Proceedings of the ASME 2022 41st International Conference on Ocean, Offshore and Arctic Engineering
OMAE2022
June 5-10, 2022, Hamburg, Germany

OMAE2022-80507

ADVANCED METHODS FOR PARTITIONED FLUID-STRUCTURE INTERACTION SIMULATIONS APPLIED TO SHIP PROPELLERS

Jorrid Lund, Lars Radtke, Alexander Düster
Numerical Structural Analysis with Application in Ship Technology
Institute for Ship Structural Design and Analysis (M-10)
Hamburg University of Technology (TUHH)
Hamburg, 21073
Germany
Email: Jorrid.Lund@tuhh.de

Daniel Ferreira González, Moustafa Abdel-Maksoud
Institute for Fluid Dynamics and Ship Theory (M-8)
Hamburg University of Technology (TUHH)
Hamburg, 21073
Germany

ABSTRACT

In naval architecture, fluid-structure interaction is highly important for many applications. The accurate and fast computation of fluid-structure interaction problems is for this reason a major challenge for a simulation engineer working on flexible structures interacting with water and wind.

For ship propellers, steel and metal alloy have long been the dominating choice of material. With the advancement in the development of fiber-reinforced polymers such as carbon fiber reinforced polymers the consideration of fluid-structure interaction for ship propellers becomes increasingly important.

This work presents a partitioned coupled solution approach for the simulation of fluid-structure interaction problems on the example of a large ship propeller. The in-house developed software library comana is used as coupling manager together with the commercial finite element software ANSYS as structural solver and the boundary element method code panMARE as fluid solver. comana offers the possibility to couple a number of existing and highly specialized solvers to solve multifield problems.

For partitioned coupled fluid-structure interaction problems the increased computational effort due to the necessary coupling iterations and possible instabilities due to the partitioned coupling should be reduced by suitable predictor and convergence acceleration methods. For convergence acceleration, the Aitken method is one of the most common choices even though Quasi-Newton methods such as the Quasi-Newton least-squares method show promising results for the acceleration of fluid-structure interaction simulations.

The simulation of the dynamic structural behaviour of a ship propeller is introduced and the advantages and disadvantages of the partitioned fluid-structure interaction simulation approach are shown. Predictor and convergence acceleration schemes to improve the solution process are discussed and results for flexible ship propellers are presented.

INTRODUCTION

The accurate and fast solution of flexible marine structures deforming under the influence of water and wind is still a major challenge for simulation engineers. Fluid-structure interaction (FSI) plays a key role in the modelling of flexible marine structures. For the numerical simulation of FSI problems, different solution strategies exist. Monolithic and partitioned solution strategies can be used to simulate FSI problems. Monolithic solution strategies solve the fluid and structural sub-problems at once, which makes the development of a dedicated integrated FSI solver necessary [1].

Partitioned solution strategies have, compared with this, the

advantage of reusing existing advanced and specialized solvers with only small adaptations to the non-coupled solution process in these solvers. With the partitioned solution approach it is usually necessary to solve each of the sub-fields multiple times until all field are in equilibrium which increases the computational effort for the solution. On the other hand, the computational effort for the solution of the sub-problems in a partitioned approach is lower compared to a monolithic approach since the equation systems of the sub-problems are smaller. The major disadvantage of partitioned solution approaches is that they are less stable compared to monolithic approaches [1]. The disadvantage of the increased computational effort and the instability in the partitioned solution approach can be significantly reduced by using prediction and convergence acceleration techniques [2]. In this paper different convergence acceleration techniques are compared on the example of a flexible ship propeller. The necessary number of implicit coupling iterations is used as a measure of the performance of the convergence acceleration method. The in-house developed software library *comana* [3] is used together with the commercial finite element solver ANSYS [4] for the solution of the structural sub-problem and the in-house developed boundary element solver *pan*MARE [5] for solution of the fluid sub-problem.

PARTITIONED SOLUTION APPROACH

In this section the partitioned solution approach for FSI problems used in this paper is introduced and explained. With the partitioned solution approach for FSI problems the fluid and structural problems are solved separately with the exchange of quantities at the interface between the fluid and the structure. Both sub-problems must be solved multiple times in an implicit iteration loop until the quantities that are exchanged at the interface are converged.

The in-house developed software library *comana* was developed as a coupling manager to enable and control the exchange of data between different solvers and to manage the solution process. With *comana* [3] adaptations to the existing solvers are kept small and the existing highly specialized solvers are used as black-boxes. This means that, except for the repetition of time steps and the exchange of the coupling quantities at the interface, the solution process in the solvers is not modified. *comana* is not only capable to solve FSI problems but also multi-physics problems in general [6].

For the implicit coupling of FSI problems, the quantities that are exchanged at the interface between the fluid and structural sub-problems are chosen to be displacements $\mathbf{d}$ computed by the structural solver and tractions $\mathbf{t}$ computed by the fluid solver.

