Improving the performance of the partitioned QN-ILS procedure for fluid-structure interaction problems: filtering

R. Haelterman [a,*], A.E.J. Bogaers [b], B. Uekermann [c], K. Scheufele [d], M. Mehl [d],

[a] *Royal Military Academy, Dept. of Mathematics (MWMW), Renaissancelaan 30, B-1000 Brussels, Belgium*

[b] *Advanced Mathematical Modelling, CSIR, Meiring Naudé Road; Brummeria, Pretoria, South Africa*

[c] *Institute for Advanced Study, Technische Universität München, Lichtenbergstraße 2a, D-85748 Garching b. München*

[d] *Universität Stuttgart, Universitätsstraße 38, D-70569 Stuttgart*

Abstract

The Quasi-Newton Inverse Least Squares method has become a popular method to solve partitioned interaction problems. Its performance can be enhanced by using information from previous time-steps if care is taken of the possible ill-conditioning that results. To enhance the stability, filtering has been used. In this paper we show that a relatively minor modification to the filtering technique can substantially reduce the required number of iterations.

1 Introduction

Often in nature different systems interact, like fluids and structures, heat and electricity, populations of species, etc. From the growing number of conferences, publications and software releases it is clear that *in silico* simulations of these kinds of coupled systems are becoming increasingly important in the

* Corresponding author.
Email addresses: Robby.Haelterman@mil.be (R. Haelterman), abogaers@csir.co.za (A.E.J. Bogaers), uekerman@in.tum.de (B. Uekermann), klaudius.scheufele@ipvs.uni-stuttgart.de (K. Scheufele), miriam.mehl@ipvs.uni-stuttgart.de (M. Mehl).

engineering community. Examples can be found in aeronautics (e.g. [38,40–43,62,78,84]), bio-medical science (e.g. [3,7,16,31,32,48,58,79,87,89]), civil engineering (e.g. [13,33,46,60,61,77,81,82,92,93]), plasma physics (e.g. [55,56]), to name but a few. In this paper we will focus solely on fluid-structure interactions.
For these types of problems powerful solvers often already exist for each physical domain (e.g. structural or fluid). Even so, development of similar tools for multi-physics problems is still ongoing and the paths followed to obtain such a solver can be broadly put in one of the following categories:

- *Monolithic or simultaneous solution*: the whole problem is treated as a monolithic entity and solved simultaneously with a specialized *ad hoc* solver.
- *Partitioned solution*: the physical components are treated as isolated entities that are solved separately. Interaction is modeled as forcing terms and/or boundary conditions.

The relative merits of these methods are very problem dependent. The advantage of the monolithic approach is the enhanced stability [72]. This however comes at the cost of having to develop specialized software for each type of interaction problem, where the resulting solution systems are often very large. Furthermore, it can be inappropriate to use the same basic formulation for both types of problems, which further forces the user to treat non-linearities in the same way for all components. Despite this the monolithic approach has proven to be a very popular method, e.g. [4,7,14,59,80,85].

The partitioned approach allows for the use of available specialized solvers for each physical component, on the condition that the coupling effects can be treated efficiently. The latter is often feasible for problems where the systems only weakly interact. Strongly coupled problems, on the other hand, still pose a real challenge. Many articles can be found on partitioned methods in the literature, e.g. [13,23,44,69–71,76,78,88].

In this paper we will solely focus on the partitioned approach, as the main aim of this work is to improve the performance of the Quasi-Newton Least Squares family of methods, in particular QN-ILS [21], which has found widespread acceptance as a means to accelerate the convergence of partitioned solvers. We are not concerned with the solution process of the constituent physical problems as these are assumed to be handled by specialized solvers which we assume to be *black box* operations of which no specific details can be modified or even assessed (e.g. the Jacobian). Furthermore we will assume that the computing time of these black boxes is sufficiently high that the actual computing time of the quasi-Newton step can be neglected.

To accelerate the convergence of a time-dependent problem it can be beneficial to use "histories" from previous time-steps to construct an approximation of the (inverse) Jacobian at the current time-step. This might however lead to numerical breakdown, as the risk of constituent vectors becoming (nearly) linearly dependent increases. To avoid this, QR-filtering has been applied in the past [22]. In this paper we try to improve the convergence performance of QN-ILS by introducing a better form of filtering.

This paper is organized as follows: in §2 we give a short overview of QN-ILS; in §3 we explain the idea of recovery of data from previous time-steps, while in §4 we address the topic of filtering. The currently used method of QR-filtering is explained, together with two new approaches, one based on the ideas of the orthogonalization performed in QR-decomposition, the other on that of Proper Orthogonal Decomposition. In §5 the performance of the different filtering techniques is tested and compared using different numerical applications, after which we end with a short conclusion.

2 The Quasi-Newton Inverse Least Squares (QN-ILS) method

2.1 Problem setting

In general we are interested in non-linear surface-coupled problems that can be mathematically stated in the following form:

$$\begin{cases} F(g) = p \\ S(p) = g, \end{cases} \tag{1}$$

where $F : D_F \subset \mathbb{R}^m \to \mathbb{R}^n : g \mapsto F(g)$ and $S : D_S \subset \mathbb{R}^n \to \mathbb{R}^m : p \mapsto S(p)$.
Each equation describes (the discretized equations of) a physical problem that is spatially decomposed. In fluid-structure interaction problems $F(g) = p$ could give the pressure p on the interface between fluid and structure for a given geometry g, while $S(p) = g$ could give the deformed geometry of that same interface under influence of the pressure exerted on it by the fluid.
We limit p and g to values on the interface between the two physical problems. In this way the physically decoupled nature of the problem is exploited. This approach can be regarded as a special case of *heterogeneous domain decomposition methods* [28] and limits the number of variables that the coupling technique will be dealing with, even though the black box solvers that give $F(g)$ and $S(p)$ might use a substantially higher number of internal variables; for instance in the case of a fluid-structure interaction problem where the pres-

sure is passed from the fluid to the structure, the fluid velocity is an internal variable for the flow solver as are all nodal values of the pressure that are not on the interface.

Alternatively, (1) can be written as the fixed point problem

$$F(S(p)) = H(p) = p \tag{2}$$

or the root-finding problem

$$H(p) - p = K(p) = 0. \tag{3}$$

Using (2) or (3) means that we have actually lumped both systems F and S together into one system (either H or K), which in general has a lower number of variables than the sum of the number of variables of both constituent systems F and S.

We assume that H and K satisfy the following hypotheses, which are typical when working with Newton and quasi-Newton type methods [74]:

(1) H (and hence K) are continuously differentiable in an open set D.
(2) $K(p) = 0$ has one solution p^* in D_K.
(3) $(K'(p))^{-1}$ exists and is continuous in an open set containing p^*.

Remark 2.1 *We assume the operations $F(g)$ and $S(p)$ (and hence $H(p)$ and $K(p)$) are black box systems, representing the propriety solvers, with a high computational cost and of which nothing is known about the Jacobian; neither do we make assumptions about this Jacobian like sparseness, symmetry, etc. For that reason we count the performance of a method by the number of times $F(g)$ or $S(p)$ are executed. Requirements like actual CPU-time or storage are not taken into consideration.*

Remark 2.2 *While we write the equations in (1) in explicit form, this is only for convenience; any form is usable as long as for a given value of g (resp. p) a corresponding value of p (resp. g) can be computed that satisfies the equations in (1).*

Remark 2.3 *We could have used (and sometimes will use)*

$$S(F(g)) - g = 0 \tag{4}$$

instead of (2). The choice between both can depend on

- *practical implementation issues due to the solvers used;*
- *the relative sizes of n and m. If $n < m$, resp. $n > m$, the use of (3), resp. (4), will result in a problem that is defined on a space with the lowest dimension.*

Remark 2.4 *When (1) is derived from a physical problem, it often represents the equations obtained after discretizing the continuous equations in time and space, and thus only represents the evolution over one time-step (see §3). This is an example of how we could be presented with a series of related problems. In this context we can write (1) as*

$$\begin{cases} F_{t+1}(g, p_t, g_t) = p \\ S_{t+1}(p, p_t, g_t) = g, \end{cases} \tag{5}$$

where the subscript $t+1$ ($t = 0, 1, \dots$) denotes the time-level at which the problem is solved. The solution of (5) will give the values of p and g at that time-level (p_{t+1}, resp. g_{t+1}); the extra arguments p_t and g_t are added to show that the solution at the next time-level depends on the values at the previous time-level[1] *.*
In what follows we will almost always simply write $F(g)$ and $S(p)$ and assume it is clear from the context that this either describes an isolated problem or a problem solved over one time-step.

