PARALLEL TIME-STEPPING FOR FLUID–STRUCTURE INTERACTIONS

NILS MARGENBERG¹ AND THOMAS RICHTER^{2,*}

Abstract. We present a parallel time-stepping method for fluid–structure interactions. The interaction between the incompressible Navier-Stokes equations and a hyperelastic solid is formulated in a fully monolithic framework. Discretization in space is based on equal order finite element for all variables and a variant of the Crank-Nicolson scheme is used as second order time integrator. To accelerate the solution of the systems, we analyze a parallel-in time method. For different numerical test cases in 2d and in 3d we present the efficiency of the resulting solution approach. We also discuss some challenges and limitations that are connected to the special structure of fluid–structure interaction problem. In particular, we will investigate stability and dissipation effects of the time integration and their influence on the convergence of the parareal method. It turns out that especially processes based on an internal dynamics (*e.g.* driven by the vortex street around an elastic obstacle) cause great difficulties. Configurations however, which are driven by oscillatory problem data, are well-suited for parallel time stepping and allow for substantial speedups.

Mathematics Subject Classification. 65L05, 65M22, 65Y05, 76D05, 74F10.

Received July 2, 2019. Accepted January 11, 2021.

1. INTRODUCTION

Fluid structure interactions appear in various problems ranging from classical applications in engineering like the design of ships or aircrafts, the design of wind turbines, but they are also present in bio/medical systems describing the blood flow in the heart or in general problems involving the cardiovascular system. The typical challenge of fluid–structure interactions is two-fold. First, the special coupling character that stems from the coupling of a hyperbolic-type equation – the solid problem – with a parabolic-type equation – the Navier-Stokes equations. Second, the moving domain character brings along severe nonlinearities that have a non-local character, as geometrical changes close to the moving fluid-solid interface might have big impact on the overall solution.

Numerical approaches can usually be classified into *monolithic approaches*, where the coupled fluid–structure interaction system is taken as one entity and into *partitioned approaches*, where two separate problems – for fluid and solid – are formulated and where the coupling between them is incorporated in terms of an outer (iterative) algorithm. This second approach has the advantage that difficulties are isolated and that perfectly

Keywords and phrases: Fluid-structure interactions, parallel time stepping, finite elements, monolithic, parareal.

¹ Helmut Schmidt University, Hamburg, Germany.

² Otto-von-Guericke Universität Magdeburg, Magdeburg, Germany.

* Corresponding author: thomas.richter@ovgu.de

suitable numerical schemes can be used for each of the subproblems. There are however application classes where partitioned approaches either fail or lack efficiency. The *added mass effect* [12] exactly describes this special stiffness connected to fluid–structure interactions. It is typical for problems with similar densities in the fluid and the solid – as it happens in the interaction of blood and tissue or in the interaction of water and the solid structure of a vessel. Here, monolithic approaches are considered to be favourable.

Monolithic approaches all give rise to strongly coupled, usually very large and nonlinear algebraic systems of equations. Although there has been substantial progress in designing efficient numerical schemes for tackling the nonlinear problems [17, 24, 27] (usually by Newton’s method) and the resulting linear systems [3, 21, 31, 38, 48], the computational effort is still immense. Numerically accurate results and efficient approaches for 3d problems are still very rare [16, 39].

In this contribution we exploit the perspectives (and limitations) of parallel time-stepping schemes for fluid–structure interaction systems. We will have to face various difficulties that come from the special type of equations and coupling, such as the hyperbolic property of the solid problem and the saddle-point structure of the fluid system. Parallel time-stepping methods are well established in the literature [7, 11, 13, 18, 23, 30, 44], but up to now there is little experience with fluid–structure interactions. However Parareal has shown to be unstable for hyperbolic problems, where [43] found the phase error between the fine and coarse propagators to be responsible, which already points to possible complications for our use case. In Section 4.2 we will analyse a benchmark configuration where exactly such a phase error will prevent convergence of the Parareal scheme. We also find that the moving domain formulation, which is characteristic for fluid–structure interactions, gives rise to further issues when an incompressible fluid is considered. Our discrete formulation will be based on Arbitrary Eulerian Lagrangian coordinates by mapping the moving fluid domain onto a fixed reference domain. Hereby geometric nonlinearities are introduced and when solutions from different discretizations, *i.e.* fine and coarse propagators, are linearly combined, incompressibility is disturbed.

Parareal has been investigated for the solution of the Navier-Stokes equations [13, 18, 46]. In [46] it has been shown that high Reynolds numbers lead to poor convergence of the algorithm. For low Reynolds numbers, as will be tested here, we expect that the viscous dissipation is dominant. The dissipation properties of coupled fluid–structure interactions are not fully understood and they depend on the material parameters but also on the geometry. Stability can be increased by adding damping within the structure or by damping the coupling between fluid and structure. The analysis is mostly restricted to linear model problems where the Stokes equation is coupled to linear elasticity, see [4–6].

In [13], the potential speedup of parareal beyond the saturation of spatial parallelization for the Navier-Stokes equations is investigated. Since we don’t use other means of parallelization a comparison to the present work seems difficult.

In the following section we will shortly describe a monolithic Arbitrary Lagrangian Eulerian formulation for fluid–structure interaction problems. Further, we detail on the discretization of this system in space (with continuous finite elements) and time (with classical time-stepping methods). The following 3rd section will focus on describing a parallel time-stepping scheme and the special requirements for a realization in terms of fluid–structure interactions. Finally, in Section 4 we present numerical results and discuss possible applications but we also address some limitations.

2. FLUID–STRUCTURE INTERACTIONS

The presentation within this section mainly follows [39]. We consider fluid–structure interaction problems coupling an incompressible fluid with a hyperelastic solid. By $\hat{\mathcal{S}}$ we denote the Lagrangian reference framework of the solid, by $\mathcal{S}(t)$ its current configuration in Eulerian coordinates. By $\mathcal{F}(t)$ we denote the fluid domain at time t matching the solid at the common interface $\mathcal{I}(t) = \partial\mathcal{S}(t) \cap \partial\mathcal{F}(t)$. By $\Omega(t) := \mathcal{F}(t) \cup \mathcal{I}(t) \cup \mathcal{S}(t)$ we denote the Eulerian fluid–structure interaction domain. The domains $\mathcal{F}(t)$, $\mathcal{S}(t)$ and $\Omega(t)$ are all either two-dimensional or three-dimensional. The boundary of the fluid domain $\partial\mathcal{F}(t) = \mathcal{I}(t) \cup \Gamma_f^D(t) \cup \Gamma_f^{out}(t)$ is split into the interface, a Dirichlet part $\Gamma_f^D(t)$ (usually inflow or rigid walls) and an outflow part $\Gamma_f^{out}(t)$, where we ask for the do-nothing condition [25]. For simplicity we assume Dirichlet conditions at the solid boundary apart from the interface

