Fig. 6a for the Gauss–Seidel, QN-ILS and the ASM-ILS algorithm. The number of low fidelity fluid operator evaluations used in the ASM-ILS algorithm are plotted in Fig. 6b. Several inner iterative algorithms are investigated: the Gauss–Seidel, Broyden's and the QN-ILS algorithm. The number of low fidelity fluid operator evaluations does not depend too much on the type of inner iterative algorithm. This can be explained by the fact that only two or three low fidelity fluid operator evaluations are required per evaluation of the space-mapping function. In that case, Broyden's algorithm and the QN-ILS algorithm converge at a similar rate.

The estimated speedup per time step is shown in Fig. 7a. The speedup is around $S_p \approx \frac{n_f^Q}{n_f^A} = \frac{6}{4}$ for most time steps in the simulation since $\frac{w_c}{w_f}$ is negligible, see Fig. 7b.

Table 3 lists the total speedup of the simulations for all considered time step sizes, grid sizes and FSI cases. The influence of the time step size is large and demonstrates that no speedup is obtained when the time step size becomes too small. In this case only three iterations are sufficient to converge and it becomes difficult to obtain a speedup
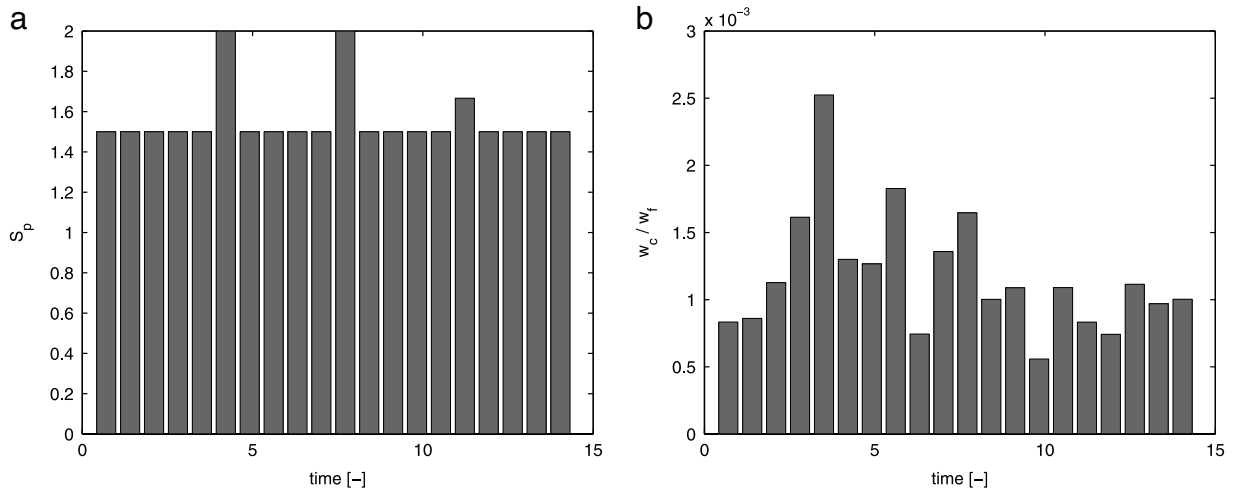
Fig. 7. (a) Estimated speedup per time step for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Average work ratio per time step for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$.

Table 3
Estimated total speedup $\hat{S}_p$ of the simulations.

Grid	FSI	$\Delta t = P_{cr}/10$	$\Delta t = P_{cr}/30$	$\Delta t = P_{cr}/70$
	Weak	1.26	1.30	0.991
Small	Medium	1.27	1.29	0.991
	Strong	1.26	1.18	0.992
	Weak	1.56	1.30	0.997
Medium	Medium	1.54	1.31	0.998
	Strong	1.44	1.30	0.999
	Weak	1.54	1.30	1.000
Large	Medium	1.51	1.31	1.000
	Strong	1.46	1.31	1.000

Table 4
Observed speedup computed using total CPU times.