2.2 QN-ILS

The Quasi-Newton Least Squares family of algorithms was developed starting from the Block Quasi-Newton Least Squares algorithm (BQN-LS) [19,20,89] even though the name BQN-LS was not used until later [21]. From BQN-LS, the quasi-Newton Least Squares algorithm (QN-LS) was developed, which was first introduced in [51]. BQN-LS and QN-LS are very closely related to the point that for affine problems both are algebraically identical [54].

The Quasi-Newton Inverse Least Squares algorithm (QN-ILS) was derived from QN-LS in [21] and further generalized in [24]. It quickly overtook QN-LS in popularity, probably as it directly gives the inverse of the Jacobian, which is actually needed [2,6,10–12,17,25,27,29,34–36,49,50,52,64,65] [2] .

[1] It is possible that it depends on more than one of the previous time-levels.
[2] The method is also known as Anderson acceleration going back to 1965 [1] and was mainly applied in electronic structure computation (e.g. [73]). Under this name is method recently re-attracted new attention [90,37] and showed potential in further

QN-ILS is applied to equation (3) and can be described as follows.

1. Startup:
   a. Take an initial value p_o.
   b. Compute $p_1 = (1-\omega)p_o + \omega H(p_o)$.
   c. Set $s = 1$.
2. Loop until sufficiently converged:
   a. Compute $K(p_s)$.
   b. Construct the approximate inverse Jacobian $\hat{M}'_s$ as given below [a].
   c. Quasi-Newton step: $p_{s+1} = p_s - \hat{M}'_s K(p_s)$.
   d. Set $s = s + 1$.

[a] I.e. $\hat{M}'_s$ is an approximation of $(K'(p_s))^{-1}$, where $K'(p_s)$ is the Jacobian of K evaluated for p_s.

In this algorithm ω represents a relaxation parameter, which we apply to avoid excessive initial divergence.

The construction of the approximate inverse Jacobian is obtained as follows.

Let $G(y) = H(K^{-1}(y))$, then

$$\begin{aligned} K^{-1}(y) &= \mathcal{I}(K^{-1}(y)) \\ &= (H - K)(K^{-1}(y)) \\ &= H(K^{-1}(y)) - K(K^{-1}(y)) \\ &= H(K^{-1}(y)) - \mathcal{I}(y) \\ &= G(y) - \mathcal{I}(y) \end{aligned} \tag{6}$$

It follows that $(K^{-1})'(y) = G'(y) - I$.

Thus, we approximate $(K^{-1})'$ using an approximation $\hat{G}'_s$ of G' that is computed based on recorded input and output values:

$$\begin{aligned} \delta y_i^s &= y_s - y_i \;\; (i = 0, \ldots, s-1), \\ V_s^{yG} &= [\delta y_{s-1}^s \; \delta y_{s-2}^s \; \ldots \; \delta y_0^s] \in \mathbb{R}^{n \times s}, \\ \delta G_i^s &= G(y_s) - G(y_i) \;\; (i = 0, \ldots, s-1), \\ W_s^{yG} &= [\delta G_{s-1}^s \; \delta G_{s-2}^s \; \ldots \; \delta G_0^s] \in \mathbb{R}^{n \times s}. \end{aligned}$$

applications like e.g. groundwater flow [67].

$$\hat{G}'_s = W_s^{wG} \left((V_s^{wG})^T V_s^{wG} \right)^{-1} (V_s^{wG})^T,$$
$$\widehat{(K^{-1})'}_s = \hat{G}'_s - I = W_s^{wG} \left((V_s^{wG})^T V_s^{wG} \right)^{-1} (V_s^{wG})^T - I.$$

Setting $K^{-1}(y_i) = p_i$ for $i = 0, 1, \ldots$ (i.e., $y_i = K(p_i)$ and $G(y_i) = H(K^{-1}(y_i)) = H(p_i)$) and modifying the notation accordingly, we get

$$\widehat{(K^{-1})'}_s = \hat{M}'_s = W_s \left((V_s)^T V_s \right)^{-1} (V_s)^T - I, \tag{7}$$

where

$$\delta K_i^s = K(p_s) - K(p_i) \quad (i = 0, \ldots, s-1), \tag{8a}$$
$$V_s = [\delta K_{s-1}^s \; \delta K_{s-2}^s \; \ldots \; \delta K_0^s] \in \mathbb{R}^{n \times s}, \tag{8b}$$
$$\delta H_i^s = H(p_s) - H(p_i) \quad (i = 0, \ldots, s-1), \tag{8c}$$
$$W_s = [\delta H_{s-1}^s \; \delta H_{s-2}^s \; \ldots \; \delta H_0^s] \in \mathbb{R}^{n \times s}. \tag{8d}$$

Remark 2.5 *This method is sometimes called Interface Quasi-Newton Inverse Least Squares to emphasize the use of interface variables, even though this does not change the method itself [15,26,66,86].*

Remark 2.6 *If needed, filtering will be applied to V_s and W_s (cf. §4).*

3 Recovery of data from previous time-levels

When the problem to be solved is time-dependent, and if we assume that the input-output pairs of previous time-levels are representative enough for the current time-level, we might think of enhancing the Jacobian by taking these into account as follows:

$$\mathbf{V}_{s,t} = [V_{s,t} \mid V_{\text{final},t-1} \mid \ldots \mid V_{\text{final},t-\zeta}] \tag{9a}$$
$$\mathbf{W}_{s,t} = [W_{s,t} \mid W_{\text{final},t-1} \mid \ldots \mid W_{\text{final},t-\zeta}], \tag{9b}$$

where $V_{s,t}$ and $W_{s,t}$ are constructed at the current time-level t and current iteration s as in §2.2, $V_{\text{final},t-i}$ and $W_{\text{final},t-i}$ $(i = 1, \ldots, \zeta)$ are the input and output matrices constructed as in §2.2 at the end of the iteration process at

time-level $t-i$ and ζ is a parameter that determines how many time-levels are kept.

The Jacobian at iteration s of time-level t is then constructed as:

$$\hat{M}'_{s,t} = \mathbf{W}_{s,t} \left((\mathbf{V}_{s,t})^T \mathbf{V}_{s,t} \right)^{-1} (\mathbf{V}_{s,t})^T - I. \tag{10}$$

We start the first iteration of a new time-level ($t > 1$) by computing

$$p_{1,t} = p_{o,t} - \hat{M}'_{\text{final},t-1} K_t(p_{o,t}),$$

where $\hat{M}'_{\text{final},t-1}$ is the approximate Jacobian at the last iteration of the previous time-level; this means that we set $\hat{M}'_{o,t} = M'_{\text{final},t-1}$. (For the first time-level we implicitly used $\hat{M}'_{o,1} = -I$, possibly combined with under-relaxation.) A first input-output pair for the new time-level can then be computed based on $p_{1,t} - p_{o,t}$ and $H_t(p_{1,t}) - H_t(p_{o,t})$.
The value $p_{o,t}$ is obtained by linear extrapolation based on $p_{\text{final},t-1}$ and $p_{\text{final},t-2}$ (if available), which are the last iterates at time step $t-1$ and $t-2$ respectively [3].

