VI International Conference on Computational Methods for Coupled Problems in Science and Engineering
COUPLED PROBLEMS 2015
B. Schrefler, E. Oñate and M. Papadrakakis(Eds)

A COMPARISON OF VARIOUS QUASI-NEWTON SCHEMES FOR PARTITIONED FLUID-STRUCTURE INTERACTION

FLORIAN LINDNER[†], MIRIAM MEHL[†], KLAUDIUS SCHEUFELE[†], AND BENJAMIN UEKERMANN[*]

[†]Institute for Parallel and Distributed Systems, Universität Stuttgart
Universitätsstraße 38, 70569 Stuttgart, Germany
e-mail: miriam.mehl@ipvs.uni-stuttgart.de, web page: http://www.ipvs.uni-stuttgart.de/

[*]Institute for Advanced Study (IAS), Technische Universität München
Lichtenbergstraße 2a, 85748 Garching b. München, Germany
e-mail: uekerman@in.tum.de, web page: http://www5.in.tum.de

Key words: Fluid-Structure Interaction, Quasi-Newton, Partitioned Coupling

Abstract. During the last 5 years, quasi-Newton schemes have proven to be a robust and efficient way to couple partitioned fluid-structure interaction. We showed in previous work that they also allow to perform a parallel coupling. Bogaers et al. introduced a new variant based on a multi-vector update [14]. This variant renders a tuning of the reuse of old information unnecessary as all old iterations are implicitly covered in a Jacobian update. In this work, we compare this multi-vector variant in an inverse formulation to the classical IQN-ILS algorithm for serial as well as parallel coupling.

1 INTRODUCTION

The simulation of fluid-structure interactions is an important contribution to many fields in science and engineering – from aero-elasticity in aerospace engineering to hemodynamics in medical applications. At the same time, it is a very challenging type of multi-physics application as it tends to be ill-conditioned and unstable in particular for incompressible fluids.

Since fluid-structure interactions have been among the first multi-physics models considered for numerical simulation, the respective methods are in the meantime very sophisticated and, in particular, also a lot of very powerful monolithic solvers are available [1, 2, 3]. We focus, however, on partitioned approaches as our aim is to provide a general coupling tool, not only for fluid-structure interactions but also for multi-field coupling [4] and situation where flexible ad-hoc solutions combining different existing solvers are required. We have to design coupling numerics that are suited for black-box solvers where

nothing but the pure input and output data of the respective solver are accessible to the user. This enables us to use also commercial closed-source software as solvers can be coupled to our coupling software with minimal code changes or code wrappers. Thus, we consider the structural solver S and the flow solver F as mappings taking certain boundary values as an input and delivering other values at the coupling surface between fluid and structure as an output:

$$\begin{aligned} S : x_f &\mapsto x_d, \\ F : x_d &\mapsto x_f. \end{aligned}$$

Here, x_d denotes interface displacements or velocities at the wet surface between fluid and structure, x_f the forces or stresses modelling the impact of the fluid on the structure at the wet surface. This notation corresponds to the well-known Dirichlet-Neumann coupling between fluid and structure solvers. We are going to focus on this type of coupling throughout this paper. Alternative ways to define boundary conditions for fluid and structure solvers often require at least some knowledge on discretization details or Jacobians of the involved solvers. This is the case, e.g., for Robin-type boundary conditions [5].

Stability issues due to the physically strong coupling between fluid and structure, also referred to as the added mass effect (see [6, 7]), enforce implicit coupling within each time step. There are several ways how to realize the respective iterations which are different

1. in terms of the execution order of fluid and structure solvers,

2. in terms of the method used to stabilize and accelerate the coupling iteration.

Regarding the first, a staggered execution of fluid and structure solver, resulting in a Gauss-Seidel type iteration has been the common approach for a long time (cf. e.g. [8]). For compressible fluids, however, also the simultaneous execution of fluid and structure solvers, i.e., a Jacobian-type coupling, in many cases even as an explicit version has been in use ([8, 9]). We have presented a coupling based on the simultaneous execution of both solvers also for incompressible fluids and examples with strong instabilities in [10]. This iteration type is not feasible for a pure fixed-point iteration. Thus, sophisticated stabilization methods from 2. have to be used. We showed in [10] that neither constant nor adaptive Aitken underrelaxation was sufficient to accelerate the simultaneous coupling enough to make it competitive with the staggered iteration. Only quasi-Newton methods, that showed the best results also for the staggered coupling over the last years ([11, 12, 13]) showed a very good performance as presented in [10], even for the parallel coupling of more than two physical fields [4].