Grid	FSI	$\Delta t = P_{cr}/10$	$\Delta t = P_{cr}/30$	$\Delta t = P_{cr}/70$
	Weak	1.44	1.23	0.94
Medium	Medium	1.45	1.19	0.95
	Strong	1.29	1.19	0.97

larger than 1. In contrast to fluid–structure interaction problems involving incompressible fluids the added mass effect is not causing problems when small time steps are considered, see [36,37]. The speedup obtained with the ASM-ILS method is therefore limited in this case. The use of a smaller fluid grid results in a smaller speedup when large time step sizes are used in the simulation. This can be attributed to a better numerical conditioning of the FSI problem in this case. There is however almost no difference in speedup between the use of the medium and the large fluid grid, indicating that it is matter of convergence rather than numerical conditioning. The influence of physical parameters on the speedup achieved with the ASM-ILS algorithm is surprisingly small. This requires further investigations.

Table 4 shows the observed speedup measured by the total CPU time ratio for test case FSI-medium on the medium fluid grid. Slightly lower efficiencies are measured due to the overhead costs which are higher for the ASM-ILS algorithm than for the QN-ILS algorithm.

The evaluation of the space-mapping function converges to the desired inner tolerance $\epsilon_s = 1 \cdot 10^{-7}$ in all cases. The low fidelity model is therefore more flexible or equally flexible when compared to the high fidelity model in
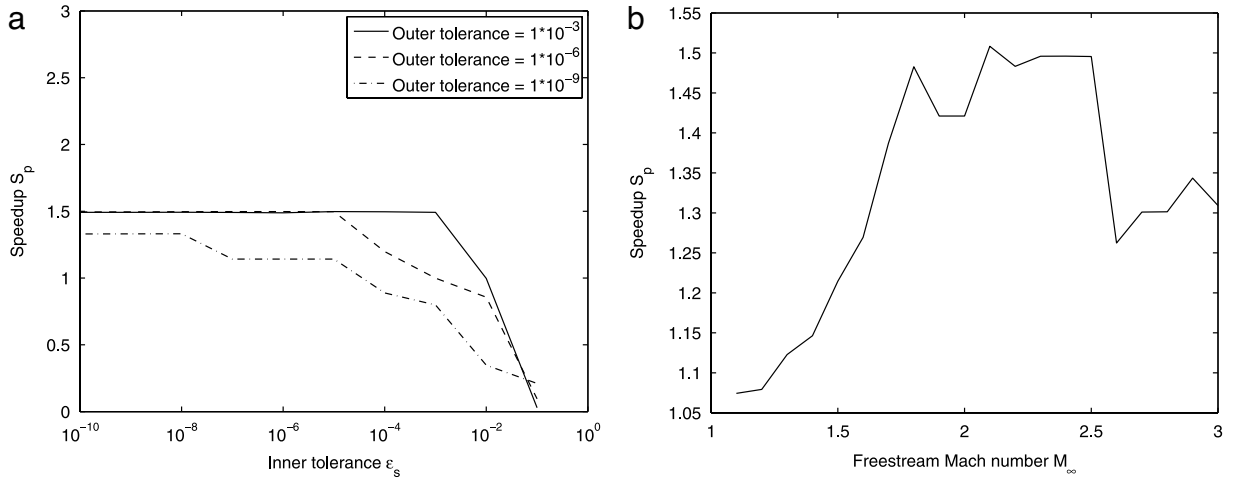
Fig. 8. (a) Total speedup of the ASM-ILS algorithm versus the inner tolerance for several outer tolerances using test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$. (b) Total speedup of the ASM-ILS algorithm versus the freestream Mach number for test case FSI-medium on the medium grid with $\Delta t = P_{cr}/10$.
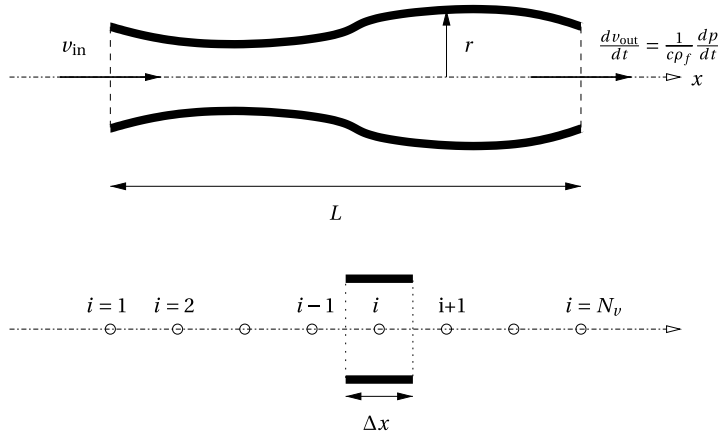

Fig. 9. The conceptual (top) and computational domain (bottom) of the 1D flexible tube problem.