$\partial\mathcal{S}(t) = \mathcal{I}(t) \cup \Gamma_s^D(t)$. Finally, by $I = [0, T]$ we denote the time interval. With the density ρ_f the incompressible Navier-Stokes equations are given by

$$\begin{aligned} \operatorname{div} \mathbf{v}_f &= 0, \quad \rho_f (\partial_t \mathbf{v}_f + (\mathbf{v}_f \cdot \nabla) \mathbf{v}_f) - \operatorname{div} \boldsymbol{\sigma}_f(\mathbf{v}_f, p_f) = \rho_f \mathbf{f}_f, & \text{in } I \times \mathcal{F}(t) \\ \mathbf{v}_f &= \mathbf{v}_f^D & \text{on } I \times \Gamma_f^D(t) \\ \rho_f \nu_f \nabla \mathbf{v}_f \vec{n}_f - p_f \vec{n}_f &= 0 & \text{on } I \times \Gamma_f^{\text{out}}(t) \\ \mathbf{v}_f &= d_t \hat{\mathbf{u}}_s & \text{on } I \times \mathcal{I}(t) \\ \mathbf{v}_f &= \mathbf{v}_f^0 & \text{on } \{0\} \times \mathcal{F}(0), \end{aligned} \quad (2.1)$$

with the right hand side field $\mathbf{f}_f$, boundary data $\mathbf{v}_f^D(t)$ and the interface velocity $d_t \mathbf{u}_s$ coming from the coupling to the solid equation. By $\mathbf{v}_f^0$ we denote the initial velocity. The solid problem in term is given in Lagrangian reference formulation on $\hat{\mathcal{S}}$ as

$$\begin{aligned} \hat{\rho}_s d_{tt} \hat{\mathbf{u}}_s - \widehat{\operatorname{div}}(\hat{\mathbf{F}}_s \hat{\boldsymbol{\Sigma}}_s(\hat{\mathbf{u}}_s)) &= \hat{\rho}_s \hat{\mathbf{f}}_s & \text{in } I \times \hat{\mathcal{S}} \\ \hat{\mathbf{u}}_s &= \hat{\mathbf{u}}_s^D & \text{on } I \times \hat{\Gamma}_s^D \\ \hat{\mathbf{F}}_s \hat{\boldsymbol{\Sigma}}_s(\hat{\mathbf{u}}_s) \hat{\vec{n}}_s &= \boldsymbol{\sigma}_f(\mathbf{v}_f, p_f) \vec{n}_f & \text{on } I \times \hat{\mathcal{I}}, \\ d_t \hat{\mathbf{u}}_s &= \hat{\mathbf{v}}_s^0, \quad \mathbf{u}_s = \hat{\mathbf{u}}_s^0 & \text{on } \{0\} \times \hat{\mathcal{S}}, \end{aligned} \quad (2.2)$$

where we denote by $\hat{\mathbf{F}}_s = I + \hat{\nabla} \hat{\mathbf{u}}_s$ the deformation gradient, by $\hat{\rho}_s$ the reference density, by $\hat{\mathbf{f}}_s$ the right hand side vector, boundary data by $\hat{\mathbf{u}}_s^D$ and by $\boldsymbol{\sigma}_f(\mathbf{v}_f, p_f) \vec{n}_f$ the normal stresses from the coupling to the fluid equations. The hats “ $\hat{\cdot}$ ” are added to distinguish Lagrangian variables from their counterparts in the reference framework. The attribution of the kinematic interface condition $\mathbf{v}_f = d_t \mathbf{u}_s$ to the fluid problem and the dynamic condition to the solid problem is artificial, as the coupled system of (2.1) and (2.2) must be considered as one entity. Finally, as material models we consider a Newtonian fluid and a St. Venant Kirchhoff solid

$$\boldsymbol{\sigma}_f(\mathbf{v}_f, p_f) = \rho_f \nu_f (\nabla \mathbf{v}_f + \nabla \mathbf{v}_f^T) - p_f I, \quad \hat{\boldsymbol{\Sigma}}_s = 2\mu_s \hat{\mathbf{E}}_s + \lambda_s \operatorname{tr}(\hat{\mathbf{E}}_s) I, \quad (2.3)$$

where we denote by $\hat{\mathbf{E}}_s = \frac{1}{2}(\hat{\mathbf{F}}_s^T \hat{\mathbf{F}}_s - I)$ the Green-Lagrange strain tensor and by μ_s, λ_s the Lamé parameters, by ν_f the kinematic viscosity. By (2.1), (2.2) and (2.3) we denote the fluid-structure interaction problem.

2.1. Arbitrary Lagrangian Eulerian coordinates

To overcome the discrepancy between the Eulerian fluid framework and the Lagrangian solid framework, we map the flow problem to a fixed reference domain $\hat{\mathcal{F}}$ that fits the Lagrangian solid domain $\hat{\mathcal{S}}$. Here we assume that $\hat{\mathcal{F}} = \mathcal{F}(0)$ is just the known fluid domain at initial time $t = 0$. By $\hat{T}_f(t) : \hat{\mathcal{F}} \rightarrow \mathcal{F}(t)$ we denote the reference map, by $\hat{\mathbf{F}}_f := \hat{\nabla} \hat{T}_f$ its gradient and by $\hat{J}_f := \det \hat{\mathbf{F}}_f$ its determinant. By $\hat{\mathbf{v}}_f(\hat{x}, t) = \mathbf{v}_f(x, t)$ with $x = \hat{T}_f(\hat{x}, t)$ we denote the ALE representation of the Eulerian variable $\mathbf{v}_f$ (same for the pressure). The Arbitrary Lagrangian Eulerian formulation (ALE) goes back to the 70s [14, 26, 29], and usually consists of adding convective terms with respect to the motion of the domain. We use a strict mapping to the fixed reference system and formulate the coupled variational formulation on this arbitrary framework $\hat{\mathcal{F}}$. Details are given in [39, 40]. To close the ALE-formulation we construct the map $\hat{T}_f := \operatorname{id} + \hat{\mathbf{u}}_f$ by means of an extension of the solid deformation to the fluid domain, denoted by $\hat{\mathbf{u}}_f$. For small deformations of the fluid domain a simple harmonic extension is sufficient

$$-\hat{\Delta} \hat{\mathbf{u}}_f = 0 \text{ in } \hat{\mathcal{F}}, \quad \hat{\mathbf{u}}_f = \hat{\mathbf{u}}_s \text{ on } \hat{\mathcal{I}}, \quad \hat{\mathbf{u}}_f = 0 \text{ on } \partial \hat{\mathcal{F}} \setminus \hat{\mathcal{I}},$$

while problems with large changes in the fluid domain require more care in extending the deformation. Details are given in Section 5.3.5 of [39] and the references therein. We finally give the complete system. From hereon, all problems are given in the reference system such that we skip the hat for better readability