Some caution is needed when using this method. First of all, the choice of the parameter ζ is difficult, as it is not always clear *a priori* how many time-levels can be kept, i.e. how long old data will be representative for the problem at the current time-level. Secondly, as explained in §4, filtering becomes even more important when previous histories are retained.

4 Avoiding (near-)singularity through filtering

4.1 The need for filtering

For reasons of stability the following formula for QN-ILS, (cf. §2.2) is not used in the actual computation [4]

$$p_{s+1} = p_s - (W_s \left((V_s)^T V_s \right)^{-1} (V_s)^T - I) K(p_s) \tag{11}$$

[3] We assume that at the last iteration convergence has been reached.
[4] We will use the notation for the case where no histories are kept to enhance readability. When histories are kept, the notation should be modified accordingly.

but replaced by

$$p_{s+1} = p_s - (W_s(R_s)^{-1}(Q_s)^T - I)K(p_s), \tag{12}$$

where $V_s = Q_s R_s$ is the economy size QR-decomposition, i.e. $Q_s \in \mathbb{R}^{n \times s}$ and $R_s \in \mathbb{R}^{s \times s}$.

Although we have shown in [51] and [53] that singularities in the construction of the approximate (inverse) Jacobian of QN-LS and QN-ILS cannot occur when the mappings are affine and when working in exact arithmetic, it is quite possible that the columns of V_s become linearly dependent of one another when the mappings are non-linear, when rounding errors are present or when the total number of (column-)vectors in V_s exceeds n, e.g. when many previous histories are retained.
For that reason filtering is sometimes necessary to allow the construction of the Jacobian (when $s > n$) and/or to avoid singularity of the approximate Jacobian (when the columns of V_s do not form a linearly independent set).

4.2 QR-filtering

QR-decomposition of V_s is an effective way to indicate which columns of V_s are (nearly) linearly dependent [5]. Currently most applications that use QN-ILS use QR-filtering based on the method first applied in [22]. This method can be described as follows [6].

(1) Make an economy size QR-decomposition of V starting from the newest vector and choose a threshold ϵ.
(2) if $|R_{ii}| < \epsilon \cdot \|R\|_2$, then eliminate the i-th column of V and W and re-start the procedure.

We will call this method *old QR*(-filtering) in the remainder of the paper.

[5] In this context, it is important that the QR-decomposition is started with the vector that is added last, as this one is probably a better representation of the current iterate than earlier vectors. This, however, incurs an extra cost as the QR-decomposition needs to be recomputed every time, while starting with the oldest vector would allow the re-use of previous QR-decompositions.
[6] We will drop the subscripts in this part to enhance readability.

4.3 An alternative approach to QR-filtering

We will compare the approach in §4.2 with the following, *new QR* filtering, which is also based on the (economy size) QR-decomposition.

(1) Let η be the number of columns of $V = [\bar{V}_1|\bar{V}_2|\dots|\bar{V}_\eta]$ (numbered so that the lowest index corresponds to the newest vector), and ϵ a certain threshold value.
(2) Set $i = 1$. $R_{11} = \|\bar{V}_1\|_2$ and $\bar{Q}_1 = \frac{\bar{V}_1}{R_{11}}$.
(3) for $i = 2, \dots, \eta$ do:
  (a) $\bar{v} = \bar{V}_i$;
  (b) for $j = 1, 2, \dots, i-1$ do:
    (i) $R_{ji} = \bar{Q}_j^T \cdot \bar{v}$.
    (ii) $\bar{v} = \bar{v} - R_{ji} \cdot \bar{Q}_j$
  (c) if $\|\bar{v}\|_2 < \epsilon \|\bar{V}_i\|_2$ then remove the i-th column of V and W and restart the procedure,
  otherwise $R_{ii} = \|\bar{v}\|_2$ and $\bar{Q}_i = \frac{\bar{v}}{R_{ii}}$.

In the previous approach (§4.2) ϵ depends on how accurately the flow and structural equations are solved. An appropriate value for ϵ can be determined by analyzing the change of the output of K due to a small perturbation of the input of K. If the perturbation is too small, the resulting change will be numerical noise [22]. The approach is thus numerical in nature.

In the newly proposed method we compare the norm of a column of a matrix with that of other columns to indicate possible linear dependencies; the approach is thus purely algebraic.

Note also that the criterion of the method in §4.2 can be written as

$$|R_{ii}| < \epsilon \cdot \|R\|_2 = \sqrt{\sum_{j=1}^{\eta} \sum_{i=1}^{\eta} {R_{ji}}^2} \tag{13}$$

while the criterion in the newly proposed filtering algorithm is

$$|R_{ii}| < \epsilon \cdot \|\bar{R}_i\|_2 = \sqrt{\sum_{j=1}^{\eta} {R_{ji}}^2}. \tag{14}$$

An inconvenience of the method in §4.2 is that it would filter out a completely linearly independent column just because it is small compared to other

columns, e.g. when values change little between iterations.

4.4 POD-filtering

A completely different approach to filtering out (nearly) linearly dependent vectors is based on Proper Orthogonal Decomposition (POD) [9,83]. It can be written as follows.

(1) Let η be the number of columns of V_s and ϵ a certain threshold value.
(2) Create the autocorrelation matrix $\Sigma = \frac{1}{\eta}(V_s)^T V_s$.
(3) Compute the eigenvalues λ_i and eigenvectors $\bar{X}_i$ $(i = 1, \ldots, \eta)$ of Σ, such that $\Sigma X = X\Lambda$, with $X = [\bar{X}_1|\bar{X}_2|\ldots|\bar{X}_\eta]$, $\Lambda = \text{diag}\{\lambda_1, \ldots, \lambda_\eta\}$ and $\lambda_1 \geq \lambda_2 \geq, \ldots, \geq \lambda_\eta$.
(4) Determine c such that $\frac{\lambda_c}{\lambda_1} \leq \epsilon < \frac{\lambda_{c-1}}{\lambda_1}$. If such a value of c does not exist, take $c = \eta$.
(5) Create $\mathcal{V} = [\overline{\mathcal{V}_1}|\overline{\mathcal{V}_2}|\ldots|\overline{\mathcal{V}_\eta}] = VX$ and $\mathcal{W} = [\overline{\mathcal{W}_1}|\overline{\mathcal{W}_2}|\ldots|\overline{\mathcal{W}_\eta}] = WX$
(6) Truncate: $V_{s,POD} = [\overline{\mathcal{V}_1}|\overline{\mathcal{V}_2}|\ldots|\overline{\mathcal{V}_c}]$ and $W_{s,POD} = [\overline{\mathcal{W}_1}|\overline{\mathcal{W}_2}|\ldots|\overline{\mathcal{W}_c}]$
(7) Create the (I)LS Jacobian $W_{s,POD}((V_{s,POD})^T V_{s,POD})^{-1}(V_{s,POD})^T - I$ and use in QN-LS or QN-ILS.

5 Test-cases

We now present the key results for two test-cases. More data for these test-cases and two extra test-cases can be found in [57].
Note that in all test-cases the results obtained with any of the methods were identical (within convergence tolerance) when they converged. The only difference lies with the convergence speed of the methods.

5.1 Wave Propagation in a Three-Dimensional Elastic Tube

The flexible tube example [5,19,45] simulates a wave propagating in a straight elastic tube with a length of 0.05 m, a wall thickness of 0.001 m and an inner diameter of 0.01 m. Both ends of the tube are fixed. A pressure driven flow is generated by setting the boundary condition for the pressure inlet to a peak value of 1333.2Pa for a initial duration of 0.003s. After this time, the inlet pressure is set to zero. At the right boundary, the pressure is set to zero at all times. The fluid has a density ρ_f of $10^3\,\text{kg/m}^3$ and a dynamic viscosity ν of $3.0 \cdot 10^{-3}$ or $3.0 \cdot 10^{-2}$ $\text{Pa}\cdot\text{s}$ (as indicated in Tables 1-3). The structure is

assumed to be elastic and compressible with a density ρ_s of $1.2 \cdot 10^3$ or $1.2 \cdot 10^2$ kg/m^3 (as indicated in Tables 1-3), a Young's modulus of $3.0 \cdot 10^5$ N/m^2, and a Poisson's ratio of 0.3.