The solution process is depicted in Figure 1. After the initialisation of the fluid and structural solver, the time step loop is entered. At the beginning of the simulation no previous data exists, therefore the displacements at the interface are set to

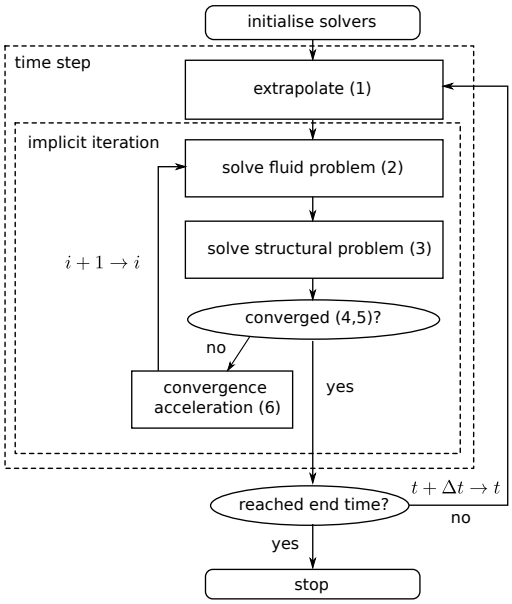

FIGURE 1. Partitioned solution approach for FSI problems

zero. When converged displacements from previous time steps are available, an extrapolation $\mathscr{E}$ of the displacement from the previous time steps

$$\tilde{\mathbf{d}}_1^t = \mathscr{E}(\mathbf{d}^{t-1}, ..., \mathbf{d}^{t-m}) \tag{1}$$

can be performed to obtain a prediction of the displacement for the current time step in the first implicit iteration $\tilde{\mathbf{d}}_1^t$. The extrapolation methods are described in more detail in the next section.

With the extrapolated displacements the implicit iteration loop can be entered and the sub-problems can be solved in a segregated way. The displacements $\tilde{\mathbf{d}}_i^t$ are transferred to the fluid solver $\mathscr{F}$ which computes depending on these displacements tractions on the interface

$$\mathbf{t}_i^t = \mathscr{F}(\tilde{\mathbf{d}}_i^t). \tag{2}$$

These tractions are transferred in the next step to the structural solver $\mathscr{S}$ to compute new displacements

$$\mathbf{d}_i^t = \mathscr{S}(\mathbf{t}_i^t). \tag{3}$$

The combination of Eq. (2) and Eq. (3) leads to a fixed-point iteration that can be repeated until convergence is obtained. A residual is introduced to check if the result is converged. Two different convergence criteria are used to check for convergence.

The first criterion is an absolute convergence criterion which is fulfilled if the L_2-norm of the residual $\mathbf{r}_i$ is below a predefined absolute tolerance ε_a.

$$\mathbf{r}_i = \mathbf{d}_i^t - \tilde{\mathbf{d}}_i^t, \quad \|\mathbf{r}_i\| < \varepsilon_a. \tag{4}$$

The second criterion checks a relative convergence tolerance ε_r that compares the residual in the first implicit iteration $\mathbf{r}_1$ with the residual of the current iteration $\mathbf{r}_i$ according to

$$\frac{\|\mathbf{r}_i\|}{\|\mathbf{r}_1\|} < \varepsilon_r. \tag{5}$$

If one of the two convergence criteria is fulfilled, the fixed-point iteration is considered to be converged. When the fixed-point iteration is converged or the maximum number of implicit iterations is reached, the displacements $\mathbf{d}^t$ for the current time step are saved and the next time step is started with Eq. (1).

If not, a convergence acceleration method $\mathscr{A}$ (6) is used to improve the estimated displacement in the next implicit iteration

$$\tilde{\mathbf{d}}_{i+1}^t = \mathscr{A}(\mathbf{d}_i^t, \tilde{\mathbf{d}}_i^t, \mathbf{d}_{i-1}^t, ..., \mathbf{d}_{i-n}^t). \tag{6}$$

EXTRAPOLATOR

A good prediction of the displacement in the current time step can reduce instabilities and the number of necessary implicit iterations and with this also the computational effort to solve the whole system [7]. Well known classes of predictors are for example polynomial predictors and predictors based on the Taylor-series expansion. With polynomial predictors a polynomial of order p is constructed that takes the values of the previous displacements $(d^{t-1}, ..., d^{t-(p+1)})$ at the previous time steps as

$$d^{t+j} = \sum_{i=0}^{p} \beta_i t_j^i \qquad \text{for } j = -1, ..., -(p+1). \tag{7}$$

The components of the displacement are treated independently in x-, y- and z-direction. With Eq. (7) a system of equations is constructed with that the coefficients β_i can be determined. The resulting polynomial can be evaluated at the current time t. This gives a relation between the previous displacements and the current displacement that is constant if the time step size is constant. Since every point on the interface is treated separately, the approach can easily be extended to vector fields $\mathbf{d}$.

The resulting equations for the polynomial predictor are listed in Table 1.

TABLE 1. Equations for polynomial extrapolation.

order p	equation
0	$\tilde{\mathbf{d}}_1^t = \mathbf{d}^{t-1}$
1	$\tilde{\mathbf{d}}_1^t = 2\mathbf{d}^{t-1} - \mathbf{d}^{t-2}$
2	$\tilde{\mathbf{d}}_1^t = 3\mathbf{d}^{t-1} - 3\mathbf{d}^{t-2} + \mathbf{d}^{t-3}$

CONVERGENCE ACCELERATOR

The purpose of convergence acceleration is to stabilize and accelerate the implicit iteration loop.

Static relaxation

The simplest method of convergence acceleration is a static relaxation which multiplies the residual $\mathbf{r}_i$ with a constant relaxation factor ω. The aim of this method is mainly to stabilize the fixed-point acceleration originated from the combination of Eq. (2) and (3). This approach specifies the accelerator operator $\mathscr{A}$ as

$$\tilde{\mathbf{d}}_{i+1}^t = \mathscr{A}(\mathbf{d}_i^t, \tilde{\mathbf{d}}_i^t) = \tilde{\mathbf{d}}_i^t + \omega(\mathbf{d}_i^t - \tilde{\mathbf{d}}_i^t) = \tilde{\mathbf{d}}_i^t + \omega\mathbf{r}_i. \tag{8}$$

The choice of an appropriate relaxation factor is of course important for the performance of this method and it is not trivial since for different problems the optimal relaxation factor might be different. For the simulations with static relaxation in this work, the relaxation factor is set to $\omega = 0.5$.