$$\begin{aligned}
J_f \mathbf{F}_f^{-1} : \nabla \mathbf{v}^T &= 0, \quad \rho_f J_f (\partial_t \mathbf{v}_f + \nabla \mathbf{v}_f \mathbf{F}_f^{-1} (\mathbf{v}_f - \partial_t \mathbf{u}_f)) \\
&\quad - \operatorname{div} (J \boldsymbol{\sigma}_f(\mathbf{v}_f, p_f) \mathbf{F}_f^{-T}) = \rho_f \mathbf{f}_f \quad \text{in } I \times \mathcal{F} \\
\partial_t \mathbf{u}_s &= \mathbf{v}_s, \quad \rho_s \partial_t \mathbf{v}_s - \operatorname{div} (\mathbf{F}_s \boldsymbol{\Sigma}_s(\mathbf{u}_s)) = \rho_s \mathbf{f}_s \quad \text{in } I \times \mathcal{S} \\
\mathbf{v}_f &= \mathbf{v}_s, \quad J_f \sigma_f(\mathbf{v}_f, p_f) \mathbf{F}_f^{-T} \vec{n}_f + \mathbf{F}_s \boldsymbol{\Sigma}_s(\mathbf{u}_s) \vec{n}_s = 0 \quad \text{on } I \times \mathcal{I}, \\
\mathbf{v}_f &= \mathbf{v}_f^D \text{ on } I \times \Gamma_f^D(t), \quad \rho_f \nu_f J_f \nabla \mathbf{v}_f \mathbf{F}_f^{-1} \mathbf{F}_f^{-T} \vec{n}_f - J_f \mathbf{F}_f^{-T} p_f \vec{n}_f = 0 \quad \text{on } I \times \Gamma_f^{\text{out}}(t) \\
&\quad \mathbf{u}_s = \mathbf{u}_s^D \quad \text{on } I \times \Gamma_s^D \\
\mathbf{v}_f &= \mathbf{v}_f^0 \text{ on } \{0\} \times \mathcal{F}(0), \quad \mathbf{v}_s = \mathbf{v}_s^0, \quad \mathbf{u}_s = \mathbf{u}_s^0 \quad \text{on } \{0\} \times \mathcal{S}.
\end{aligned} \tag{2.4}$$

Note that $\vec{n}_f$ and $\vec{n}_s$ are the outward facing normals in the reference framework such that the fluid stresses are given in terms of the Piola transform. The fluid stress tensor in ALE formulation reads

$$\boldsymbol{\sigma}_f(\mathbf{v}_f, p_f) = \rho_f \nu_f (\nabla \mathbf{v}_f \mathbf{F}_f^{-1} + \mathbf{F}_f^{-T} \nabla \mathbf{v}_f^T) - p_f I, \quad \hat{\boldsymbol{\Sigma}}_s = 2\mu_s \mathbf{E}_s + \lambda_s \operatorname{tr}(\mathbf{E}_s) I, \tag{2.5}$$

In (2.4) we have split the hyperbolic solid problem into a system of first order (in time) equations by introducing the solid velocity $\mathbf{v}_s$ that – on the interface – matches the fluid velocity.

2.2. Variational formulation and finite element discretization

To prepare for a discretization with finite elements we briefly sketch the variational formulation of the fluid–structure interaction problem (2.4) that also embeds the coupling conditions in a variational sense. Kinematic and geometric coupling conditions are taken care of by choosing global function spaces for the velocity and deformation, *i.e.*

$$\mathbf{v} \in \mathbf{v}^D + H_0^1(\Omega; \Omega^D)^d, \quad \mathbf{u} \in \mathbf{u}^D + H_0^1(\Omega; \Omega^D)^d,$$

where $d \in \{2, 3\}$ is the spatial dimension, $\Omega = \mathcal{F} \cup \mathcal{I} \cup \mathcal{S}$ and $\Omega^D = \Gamma_f^D \cup \Gamma_s^D$ the combined Dirichlet boundary and $\mathbf{u}^D, \mathbf{v}^D$ are extensions of the Dirichlet data into the domain. Solid and fluid velocity (and deformation) are defined as the restrictions of $\mathbf{v}$ and $\mathbf{u}$ to the respective domain. The dynamic condition is realized by testing the momentum equations for both subproblems by one common and continuous (in the H^1 -sense) test functions $\phi \in H_0^1(\Omega; \Omega^D)^d$ and summing up both equations.

$$\begin{aligned}
& (J \mathbf{F}^{-1} : \nabla \mathbf{v}^T, \xi)_{\mathcal{F}} + (\rho_f J (\partial_t \mathbf{v} + \nabla \mathbf{v} \mathbf{F}^{-1} (\mathbf{v} - \partial_t \mathbf{u})), \phi)_{\mathcal{F}} \\
& + (J \boldsymbol{\sigma}_f(\mathbf{v}, p) \mathbf{F}^{-T}, \nabla \phi)_{\mathcal{F}} - \langle \rho_f \nu_f J \mathbf{F}^{-T} \nabla \mathbf{v}^T \mathbf{F}^{-T} \vec{n}_f, \phi \rangle_{\Gamma_f^{\text{out}}} \\
& + (\rho_s \partial_t \mathbf{v}, \phi)_{\mathcal{S}} + (\mathbf{F} \boldsymbol{\Sigma}_s(\mathbf{u}), \nabla \phi)_{\mathcal{S}} = (\rho_f \mathbf{f}_f, \phi)_{\mathcal{F}} + (\rho_s \mathbf{f}_s, \phi)_{\mathcal{S}} \\
& (\partial_t \mathbf{u} - \mathbf{v}, \psi_s)_{\mathcal{S}} + (\nabla \mathbf{u}, \nabla \psi_f)_{\mathcal{F}} = 0,
\end{aligned} \tag{2.6}$$

for all

$$\phi \in H_0^1(\Omega; \Omega^D)^d, \quad \xi \in L^2(\mathcal{F}), \quad \psi_s \in L^2(\mathcal{S})^d, \quad \psi_f \in H_0^1(\mathcal{F}). \tag{2.7}$$

By $\langle \cdot, \cdot \rangle_\Gamma$ we denote the L^2 -inner product on a boundary segment Γ . The boundary term is introduced to correct the normal stresses to comply with the do-nothing outflow condition [25]. Dirichlet boundary values are embedded in the trial spaces, the initial conditions will be realized in the time stepping schemes.

Next, let Ω_h be a triangulation (or more general a mesh) of the domain Ω satisfying the usual conditions of structure-, shape- and form-regularity. In addition we assume that the mesh Ω_h matches the interface $\mathcal{I}$ in the following sense: for each element $K \in \Omega_h$ it holds $K \cap \mathcal{I} = \emptyset$. We refer to Section 5.3.1 of [39] for details on the construction of such meshes. Details on the specific implementation in the finite element library *Gascoigne 3D* [9] are given in Section 4.2 of [39]. The assumption of matching meshes is important to guarantee good approximation properties, as fluid-structure interactions are *interface problems*, where the solution has limited regularity across the interface and non-fitted meshes give rise to a breakdown in accuracy [19, 42].