the region of interest. The space-mapping function is therefore either injective or bijective and the ASM algorithm converges to the high fidelity solution. To study the sensitivity of the speedup with respect to the inner tolerance we plot the speedup versus the inner tolerance for a number of outer tolerances, see Fig. 8a. From the figure it becomes clear that $\epsilon_s \leq \epsilon_I$ is a safe choice for the inner tolerance. Choosing a larger inner tolerance has a direct negative consequence on the speedup of the simulation. The speedup is plotted versus the freestream Mach number M_∞ in Fig. 8b. The speedup increases with the freestream Mach number since the low fidelity fluid model becomes more accurate for $M_\infty > 1.6$ while the computational cost of the model remains constant. However for $M_\infty > M_{cr}$, the low fidelity and high fidelity model responses are not similar anymore since the low fidelity model starts to flutter at a slight different freestream Mach number than the critical speed M_{cr} of the high fidelity model. This results in a decrease of the speedup. However, the speedup is found to be larger than one for all Mach numbers considered, indicating that the ASM algorithm is robust.

5.2. 1-D flexible tube problem

In this section we apply the space-mapping algorithms on the fluid–structure interaction between an incompressible flow in a flexible tube and a massless elastic tube wall. This test case is described in detail in [12,38]. Fig. 9 shows the conceptual and computational domain of the flexible tube problem.

5.2.1. Structure model

The cross-sectional area $a = \pi r^2$ and the pressure p in the tube are related by

$$a = a_0 \left(\frac{\frac{p_0}{2\rho_f} - c_{MK}^2}{\frac{p}{2\rho_f} - c_{MK}^2} \right)^2, \tag{54}$$

where ρ_f is the fluid density and p_0 and a_0 are the initial pressure and initial cross-sectional area respectively. The Moens–Korteweg wave speed is given by

$$c_{MK} = \sqrt{\frac{Eh}{2\rho_f r_0}}, \tag{55}$$

where E denotes the Young's modulus and h and r_0 are the wall thickness and initial radius of the tube.

5.2.2. Fluid model

The fluid is considered an incompressible Newtonian fluid. Conservation of mass and momentum yield

$$\frac{\partial a}{\partial t} + \frac{\partial av}{\partial x} = 0, \tag{56}$$

$$\frac{\partial av}{\partial t} + \frac{av^2}{\partial x} + \frac{1}{\rho_f} \left(\frac{\partial ap}{\partial x} - p \frac{\partial a}{\partial x} \right) = 0, \tag{57}$$

which are the continuity equation and the Navier–Stokes equation in conservative form. Here, x denotes the coordinate along the axis of the tube. The velocity along the axis of the tube is denoted by v and p denotes the fluid pressure in the tube. A finite volume discretization of Eqs. (56) and (57) yield

$$\frac{\Delta x}{\Delta t} \left(a_i - a_i^n \right) + v_{i+1/2} a_{i+1/2} - v_{i-1/2} a_{i-1/2} - \frac{\alpha}{\rho_f} \left(p_{i+1} - 2p_i + p_{i-1} \right) = 0, \tag{58}$$

$$\frac{\Delta x}{\Delta t} \left(v_i a_i - v_i^n a_i^n \right) + v_i v_{i+1/2} a_{i+1/2} - v_{i-1} v_{i-1/2} a_{i-1/2}$$

$$\times \frac{1}{2\rho_f} \left(a_{i+1/2}(p_{i+1} - p_i) + a_{i-1/2}(p_i - p_{i-1}) \right) = 0, \tag{59}$$

with $\alpha = a_0/(v_0 + \Delta x/\Delta t)$ a pressure stabilization term. The inlet velocity

$$v_{in} = v_0 + \frac{v_0}{10} \sin^2(\pi n \tau) \quad \text{with } \tau = \frac{v_0 \Delta t}{L}, \tag{60}$$

where n is the current time level.