Aitken method

Many extensions of the Aitken method for scalar valued problems [8] to vector valued problems have been developed [9]. Multidimensional Aitken methods compute the relaxation factor ω in contrast to the previous acceleration method now as a dynamic factor. The idea of the method used here is to choose the relaxation factor ω such that in

$$\tilde{\mathbf{d}}_i^t = \mathbf{d}_{i-1}^t + \omega_i(\mathbf{d}_{i-2}^t - \mathbf{d}_{i-1}^t)$$
$$\tilde{\mathbf{d}}_{i+1}^t = \mathbf{d}_i^t + \omega_i(\mathbf{d}_{i-1}^t - \mathbf{d}_i^t)$$

the norm of the difference $\|\tilde{\mathbf{d}}_i^t - \tilde{\mathbf{d}}_{i-1}^t\|$ is minimized [9].

After introducing a new residual $\hat{\mathbf{r}}_i = \mathbf{d}_i^t - \mathbf{d}_{i-1}^t$, the expression for the computation of the dynamic relaxation factor is

$$\omega_i = -\frac{\hat{\mathbf{r}}_{i-1} \cdot (\hat{\mathbf{r}}_i - \hat{\mathbf{r}}_{i-1})}{\|\hat{\mathbf{r}}_{i-1} - \hat{\mathbf{r}}_i\|^2}. \tag{9}$$

The disadvantage of this method is that the relaxation factor ω_i can only be computed every 3rd implicit iteration since the

residuals of the two previous iterations must be known to compute ω_i according to (9). For the intermediate iterations an relaxation factor ω_m must be chosen. ω_m is set to 1 for the simulations in this work.

Irons-Tuck relaxation method

Irons and Tuck [10] proposed a modification to the multidimensional Aitken method that enables the computation of the relaxation factor ω in every iteration by using the residuals of the previous iterations to update rule to the relaxation factor. For FSI problems this method is also often also referred to as Aitken method [11] which would cause here confusion with the previous method. In this work the method is therefore referred to only as Irons-Tuck relaxation method. The resulting equation for the update of the relaxation factor ω is

$$\omega_i = -\omega_{i-1}\frac{\hat{\mathbf{r}}_{i-1}\cdot(\hat{\mathbf{r}}_i-\hat{\mathbf{r}}_{i-1})}{\|\hat{\mathbf{r}}_{i-1}-\hat{\mathbf{r}}_i\|^2}.$$

In the first iteration the relaxation factor ω_0 must be initialized. For the simulations in this work ω_0 is set to 0.5.

Quasi-Newton least squares method

The quasi-Newton least squares (QNLS) method for the solution of FSI problems has been introduced by Haelterman [12] and Degroote [13].

The idea of the QNLS method will be briefly illustrated here. Since QNLS method belongs to the class of the quasi-Newton methods an approximation of the Jacobi matrix in the Newton method is computed.

In the first step the fixed point iteration in Eq. (2) and (3) can be reformulated as

$$\mathbf{R}(\tilde{\mathbf{d}}_i^t) = \mathscr{S}(\mathscr{F}(\tilde{\mathbf{d}}_i^t)) - \tilde{\mathbf{d}}_i^t = \mathbf{r}_i \tag{10}$$

to be solved with the Newton-Raphson method:

$$\begin{aligned}\frac{\partial \mathbf{R}(\tilde{\mathbf{d}}_i^t)}{\partial \tilde{\mathbf{d}}_i^t}\Delta\tilde{\mathbf{d}}_i^t &= -\mathbf{R}(\tilde{\mathbf{d}}_i^t)\\ \tilde{\mathbf{d}}_{i+1}^t &= \tilde{\mathbf{d}}_i^t + \Delta\tilde{\mathbf{d}}_i^t.\end{aligned} \tag{11}$$

Since the Jacobi matrix $\frac{\partial \mathbf{R}(\tilde{\mathbf{d}}_i^t)}{\partial \tilde{\mathbf{d}}_i^t}$ in Eq. (11) is unknown and a computation of the Jacobi matrix e.g. with the finite difference method is prohibitively expensive, the inverse of the Jacobi matrix in the quasi-Newton methods is approximated based on the already existing data of the previous implicit iterations. The idea of the QNLS method is to minimize the residual in the next implicit iteration $\mathbf{r}_{i+1}$. To this end the unknown residual difference $\Delta\mathbf{r}_{i,i+1} = \mathbf{r}_{i+1} - \mathbf{r}_i$ is approximated by a linear combination of the already known residual difference $\Delta\mathbf{r}_{i,j} = \mathbf{r}_j - \mathbf{r}_i$ as

$$\Delta\mathbf{r}_{i,i+1} \approx \sum_{j=0}^{i-1}\alpha_j\Delta\mathbf{r}_{i,j}. \tag{12}$$

The minimization problem for the current residual

$$\text{argmin}\left\|\mathbf{r}_i + \sum_{j=0}^{i-1}\alpha_j\Delta\mathbf{r}_{i,j}\right\|^2 \rightarrow \boldsymbol{\alpha}_i \tag{13}$$

is to this end solved in a least squared sense to compute the coefficient vector $\boldsymbol{\alpha}_i = \begin{bmatrix}\alpha_0 \ \dots \ \alpha_{i-1}\end{bmatrix}^{\mathrm{T}}$.