The finite element discretization of (2.6) is straightforward. We choose continuous polynomial spaces assembled on the mesh Ω_h as

$$V_h^{(r)} := \{\phi \in C(\bar{\Omega}) \mid \phi|_T \circ \hat{T}_T^{-1} \in Q^{(r)}\},$$

where $Q^{(r)}$ is the space of bi- or tri-polynomial of degree r on quadrilateral or hexahedral meshes and where the *iso-parametric* reference map $\hat{T}_T \in [Q^{(r)}]^d$ comes from this same space. The *iso-parametric* setup is able to give optimal order approximations on domains with curved boundaries, see [32] or Section 4.2.3 of [39]. Throughout this paper we choose piecewise quadratic discrete trial-spaces for the discretization of velocity, deformation and pressure in (2.6) and (2.7)

$$\mathbf{v}_h \in \mathbf{v}_h^D + \mathbf{V}_h \quad (\mathbf{V}_h := [V_h^{(2)}]^d), \quad \mathbf{u}_h \in \mathbf{u}_h^D + \mathbf{V}_h, \quad p_h \in Q_h := V_h^{(2)}(\mathcal{F}),$$

and also piecewise quadratic test-spaces for all equations

$$\phi_h \in \mathbf{V}_h, \quad \xi_h \in Q_h, \quad \psi_{s,h} \in [V_h^{(2)}(\mathcal{S})]^2, \quad \psi_{f,h} \in [V_h^{(2)}(\mathcal{F})]^2,$$

with the necessary modifications to implement Dirichlet values.

As the equal order finite element pair $\mathbf{V}_h \times Q_h$ does not fulfil the inf-sup condition we add additional stabilization terms of local projection [8] type

$$\sum_{T \in \Omega_h \cap \mathcal{F}} \delta_T (\nabla(p_h - i_h^{(1)} p_h), \nabla(\xi_h - i_h^{(1)} \xi_h))_{\mathcal{F}},$$

where $i_h^{(1)} : V_h^{(2)} \rightarrow V_h^{(1)}$ is the interpolation into the space of piecewise linear finite elements and where $\delta_T = 0.1(\nu_f h_T^{-2} + |\mathbf{v}_f|_{\infty, T} h_T^{-1})^{-1}$ is the local stabilization parameter. Required small modifications in terms of the ALE formulation in fluid-structure interactions are discussed in [35] or Section 5.3.3 of [39]. For simplicity we assume that no mechanisms for stabilizing dominant transport are required.

2.3. Time discretization

For temporal discretization in the time interval $I = [0, T]$, we start by a splitting into discrete time-steps

$$0 = t_0 < t_1 < \dots < t_N = T, \quad k := t_n - t_{n-1}.$$

To simplify notation we assume that this distribution is uniform with k being the same in all time steps $t_{n-1} \mapsto t_n$. Modifications are however straightforward. At time t_n we denote by $\mathbf{v}_n = \mathbf{v}_h(t_n)$, $\mathbf{u}_n = \mathbf{u}_h(t_n)$, $p_n = p_h(t_n)$ the discrete approximations and by $\mathbf{F}_n := I + \nabla \mathbf{u}_n$, $J_n := \det \mathbf{F}_n$, the deformation gradient and

its determinant. By the index $n - \frac{1}{2}$ we denote the mean values on $I_n = [t_{n-1}, t_n]$, *e.g.* $\mathbf{v}_n := (\mathbf{v}_{n-1} + \mathbf{v}_n)/2$ or $J_n := (J_{n-1} + J_n)/2$.

As a compromise between accuracy (second order), very good stability properties (globally A-stable), simplicity and efficiency (simple one-step scheme) we consider an implicitly shifted version of the Crank-Nicolson method [34, 36] and Section 5.1.2 of [39].

Applied to the finite element approximation to (2.6), each time step $t_{n-1} \mapsto t_n$ of the fully discrete problem reads:

$$\begin{aligned} & (\rho_f J_{n-\frac{1}{2}} ((\mathbf{v}_n - \mathbf{v}_{n-1}) - \nabla \mathbf{v}_{n-\frac{1}{2}} \mathbf{F}_{n-\frac{1}{2}}^{-1} (\mathbf{u}_n - \mathbf{u}_{n-1})), \phi)_{\mathcal{F}} \\ & + (\rho_s (\mathbf{v}_n - \mathbf{v}_{n-1}), \phi)_{\mathcal{S}} + (\mathbf{u}_n - \mathbf{u}_{n-1}, \psi_s)_{\mathcal{S}} \\ & + k A_{div}(\mathbf{v}^n, \mathbf{u}^n)(\xi) + k \theta A_{NS}(\mathbf{v}^n, \mathbf{u}^n, p^n)(\phi) + k(1 - \theta) A_{NS}(\mathbf{v}^{n-1}, \mathbf{u}^{n-1}, p^n)(\phi) \\ & + k \theta A_{ES}(\mathbf{v}^n, \mathbf{u}^n)(\phi) + k(1 - \theta) A_{ES}(\mathbf{v}^{n-1}, \mathbf{u}^{n-1})(\phi) \\ & - k \theta (\mathbf{v}_n, \psi_s)_{\mathcal{S}} - k(1 - \theta) (\mathbf{v}_n, \psi_s)_{\mathcal{S}} + k \theta (\nabla \mathbf{u}_n, \nabla \psi_f)_{\mathcal{F}} + k(1 - \theta) (\nabla \mathbf{u}_{n-1}, \nabla \psi_f)_{\mathcal{F}} = F(\phi) \end{aligned} \quad (2.8)$$

for all $\phi \in \mathbf{V}_h, \xi \in Q_h, \psi_s \in \mathbf{L}_h$ and $\psi_f \in \mathbf{W}_h$, where for simplicity of notation we introduced the notation

$$\begin{aligned} A_{div}(\mathbf{v}, \mathbf{u})(\xi) &= (J \mathbf{F}^{-1} : \nabla \mathbf{v}^T, \xi)_{\mathcal{F}} \\ A_{NS}(\mathbf{v}, \mathbf{u}, p)(\phi) &= (\rho_f J \nabla \mathbf{v} \mathbf{F}^{-1} \mathbf{v}, \phi)_{\mathcal{F}} + (J \boldsymbol{\sigma}_f(\mathbf{v}, p) \mathbf{F}^{-T}, \nabla \phi)_{\mathcal{F}} - \langle \rho_f \nu_f J \mathbf{F}^{-T} \nabla \mathbf{v}^T \mathbf{F}^{-T} \vec{n}_f, \phi \rangle_{\Gamma_f^{out}} \\ A_{ES}(\mathbf{v}, \mathbf{u})(\phi) &= (\mathbf{F} \boldsymbol{\Sigma}_s(\mathbf{u}), \nabla \phi)_{\mathcal{S}} \\ F(\phi) &= k(\rho_f (\theta \mathbf{f}_f(t_n) + (1 - \theta) \mathbf{f}_f(t_{n-1})), \phi)_{\mathcal{F}} + k(\rho_s (\theta \mathbf{f}_s(t_n) + (1 - \theta) \mathbf{f}_s(t_{n-1})), \phi)_{\mathcal{S}}. \end{aligned} \quad (2.9)$$