The combination of simultaneous solver execution with efficient quasi-Newton solvers results in an almost optimal partitioned fluid-structure interaction simulation method. However, the performance of the quasi-Newton methods depends on parameters such as the number of old time steps used to estimate Jacobians, for which only experience or try-and-error can be used to determine the optimal value. Therefore, we compare the

quasi-Newton approach used in [11] and [10] with an alternative approach presented in [14] for a different, block-iterative Newton solver. For this approach, information from passed time steps is implicitly used in the Jacobian estimation by suitable norm minimization conditions. The drawback, however, is that these methods explicitly determine and store Jacobians whereas the quasi-Newton approach from [11] only estimates the results of matrix-vector products involving parts of the Jacobian.

In the following[1], we first present the respective different coupling methods in Sect. 2, followed by numerical results in Sect. 3.

2 QUASI-NEWTON SCHEMES FOR PARTITIONED FLUID-STRUCTURE INTERACTION

We consider two different implicit coupling systems, namely a serial or staggered coupling and a parallel or vectorial coupling systems as mentioned above in Sect. 1. The parallel or vectorial system is preferable in case of massively parallel simulations in order to achieve a reasonable load balancing as described in [10]. We repeat the basics of both systems in Sect. 2.1.

Both equation systems are solved in two different ways: Using the known interface-quasi-Newton least squares approach as in [10, 11] and a multi-vector Jacobian approximation as described in [14] for a different Newton- iteration type, a block-iterative Newton method. The two solver alternatives are introduced in Sect. 2.2.

2.1 Execution Orders and Fixed Point Equations

In this section, we shortly recapitulate the introduction of fixed-point equations at the wet surface between fluid and structure depending on the execution order of the solvers.

Serial Implicit Coupling Scheme (S-System). If we execute the solvers in a staggered way, i.e., the flow solver first computes stresses or forces that are communicated to the structure solver that afterwards computes new wet surface displacements or velocities, the corresponding interface equation reads

$$x_d \stackrel{!}{=} S \circ F(x_d) \; . \tag{1}$$

Parallel Implicit Coupling Scheme (V-System) The above serial implicit coupling scheme offers some remarkable drawbacks regarding efficient parallelization. There is a substantial mismatch of work load between the structure and the fluid field solver, which does not allow for an efficient parallelization using the S-System. The only way to overcome these limitations in parallel efficiency is to evaluate the fluid and structure solver in parallel. Hereby, the V-System uses the original input/output relation for both

[1] A more detailed presentation of the findings of this paper including tables and figures has already been published in the master thesis of Klaudius Scheufele [15].

solvers but the boundary values are exchanged after each solve of the solvers executed in parallel. This leads to the vectorial fixed-point equation

$$\begin{pmatrix} x_f \\ x_d \end{pmatrix} \overset{!}{=} \begin{pmatrix} 0 & F \\ S & 0 \end{pmatrix} \begin{pmatrix} x_f \\ x_d \end{pmatrix} . \tag{2}$$

The vectorial system results in two independent instances of the S-System if solved by a pure fixed-point iteration, but quasi-Newton solvers turn out to be powerful enough such that one iteration of the V-System is comparable to one iteration of the S-System (cf. [4, 10]).

2.2 Quasi-Newton Solvers for the Interface Equations

In this section, we describe two quasi-Newton schemes, which can be applied to both aforementioned fixed-point equations (1) and (2). For sake of clarity, we introdue a unified notation:

$$x := \begin{cases} x_d \\ \begin{pmatrix} x_f \\ x_d \end{pmatrix} \end{cases} \quad \text{and} \quad H := \begin{cases} S \circ F & \text{for the staggered equation (1) ,} \\ \begin{pmatrix} 0 & F \\ S & 0 \end{pmatrix} & \text{for the parallel equation (2) .} \end{cases}$$