5.2.3. Space mapping function

The low fidelity model can be constructed using a coarser space discretization than the high fidelity model. For the low fidelity model we use $N_v = N_H$ with $\Delta x = H$ while for the high fidelity model we use $N_v = N_h$ with a smaller $\Delta x = h$. We then have $\mathbf{z} \in \mathbb{R}^{N_H}$ and $\mathbf{x} \in \mathbb{R}^{N_h}$ with $N_H < N_h$. The space mapping function can be chosen as

$$\mathcal{P}(\mathbf{x}) = I_H^h \cdot \arg \min_{\mathbf{z}} \| \tilde{\mathcal{R}}(\mathbf{z}) - I_h^H \cdot \mathcal{R}(\mathbf{x}) \|, \tag{61}$$

where the prolongation operator is denoted by $I_H^h : \mathbb{R}^{N_H} \to \mathbb{R}^{N_h}$ and the restriction operator by $I_h^H : \mathbb{R}^{N_h} \to \mathbb{R}^{N_H}$. Although the space mapping function in (61) is a perfect mapping the function is surjective in the region of interest since high frequency components in the fine model residual $\mathcal{R}$ are filtered out by the restriction operator I_h^H such that it happens that $\mathcal{P}(\hat{\mathbf{x}}) = \mathbf{z}^*$ with $\hat{\mathbf{x}}$ close but not identical to the fine model solution $\mathbf{x}^*$. The high frequency components of the error $\epsilon = \| \hat{\mathbf{x}} - \mathbf{x}^* \|$ cannot be removed by the ASM algorithm as a consequence. It was shown in [22] that standard subiteration provides a good smoother for multi-grid methods. A decision criterium to switch to the smoothing algorithm during the coupling iterations is proposed in [39] where a similar problem was observed

Table 5
Physical parameters.

Description	Symbol	Value
Constant fluid density	ρ	1
Young's modulus	E	1
Tube wall thickness	h	$\sqrt{\frac{4}{\pi}}$
Tube length	L	1

Table 6
Initial conditions and numerical parameters.

Description	Symbol	Value
Initial fluid velocity	v_0	0
Initial fluid pressure	p	0
Switch criterion	D	1
Finite volumes (course/fine)	N_v	$80 \cup 250$
Number of time steps	N_t	400
Time step	Δt	P/N_t
Outer tolerance	ϵ_I	$1 \cdot 10^{-9}$
Inner tolerance	ϵ_s	$1 \cdot 10^{-12}$

with the use of multi-level acceleration and Quasi-Newton acceleration for partitioned strongly coupled fluid–structure interaction. The idea is to split the high fidelity model residual in a part corresponding to the coarse grid and a part corresponding to the fine grid

$$\mathcal{R} = \mathcal{R}_c + \boldsymbol{\delta}_f, \tag{62}$$

with $\mathcal{R}_c = I_H^h \cdot I_h^H \cdot \mathcal{R}$ and $\boldsymbol{\delta}_f$ the high frequency components of the residual. When performing space mapping iterations we have $\mathcal{R}_c \to \mathbf{0}$, while $\mathcal{R} \to \boldsymbol{\delta}_f$ and $\mathbf{x} \to \hat{\mathbf{x}}$ with $\hat{\mathbf{x}} \neq \mathbf{x}^*$. When this happens we switch to the smoothing algorithm, e.g. Gauss–Seidel or Quasi-Newton iterations in order to remove high frequency components of the error. The decision to switch to the smoothing algorithm is then based on

$$D = \frac{\|\mathcal{R}_c\|}{\|\boldsymbol{\delta}_f\|} = \frac{\|\mathcal{R}_c\|}{\|\mathcal{R} - \mathcal{R}_c\|}, \tag{63}$$

which measures the ratio between the coarse and high frequency components of the fine model residual. When $D = 1$ the coarse residual and high frequency components are of the same order. When $D \leq 1$ it is not useful anymore to continue the ASM algorithm, making it is necessary to smooth the error with the smoothing algorithm.

5.2.4. Numerical experiments

The physical parameters of this test case are collected in Table 5. The high fidelity model consists of a finite volume discretization with 250 cells and the low fidelity model discretization uses 80 cells.

Other numerical parameters are collected in Table 6.