For $\theta = \frac{1}{2}$ this is the typical Crank-Nicolson scheme, $\theta = 1$ would give the backward Euler method of first order. We usually consider $\theta = \frac{1}{2} + \theta_0 k$ with a small fixed parameter θ_0 , which results in a second order accurate scheme, see [34, 37] and [41] for a numerical study applied to fluid-structure interactions. A proper combination of three substeps with different choices of θ would give the fractional step theta method which is of second order and strongly A-stable, see [49] or [15] for an application to fluid-structure interactions. The discretization of the nonlinear terms including time derivatives (*e.g.* coming from the domain convection) is still discussed in literature. However, different choices give similar stability and accuracy results, see [41] and Section 5.1.2 of [39].

3. PARALLEL TIME-STEPPING

The motivation for increased parallelism in numerical algorithms arises from the architecture of modern computers with an ever-increasing number of processing units. Most numerical solvers for differential equations are formulated as sequential algorithms, which denies exploitation of parallelism.

The Parareal algorithm, which is probably the best known parallel time stepping method, promises to bypass this problem by basically providing a wrapper around common sequential algorithms. Although it is known in its fundamentals since 2001, it hasn't been widely applied to fluid structure interactions.

The Parareal algorithm can be derived in multiple ways, see [20] for an overview. The first approach introduced in [33], is based on a predictor corrector scheme. First the time domain I is divided into L subintervals of equal length. Predictions with coarse timesteps h_C are used to provide initial values for parallel computations with fine timesteps $h_C \ll h_F$. These results are then used together with old coarse predictions for correcting the solution, providing new initial values. Timesteps are chosen such that $\frac{T}{L} = m h_F = M h_C$. The algorithm is defined by using two different propagators for the solution over one subinterval. While C is the propagator using the coarse time step size, F is the fine counterpart.

With these definitions and the remarks from the last chapter in mind, the Parareal-algorithm for fluid-structure-interactions is defined by a simple recursive formula

$$\begin{aligned} (\mathbf{v}, \mathbf{u}, p)_i^0 &= (\mathbf{v}^0, \mathbf{u}^0) \\ (\mathbf{v}, \mathbf{u}, p)_{i+1}^{l+1} &= \underbrace{C((\mathbf{u}, \mathbf{v})_{i+1}^l, t_{l+1}, t_l)}_{\text{predictor}} + \underbrace{F((\mathbf{u}, \mathbf{v})_i^l, t_{l+1}, t_l) - C((\mathbf{u}, \mathbf{v})_i^l, t_{l+1}, t_l)}_{\text{corrector}}, \end{aligned} \quad (3.1)$$

where $l \in \{0, \dots, L-1\}$ is the index of the subinterval and $i \in \mathbb{N}$ is the iteration count. While the predictor part is sequential (particularly the first iteration for initialization), the fine propagations on each subinterval can be parallelized. Due to the sequential part the parallel speedup will never be optimal, but by reducing the computational effort for the predictor the sequential part can be minimized. However to achieve fast convergence of the algorithm, the predictors accuracy shouldn't differ too much from the fine propagators accuracy. This criterion is hard to quantify since there is no way to predict the optimal corrector for a given fine propagator. In Figure 1 a shared memory implementation with OpenMP is sketched, following the distributed task scheduling from [2]. This results in a (theoretically) nearly optimal runtime since the sequential part of the algorithm is minimized. The theoretically achievable speedup with $r = \frac{h_F}{h_C}$ is

$$S = \frac{1}{r + \frac{K}{N}(1+r)} \quad (3.2)$$

where h_F and h_C are the timestep lengths of the fine and coarse propagator, respectively. The algorithm requires very little communication, so there is little overhead and the estimate holds well in practice. The main bottleneck are the sequential computations which can be minimized by optimal task scheduling and reducing the ratio r . In (3.2) the first term in the denominator corresponds to the initialization phase with the corrector and $\frac{K}{N}r$ corresponds to the sequential part of each further iteration.

From (3.2), we conclude that r and K should be minimized. In the impossible scenario that $r = 0$ and $K = 1$ we would get optimal speedup. Upon reducing r , the algorithm will require more iterations to converge, reducing K requires a more accurate predictor, resulting in a higher r . Since r and K interact this way, optimizing both at the same time is the main difficulty in applying the parareal-method effectively. As we will see in our examples, we cannot rely on the fact that r is actually the ratio between the computational costs of the coarse and fine propagator, but it still provides a useful guidance.

To our knowledge, there is no previous application of the Parareal idea to fluid-structure interactions. But from the experience with the Navier-Stokes equations we must expect to meet several problems: The nonlinearity of the will have an impact on the ratio between the effort on fine and coarse problem f , since larger time-steps will make the effect of the nonlinearity more severe. Considering higher Reynolds numbers a deterioration of the Parareal convergence rate is reported [46]. This is attributed to dominant imaginary Eigenvalues. The spectrum of the coupled fluid-structure interaction problem including the hyperbolic structure problem is even more diverse and less understood. Nevertheless, the numerical results presented below will show speedups up to about 4 (for a 2d example using 20 cores), which is comparable to a study on the Navier-Stokes equations [47] (showing a speedup up to 4.1 on 15 cores).

Recently a weighted version of the Parareal scheme, the θ -Parareal scheme has been developed [1], introducing weights for the contributions of the coarse propagators. To avoid confusion with the unrelated θ -timestepping scheme, we name the weights Λ :