Thus, we have to solve the fixed-point equation

$$H(x) = x \;\Leftrightarrow\; R(x) := H(x) - x \overset{!}{=} 0 . \tag{3}$$

As a pure fixed-point iteration tends to be unstable and, in particular, two times slower for the parallel fixed-point equation compared to the staggered one (compare [10]), we use a quasi-Newton scheme as a stabilization and acceleration after each iteration $\tilde{x}^k = H(x^k)$. We, therefore, rewrite (3) to the equivalent inverse form

$$\tilde{R}(\tilde{x}) := \tilde{x} - H^{-1}(\tilde{x}) \overset{!}{=} 0 .$$

Now, the Newton iteration reads

$$\textbf{solve} \quad \left[I - J_{H^{-1}}(\tilde{x}^k)\right] \Delta\tilde{x}^k = J_{\tilde{R}}(\tilde{x}^k)\Delta\tilde{x}^k = -\tilde{R}(\tilde{x}^k) , \tag{4}$$

$$\textbf{set} \quad x^{k+1} = \tilde{x}^k + \Delta\tilde{x}^k . \tag{5}$$

As the exact Jacobian $J_{\tilde{R}}(x^k) := I - J_{H^{-1}}(\tilde{x}^k)$, however, is not accessible for black-box solvers, we work with an approximation $\widehat{J_{\tilde{R}}}(\tilde{x}^k)$. To minimize the computational cost, in particular of solving the system (4), we do not approximate the Jacobian itself, but its inverse $\widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k)$. To do so, we collect input-output data throughout our iterations within a time step and generate the following matrices:

$$W_k = (w_i^k)_{i=0}^{k-1} = \left[\Delta\tilde{x}_0^k, \Delta\tilde{x}_1^k, \cdots, \Delta\tilde{x}_{k-1}^k\right] , \qquad \text{with} \;\; \Delta\tilde{x}_i^k = \tilde{x}^k - \tilde{x}^i ,$$

$$V_k = (v_i^k)_{i=0}^{k-1} = \left[\Delta R_0^k, \Delta R_1^k, \cdots, \Delta R_{k-1}^k\right] , \qquad \text{with} \;\; \Delta R_i^k = R(x^k) - R(x^i) .$$

The inverse Jacobian approximately fulfills the secant equation, i.e.,

$$J_{\tilde{R}}^{-1}(\tilde{x}^k)\, V_k \approx W_k \ .$$

We use this as a system of equations for the entries of our approximate $\widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k)$. As k is in general much smaller than the number of degrees of freedom at the coupling interface, W_k and V_k are tall and thin matrices. Thus, we get an underdetermined system of equations for the entries of $\widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k)$:

$$\widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k) V_k = W_k \ . \tag{6}$$

In the following, we present two approaches to compute $\widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k)$ or the result of applying part of the approximate Jacobian to $-\tilde{R}(\tilde{x}^k) = -R(x^k)$, respectively, based on this secant equation.

2.2.1 Interface quasi-Newton Least Squares (IQN-LS)

The interface quasi-Newton least squares method presented in [11] uses the norm minimization

$$\left\| \widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k) \right\|_F \rightarrow \min \tag{7}$$

to enhance the secant equation (6) to a system with a unique solution for $\widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k)$. Here, $\|\cdot\|_F$ is the Frobenius norm. This gives the approximate inverse Jacobian

$$\widehat{J_{\tilde{R}}}^{-1}(x^k) = W_k \left(V_k^T V_k\right)^{-1} V_k^T$$

and the update formula

$$x^{k+1} = \tilde{x}^k + W_k \underbrace{(V_k^T V_k)^{-1} V_k \left(-R(x^k)\right)}_{=:\,\alpha} .$$

We do not have to explicitely compute the inverse Jacobian, but can restrict ourselves to compute only the vector α. This can be realized very efficiently by solving the least squares problem

$$\min_{\alpha \in \mathbb{R}^k} \|V_k \alpha + R(x^k)\|_2 \ ,$$

where $\|\cdot\|_2$ denotes the Euclidian norm.[2]

The convergence properties of the IQN-LS method can be greatly improved, if the input/output informations from previous time steps are incorporated into the secant equation, i.e., into W_k and V_k. To achieve this, the difference matrices $V^{n+1-R}, \cdots, V^n, V^{n+1}$

[2]Note that the update formula for x^{k+1} also shows that skipping the fixed point iteration step (computing $\tilde{x}^k = H(x^k)$) before using a quasi-Newton step would have lead to linearly dependent columns in W_k: We then would always correct x^k to x^{k+1} by adding multiples of differences $x^k - x^i$ from previous iterations as we would have to use $W_k = (\Delta x_0^k, \Delta x_1^k, \ldots, \Delta x_{k-1}^k)$ with $\Delta x_i^k = x^k - x^i$ in this case. Using induction over the iterations, we see that all columns of W_k would be in the space spanned by x^0 and $x^1 - x^0$.