In order to demonstrate the effectiveness of space mapping when small time steps are used we choose 400 time steps per period using a simple backward Euler time-integration scheme. The tube area as a function of time at the locations $x = 0$, $x = 1/4$, $x = 3/4$ and $x = 1$ is given in Fig. 10a. As can be seen the area of the tube changes gradually with the inlet velocity of the tube. The maximum theoretical speedup of the ASM-ILS algorithm – the speedup that would be achieved if the cost of evaluating the low fidelity model would be zero – is plotted in Fig. 10b. The maximum theoretical speedup is larger than 1 for all time steps of the simulation which means that solving $\mathcal{P}(\mathbf{x}) - \mathbf{z}^*$ indeed results in a root-finding problem which can be solved faster. The actual coarse model work is however not negligible in this particular test case since there is no exterior domain and the test case is only 1-D. In order to demonstrate the sensitivity of the speedup with respect to the coarse to fine work ratio, Fig. 11a and b show the estimated speedup if the coarse to fine work ratio would have been in the order of a few percent.
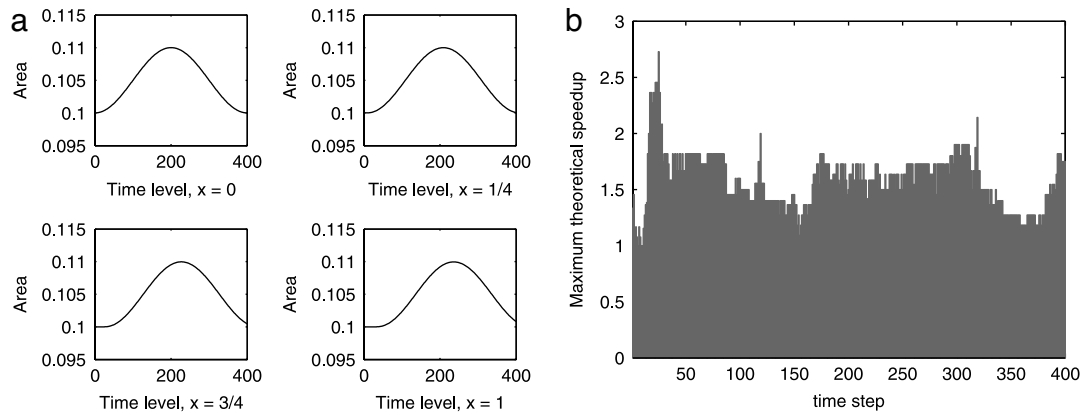
Fig. 10. (a) Tube area as a function of time, $\Delta t = P/400$. (b) Maximum theoretical speedup.

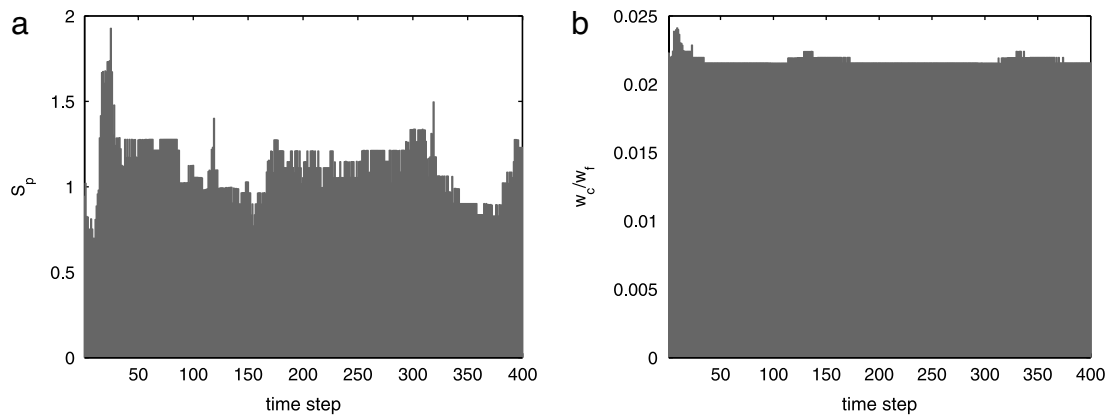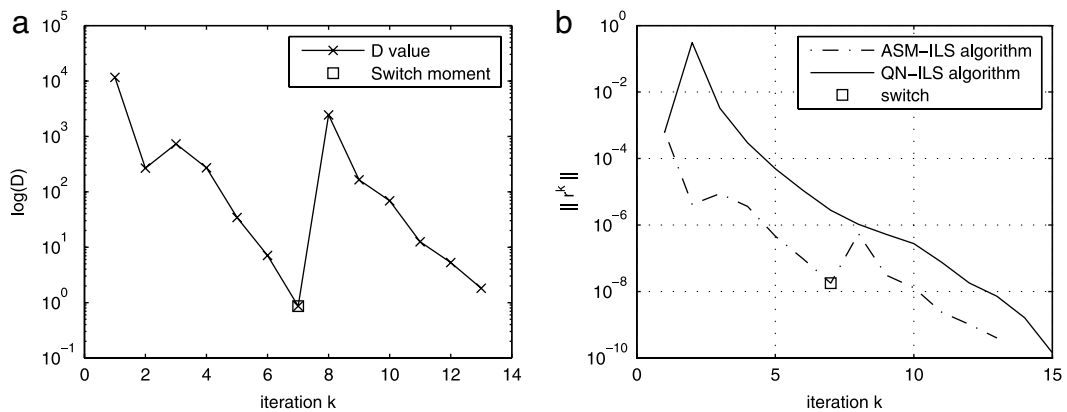Fig. 11. (a) Estimated speedup per time step. (b) Average work ratio per time step.