With the known coefficients $\boldsymbol{\alpha}_i$, an update rule for the displacements can be formulated. To this end, two new displacement differences are introduced: $\Delta\mathbf{d}_{i,j} = \mathbf{d}_j^t - \mathbf{d}_i^t$ and $\Delta\tilde{\mathbf{d}}_{i,j}^t = \tilde{\mathbf{d}}_j^t - \tilde{\mathbf{d}}_i^t$. With these definitions, the residual difference $\Delta\mathbf{r}_{i,j}$ can be reformulated as

$$\Delta\mathbf{r}_{i,j} = \mathbf{r}_j - \mathbf{r}_i = \mathbf{d}_j^t - \tilde{\mathbf{d}}_j^t - \mathbf{d}_i^t + \tilde{\mathbf{d}}_i^t = \Delta\mathbf{d}_{i,j}^t - \Delta\tilde{\mathbf{d}}_{i,j}^t. \tag{14}$$

Eq. (13) can be now expressed as

$$\mathbf{r}_i + \sum_{j=0}^{i-1}\alpha_j\Delta\mathbf{r}_{i,j} = \mathbf{r}_i + \sum_{j=0}^{i-1}\alpha_j\Delta\mathbf{d}_{i,j}^t - \sum_{j=0}^{i-1}\alpha_j\Delta\tilde{\mathbf{d}}_{i,j}^t \approx \mathbf{0} \tag{15}$$

by assuming that the coefficients resulting from the minimization problem (13) can be transferred to the displacement differences, the update rule to compute the displacement increment in Eq. (11) is

$$\mathbf{r}_i + \sum_{j=0}^{i-1}\alpha_j\Delta\mathbf{d}_{i,j}^t \approx \sum_{j=0}^{i-1}\alpha_j\Delta\tilde{\mathbf{d}}_{i,j}^t \approx \Delta\tilde{\mathbf{d}}_i^t. \tag{16}$$

Even though the inverse of the Jacobi matrix in Eq. (11) is not calculated directly the method provides an approximation of the displacement increment. In this work the QNLS method is used in the single time step version without resetting.

Broyden method

Like the QNLS method, the Broyden method [14] belongs to the class of quasi-Newton method and is therefore based on

the idea to approximate the Jacobi matrix in Eq. (11). Several versions of the Broyden method exist. The problem of some versions of the Broyden method is that they require to store a large matrix for a high number of degrees of freedom of the vector $\mathbf{d}$. This disadvantage can be overcome by a restart formulation as described e.g. in [15]. In this work the version described in [16] is used.

SIMULATION SETUP

The example considered in this work is the KRISO container ship (KCS) propeller KP505. The propeller is simulated with the partitioned solution approach with the commercial finite element solver ANSYS [4] for the structural domain and the boundary element solver *pan*MARE [5] for the fluid domain. The coupling manager *comana* is used to manage the solution process including the extrapolation, interpolation and the convergence acceleration used for the implicit coupling.

The KCS propeller is a well investigated example of a large scale container ship propeller. The five bladed propeller has a diameter of $d = 7.9\,\mathrm{m}$. The advance ratio J of the propeller is defined as

$$J = \frac{v}{n\,d}\,, \tag{17}$$

with v being the averaged inflow velocity and n being the number of rotations per second. The number of rotations is set to $n = 1.5\,\frac{1}{\mathrm{s}}$.

The startup of the simulation is performed such that the angle increment is increased only slowly by multiplying it with a factor f. This is done from the beginning of the simulation at $t_s^1 = 0$ until full rotational speed is obtained at time $t_e^1 = 90\Delta t$.

The factor f is computed as

$$f = 0.5\left(1 - \cos\left(\frac{t - t_s}{t_e}\pi\right)\right). \tag{18}$$

The displacements computed with the angle increment are prescribed in ANSYS on the hub of the propeller blades. The time step size Δt in the simulation is chosen such that the angle increment is equal to 2° when full rotational speed is obtained. The time step size is therefore $\frac{2^\circ}{360^\circ}\frac{1}{n} = \frac{1}{270}\,\mathrm{s}$.

Similar to the angle increment the tractions are also applied with a cosine function like in Eq. (18) with a factor f_t such that Eq. (3) changes to $\mathbf{d}_i^t = \mathscr{S}(f_t\mathbf{t}_i^t)$. The tractions are applied after the full rotational speed is obtained and possible oscillations are faded out beginning from $t_s^2 = 135\Delta t$ until $t_e^2 = 180\Delta t$.

Fluid simulation

The BEM code *pan*MARE that is used for the fluid simulation is a low order order panel method solver that has been developed by the Institute for Fluid Dynamics and Ship Theory.