and $W^{n+1-R}, \cdots, W^n, W^{n+1}$ from the previous $R \in \mathbb{N}$ time steps are stored and included in the secant equation of the current time step, i. e., we replace W_k and V_k by the enhanced versions

$$W_k^{(R)} = \left[W^{\{n+1-R\}}, W^{\{n-R\}}, \cdots, W_k^{\{n+1\}}\right],$$
$$V_k^{(R)} = \left[V^{\{n+1-R\}}, V^{\{n-R\}}, \cdots, V_k^{\{n+1\}}\right].$$

This additional information significantly improves the convergence as shown in [11] and Sect. 3. However, the optimal parameter R of reused time steps is highly problem dependent and there is no analytical method available to determine the optimal R. Thus, in practice, R has to be determined based on experiences and in a costly try-and-error process. Also, linear dependencies and contradicting information within the accumulated difference matrices need to be handled properly. The alternative quasi-Newton approach presented in the next section provides an automatic implicit incorporation of information from passed time steps and, thus, avoids these drawbacks of the IQN-LS method. However, this requires to explicitly compute $\widehat{J_{\tilde{R}}}^{-1}$ instead of only the short vector $\alpha \in \mathbb{R}^k$.

2.2.2 Interface quasi-Newton Multiple Vector Jacobian (IQN-MVJ)

The IQN-MVJ method presented here is a newly developed quasi-Newton fluid-structure coupling approach. It combines the idea of approximating the Newton iteration defined by (4) and (5) based on the secant equation (6) with the ideas presented in [14] for Jacobian approximations in the context of a block-iterative Newton method. To implicitly use information from previous time steps, the IQN-MVJ method uses a different norm minimization than the IQN-LS method in order to achieve uniqueness of the inverse Jacobian approximation:

$$\left\| \widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k) - \widehat{J_{\tilde{R}prev}}^{-1} \right\|_F \rightarrow \min , \tag{8}$$

where $\widehat{J_{\tilde{R}prev}}^{-1}$ denotes the last inverse Jacobian approximation of the previous time step. Thus, our approximations always stay as close as possible to the approximation from the last time step. This automatically guaranties that we profit from past information without having to explicitely use old W and V matrices again. We get the approximate inverse Jacobian

$$\widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k) = \widehat{J_{\tilde{R}prev}}^{-1} + \left(W_k - \widehat{J_{\tilde{R}prev}}^{-1} V_k\right) \left(V_k^T V_k\right)^{-1} V_k^T$$

and the update formula

$$\begin{aligned} x^{k+1} &= \tilde{x}^k + \widehat{J_{\tilde{R}}}^{-1}(\tilde{x}^k) \left(-R(x^k)\right) \\ &= x^k + \left(\widehat{J_{\tilde{R}prev}}^{-1} + \left(W_k - \widehat{J_{\tilde{R}prev}}^{-1} V_k\right) \left(V_k^T V_k\right)^{-1} V_k^T\right) \left(-R(x^k)\right). \end{aligned}$$

There is a close relation of the IQN-MVJ update scheme to the Broyden method: The Broyden method also minimizes distances between successively computed Jacobian approximations within a Newton iteration. However, Broyden minimizes the distance of approximations between two successive iterations, whereas we minimize the distance between Jacobian approximations in two successive time steps. This also implies that our updates are not only rank-one modifications as more than one new input/output pair is added for the update of the Jacobian approximation from the previous time step to the Jacobian approximation of the current time step.

In Algorithm 1, we oppose the two quasi-Newton approaches, the established IQN-LS method from [11] and our new variant. Summarizing Sect. 2, we state that we have described four different coupling methods, which are going to be considered for numerical test cases in Sect. 3. For clarity reasons, we give an overview of our combinations in Tab. 1.

Algorithm 1 Quasi-Newton schemes, **left:** the IQN-LS algorithm, a matrix free approach that approximates for the inverse Jacobian of the residual operator, **right:** the IQN-MVJ algorithm which explicitly stores the matrix of the Jacobian estimation. The inverse of the Jacobian of the residual operator is approximated using information from previous time steps implicitly.