$$\begin{aligned} (\mathbf{v}, \mathbf{u}, p)_i^0 &= (\mathbf{v}^0, \mathbf{u}^0) \\ (\mathbf{v}, \mathbf{u}, p)_{i+1}^{l+1} &= \Lambda_{i+1}^{l+1} C((\mathbf{u}, \mathbf{v})_{i+1}^l, t_{l+1}, t_l) + F((\mathbf{u}, \mathbf{v})_i^l, t_{l+1}, t_l) - \Lambda_{i+1}^{l+1} C((\mathbf{u}, \mathbf{v})_i^l, t_{l+1}, t_l). \end{aligned} \quad (3.3)$$

```

#pragma omp parallel
{
    #pragma omp ordered nowait schedule(static)
     $(\mathbf{u}, \mathbf{v})_0^F = (\mathbf{u}, \mathbf{v})_0$ 
    for m = 0 to N-1
        #pragma omp ordered
        {
            omp_set_lock(&interval_locker[i])
             $(\mathbf{u}, \mathbf{v})_m^C = C((\mathbf{u}, \mathbf{v})_m, t_{m+1}, t_m)$ 
             $(\mathbf{u}, \mathbf{v})_m^{\text{final}} = (\mathbf{u}, \mathbf{v})_{m+1}^C$ 
             $(\mathbf{u}, \mathbf{v})_{m+1}^F = (\mathbf{u}, \mathbf{v})_{m+1}^C$ 
            omp_unset_lock(&interval_locker[i])
        }
    for k = 1 to K
        #pragma omp for nowait schedule(static)
        for m = 0 to N - k + 1
            omp_set_lock(&interval_locker[m])
             $(\mathbf{u}, \mathbf{v})_{m+k}^F = F((\mathbf{u}, \mathbf{v})_{m+1}^F, t_{m+k}, t_{m+k-1})$ 
            omp_unset_lock(&interval_locker[m])
     $(\mathbf{u}, \mathbf{v})_0^{\text{final}} = (\mathbf{u}, \mathbf{v})_0^F$ 
    #pragma omp for ordered nowait schedule(static)
    for m = 0 to N - k
        #pragma omp ordered
        {
            omp_set_lock(&interval_locker[m])
             $(\mathbf{u}, \mathbf{v})_{m+k-1}^C = (\mathbf{u}, \mathbf{v})_{m+k-1}^{\text{final}}$ 
            omp_unset_lock(&interval_locker[m])
             $(\mathbf{u}, \mathbf{v})_{m+k-1}^C = C((\mathbf{u}, \mathbf{v})_{m+k-1}^C, t_{l+1}, t_l)$ 
            omp_set_lock(&interval_locker[m + 1])
             $(\mathbf{u}, \mathbf{v})_{m+k}^{\text{final}} = (\mathbf{u}, \mathbf{v})_{m+k-1}^C + (\mathbf{u}, \mathbf{v})_{m+k}^F - (\mathbf{u}, \mathbf{v})_{m+k}^C$ 
             $(\mathbf{u}, \mathbf{v})_{m+k}^F = (\mathbf{u}, \mathbf{v})_{m+k}^{\text{final}}$ 
            omp_unset_lock(&interval_locker[m + 1])
        }
    }
}

```

FIGURE 1. Pseudo-code for Parareal algorithm with OpenMP directives, following the distributed task scheduling.

The idea here was to replace the predictor term by

$$\Lambda_{i+1}^{l+1} C((\mathbf{u}, \mathbf{v})_{i+1}^l, t_{l+1}, t_l) + (1 - \Lambda_{i+1}^{l+1}) C((\mathbf{u}, \mathbf{v})_i^l, t_{l+1}, t_l),$$

yielding the formula above. One idea for obtaining Λ_i^l is to minimize the discrepancy of fine and coarse solution:

$$\Lambda_{i+1}^l = \arg_{\Lambda \in \mathbb{R}} \min \|F((\mathbf{u}, \mathbf{v})_i^l, t_{l+1}, t_l) - \Lambda C((\mathbf{u}, \mathbf{v})_{i+1}^l, t_{l+1}, t_l)\|.$$

In which case Λ_i^l can be computed as $\frac{\langle F, C \rangle}{\langle C, C \rangle}$ assuming the same arguments as above. We will use $\frac{\langle F, C \rangle}{\langle C, C \rangle \langle F, F \rangle}$, which is not a least squares solution, but can be interpreted as angle penalization. The factors are computed for each component separately and averaged to obtain a single scaling factor. This way we make sure that $\Lambda_i^l \leq 1$ by construction. Later on we will shortly investigate the different approaches. In Section 4 the speedup and the convergence of Parareal will be investigated.

4. NUMERICAL EXAMPLES

In this section we will apply the Parareal algorithm to two FSI problems. Further we discuss various issues that we were facing when applying the Parareal algorithm to different configurations. Based on these examples we will measure the error introduced by the Parareal algorithm as well as the speedup and efficiency. Measurements are taken on a two socket machine with two Intel Xeon E5-2640 v4 each having 10 cores. The number of subintervals in the Parareal algorithm are chosen as the total number of cores, 20. Other discretization parameters are depicted in Table 2

Figures 2 and 3 show the geometry of the test cases in the 2d and 3d configuration. Both test cases assemble the flow around a wall-mounted elastic obstacle that will undergo a deformation. The problem is driven by a parabolic (bi-parabolic in 3d) inflow profile on Γ_{in} which is oscillating in time

$$\begin{aligned} \mathbf{v}_{in}^{2d}(t, y) &= s(t) \frac{y(H-y)}{(H/2)^2} \frac{3}{2} \bar{\mathbf{v}}_{2d}, \\ \mathbf{v}_{in}^{3d}(t, y, z) &= s(t) \frac{y(H-y)(H^2 - z^2)}{(H/2)^2 H^2} \frac{9}{8} \bar{\mathbf{v}}_{3d}, \quad s(t) = \frac{1}{2}(1 - \cos(\pi t)). \end{aligned}$$

By $H_{3d} = 0.4$ and $H_{2d} = 1$ we denote the diameter of the flow domain and by $\bar{\mathbf{v}}_{2d}$ and $\bar{\mathbf{v}}_{3d}$ the average flow rates. The scalar function $s(t)$ oscillates with the period $T = 1$ s. In both cases the time interval is $I = [0, 8]$. All further parameters of this benchmark problem as shown in Table 1. The configurations yield the maximum Reynolds number $Re_{2d} = 30$ in 2d and $Re_{3d} = 20$ in 3d, where $L_{2d} = 0.5$ and $L_{3d} = 0.2$, the height of the obstacles is chosen as characteristic length scale. On the wall boundary Γ_{wall} we prescribe homogeneous Dirichlet conditions $\mathbf{v} = 0$ and on the outflow boundary Γ_{out} the do-nothing condition, which, in ALE formulation, looks like

$$J(\rho_f \nu_f \mathbf{F}^{-T} \nabla \mathbf{v}_f - p_f) \mathbf{F}^{-T} \tilde{\mathbf{n}}_f = 0 \quad (4.1)$$

FIGURE 2. 2d configuration. Dirichlet inflow on Γ_{in} , do-nothing outflow condition on Γ_{out} and no-slip Dirichlet condition on Γ_{wall} . By $\mathcal{I}$ we denote the fluid-structure interface.

FIGURE 3. 3d configuration. Like in 2d with additional symmetry boundary Γ_{sym} , where $\mathbf{v} \cdot \vec{n} = 0$ is prescribed.

TABLE 1. Configuration of the test problems in two and three dimensions. We indicate the parameters defining the different cases.