Fig. 12. (a) D value of the ASM-ILS algorithm at time $n = 3$. (b) Interface residual convergence at time $n = 3$.

Fig. 12a and b show the decision value D and the residual convergence plot at time step $n = 3$. In this particular time step it can be seen that it is necessary to switch to the QN-ILS algorithm after 7 iterations. A jump of the residual norm is present in the ASM-ILS algorithm directly after switching but the ASM-ILS algorithm still converges faster than the QN-ILS algorithm. Fig. 13a and b show the same plots for time step $n = 200$. It can be seen that it is not necessary to switch in this case since the ASM-ILS algorithm converges to the high fidelity solution.
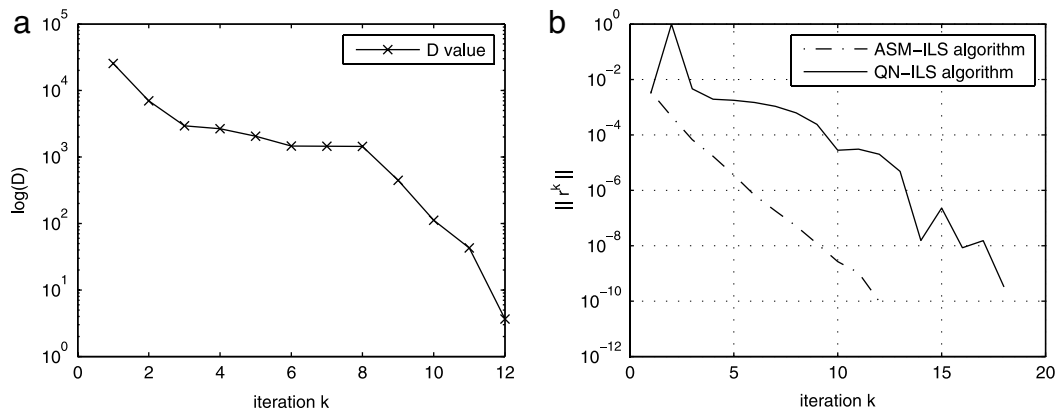
Fig. 13. (a) D value of the ASM-ILS algorithm at time $n = 200$. (b) Interface residual convergence at time $n = 200$.

6. Conclusions

Space-mapping is a general multi-fidelity technique that can be applied to a wide variety of engineering problems. The purpose of this contribution is:

1. To provide a framework for the application of the space-mapping technique to partitioned FSI problems in order to speed up high fidelity simulations.
2. To demonstrate the application of the Aggressive Space-Mapping technique to various academic fluid–structure interaction problems.

Aggressive Space-Mapping has been successfully applied to two academic fluid–structure interaction problems of increasing complexity in order to speed up the partitioned algorithm. This is achieved by the definition of a space-mapping function on the FSI interface together with a proper way of constructing the space-mapping Jacobian using input/output information. For the test cases under consideration we found that:

1. The estimated and observed speedup of the Aggressive Space-Mapping algorithm with respect to the Quasi-Newton algorithm is larger than 1 for most test cases considered.
2. The influence of the time step on the achieved speedup is large compared to other parameters. When compressible flows are studied the speedup is close to 1 for small time steps while for larger time steps the speedup can rise to 1.5. For incompressible flows the fluid–structure interaction is much stronger and for this reason higher speedups can be obtained.

The space-mapping framework allows for the design of new coupling algorithms that – using the information of simplified fluid models – can significantly speed up partitioned strongly coupled FSI simulations. Off the shelf solvers can be reused since the space-mapping technique is solely based on the use of input/output information. This makes the framework especially useful in an industrial environment.