IQN-LS
initial value x^0
$\tilde{x}^0 = H(x^0)$ and $R^0 = \tilde{x}^0 - x^0$
$x^1 = x^0 + 0.1 \cdot R^0$
for $k = 1 \dots$ **do**
  $\tilde{x}^k = H(x^k)$ and $R^k = \tilde{x}^k - x^k$
  $V_k = [\Delta R_0^k, \dots, \Delta R_{k-1}^k]$ with $\Delta R_i^k = R^i - R^k$
  $W_k = [\Delta \tilde{x}_0^k, \dots, \Delta \tilde{x}_{k-1}^k]$ with $\Delta \tilde{x}_i^k = \tilde{x}^i - \tilde{x}^k$
  decompose $V^k = Q^k U^k$
  solve the first k lines of $U^k \alpha = -{Q^k}^T R^k$
  $\Delta \tilde{x}^k = W\alpha$
  $x^{k+1} = \tilde{x}^k + \Delta \tilde{x}^k$
end for

IQN-MVJ(x)
initial value x^0 and $\widehat{J}_{\tilde{R}prev}^{-1} = 0$
$\tilde{x}^0 = H(x^0)$ and $R^0 = \tilde{x}^0 - x^0$
$x^1 = x^0 + 0.1 \cdot R^0$
for $k = 1 \dots$ **do**
  $\tilde{x}^k = H(x^k)$ and $R^k = \tilde{x}^k - x^k$
  $V_k = [\Delta R_0^k, \dots, \Delta R_{k-1}^k]$ with $\Delta R_i^k = R^i - R^k$
  $W_k = [\Delta \tilde{x}_0^k, \dots, \Delta \tilde{x}_{k-1}^k]$ with $\Delta \tilde{x}_i^k = \tilde{x}^i - \tilde{x}^k$
  $\widehat{J}_{\tilde{R}}^{-1}(\tilde{x}^k) = \widehat{J}_{\tilde{R}prev}^{-1} + \left(W_k - \widehat{J}_{\tilde{R}prev}^{-1} V_k\right) \left(V_k^T V_k\right)^{-1} V_k^T$
  $\Delta \tilde{x}^k = -\widehat{J}_{\tilde{R}}^{-1}(\tilde{x}^k) R^k$
  $x^{k+1} = \tilde{x}^k + \Delta \tilde{x}^k$
end for

	Serial System	**Vectorial System**
IQN-LS	S-IQN-LS	V-IQN-LS
IQN-MVJ	S-IQN-MVJ	V-IQN-MVJ

Table 1: Numerical coupling methods examined in this paper: each method is defined by choosing either the serial fixed-point equation (1) or the vectorial fixed-point equation (2) and either the least-squares quasi-Newton solver described in Sect. 2.2.1 or the multi-vector Jacobian approximation quasi-Newton approach from Sect. 2.2.2.

3 NUMERICAL RESULTS

3.1 Software

All numerical experiments were conducted using the coupling library preCICE and the simulation toolbox OpenFOAM. We give a brief summary of both below.

preCICE is a library for flexible numerical coupling of single-physics solvers. preCICE[3] is developed at the Technische Universität München and the Universität Stuttgart. It uses a partitioned black-box coupling approach, thus requiring only minimal modifications to existing solvers. Its software architecture as a library in conjunction with a high-level API fosters quick and minimal-invasive integration into existing codes. Integration can be realized in less than 30 lines of code. preCICE offers a wide variety of runtime configurable aspects of numerical coupling like serial and parallel as well as explicit and implicit coupling schemes. The latter category includes the schemes presented in this paper. Subcycling enables to combine problems that show convergence on different time scales. Coupling between non-matching grids can be achieved by data mapping methods ranging from simple projection methods to approaches based on radial-basis functions. preCICE is written in C++ and features a clean and modern software design with extensive unit and integration testing while maintaining minimal external dependencies and easy extensibility. For more information, refer to [12].

OpenFOAM The fluid and structure simulations were done using a software based on OpenFOAM[4] resp. the foam-extend-3.1 project[5]. The preCICE adapter and the actual solver were developed by David Blom et.al. from TU Delft. The fluid solver uses a 2^{nd} order finite volume discretization of the incompressible Navier-Stockes equation. Instead of the standard PISO (pressure implicit with splitting of operator) algorithm it uses a coupled solution algorithm as described in [16]. Time integration is performed by the second order backward differencing scheme. The fluid equations are formulated in the arbitrary-lagrangian-eulerian perspective whereas the mesh movement uses radial basis function interpolation [17]. The structural domain is modeled using a fully langrangrian formulation and a Saint Venant-Kirchhoff model. For more information, refer to e.g. [13].