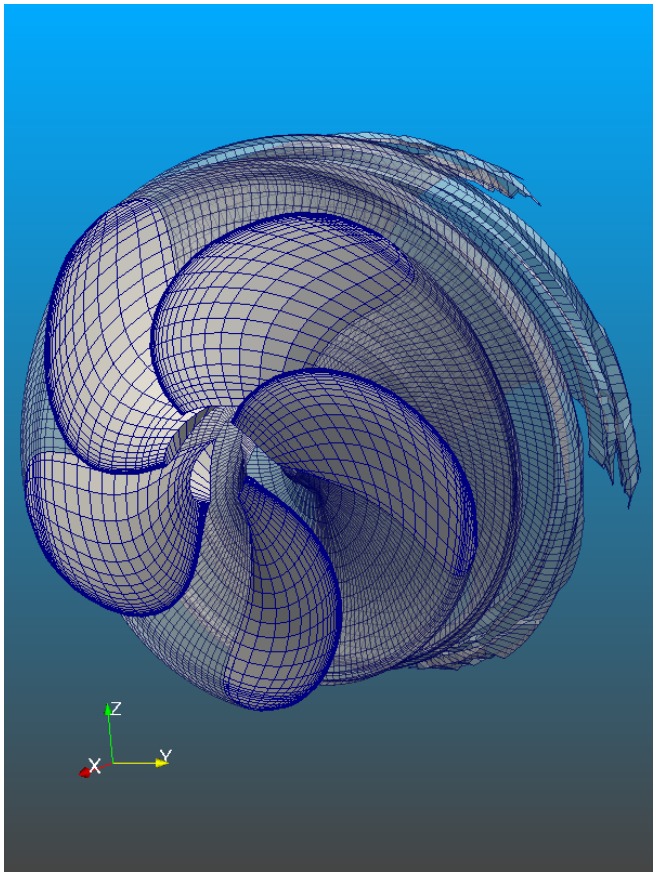

FIGURE 2. Fluid mesh of the propeller with wake

The panel method is based on the potential theory which means that the assumptions for the fluid field are that it is incompressible, irrotational and inviscid. Even though the viscous force can not be directly used for the fluid solver, a friction model is used that estimates the friction based on the *Schoenherr line* as described in [17]. More details about the panel method and its implementation can be found in [18].

The fluid mesh as shown in Fig. 2 has in span-wise direction 25 panels and in chord-wise direction 18 panels on each side. The wake of the propeller is discretized with panels as described in [18]. The panels of the wake deform to follow the streamlines which leads to roll-up of the wake as described in [19, p. 22]. In Figure 2 also the wake panels are visualized.

In *pan*MARE the pressure forces and the friction forces are evaluated at the panel center and summed up to obtain the total force for each panel. The total force is divided by the panel area in order to calculate the tractions that must be computed for the implicit coupling according to Eq. (2). *comana* is coupled to *pan*MARE such that the time steps in *pan*MARE can be repeated and the displacement boundary conditions computed by

structural solver can be set in every time step and implicit iteration. Since the implicitly coupled FSI solution approach requires the multiple solution of the fluid domain in every time step, a reduction in computational time is especially important in this approach.

The propeller operates in an inhomogeneous ship wake field to include the velocity signature of the ship, leading to a varying loading condition during the rotation. The advance ratio J defined in Eq. (17) in the fluid simulation is approximately 0.85.

Compared to approaches to simulate the fluid domain based on the finite volume method, the panel method has the advantage that only the surface of the propeller and the wake surface must be discretized which enables a solution of the fluid problem within a reasonable amount of time. The computation of the fluid domain with the finite volume method requires the discretization of the whole fluid volume. The Reynolds averaged Navier-Stokes approach as described in [20] is expensive since the domain must be sufficiently large and the resolution close to the structural surface must be relatively high to account for wall effects in the boundary layer. Furthermore problems with mesh distortion can more easily be avoided with the panel method.

Structural simulation

The commercial FEM solver ANSYS is used on the structural side for the dynamic FEM simulation. In ANSYS, a structured mesh as depicted in Figure 3 with linear hexahedral elements is used. Since the structural mesh is in contrast to the fluid mesh in *pan*MARE a volumetric mesh, the meshing on the leading and trailing edge as well as on the tip propeller is challenging. Wedge elements at the trailing and leading edge of the propeller blades are necessary when a structured meshing approach is used.

A geometrically nonlinear approach is used with a linear elastic material model. To stabilize the structural simulation in the coupled solution process, stiffness proportional Rayleigh damping [21] is applied. As time integration method for the FEM simulation the Newmark method [22] with the constant average acceleration method is used corresponding to the parameters $\delta = \frac{1}{2}$ and $\alpha = \frac{1}{4}$. To solve the nonlinear system of equation in ANSYS the Newton-Raphson method [23] is utilized. The Poisson number in the simulations in this work is 0.3 and the density is equal to $7750 \frac{\text{kg}}{\text{m}^3}$.

The tractions computed by the fluid solver are integrated over the discretized structural surface in order to compute forces that can be applied on the structural nodes in ANSYS. In this way the structural solver can be used as defined in Eq. (3) to compute new displacements.

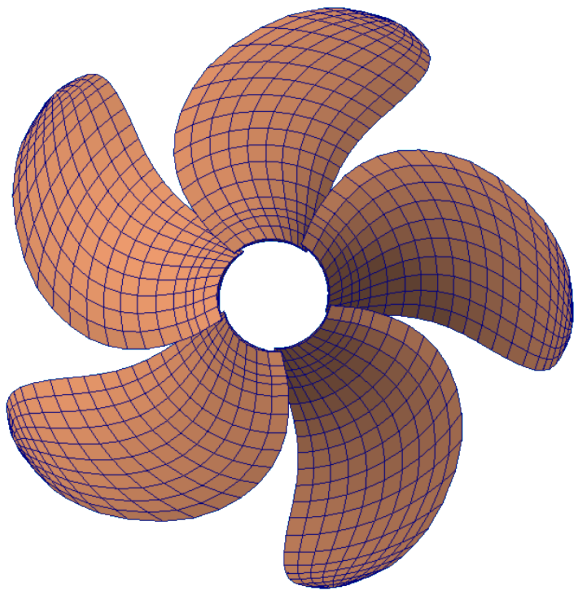

FIGURE 3. Structural mesh on the suction side of the propeller

COMPARISON OF CONVERGENCE ACCELERATORS

To compare the convergence acceleration methods, the KCS propeller setup described in the previous section is used.