The solver that is used is *OpenFOAM*[7], an open source numerical simulation toolbox for problems in continuum mechanics with emphasis on flow simulation. It was used for the fluid and the structure simulations based on a second order finite volume discretization of the incompressible Navier-Stokes equation for the fluid. Instead of the standard PISO (pressure implicit with splitting of operator) algorithm it uses a coupled solution algorithm as described in [18]. I.e., it assembles and solves as a whole a large diagonally-dominant matrix for both velocities and pressure. Hereby, the pressure equation is derived in a similar manner as in the SIMPLE method [75]. The governing fluid equations are formulated in the Arbitrary-Lagrangian-Eulerian point of view and mesh movement is done via radial-basis function interpolation. Time integration is done using a second order backward differencing scheme. The structural solver uses a full Lagrangian formulation and a Saint-Venant-Kirchhoff material model. For more information, refer to e.g. [8].
The flow solver uses a mesh with 20 800 cells, whereas the structure mesh has 6 400 cells; the time-frame consists of 100 time-steps of 10^{-4}s.
The initial solution of the displacement is determined with a state extrapolation from previous time steps for each numerical method under consideration. The pressure pulse that propagates through the tube and the arising wave in the tube wall are shown in Figure 1.

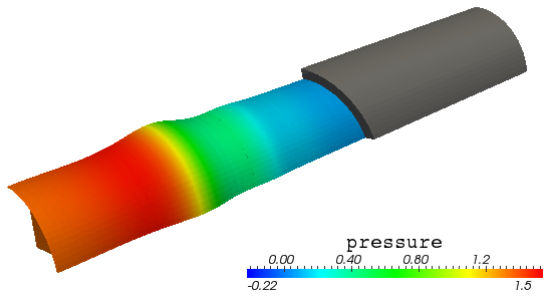

Fig. 1. Wave propagation in a three-dimensional elastic tube. Geometry and pressure contours on the fluid-structure interface at $t = 3.0 \cdot 10^{-1}$s

For the coupling between the fluid and the solid solver, we use the coupling library *preCICE*[8], while the OpenFOAM coupling adapters are provided by David Blom et al.[9]. *preCICE* is a library for flexible numerical coupling of single-physics solvers. It is developed at the Technische Universität München

[7] `http://www.openfoam.org/`
[8] `http://www.precice.org`
[9] `https://github.com/davidsblom/FOAM-FSI`

and the Universität Stuttgart [47]. preCICE uses a partitioned black-box coupling approach that allows for a flexible and minimal invasive coupling with a wide range of single-physics solvers. Its equation coupling module provides a wide variety of runtime configurable aspects of numerical coupling, like serial or parallel as well as explicit and implicit coupling schemes along with several acceleration schemes, so called post-processing methods.

Two coupling schemes were used

(1) The serial implicit coupling scheme (S-System) is the standard approach for the coupling of black box solvers in a partitioned FSI setting [8,22,39,44]. The field solvers are executed in staggered way, i. e., first the flow solver (F) computes forces (p) from the interface displacements (g) and velocities for the current time step. The structural solver takes these forces as input values and computes the new interface displacements and velocities afterwards. The corresponding fixed-point equation reads

$$g \stackrel{!}{=} S \circ F(g) \ .$$

However, for massively parallel simulations, the inherent serial calling order of the field solvers leads to performance issues due to bad load balancing. There is a significant mismatch of work load between the structural and the fluid solver, which imposes limitations on the parallel efficiency.

(2) The parallel implicit coupling scheme (V-System) was developed to overcome those limitations by evaluating the structural and fluid solver in parallel. Hereby, the V-System uses the original input/output relation for both solvers but the boundary values are exchanged after each simultaneous solving of structure and flow. The vectorial fixed-point equation reads

$$\begin{pmatrix} p \\ g \end{pmatrix} \stackrel{!}{=} \begin{pmatrix} 0 & F \\ S & 0 \end{pmatrix} \begin{pmatrix} p \\ g \end{pmatrix} .$$

If solved by a pure fixed-point iteration, the vectorial system results in two independent instances of the S-System. Quasi-Newton solvers turn out to be powerful enough that one iteration of the V-System is comparable to one iteration of the S-System [15,86].

A relative convergence criterion

$$\text{Rel.Crit.} := \max \left\{ \frac{\|K(p)\|_2}{\|H(p)\|_2}, \frac{\|K(g)\|_2}{\|H(g)\|_2} \right\} \leq 10^{-5}$$

or an absolute convergence criterion

$$\text{Abs.Crit.} := \|K(g)\|_2 \leq 10^{-7}$$

is used.

The optimal results for each filtering technique are given in Tables 1-3. More data can be found in [57]. It is seen that filtering can indeed enhance the convergence speed and that in this respect the new QR filtering gives the best results. The relative performance of the old QR and POD filtering depends on the settings of the test-case, with POD, in general, being slightly better.

	S-QN-ILS	V-QN-ILS
old QR	8.51 for $\epsilon = 10^{-11}$	9.82 for $\epsilon = 10^{-9}$
new QR	5.82 for $\epsilon = 10^{-2}$	8.67 for $\epsilon = 10^{-2}$
POD	7.40 for $\epsilon = 10^{-16}$	9.74 for $\epsilon = 10^{-16}$
no filter	8.64	9.85

Table 1
Best results for the wave propagation in a 3D tube. Mean iteration numbers over the first 100 time steps for the V- and S-QN-ILS coupling scheme and different filtering techniques. Forces are scaled with $f_{force} = 10^{-10}$ for better conditioning. The information from the past is retained for $R = 30$ time steps. **The relative convergence criterion is used.** $\rho_s = 1.2 \cdot 10^3$ **and** $\nu = 3.0 \cdot 10^{-3}$**.**

	S-QN-ILS	V-QN-ILS
old QR	7.65 for $\epsilon = 10^{-9}$	9.45 for $\epsilon = 10^{-9}$
new QR	5.74 for $\epsilon = 10^{-2}$	7.43 for $\epsilon = 10^{-6}$
POD	7.35 for $\epsilon = 10^{-14}$	9.23 for $\epsilon = 10^{-14}$
no filter	7.78	9.69

Table 2
Best results for the wave propagation in a 3D tube. Mean iteration numbers over the first 100 time steps for the V- and S-QN-ILS coupling scheme and different filtering techniques. Forces are scaled with $f_{force} = 10^{-10}$ for better conditioning. The information from the past is retained for $R = 30$ time steps. **The relative convergence criterion is used.** $\rho_s = 1.2 \cdot 10^3$ **and** $\nu = 3.0 \cdot 10^{-2}$**.**

5.2 Flow induced oscillating flexible beam

We consider the popular flexible beam benchmark problem first introduced by Wall [91]. The problem consists of analyzing flow around a flexible tail attached to a rigid bluff body, with large deformations induced by oscillating vortices formed by flow around the square bluff body. The geometry and discretization

	S-QN-ILS
old QR	8.31 for $\epsilon \leq 10^{-10}$
new QR	7.20 for $\epsilon = 10^{-2}$
POD	8.37 for $\epsilon = 10^{-18}$
no filter	8.31

Table 3
Best results for the wave propagation in a 3D tube. Mean iteration numbers over the first 100 time steps for the S-QN-ILS coupling scheme and different filtering techniques. The information from the past is retained for $R = 30$ time steps. **The absolute convergence criterion 1 is used.** $\rho_s = 1.2 \cdot 10^2$ **(for fostering instabilities) and** $\nu = 3.0 \cdot 10^{-2}$**.** V-QN-ILS is not shown as it failed to converge.

used for the test is illustrated in Figure 2 along with the beam displacement at $t = 4.29$s in Figure 3.