3.2 Three-dimensional flow over an elastic structure

Scenario description. This case depicts a three-dimensional laminar and incompressible flow over an elastic structure. The scenario is assumed to be symmetric in the x/y-plane, hence, the simulation is only performed in one half of the domain. The geometry of the scenario and physical results are shown in Fig. 1 (a). The computational

[3] `http://www5.in.tum.de/wiki/index.php/PreCICE_Webpage`

[4] `http://www.openfoam.org/`

[5] `http://www.extend-project.de/`

Table 2: Three-dimensional flow over an elastic structure. Average numbers of coupling iterations for IQN-LS and IQN-MVJ, and for serial and parallel coupling. The iteration numbers are averaged over the first 40 time steps. Different numbers of reused time steps are evaluated.† For a well-conditioned V-system we scaled the forces by a factor of $4.0 \cdot 10^4$ for IQN-LS and 10^4 for IQN-MVJ.

Reuse (R)	0	1	2	3	4	5	6	7	8
S-IQN-LS(R)	5.15	4.75	4.50	4.45	4.65	5.05	5.65	5.70	6.15
S-IQN-MVJ(R)	5.15	5.22	5.57	5.87	6.32	6.70	7.00	7.45	7.62
V-IQN-LS(R)†	9.30	7.85	7.07	6.90	6.70	6.80	7.05	7.15	7.45
V-IQN-MVJ(R)†	6.65	6.75	6.82	7.27	7.47	7.72	8.02	8.10	8.52

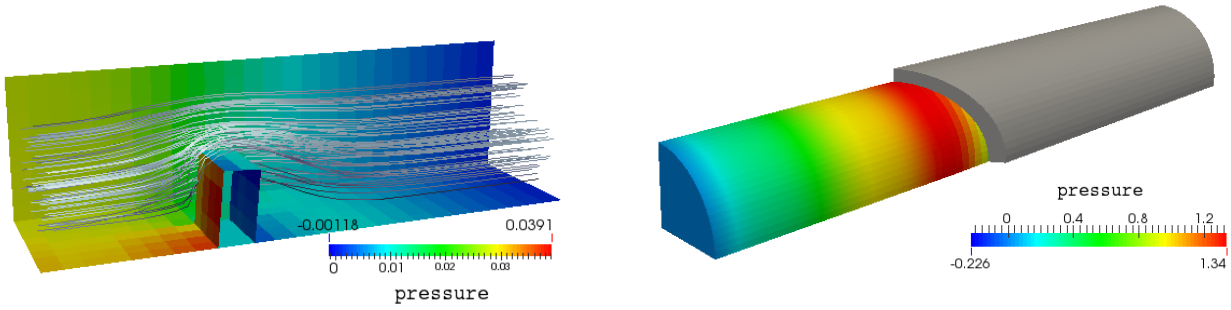
domain has a length of 1.5 m along with a width and height of 0.4 m, whereas the elastic rectangular structure with dimensions 0.2 m × 0.2 m × 0.2 m is mounted on the wall. The fluid flow is driven by a parabolic velocity profile v with peak velocity $v_{max} = 0.2\,\frac{\mathrm{m}}{\mathrm{s}}$ which is imposed as a Dirichlet boundary condition at the inflow boundary. The profile is faded in smoothly for $t < 2.0$. At the opposing side standard outflow conditions are applied, on the symmetry surface free-slip conditions as well as no-slip conditions on all remaining boundaries. The fluid density is $1 \times 10^3\,\frac{\mathrm{kg}}{\mathrm{m}^3}$, the dynamic viscosity $1 \times 10^{-3}\,\mathrm{Pa\,s}$. The density of the structure is $1 \times 10^3\,\frac{\mathrm{kg}}{\mathrm{m}^3}$, the Poisson ratio 0.4 and the Young's modulus is set to $1 \times 10^4\,\frac{\mathrm{N}}{\mathrm{m}^2}$.

Results. The simulation domain is decomposed into 1632 cells for the fluid and 32 cells for the structure mesh. Forty time steps with a time step size of 0.1 s are performed. A relative convergence measure of 10^{-5} for the forces as well as for the displacements at the fluid-structure interface is used. The results of the simulation are shown in Tab. 2. The average number of iterations over the first 40 time steps are shown for the IQN-LS as well as for the IQN-MVJ quasi-Newton scheme with different numbers of reused time steps. To precondition the V-system, we scale the forces in advance, such that forces and displacements have the same order of magnitude. It can be seen from Tab. 2, that the IQN-LS and the IQN-MVJ approach yield quite similar results for the S-system (serial coupling) as well as for the V-system (parallel coupling). As the IQN-MVJ implicitly incorporates information from previous time steps via the norm minimization in eq. (8) there is no benefit in collecting columns from previous time steps for the V and W matrices. Hence, the IQN-MVJ scheme is more robust and there is no need to tune for an optimal number of reused time steps (R).