Problem configuration	2d	3d
Fluid density	10^3 kg m^{-3}	10^3 kg m^{-3}
Kinematic viscosity	$2 \times 10^{-2} \text{ m}^2 \text{ s}^{-1}$	$10^{-2} \text{ m}^2 \text{ s}^{-1}$
Average inflow velocity	$1.2 \text{ m} \cdot \text{s}^{-1}$	$1.0 \text{ m} \cdot \text{s}^{-1}$
Solid density	10^3 kg m^{-3}	10^3 kg m^{-3}
Shear modulus	$1 \times 10^6 \text{ kg m}^{-1} \text{ s}^{-2}$	$5 \times 10^5 \text{ kg m}^{-1} \text{ s}^{-2}$
Poisson ratio	$\nu = 0.4$	$\nu = 0.4$

TABLE 2. Discretization parameters of the test problems in two and three dimensions. The coarse timesteps are the same for 2d and 3d.

Problem configuration	2d	3d
unknowns	5 360	93 925
coarse timestep h_C	0.01 0.02	0.05 0.1
fine timestep h_F	0.001	0.001

compare (2.6), where we subtract the transposed part of the gradient such that (4.1) remains. In 3d, we split the domain and introduce a symmetry boundary, where we prescribe a free-slip, no-penetration condition $\mathbf{v} \cdot \vec{n} = 0$ and $\boldsymbol{\sigma} \vec{n} \cdot \vec{t} = 0$, where $\vec{t}$ are the tangential vectors.

4.1. 2d example

Figure 4 shows the solutions of the problems at time $t_{17} = 6.8$, $t_{18} = 7.2$, $t_{19} = 7.6$, $t_{20} = 8$. The dynamics of this problem is enforced by the oscillating right hand side that causes a periodic motion of the elastic obstacle.

FIGURE 4. Solutions on the last 4 subintervals (from left to right), the deformation is magnified by a factor of 2.

TABLE 3. Parareal error (2d example). We show the maximum Parareal error over all 20 subintervals. In addition we indicate the discretization error using the fine step size $h_F = 0.001$ on a mesh with 5 360 unknowns and mark the first step, where the maximum Parareal error is below the maximum discretization error.

k	1	2	3	4	5	Discretization error
$h_C = 0.01$	8.3×10^{-5}	1.3×10^{-5}	2.9×10^{-6}	1.1×10^{-6}	2.4×10^{-7}	8.5×10^{-5}
$h_C = 0.02$	3.9×10^{-4}	8.7×10^{-5}	1.1×10^{-5}	1.3×10^{-6}	1.3×10^{-6}	8.5×10^{-5}
$h_C = 0.05$	5.2×10^{-4}	3.5×10^{-5}	2.1×10^{-5}	8.8×10^{-6}	1.9×10^{-6}	8.5×10^{-5}
$h_C = 0.10$	4.1×10^{-3}	3.0×10^{-4}	1.1×10^{-4}	9.9×10^{-6}	3.0×10^{-6}	8.5×10^{-5}

The problem comprises a large deformation such that the nonlinearity of the ALE mapping and the structure model plays an important part in the complexity of this test case.

In Figure 5 we show the convergence behavior of the Parareal method for four choices of the large time step h_C . For each iteration k of the Parareal method we plot the relative error between the Parareal approximation and a sequential solution (using the same small step size $h_F = 0.001$). The time interval is split into 20 subintervals and we clearly recognize that after the k -th iteration, the solution on the k -th interval does not get any better. All plots also include a dashed line (this is exactly the same line in every plot) which indicates the discretization error, *e.g.* the error between the sequential solution with stepsize k and a refined reference solution. If the Parareal error is below this line, the Parareal approximation can be considered sufficiently accurate, *e.g.* considering $h_C = 0.01$, two iterations $k = 2$ already yield a Parareal approximation error that is smaller than the discretization error.

Table 3 shows the maximum (taken over all 20 subintervals) Parareal approximation for the first five Parareal iterations. We also indicate the discretization error for a sequential iteration using the time step size h_F , which could serve as stopping criteria for the Parareal iteration. Parareal approximation errors far below the discretization error cannot reduce the overall error such that the simulation should optimally be stopped, once the Parareal error is below this threshold.

To test the robustness of the Parareal algorithm with respect to the discretization we indicate further results in Table 4. These show that the convergence of the Parareal method is not affected by the choice of the discretization, neither the fine time step h_F nor the spatial mesh resolution has a significant impact on the convergence rate.

Furthermore, in Figure 6 we give an overview of the speedup that is obtained in comparison to the sequential simulation. In this example the optimal choice would be $h_C = 0.05$, stopping after 2 iterations such that the discretization error is dominant and resulting in a speedup of 4.44.

(a) Coarse stepsize $h_C = 0.01$ (b) Coarse stepsize $h_C = 0.02$ (c) Coarse stepsize $h_C = 0.05$ (d) Coarse stepsize $h_C = 0.1$

FIGURE 5. Velocity error between sequential (computed with the fine stepsize k) and Parareal solution with different coarse step sizes calculated at the interval boundaries as the relative error in the Frobenius norm $\frac{\|v_p - v_s\|_F}{\|v_s\|_F}$. Where v_s is the sequential solution and v_p the Parareal solution. The dashed line is the difference between the sequential solution and a reference solution. All Parareal approximations below this line can be considered sufficiently accurate.

That is below the theoretical value of 5.78 estimated by from (3.2). The sub optimal performance is mostly due to the strong nonlinearity of the fluid–structure interaction problem. The nonlinear problems are approximated with a Newton’s method that benefits from small time-steps as they limit the role of the ALE map. Using coarser time steps for the prediction (that should theoretically improve the speedup) we require more Newton iterations such that the efficiency is reduced. The theoretical predictions can only be reached when the computational costs of each time step are exactly the same, which we cannot guarantee for a nonlinear problem.

TABLE 4. Parareal error (2d example) for fixed coarse time step $h_C = 0.05$ and variations of the fine time step h_F on two different spatial discretizations. We show the maximum Parareal error over all 20 subintervals. The tests show that the Parareal convergence is basically independent from the spatial discretization.

h_F	h_C	unknowns	1	2	3	Discretization error
0.001	0.05	5 360	5.2×10^{-4}	3.5×10^{-5}	2.1×10^{-5}	8.5×10^{-5}
0.001	0.05	21 285	5.5×10^{-4}	3.3×10^{-5}	1.2×10^{-5}	3.9×10^{-5}
0.002	0.05	21 285	5.6×10^{-4}	3.3×10^{-5}	1.4×10^{-5}	1.6×10^{-4}

FIGURE 6. 2d. Speedup of the Parareal algorithm on 20 cores. We show the speedup for the first 8 Parareal iterations. We mark the iteration, where the Parareal error is of the same order as the discretization error. On the right we show the raw data with all speedups.