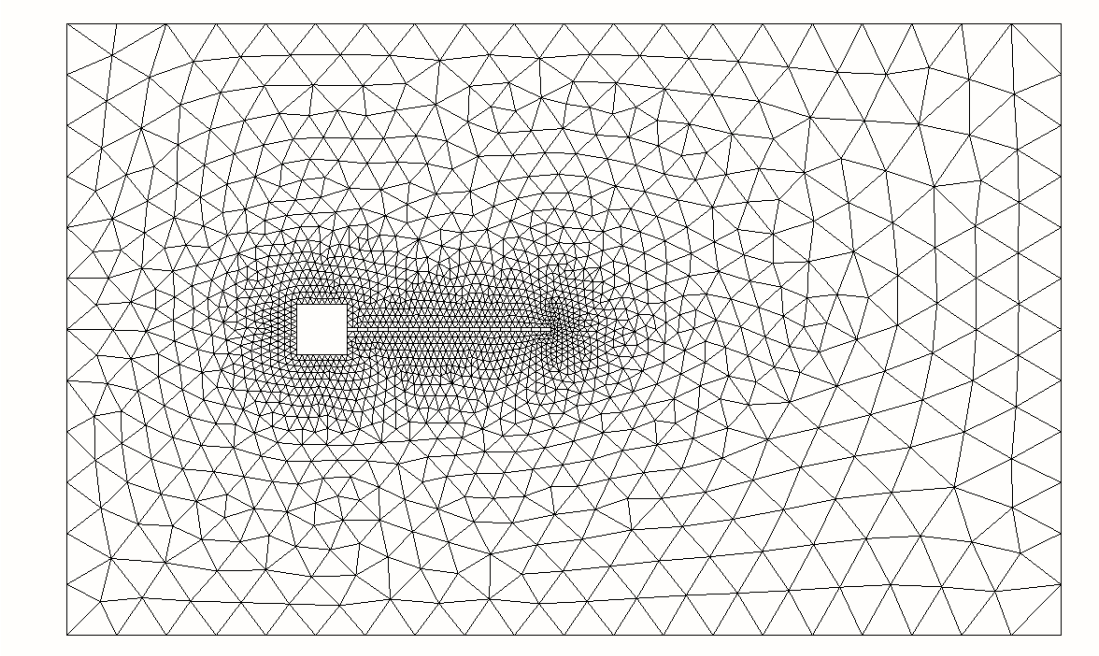

Fig. 2. Discretization used for flexible tail benchmark problem.

OpenFOAM was once again used, and analysed here using S-QN-ILS. While not solved using *preCICE*, it does follow the same methodology.
The time step size is $\Delta t = 0.01$s with a total of 620 DOFs along the interface where the QN approximation is constructed.
The convergence criterion is $\|K(g)\|_2 \leq 10^{-7}$.

Figure 4 shows the the tip displacement for 3 different time-steps including the steady state mean amplitude from [30,63].

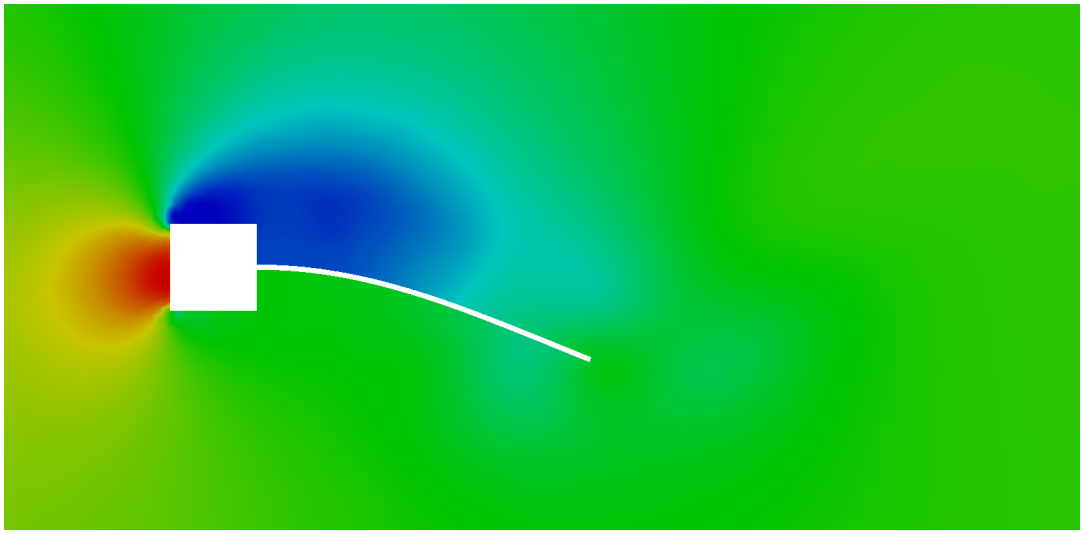

Fig. 3. Deformation and pressure contours shown here for $t = 4.29$s.

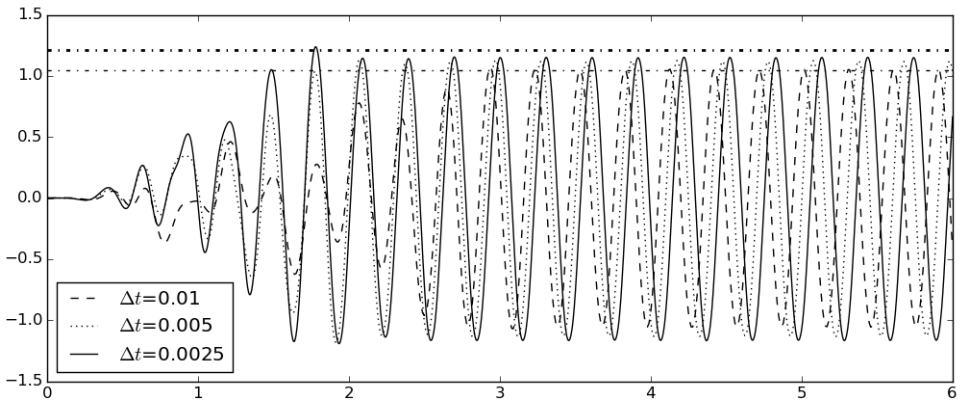

Fig. 4. Tip displacement for 3 different time-steps including the steady state mean amplitude (horizontal lines):
thick -. line is from [30]: 4336 fluid elements, 20 nine-noded quadratic solid elements, $\Delta t = 0.005$s;
thin -. line is from [63]: 5080 fluid elements, 20 nine-noded quadratic solid elements, $\Delta t = 0.004$s

The optimal results for each filtering technique and different numbers of retained histories are given in Table 4. The number of observation vectors filtered out in the last time step is also indicated. More data can be found in [57]. It is seen that filtering can indeed enhance the convergence speed. When a low number of histories is kept, the filtering techniques all give roughly the same performance. When more histories are kept (and thus the risk of linear dependence becomes bigger), the performance gain of the new QR filtering technique becomes more obvious. It is remarkable that the new QR filtering technique obtains the best results with aggressive filtering, retaining only 4 or 5 vectors, while the other filtering techniques score best with moderate filtering. This

shows that the filtering procedure is more important than the actual amount of filtering [10] . POD filtering, while better than the old QR-based filtering, does not match the performance of the new QR-based filtering.

	No histories	5 Histories	10 Histories	20 Histories
old QR	5.27 for $\epsilon = 10^{-6}$ (1/4)	5.52 for $\epsilon = 10^{-10}$ (6/34)	6.33 for $\epsilon = 10^{-11}$ (17/74)	7.56 for $\epsilon = 10^{-10}$ (101/178)
new QR	5.26 for $\epsilon = 10^{0}$ (0/5)	5.03 for $\epsilon = 10^{0}$ (25/29)	5.11 for $\epsilon = 10^{0}$ (50/54)	5.18 for $\epsilon = 10^{0}$ (96/100)
POD	5.26 for $\epsilon = 10^{-14}$ (0/4)	5.25 for $\epsilon = 10^{-20}$ (9/32)	5.57 for $\epsilon = 10^{-22}$ (16/61)	6.72 for $\epsilon = 10^{-20}$ (98/157)
no filter	5.30	5.61	6.38	div

Table 4
Best results for the the flexible beam benchmark problem for varying numbers of retained histories and different filtering techniques. Mean iteration numbers over the first 500 time steps. The quantities in brackets indicate the number of vectors filtered out in the final time step.