3.3 Wave propagation in a three-dimensional elastic tube

Scenario description. This test case simulates the wave propagation in a straight, three-dimensional elastic tube (cf. e.g. [11]). The length of the tube is 0.05 m. The fluid

domain has a diameter of 0.003 m, whereas the tube thickness is 0.001 m. Both ends of the tube are fixed. For the initial duration of 0.003 s, the boundary condition for the pressure inlet is set to a fixed value of 1333.2 Pa. Thereafter, the inlet pressure is set to zero. At the outlet, the pressure is set to zero. The fluid has a density of $1 \times 10^3 \frac{\text{kg}}{\text{m}^3}$, and a dynamic viscosity of 3×10^{-3} Pa s. The density of the elastic structure is $1.2 \times 10^3 \frac{\text{kg}}{\text{m}^3}$, the Young's modulus $3 \times 10^5 \frac{\text{N}}{\text{m}^2}$, and the Poisson's ratio 0.3. A pressure pulse propagates through the tube as shown in Fig. 1 (b).



(a) (b)

Figure 1: Geometry and physics of the simulated scenarios. (a) Three-dimensional flow over an elastic structure. Streamlines and pressure contours in the fluid domain at $t = 2.8\,s$. (b) Wave propagation in a three-dimensional elastic tube. Geometry and pressure contours on the fluid-structure interface at $t = 8.9 \cdot 10^{-3}\,s$

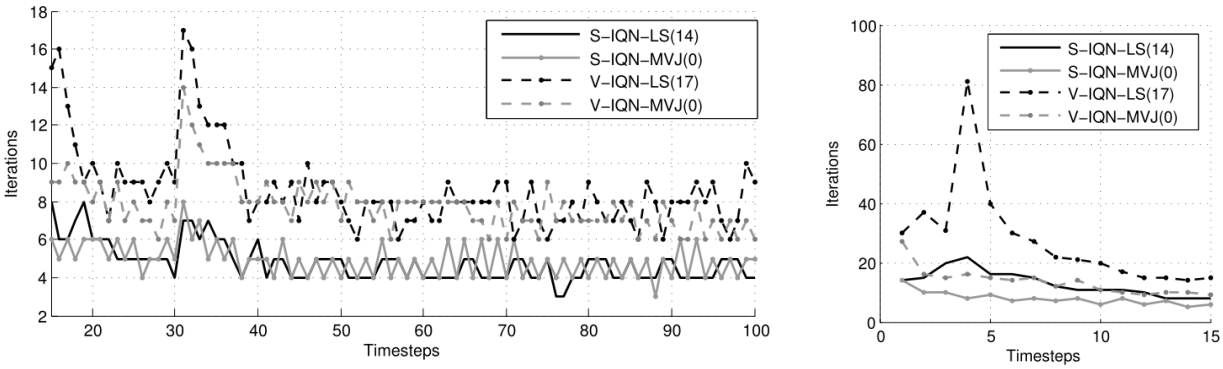
Results. The fluid mesh consists of 17600 cells, and the structure mesh contains 800 cells. Hundred time steps of 1×10^{-4} s are performed. A relative convergence measure of 10^{-5} is used for both, the displacements and the traction on the fluid-structure interface. The results of the simulation are shown in Tab. 3. It can be observed that the behavior of S-IQN-LS(9) and S-IQN-MVJ(0) is quite similar, cf. Fig. 2 (a), which confirms the results of Sect. 3.2. Also, it seems to be advantageous to reuse more time steps for the parallel coupling than for the serial coupling. Furthermore, for parallel coupling, it turns out that the IQN-MVJ scheme is more robust and performs slightly better than the IQN-LS. This dominance of the IQN-MVJ(0) over the IQN-LS(17) is mainly due to the fact that the IQN-MVJ is far more robust during the attack time of the simulation, as can be seen from Fig. 2 (b).

4 CONCLUSIONS

We applied the multi-vector update scheme from Bogaers et al. [14] on the inverse Jacobian formulation – the similar setting that was used by Degroote et al. to derive the classical IQN-ILS scheme [11]. We tested this new coupling scheme, the IQN-MVJ, as well as the IQN-ILS for serial and parallel coupling on two 3D scenarios. The multi-vector

Table 3: Wave propagation in a three-dimensional elastic tube. Average numbers of coupling iterations over first 100 time steps for IQN-ILS and IQN-MVJ, and for serial and parallel coupling. Different numbers of reused time steps are evaluated. † For a well-conditioned V-system, we scale the forces by a factor of 10^{10} for IQN-LS and $4.0 \cdot 10^8$ for IQN-MVJ.