The number of implicit iteration i that are necessary to obtain convergence according to the absolute criterion Eq. (4) or to the relative criterion Eq. (5) are used as a measure of performance of the accelerator. The assumption is here that the computational effort for the solution of the fluid and structural field according to Eq. (2) and (3) is much higher than the necessary effort for the convergence accelerator. The maximum number of implicit iterations is set to 50 for all the simulations shown here.

TABLE 2. Relative and absolute tolerances (Eq. (4) and (5)) used in the simulations

	rel. tol. ε_r	abs. tol. ε_a
low tolerance	$5 \cdot 10^{-3}$	10^{-6}
middle tolerance	$5 \cdot 10^{-8}$	10^{-11}
high tolerance	$5 \cdot 10^{-11}$	10^{-15}

The extrapolation method chosen for all the following simulations is the quadratic (p=2) extrapolation with the factors given in Table 1. The tolerances are varied to the values listed in Table 2. Concerning the stopping criteria, the relative convergence criterion is usually the one which is fulfilled first for the simulation

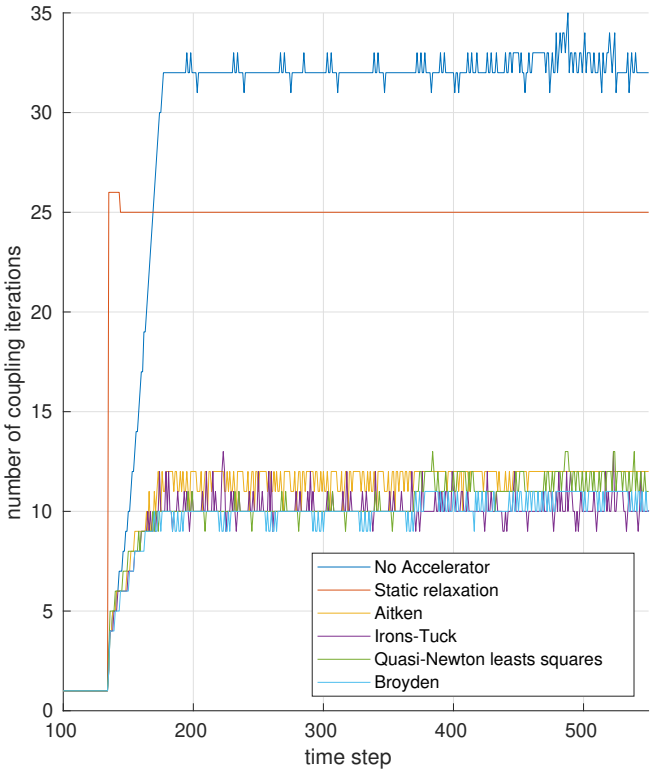

FIGURE 4. Comparison of acceleration methods for a Young's modulus of 210 GPa

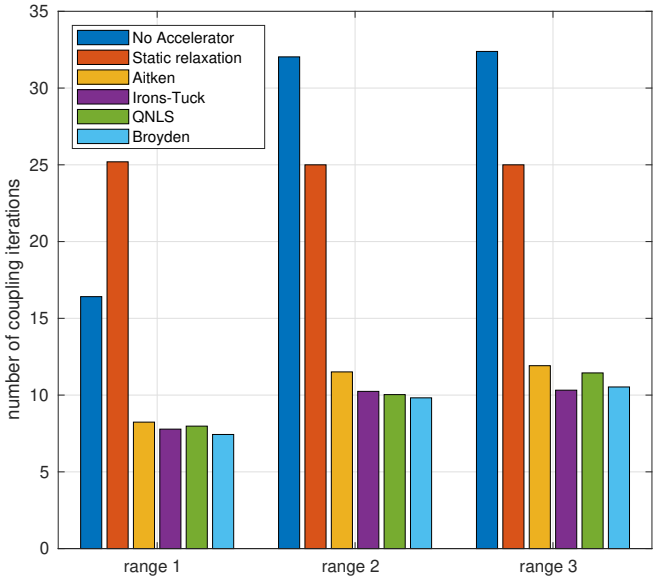

FIGURE 5. Number of necessary implicit coupling iterations for a Young's modulus of 210 GPa

setup applied in this work.

The Young's modulus is varied to 2 GPa, 5 GPa, 20 GPa and 210 GPa leading to a total number of 12 setups for which the 5 different convergence accelerators and the simulation without convergence acceleration are compared. The simulations are carried out on the high performance cluster (HPC) of the Hamburg University of Technology. A selection of the results of the simulations will be presented here.

The Young's modulus in the structural simulation is varied since it strongly influences the interaction between the fluid and the structure and with this the necessary number of internal iterations. For a propeller with a high Young's modulus made from a material like steel or a metal alloy, the deflection of the propeller blade due to the fluid loads is small and therefore the number of necessary coupling iterations is lower.

In Figure 4 the comparison between the different acceleration methods is shown for a Young's modulus of 210 GPa and the middle tolerance criterion.

Before the tractions are applied at time step 135, the number of implicit coupling iterations is set to 1 since there are no external forces acting on the structure. The displacements of the structure are still transferred to the fluid solver therefore the coupling is only one way from the structure to the fluid until time step 135.

In Figure 5 the number of implicit coupling iterations is plotted for the same simulations visible in Figure 4. Three different time step ranges are distinguished here. In range 1 from time step 135 to time step 179 the tractions are slowly applied like described in the previous section. Range 2 captures the simulation with fully transferred tractions $f_t = 1$ but with no wake deformation enabled from time step 180 to 369. In range 3 from time step 370 to the end of the simulation, the wake deformation is also enabled. The number of implicit coupling iterations is averaged over the considered range.