6 Conclusions

When QN-ILS is used as a tool to solve partitioned fluid-structure interaction problems, it can be beneficial to use data from previous time-steps, while, at the same time, creating the risk of destabilizing the method due to bad conditioning. This can be solved by judicially filtering out data that is (nearly) linearly dependent. However, this is a compromise, as discarding data might impair the convergence speed. In this paper, we introduced two new ways of filtering. While all filtering techniques that were shown are dependent on a parameter that needs to be tuned depending on the application, we show that better results can be obtained with a new way of QR-based filtering, without incurring extra complexity.

[10] While the best values of ϵ are still obtained by trial-and-error for each filtering technique, we only compare best results with best results. The reason why the new QR filtering uses substantially higher values of ϵ which can be deduced from Equations (13) and (14).

Acknowledgements

The financial support of the Institute for Advanced Study (IAS) of the Technische Universität München and of SPPEXA, the German Science Foundation Priority Programme 1648 – Software for Exascale Computing, is thankfully acknowledged.

References

[1] D. Anderson, Iterative procedures for nonlinear integral equations. *J. ACM* **12**, pp. 547–560 (1965)

[2] J.W. Banks, W.D. Henshaw, D.W. Schwendeman, An Analysis of a New Stable Partitioned Algorithm for FSI Problems. Part I: Incompressible Flow and Elastic Solids, *Journal of Computational Physics*, **269**, pp 108–137 (2014)

[3] M. Bathe, R. Kamm, A fluid-structure interaction finite element analysis of pulsatile blood flow through a compliant stenotic artery. *Journal of Biomechanical Engineering* **121**, pp. 361–369 (1999)

[4] K.-J. Bathe, H. Zhang, Finite element developments for general fluid flows with structural interactions. *Internat. J. Numer. Methods Engrg.* **60**, pp. 213–232 (2004)

[5] K.-J. Bathe, G. A. Ledezma, Benchmark problems for incompressible fluid flows with structural interactions, *Computers & Structures*, **85**, pp. 628–644 (2007)

[6] K.-J. Bathe , H. Zhang, A Mesh Adaptivity Procedure for CFD and Fluid-Structure Interactions, *Computers & Structures*, **87** (11–12), pp. 604–617 (2009)

[7] Y. Bazilevs, V.M. Calo, Y. Zhang, T.J.R. Hughes, Isogeometric fluid-structure interaction analysis with applications to arterial blood flow. *Comput. Mech.* **38**/4–5, pp. 310–322 (2006)

[8] D. S. Blom, A. H. van Zuijlen, H. Bijl, Acceleration of strongly coupled fluid-structure interaction with manifold mapping, in: *Proceedings of the 11th. World Congress on Computational Mechanics*, 5th. European Congress on Computational Mechanics, 6th. European Congress on Computational Fluid Dynamics, E. Oñate, X. Oliver, A. Huerta, (eds.), pp. 4484–4495 (2014)

[9] A.E.J. Bogaers, S. Kok, T. Franz, Strongly Coupled Partitioned FSI Using Proper Orthogonal Decomposition, Eighth South African Conference on Computational and Applied Mechanics (SACAM2012), Johannesburg, South Africa.

[10] A.E.J. Bogaers, S. Kok, B.D. Reddy, T. Franz, Quasi-Newton Methods for Implicit Black-Box FSI Coupling, *Comp. Meth. Appl. Mech. Eng.*, **279**, pp. 113–132, (2014)

[11] A.E.J. Bogaers, S. Kok, B.D. Reddy, T. Franz, Extending the Robustness and Efficiency of Artificial Compressibility for Partitioned Fluid-Structure Interactions, *Comp. Meth. Appl. Mech. Eng.*, **283**, pp. 1278–1295, (2015)

[12] S. Brändli, A. Düster, A Flexible Multi-Physics Coupling Interface for Partitioned Solution Approaches, *Proc. Appl. Math. Mech.*, **12**(1), pp. 363–364 Special Issue: 83rd Annual Meeting of the International Association of Applied Mathematics and Mechanics (GAMM), Darmstadt 2012; H.-D. Alber, N. Kraynyukova and C. Tropea (Eds.) (2012)

[13] L. Broderick, J. Leonard, Nonlinear response of membranes to ocean waves using boundary and finite elements. *Ocean Engrg.* **22**, pp. 731–745 (1995)

[14] P.N. Brown, Y. Saad, Hybrid Krylov methods for nonlinear systems of equations. *SIAM J. Sci. Statist. Comput.* **11**/3, pp. 450-481 (1990)

[15] H.-J. Bungartz, F. Lindner, M. Mehl, B. Uekermann, A plug-and-play coupling approach for parallel multi-field simulations, *Computational Mechanics*, December 2014

[16] P. Causin, J.-F. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Comput. Methods Appl. Mech. Engrg.* **194**/42-44, pp. 4506-4527 (2005)

[17] P. Crosetto, S. Deparis, G. Fourestey, A. Quarteroni, Parallel Algorithms for Fluid-Structure Interaction Problems in Haemodynamics, *SIAM J. Sci. Comput.*, **33**(4), pp. 1598–1622 (2011)

[18] M. Darwish, M, I. Sraj, F. Moukalled, A coupled finite volume solver for the solution of incompressible flows on unstructured grids, *Journal of Computational Physics* **228**, pp.180–201

[19] J. Degroote, P. Bruggeman, R. Haelterman, J. Vierendeels, Stability of a coupling technique for partitioned solvers in FSI applications. *Comput. & Structures* **86**, pp. 2224-2234 (2008)

[20] J. Degroote, P. Bruggeman, R. Haelterman, J. Vierendeels, Fluid-structure interaction coupling techniques based on sensitivities. In *Proceedings of the Sixth International Conference on Engineering Computational Technology*, M.Papadrakakis and B.H.V. Topping (Eds.), Civil-Comp Press, Stirlingshire, UK, 2008.

[21] J. Degroote, R. Haelterman, S. Annerel, A. Swillens, P. Segers, J. Vierendeels, An interface quasi-Newton algorithm for partitioned simulation of fluid-structure interaction. In *Proceedings of the International Workshop on Fluid-Structure Interaction. Theory, Numerics and Applications.* S. Hartmann, A. Meister, M. Schfer, S. Turek (Eds.), Kassel University Press, Germany, 2008

[22] J. Degroote, K.-J. Bathe, J. Vierendeels, Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. *Comp. & Struct.*, **87**, pp. 793–801 (2009)

[23] J. Degroote, P. Bruggeman, R. Haelterman, J. Vierendeels, Bubble simulations with an interface tracking technique based on a partitioned fluid-structure interaction algorithm. *J. Comput. Appl. Math.*, **234**/7, pp. 2303–2310 (2010)

[24] J. Degroote, R. Haelterman, S. Annerel, P. Bruggeman, J. Vierendeels, Performance of partitioned procedures in Fluid-Structure interaction. *Comp. & Struct.*, **88**, pp.446-457 (2010)

[25] J. Degroote, A. Souto-Iglesias, W. Van Paepegem, S. Annerel, P. Bruggeman, J. Vierendeels, Partitioned Simulation of the Interaction between an Elastic Structure and Free Surface Flow, *Comp. Meth. Appl. Mech. Eng.*, **199** (33-36), pp. 2085–2098, (2010)

[26] J. Degroote, J. Vierendeels, K. Vepa, W. Van Paepegem, Fluid-structure interaction simulation of the breaking wave slamming on an absorber for a wave-energy converter. In *Proceedings of the 8th international conference on structural dynamics*. G. De Roeck, G. Degrande, G. Lombaert, G. Müller (Eds.). Ghent, Belgium (2011).