Acknowledgements

References

[1] Y. Bazilevs, J.R. Gohean, T.J.R. Hughes, R.D. Moser, Y. Zhang, Patient-specific isogeometric fluid–structure interaction analysis of thoracic aortic blood flow due to implantation of the jarvik 2000 left ventricular assist device, Comput. Methods Appl. Mech. Engrg. 198 (2009) 3534–3550. Models and Methods in Computational Vascular and Cardiovascular Mechanics.

[2] C.M. Scotti, E.A. Finol, Compliant biomechanics of abdominal aortic aneurysms: A fluid–structure interaction study, Comput. Struct. 85 (2007) 1097–1113. Fourth MIT Conference on Computational Fluid and Solid Mechanics.

[3] C.V. Verhoosel, T.P. Scholcz, S.J. Hulshoff, Uncertainty and Reliability Analysis of Fluid–Structure Stability Boundaries, AIAA J. 47 (2009) 91–104.

[4] P.J. Schmid, E. de Langre, Transient Growth Before Coupled-Mode Flutter, J. Appl. Mech. 70 (2003) 894–901.

[5] C.V. Verhoosel, T.P. Scholcz, S.J. Hulshoff, M.A. Gutiérrez, Transient stability analysis of an aeroelastic problem with random fluid and structure properties, in: Proceedings of the 10th International Conference on Structural Safety and Reliability, ICOSSAR10, 13–17 September 2009, Osaka, Japan, pp. 377–384.

[6] D. Jones, A. Gaitonde, Future Fast Methods for Loads Calculations: The 'FFAST' project, in: D. Knörzer, J. Szodruch (Eds.), Innovation for Sustainable Aviation in a Global Environment, IOS Press BV, 2012, pp. 110–115.

[7] J. Vos, A. Rizzi, D. Darracq, E. Hirschel, Navier–Stokes solvers in European aircraft design, Prog. Aerosp. Sci. 38 (2002) 601–697.

[8] A.H. Zuijlen, S. Bosscher, H. Bijl, Two level algorithms for partitioned fluid–structure interaction computations, Comput. Methods Appl. Mech. Engrg. 196 (2007) 1458–1470.

[9] A.H. Zuijlen, H. Bijl, Multi-Level Acceleration for Sub-Iterations in Partitioned Fluid–Structure Interaction, AIP Conf. Proc. 1168 (2009) 1347–1350.

[10] E. van Brummelen, K. van der Zee, R. de Borst, Space/time multigrid for a fluid–structure-interaction problem, Appl. Numer. Math. 58 (2008) 1951–1971.

[11] J. Hron, S. Turek, A Monolithic FEM/Multigrid Solver for an ALE Formulation of Fluid–Structure Interaction with Applications in Biomechanics, in: H.-J. Bungartz, M. Schäfer (Eds.), Fluid–Structure Interaction, in: Lecture Notes in Computational Science and Engineering, 53, Springer, Berlin, Heidelberg, 2006, pp. 146–170.

[12] J. Degroote, J. Vierendeels, Multi-solver algorithms for the partitioned simulation of fluid–structure interaction, Comput. Methods Appl. Mech. Engrg. 200 (2011) 2195–2210.

[13] C. Michler, H. van Brummelen, R. de Borst, An investigation of Interface-GMRES(R) for fluid–structure interaction problems with flutter and divergence, Comput. Mech. 47 (2011) 17–29. http://dx.doi.org/10.1007/s00466-010-0519-8.

[14] C. Michler, E.H. van Brummelen, R. de Borst, An interface Newton-Krylov solver for fluid–structure interaction, Internat. J. Numer. Methods Fluids 47 (2005) 1189–1195.

[15] U. Küttler, W. Wall, Fixed-point fluid–structure interaction solvers with dynamic relaxation, Comput. Mech. 43 (2008) 61–72.

[16] U. Küttler, W.A. Wall, Vector Extrapolation for Strong Coupling Fluid–Structure Interaction Solvers, J. Appl. Mech. 76 (2009) 021205-1–021205-7.

[17] J. Degroote, K.-J. Bathe, J. Vierendeels, Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction, Comput. Struct. 87 (2009) 793–801. Fifth MIT Conference on Computational Fluid and Solid Mechanics.

[18] R. Haelterman, J. Degroote, D. van Heule, J. Vierendeels, The Quasi–Newton Least Squares method: A new and fast secant method analyzed for linear systems, SIAM J. Numer. Anal. 47 (2009) 2347–2368.