In range 1 a jump is visible in Figure 4 for the static relaxation with a relaxation factor of 0.5. Figure 4 shows that the necessary number of coupling iterations without an acceleration method and with static relaxation as convergence accelerator perform not very well. The number of coupling iterations is more than twice as high compared to the more advanced convergence acceleration methods. With this also the computational time for these two approaches is also more than twice as high. The Irons-Tuck relaxation, the QNLS method and the Broyden method perform all quite good as convergence accelerator. The Aitken method converges a little bit slower than these three methods. When comparing the three most efficient methods in the three different regions it seems that the QNLS method converges slower when the wake deformation is enabled.

Another comparison of the different convergence acceleration methods is shown in Figure 5. The results for the lower Young's moduli here are similar to the previous results. For a more flexible propeller, all simulations without a convergence

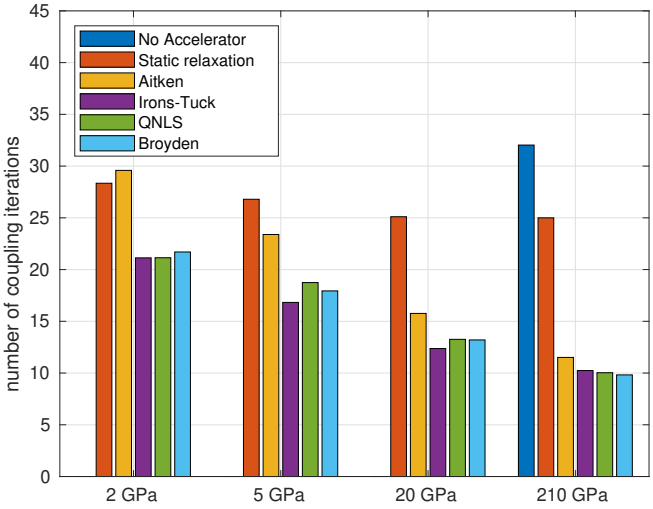

FIGURE 6. Number of necessary implicit coupling iterations for a varying Young's modulus in range 2

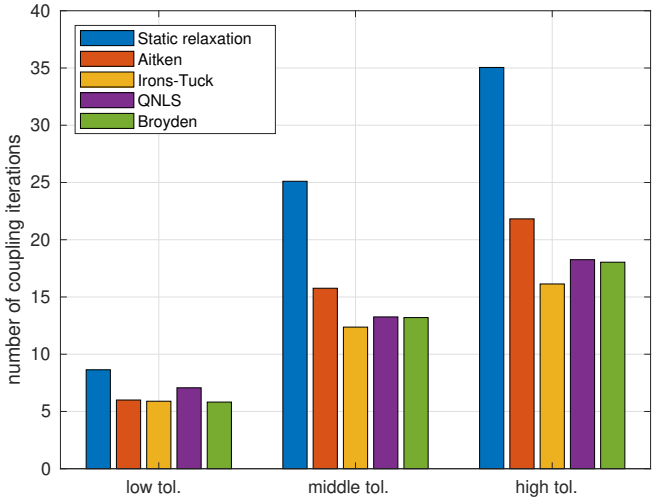

FIGURE 7. Number of necessary implicit coupling iterations for a Young's modulus of 20 GPa in range 2

accelerator fail. This illustrates the importance of convergence acceleration for partitioned coupled FSI simulations. The performance of the Irons-Tuck relaxation, the QNLS method and the Broyden method is again quite similar for the different Young's moduli. With lower Young's modulus the performance of the Aitken method detoriates significantly while the impact of the lower Young's modulus on the performance of the static relaxation seems to be comparatively smaller.

In Figure 7 a comparison for the different tolerance introduced in Table 2 is carried out for a Young's modulus of 20 GPa and range 2.

As before the importance of the usage of a convergence acceleration method becomes clear since the simulations without convergence acceleration method fail. When comparing the QNLS method for example with the Aitken method, it is visible that the QNLS method performs better for middle and high tolerances while the Aitken method performs better for the low tolerances. An explanation for this might be that the QNLS method needs a certain number of implicit iterations and with this a sufficiently large equation system that is solved in the least squares minimization in Eq. (13) before a good approximation of the displacement increment can be computed.

CONCLUSION

A comparison of different convergence acceleration methods has been conducted for the partitioned FSI simulation of the propeller of a large vessel. The study shows the importance of convergence acceleration for partitioned FSI simulations.

For the simulations without convergence acceleration, the simulations failed for a propeller with a low Young's modulus. The static relaxation accelerator performed worse than the other accelerators, but it at least enables a stable simulations for most of the considered simulations.

The Aitken method performed comparatively well for a high Young's modulus, while the performance for a lower Young's modulus detoriates quickly.

In the comparison the QNLS method, the Irons-Tuck relaxation and the Broyden method showed overall the lowest number of necessary implicit coupling iterations. The QNLS performs better with a higher tolerance compared to e.g. the Aitken method.

All of the observations made here of course only apply to the setup that was considered and described in this work. The performance of convergence accelerators strongly depends on the problem at hand. However, the comparison of convergence acceleration methods in [24] and [25] also showed that the QNLS method, the Irons-Tuck relaxation and the Broyden method perform well for artery simulation and benchmark problems. As shown in these works, the performance of the QNLS method could also be improved by using resetting and multi-time step approaches. These variants are particularly useful when combined with parallel or mixed staggered parallel approaches but they require the selection of additional parameters such as the number of reused time steps [26].