[27] J. Degroote, M. Hojjat, E. Stavropoulou, R. Wüchner, K.-U. Bletzinger, Partitioned Solution of an Unsteady Adjoint for Strongly Coupled Fluid-Structure Interactions and Application to Parameter Identification of a One-Dimensional Problem, *Structural And Multidisciplinary Optimization*, **47**(1), pp. 77–94 (2013)

[28] S. Deparis, M. Discacciati, G. Fourestey, A. Quarteroni, Fluid-structure algorithms based on Steklov-Poincaré operators. *Comput. Methods Appl. Mech. Engrg.* **195**/41–43, pp. 5797–5812 (2006)

[29] J. De Ridder , J. Degroote, K. Van Tichelen, P. Schuurmans, J. Vierendeels, Modal Characteristics of a Flexible Cylinder in Turbulent Axial Flow from Numerical Simulations, *Journal Of Fluids And Structures*, **43**, pp. 110–123 (2013)

[30] W.G. Dettmer, D. Perić, On the coupling between fluid flow and mesh motion in the modelling of fluid–structure interaction, *Computational Mechanics*, **43**/1, pp. 81–90 (2008)

[31] N. dos Santos, J.-F. Gerbeau, J.-F. Bourgat, Partitioned FSI strategy for simulations of a thin elastic valve. In: P. Wesseling, E. Oate, J. Priaux (Eds.), European Conference on Computational Fluid Dynamics ECCOMAS CFD 2006. ECCOMAS, Delft, The Netherlands, pp. 1–10 (2006)

[32] K. Dumont, J. Vierendeels, R. Kaminsky, G. Van Nooten, P. Verdonck, D. Bluestein, Comparison of the hemodynamic and thrombogenic performance of two bileaflet mechanical heart valves using a CFD/FSI model. *ASME J. of Biomechanical Engineering* **129**/4, pp. 558–565 (2007)

[33] W. Elleithy, M. Tanaka, Interface relaxation algorithms for BEM-BEM coupling and FEM-BEM coupling. *Comput. Methods Appl. Mech. Engrg.* **192**, pp. 2977–2992 (2003)

[34] P. Erbts, A. Düster, Accelerated Staggered Coupling Schemes for Problems of Thermoelasticity at Finite Strains, *Computers & Mathematics With Applications*, **64**(8), pp. 2408–2430 (2012)

[35] P. Erbts, A Düster, Acceleration of partitioned coupling schemes for problems of thermoelasticity, *Proc. Appl. Math. Mech.*, **12**(1), pp. 367–368, (2012). Special Issue: 83rd Annual Meeting of the International Association of Applied Mathematics and Mechanics (GAMM), Darmstadt 2012; Editors: H.-D. Alber, N. Kraynyukova and C. Tropea

[36] S. Etienne , A. Garon, D. Pelletier, Some Manufactured Solutions for Verification of Fluid-Structure Interaction Codes, *Computers & Structures*, **106-107**, pp. 56–67 (2012)

[37] H. Fang, Y. Saad, Two classes of multisecant methods for nonlinear acceleration, *Numer. Linear Algebra Appl.*, **16**, pp. 197–221 (2009)

[38] C. Farhat, M. Lesoinne, P. LeTallec, Load and motion transfer algorithms for Fluid/Structure Interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Comput. Methods Appl. Mech. Engrg.* **157**, pp. 95–114 (1998)

[39] C. Farhat, M. Lesoinne, Two efficient staggered algorithms for serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems. *Comput. Methods Appl. Mech. Engrg.* **182**, pp. 499–515 (2000)

[40] C. Farhat, P. Geuzaine, C. Grandmont, The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *J. Comput. Phys.* **174**, pp. 669–694 (2001)

[41] C. Farhat, P. Geuzaine, G. Brown, Application of a three–field nonlinear fluid-structure formulation to the prediction of the aeroelastic parameters of an F-16 fighter. *Comput. & Fluids* **32**, pp. 3–29 (2003)

[42] C. Farhat, CFD-based nonlinear computational aeroelasticity. In: E. Stein, R. De Borst, T.J.R. Hughes (Eds.) Encyclopedia of computational mechanics, Vol. **3**, Chapter 13. Wiley, NY (2004)

[43] C. Farhat, K. van der Zee, P. Geuzaine, Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity. *Comput. Methods Appl. Mech. Engrg.* **195**, pp. 1973–2001 (2006)

[44] C.A. Felippa, K.C. Park, C. Farhat, Partitioned analysis of coupled mechanical systems. *Comput. Methods Appl. Mech. Engrg.* **190**, pp. 3247–3270 (2001)

[45] M.A. Fernández, M. Moubachir, A Newton method using exact jacobians for solving fluid-structure coupling. *Comput. & Structures* **83**, pp. 127–142 (2005)

[46] G. Fourtesy, S. Piperno, A second-order time-accurate ALE Lagrange-Galerkin method applied to wind engineering and control of bridge profiles. *Comput. Methods Appl. Mech. Engrg.* **193**, pp. 4117–4137 (2004)

[47] B. Gatzhammer, Efficient and Flexible Partitioned Simulation of Fluid-Structure Interactions, dissertation, Institut für Informatik, Technische Universitt München, München (2015)

[48] J.-F. Gerbeau, M. Vidrascu, P. Frey, Fluid-structure interaction in blood flows on geometries coming from medical imaging. *Comput. & Structures* **83**, pp. 155–165 (2005)

[49] V. Gkanis, C. Housiadas, A Time-Dependent Numerical Analysis of Flow in a Mechanical Heart Valve: Comparison with Experimental Results, *International Journal of Computational Fluid Dynamics*, **24**(5), pp. 157–168 (2010)

[50] J.A. González, K.C. Park, I. Lee, C.A. Felippa, R. Ohayon Partitioned Vibration Analysis of Internal Fluid-Structure Interaction Problems, *Int. J. Num. Meth. Eng.*, **92**(3), pp. 268–300 (2012)

[51] 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**(3),pp. 2347–2368 (2009)

[52] 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**(6), pp.4660-4679 (2009)

[53] R. Haelterman, J. Petit, B. Lauwens, H. Bruyninckx, J. Vierendeels, On the Non-Singularity of the Quasi-Newton-Least Squares Method, *Journal of Computational and Applied Mathematics*, **257**, pp. 129–131 (2014)

[54] R. Haelterman, B. Lauwens, H. Bruyninckx, J. Petit, J. Vierendeels, Equivalence of QN-LS and BQN-LS for affine problems, *Journal of Computational and Applied Mathematics*, **278**, pp. 48–51 (2015)

[55] R. Haelterman, D. Van Eester, D.Verleyen, Accelerating the Solution of a Physics Model inside a Tokamak using the (Inverse) Column Updating Method, *Journal of Computational and Applied Mathematics* **279**, pp.133–144 (2015)

[56] R. Haelterman, D. Van Eester, Accelerating the Convergence of a Tokamak Modeling Code with Aitken's Method. To appear in *Computer Physics Communications*, *doi:10.1016/j.cpc.2015.05.018*

[57] R. Haelterman, A. Bogaers, B. Uekermann, K. Scheufele, M. Mehl, Additional results for the article: "Improving the performance of the partitioned QN-ILS procedure for fluid-structure interaction problems: filtering"; available at `https://www.researchgate.net/publication/280316662_Additional_results_for_the_article_Improving_the_performance_of_the_partitioned_QN-ILS_procedure_for_fluid-structure_interaction_problems_filtering`

[58] M. Heil, Stokes flow in an elastic tube – a large–displacement fluid–structure interaction problem. *Internat. J. Numer. Methods Fluids* **28**, pp. 243–265 (1998)