[19] T. Gallinger, K. Bletzinger, Comparison of algorithms for strongly coupled partitioned fluid–structure interaction, in: Proceedings of the 5th European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2010, 14–7 June 2010, Lisbon, Portugal.

[20] R. Haelterman, J. Degroote, D. van Heule, J. Vierendeels, On the Similarities Between the Quasi–Newton Inverse Least Squares Method and GMRes, SIAM J. Numer. Anal. 47 (2010) 4660–4679.

[21] S. Badia, A. Quaini, A. Quarteroni, Modular vs. non-modular preconditioners for fluid–structure systems with large added-mass effect, Comput. Methods Appl. Mech. Engrg. 197 (2008) 4216–4232.

[22] E.H. van Brummelen, Partitioned iterative solution methods for fluid–structure interaction, Internat. J. Numer. Methods Fluids 65 (2011) 3–27.

[23] G.P. Guruswamy, Integrated approach for active coupling of structures and fluids, AIAA J. 27 (1989) 788–793.

[24] L. Djayapertapa, C.B. Allen, Simulation of transonic flutter and active shockwave control, Internat. J. Numer. Methods Heat Fluid Flow 14 (2004) 413–443.

[25] D. Echeverría, P.W. Hemker, Space Mapping and Defect Correction, Comput. Methods Appl. Math. 5 (2005) 107–136.

[26] H. Matthies, R. Niekamp, J. Steindorf, Algorithms for strong coupling procedures, Comput. Methods Appl. Mech. Engrg. 195 (2006) 2028–2049. Fluid–Structure Interaction.

[27] J. Gerbeau, M. Vidrascu, A Quasi–Newton Algorithm Based on a Reduced Model for Fluid–Structure Interaction Problems in Blood Flow, Math. Model. Numer. Anal. 37 (2003) 631–647.

[28] J. Dennis Jr., J. More, Quasi–Newton methods, Motivation and Theory, SIAM Rev. 19 (1977) 46–89.

[29] X. chuan Cai, David, E. Keyes, Nonlinearly preconditioned inexact Newton algorithms, SIAM J. Sci. Comput. 24 (2000) 183–200.

[30] J.W. Bandler, R.M. Biernacki, S.H. Chen, R.H. Hemmers, K. Madsen, Electromagnetic optimization exploiting aggressive space mapping, IEEE Trans. Microw. Theory Tech. 43 (1995) 2874–2882.

[31] S. Minami, S. Yoshimura, Performance evaluation of nonlinear algorithms with line-search for partitioned coupling techniques for fluid–structure interactions, Internat. J. Numer. Methods Fluids 64 (2010) 1129–1147.

[32] T.P. Scholcz, A.H. Zuijlen, H. Bijl, A multi-model incremental adaptive strategy to accelerate partitioned fluid–structure algorithms using space-mapping, in: Proceedings of the 4th International Conference on Computational Methods for Coupled Problems in Science and Engineering, ECCOMAS 2011, 20–22 June 2011, Kos Island, Greece.

[33] M. Bierlaire, F. Crittin, M. Thémans, A multi-iterate method to solve systems of nonlinear equations, European J. Oper. Res. 183 (2007) 20–41.

[34] S. Piperno, C. Farhat, Partitioned procedures for the transient solution of coupled aeroelastic problems – Part ii: energy transfer analysis and three-dimensional applications, Comput. Methods Appl. Mech. Engrg. 190 (2001) 3147–3170.

[35] T.J.R. Hughes, The Finite Element Method, Prentice-Hall, 1987.

[36] E.H. van Brummelen, Added mass effects of compressible and incompressible flows in fluid–structure interaction, J. Appl. Mech. 76 (2009) 021206-1–021206-7.

[37] P. Causin, J. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid–structure problems, Comput. Methods Appl. Mech. Engrg. 194 (2005) 4506–4527.
[38] J. Degroote, P. Bruggeman, R. Haelterman, J. Vierendeels, Stability of a coupling technique for partitioned solvers in FSI applications, Comput. Struct. 86 (2008) 2224–2234.
[39] J.J. Kreeft, M. Weghs, A.H. Zuijlen, H. Bijl, Multi-level and Quasi–Newton acceleration for strongly coupled partitioned fluid–structure interaction, in: Proceedings of the 4th International Conference on Computational Methods for Coupled Problems in Science and Engineering, ECCOMAS 2011, 20–22 June 2011, Kos Island, Greece.