The Irons-Tuck relaxation showed a consistently good convergence behaviour over the considered range of parameters. If a simple approach which is also easy to implement is desired, the Irons-Tuck relaxation seems to be a suitable choice. It also has the advantage that only the initial relaxation factor must be set as parameter. In general the advanced convergence acceleration methods all show a significant reduction of the necessary number of implicit coupling iterations and stabilize the solution process.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support provided by the DFG (German Science Foundation – Deutsche Forschungsgesellschaft) under the grant number DU 405/13-2, AB 112/12-2 and the project number 344826971.

REFERENCES

[1] Bazilevs, Y., Takizawa, K., and Tezduyar, T., 2013. *Computational Fluid-Structure Interaction: Methods and Applications*. Wiley Series in Computational Mechanics. John Wiley & Sons.

[2] Bungartz, H., and Schäfer, M., eds., 2006. *Fluid-Structure Interaction, Modelling, Simulation and Optimisation*, Vol. 53 of *Lecture Notes in Computational Science and Engineering*. Springer.

[3] König, M., Radtke, L., and Düster, A., 2016. "A flexible C++ framework for the partitioned solution of strongly coupled multifield problems". *Computers & Mathematics with Applications,* ***72***, pp. 1764–1789.

[4] ANSYS, 2018. Ansys academic research mechanical, release 19.2.

[5] Bauer, M., and Abdel-Maksoud, M., 2012. "A 3-D Potential Based Boundary Element Method for the Modelling and Simulation of Marine Propeller Flows". In 7th Vienna Conference on Mathematical Modelling.

[6] Erbts, P., Hartmann, S., and Düster, A., 2015. "A partitioned solution approach for electro-thermo-mechanical problems". *Archive of Applied Mechanics,* ***85***, pp. 1075–1101.

[7] König, M., 2018. "Partitioned Solution Strategies for Strongly-Coupled Fluid-Structure Interaction Problems in Maritime Applications". PhD thesis, Fachgebiet für Numerische Strukturanalyse mit Anwendungen in der Schiffstechnik (M-10), TU Hamburg.

[8] Aitken, A. C., 1950. "Iv.—studies in practical mathematics. v. on the iterative solution of a system of linear equations". *Proceedings of the Royal Society of Edinburgh. Section A. Mathematical and Physical Sciences,* ***63***, 1, pp. 52–60.

[9] MacLeod, A., 1986. "Acceleration of vector sequences by multi-dimensional Δ^2 methods". *Communications in Applied Numerical Methods,* ***1***, pp. 3–20.

[10] Irons, B., and Tuck, R. C., 1969. "A version of the Aitken Accelerator for Computer implementation". *International Journal for Numerical Methods in Engineering,* ***1***, pp. 275–277.

[11] Küttler, U., and Wall, W., 2008. "Fixed-point fluid-structure interaction solvers with dynamic relaxation". *Computational Mechanics,* ***43***(1), pp. 61–72.

[12] Haelterman, R., Degroote, J., Heule, D., and Vierendeels, a., 2009. "The quasi-newton least squares method: A new and fast secant method analyzed for linear systems". *SIAM Journal on Numerical Analysis,* ***47***, 01, p. 2347.

[13] Degroote, J., Bathe, K.-J., and Vierendeels, J., 2009. "Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction". *Computers & Structures,* ***87***, pp. 793–801.

[14] Broyden, C. G., 1965. "A class of methods for solving nonlinear simultaneous equations". *Mathematics of computation,* ***19***(92), pp. 577–593.

[15] Kelley, C., 2003. *Solving Nonlinear Equations with Newton's Method*, 1 ed. SIAM.

[16] Minami, S., and Yoshimura, S., 2010. "Performance evaluation of nonlinear algorithms with line-search for partitioned coupling techniques for fluid-structure interactions". *International Journal for Numerical Methods in Fluids,* ***64***, pp. 1129–1147.

[17] Hoerner, S. F., 1965. *Fluid-Dynamic Drag - Practical Information on Aerodynamic Drag and Hydrodynamic Resistance*. Hoerner Fluid Dynamics.

[18] Katz, J., and Plotkin, A., 2001. *Low speed aerodynamics*, second edition ed. Cambridge aerospace series. Cambridge University Press, Cambridge.

[19] Göttsche, U., 2020. "Entwicklung einer numerischen Mthode zur Vorhersage der hydroakustischen Schallabstrahlung von Schiffspropellern". doctoralthesis, Technische Universität Hamburg.

[20] Ferziger, J., and Peric, M., 2002. *Computational Methods for Fluid Dynamics*, 3rd, revised ed. Springer-Verlag, Berlin, Heidelberg.

[21] Bathe, K.-J., 2014. *Finite Element Procedures*, 2nd ed. Prentice Hall.

[22] Newmark, N., 1959. "A numerical method for structural dynamics". *Journal of Engineering Mechanics (ASCE),* ***85***, pp. 67–94.

[23] Wriggers, P., 2001. *Nichtlineare Finite-Element-Methoden*. Springer-Verlag.

[24] Radtke, L., Larena-Avellaneda, A., Debus, E., and Düster, A., 2016. "Convergence acceleration for partitioned simulations of the fluid-structure interaction in arteries". *Computational Mechanics,* ***57***(6), pp. 901–920.

[25] Radtke, L., Lampe, T., Abdel-Maksoud, M., and Düster, A., 2018. "A partitioned solution approach for the simulation of the dynamic behaviour of flexible marine propellers". *Ship Technology Research,* ***67***(1), pp. 37–50.

[26] Uekermann, B., Bungartz, H. J., Gatzhammer, B., and Mehl, M., 2013. "A parallel, black-box coupling algorithm for fluid-structure interaction". In Proceedings of the V International Conference on Computational Methods for Coupled Problems in Science and Engineering, pp. 1–12.