[59] M. Heil, An efficient solver for the fully coupled solution of large-displacement fluid–structure interaction problems. *Comput. Methods Appl. Mech. Engrg.* **193**, pp. 1–23 (2004)

[60] B. Hübner, E. Walhorn, D. Dinkler, Numerical investigation to bridge aeroelasticity. In: H.A. Mang, F.G. Rammerstofer, J. Eberhardsteiner (Eds.), Proceedings of the Fifth World Congress on Computational Mechanics (WCCM V). Vienna University of Technology, Austria (2002)
Available from *http://wccm.tuwien.ac.at/publications/Papers/fp81407.pdf*

[61] H.Z. Jahromi, B.A. Izzuddin, L. Zdravkovic, Partitioned analysis of nonlinear soil-structure interaction using iterative coupling. *Interaction and Multiscale Mechanics* **1**/1 pp. 33–51 (2007)

[62] V. Kalro, T.E. Tezduyar, A parallel 3D computational method for fluid-structure interactions in parachute systems. *Comput. Methods Appl. Mech. Engrg.* **190**, pp. 321–332 (2000)

[63] C. Kassiotis, A. Ibrahimbegovic, R. Niekamp, H.G. Matthies, Nonlinear fluid–structure interaction problem. Part I: implicit partitioned algorithm, nonlinear stability proof and validation examples, *Computational Mechanics*, **47**/3, pp. 305–323 (2011)

[64] N. Kolekar, A. Banerjee, A Coupled Hydro-Structural Design Optimization for Hydrokinetic Turbines, *J. Renewable Sustainable Energy*, **5**, 053146 (2013)

[65] N. Kolekar, S.S. Mukherji, A. Banerjee, Numerical Modeling and Optimization of Hydrokinetic Turbine, *ASME 2011 5th International Conference on Energy Sustainability*, Washington, DC, USA, August 7–10 (2011)

[66] F. Lindner, M. Mehl, K. Scheufele, B. Uekermann, A Comparison Of Various Quasi-Newton, Schemes For Partitioned Fluid-Structure Interaction, In *Proceedings of VI International Conference On Computational Methods For Coupled Problems In Science And Engineering, Coupled Problems*, B. Schreer, E. Oñate And M. Papadrakakis(Eds) (2015)

[67] P.A. Lott, H.F. Walker, C.S. Woodward, U.M. Yang, An accelerated Picard method for nonlinear systems related to variably saturated flow, *Advances in Water Resources*, **38**, pp. 92–101 (2012)

[68] J.M. Martinez, Practical quasi-Newton method for solving nonlinear systems. *J. Comput. Appl. Math.* **124**, pp. 97–122 (2000)

[69] H.G. Matthies, J. Steindorf, Partitioned strong coupling algorithms for fluid-structure interaction. *Comput. & Structures* **81**, pp. 805–812 (2003)

[70] H. Matthies, R. Niekamp, J. Steindorf, Algorithms for strong coupling procedures. *Comput. Methods Appl. Mech. Engrg.* **195**, pp. 2028–2049 (2006)

[71] R.B. Melville, S.A. Morton, Fully-implicit aeroelasticity on overset grid systems. *AIAA Paper* **98-0251** (1998)

[72] C. Michler, E.H. Van Brummelen, R. De Borst, Error-amplification analysis of subiteration-preconditioned GMRES for fluid-structure interaction. *Comput. Methods Appl. Mech. Engrg.* **195**, pp. 2124–2148 (2006)

[73] P. Ni, Anderson acceleration of fixed-point iteration with applications to electronic structure computations, PhD Thesis, Worcester Polytechnic Institute (2009)

[74] J.M. Ortega, W.C. Rheinboldt, Iterative solution of nonlinear equations in several variables. Classics in Applied Mathematics **30**, Society for Industrial and Applied Mathematics, Philadelphia, PA (2000)

[75] S. Patanka, Numerical Heat Transfer and Fluid Flow, CRC Press (1980)

[76] S. Piperno, C. Farhat, B. Larrouturou, Partitioned procedures for the transient solution of coupled aeroelastic problems - part I: model problem, theory and two-dimensional application. *Comput. Methods Appl. Mech. Engrg.* **124**, pp. 79–112 (1995)

[77] S. Piperno, Numerical simulation of aeroelastic instabilities of elementary bridge decks. *INRIA Tech. Rep.* **3549** (1998)

[78] 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**, pp. 3147–3170 (2001)

[79] K. Riemslagh, J. Vierendeels, E. Dick, Coupling of a Navier-Stokes solver and an elastic boundary solver for unsteady problems. In: K. Papailiou, D. Tsahalis, J. Priaux, C. Hirsch, M. Pandolfi (Eds.), 4th European Computational Fluid Dynamics Conference. ECCOMAS, Athens, Greece, pp. 1040–1045 (1998)

[80] S.M. Rifai et al., Multiphysics simulation of flow-induced vibrations and aeroelasticity on parallel computing platforms. *Comput. Methods Appl. Mech. Engrg.* **174**, pp. 393–417 (1999)

[81] I. Robertson, S.J. Sherwin, P.W. Bearman, Prediction of flutter instability due to cross winds in the Second Forth Road Bridge. In: Computational Fluid Dynamics '01, Proceedings of the Eccomas CFD Conference 2001. Swansea, Wales, UK (2001)

[82] I. Robertson, S.J. Sherwin, P.W. Bearman, Flutter instability prediction techniques for bridge deck sections. *Internat. J. Numer. Methods Fluids* **43**, pp. 1239–1256 (2003)

[83] L. Sirovich, Turbulence and the dynamics of coherent structures, Part I: Coherent Structures, *Quarterly of Appl. Math.* **45**/3, pp 561–571 (1987).

[84] K. Stein, T. Tezduyar, V. Kumar, S. Sathe, R. Benney, R. Charles, Numerical simulation of soft landing for clusters of cargo parachutes. In: P. Niettaanmäki, T. Rossi, K. Majava, O. Pieronneau (Eds.), European congres on computational methods in applied sciences and engineering ECCOMAS 2004. Jyväskylä, pp. 1–14 (2004)

[85] T.E. Tezduyar, S. Sathe, K. Stein, Solution techniques for the fully-discretized equations in computation of fluid-structure interactions with the space-time formulations. *Comput. Methods Appl. Mech. Engrg.* **195**, pp. 5743–5753 (2006)

[86] B. Uekermann, H.-J. Bungartz, B. Gatzhammer, M. Mehl, A Parallel, Black-Box Coupling Algorithm For Fluid-Structure Interaction, *Proceedings of the V International Conference On Computational Methods For Coupled Problems In Science And Engineering, Coupled Problems*, S. Idelsohn, M. Papadrakakis and B. Schreer (Eds), (2013)

[87] R. van Loon, P. Anderson, F. van de Vosse, A fluid-structure interaction method with solid-rigid contact for heart valve dynamics. *J. Comput. Phys.* **217**, pp. 806–823 (2006)

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

[89] J. Vierendeels, L. Lanoye, J. Degroote, P. Verdonck, Implicit coupling of partitioned fluid-structure interaction problems with reduced order models. *Comput. & Structures* **85**, pp. 970–976 (2007)

[90] H. F. Walker, P. Ni, Anderson acceleration for fixed-point iterations, *SIAM Journal for Numerical Analysis*, **49**, pp. 1715–1735 (2011)

[91] W.A. Wall, Fluid-Struktur-Interaktion mit stabilisierten Finiten Elementen. Ph.D.-Dissertation, Report **31**, Institute of Structural Mechanics, University of Stuttgart (1999)

[92] K. Willcox, J. Paduano, J. Peraire, Low order aerodynamic models for aeroelastic control of turbomachines. In: 40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials (SDM) Conference, St. Louis, MO, pp. 1–11 (1999)

[93] R. Wüchner, A. Kupzok, K.-U. Bletzinger, A framework for stabilized partitioned analysis of thin membrane–wind interaction. *Internat. J. Numer. Methods Fluids* **54**/6–8, pp. 945–963 (2007)