Reuse(R)	0	1	3	5	7	9	11	13	15	17
S-IQN-LS(R)	9.97	9.45	6.80	6.35	6.17	6.05	6.05	6.06	6.15	6.21
S-IQN-MVJ(R)	5.36	8.97	9.47	9.32	9.25	9.21				
V-IQN-LS(R)†	21.41	18.63	13.73	12.90	12.28	11.95	11.71	13.36	13.36	11.47
V-IQN-MVJ(R)†	8.67	15.18	17.66	17.60	16.99	17.13				



(a) (b)

Figure 2: Wave propagation in a three-dimensional elastic tube. Iteration numbers for IQN-LS and IQN-MVJ, and for serial and parallel coupling. (a) timesteps 16 to 100, (b) timesteps 1 to 15.

update renders, as expected, a fine-tuning of the reuse of old information unnecessary. Similar to our previous findings for IQN-ILS [10], IQN-MVJ almost retains the convergence order, when moving from the serial coupling to the parallel one. In upcoming work, we want to automate the pre-conditioning of the parallel coupling.

Acknowledgement 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] Bazilevs, Y., Takizawa, K. and Tezduyar, T. E. *Computational Fluid-Structure Interaction - Methods and Applications.* John Wiley and Sons, Vol. I., (2013).

[2] Gee, M., Küttler, U. and Wall, W. A. Truly monolithic algebraic multigrid for fluid-

structure interaction *Int. J. for Numer. Meth. in Eng.* (2011) **85**:987–1016.

[3] Crosetto, P., Deparis, S., Fourestey, G. and Quarteroni, A. Parallel algorithms for fluid-structure interaction problems in haemodynamics *Siam SISC* (2011) **33**:1598–1622.

[4] Bungartz, H.-J., Lindner F., Mehl M. and Uekermann, B. A plug-and-play coupling approach for parallel multi-field simulations *Comp. Mech.* (2014):1–11.

[5] Nobile, F. and Vergara, C. An effective fluid-structure interaction formulation for vascular dynamics by generalized Robin conditions *SIAM SISC* (2008) **30**:73–763.

[6] Causin, P., Gerbeau, J. F. and Nobile, F. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Comput. Methods Appl. Mech. Eng.* (2005) **194**:4506–4527.

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

[8] Farhat, C. and Lesoinne, M. Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems. *Comput. Method. Appl. M.* (2000) **182**:499–515.

[9] Ross, M.R., Felippa, C.A., Park, K.C. and Sprague, M.A. Treatment of acoustic fluid-structure interaction by localized Lagrange multipliers: Formulation. *Comput. Methods Appl. Mech. Eng.* (2008) **197**:305–3079.

[10] Uekermann, B., Bungartz, H. J., Gatzhammer, B. and Mehl, M. A parallel, black-box coupling algorithm for FSI *Proc. ECCOMAS Coupl. Prob.* (2013).

[11] Degroote, J., Bathe, K.-J. and Vierendeels, J. Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction *Comput. Struct.* (2009) **87**:793–801.

[12] Gatzhammer, B. Efficient and flexible partitioned simulation of fluid-structure interactions *Ph.D. thesis, Technische Universität München, Inst. für Informatik* (2015).

[13] Blom, D. S., van Zuijlen, A. H. and Bijl, H. Acceleration of strongly coupled fluid-structure interaction with manifold mapping *Proc. 11th. WCCM* (2014) 4484–4495.

[14] Bogaers, A. E. J., Kok, S., Reddy, B. D. and Franz, T. Quasi-Newton methods for implicit black-box FSI coupling *Comp. Meth. in Appl. Mech. and Eng.* (2014) **279**:113–132.

[15] Scheufele, K. Robust quasi-Newton methods for partitioned fluid-structure interaction *Master thesis, Universität Stuttgart, Fakultät für Informatik* (2015).

[16] Darwish, M., Sraj, I. and Moukalled, F. A coupled finite volume solver for the solution of incompressible flows on unstructured grids *J. of Comp. Phys.* (2009) **228**:180–201.

[17] de Boer, A., van Zuijlen, A. H. and Bijl, H. Radial basis functions for interface interpolation and mesh deformation *Lecture Notes in CSE* (2010) **71**:143–178.