4.2. fsi-3 benchmark problem. A case where the algorithm fails

In the first experiments, we considered the *fsi-3 benchmark* problem as introduced by Hron and Turek [28]. In this case Parareal did not converge. Instead we observed pressure oscillations as shown in Figure 7. One possible explanation for these instabilities that are also known from the simulation of incompressible flow problems on moving meshes [10] is the violation of the weak divergence freeness after the Parareal update $(\mathbf{u}, \mathbf{v})_{m+k}^{\text{final}} = (\mathbf{u}, \mathbf{v})_{m+k-1}^C + (\mathbf{u}, \mathbf{v})_{m+k}^F - (\mathbf{u}, \mathbf{v})_{m+k}^C$, as the nonlinear ALE formulation $(J\mathbf{F}^{-1} : \nabla \mathbf{v}^T, \xi)_{\mathcal{F}}$ with separate updates in velocity and deformation does not conserve weak incompressibility and gives rise to pressure oscillations

$$\begin{aligned}
 (\det(I + \nabla \mathbf{u}_i)(I + \nabla \mathbf{u}_i)^{-1} : \nabla \mathbf{v}_i^T, \xi)_{\mathcal{F}} &= 0 \text{ for } i = 1 \text{ and } i = 2 \\
 &\not\Rightarrow (\det(I + \nabla(\mathbf{u}_1 + \mathbf{u}_2))(I + \nabla(\mathbf{u}_1 + \mathbf{u}_2))^{-1} : \nabla(\mathbf{v}_1 + \mathbf{v}_2)^T, \xi)_{\mathcal{F}} = 0.
 \end{aligned}$$

Different approaches to cure this problem, *e.g.* applying a Stokes projection to the initial solutions on the subintervals in every time step, see [10], did not sufficiently reduce these oscillations.

Another and more likely reason for the cause of these instabilities is the dependency of the damping property of the shifted Crank-Nicolson time-stepping scheme on both the spatial and temporal discretization parameters. Considering the Navier-Stokes flow around a rigid body, the frequency of the vortex shedding depends on the time step size. One observes higher frequencies for smaller time step sizes. In Figure 7 of [22] we have documented a similar dependency of the frequency on the spatial discretization. The Parareal update consists of combining solutions from two different time meshes which, due to the different damping, carry an offset. This leads to a systematic error at each macro step that cannot be reduced in further iterations. It would be a remedy to use a

FIGURE 7. fsi-3 benchmark problem ([28]). Drag coefficient from the Parareal method after 2 iterations in comparison to the sequential solution at time step size $h_F = 10^{-3}$. The coarse stepsize was chosen as $h_C = 10^{-2}$. Oscillations occur after each Parareal update.

time stepping scheme whose damping properties do not depend on the time step size, such as the Crank-Nicolson scheme for $\theta = \frac{1}{2}$. This however does not have sufficient stability for fluid-structure interactions, see [41, Fig. 4]. We repeated the fsi-3 test case with the BDF2 time stepping scheme, which however showed the same oscillations. In [43], Ruprecht suggests modifying the update in the parareal algorithm, such that information about the dissipation properties is taken into account.

The fsi-3 benchmark problem is driven by the inner dynamics of the system and the oscillatory motion of the solid is induced by the von Kármán vortex street arising from the flow around the obstacle. In this respect, the example discussed in Section 4.1 is simpler, since it is driven by an oscillating right hand side. For such problems we could not construct a convergent parareal method that simultaneously provides parallel acceleration. We think that the problems stem from a combination of different aspects of the interaction between the incompressible Navier-Stokes equations with an elastic structure and its discretization, and are not easily remedied.

4.3. 3d example

Next we consider the 3d configuration shown in Figure 3. This test case corresponds to the 2d case discussed in Section 4.1 and since the dynamics is driven by the oscillatory inflow profile, it should be better suited for the Parareal algorithm than the previously discussed fsi-3 problem.

Figure 8 shows the solution of the 3d problem $t_{17} = 6.8$, $t_{18} = 7.2$, $t_{19} = 7.6$, $t_{20} = 8$, analogous to the 2d case. The test case leads to a substantial displacement of the solid. Further, the Reynolds number is sufficiently high such that a complex vortex structure appears. The computational effort for the solution of such 3d fluid-structure interaction problems is very high, which is due to the size of the coupled problem (7 coupled unknowns describing pressure, velocity field and deformation field), but also due to the stiffness introduced by the coupling of two problems of different type (parabolic fluid problem vs. hyperbolic solid problem). The test case in this paper has been kept small with less than 100 000 unknowns, compare Table 2, such that a direct solver could be used for all linear problems. If finer meshes are considered, more elaborate approaches are required. Parallel solvers of Newton-Krylov type with multigrid preconditioning have shown to be most effective [3, 16, 38].

In Figure 9 we show the relative errors of the Parareal approximation in comparison to a sequential simulation for each iteration. Again, a dashed line represents the discretization error of the problem by comparison to a refined reference solution. We observe a similar behavior like in the 2d case: The Parareal error is quickly below

FIGURE 8. Solutions on the last 4 subintervals. The deformation is magnified by a factor of 10.

TABLE 5. Parareal error (3d example). We show the maximum Parareal error over all 20 subintervals. In addition we indicate the discretization error using the fine step size h_F and mark the first step, where the maximum Parareal error is below the maximum discretization error.

k	1	2	3	4	5	Discretization error
$h_C = 0.01$	1.6×10^{-5}	2.3×10^{-6}	1.4×10^{-6}	4.2×10^{-7}	2.5×10^{-8}	1.2×10^{-5}
$h_C = 0.02$	5.3×10^{-5}	5.4×10^{-6}	6.6×10^{-7}	1.4×10^{-7}	4.2×10^{-8}	1.2×10^{-5}
$h_C = 0.05$	3.8×10^{-4}	1.7×10^{-5}	4.7×10^{-6}	1.9×10^{-6}	8.1×10^{-7}	1.2×10^{-5}
$h_C = 0.10$	8.8×10^{-4}	7.9×10^{-5}	6.7×10^{-5}	4.4×10^{-6}	1.2×10^{-6}	1.2×10^{-5}

the level of the discretization error and initial intervals do not benefit from further steps. Table 5 collects these findings and here we indicate the maximum Parareal approximation error over all 20 subintervals. In addition we indicate the level of the discretization error. Figure 10 showing the speedups indicates a substantially reduced performance of the Parareal scheme when applied to this 3d test case. The main cause for this deterioration is the linearization of the problem: Newton's method takes longer to converge for big timesteps and the corrections in the Parareal algorithm only seem to aggravate this problem. Although the instabilities noted in Section 4.2 could not be observed here, they start to rise: Larger timesteps influence the dynamics and if they get too large, the problems are not feasible in a stable manner anymore. By choosing the timesteps carefully a speedup of 2.18 can still be obtained.

5. CONCLUSION

We applied the Parareal algorithm to fluid-structure interaction problems in 2d and 3d. Two cases were analyzed, in which speedup and convergence in very few iterations were achieved. In these two configurations the dynamics of the coupled problem was driven by oscillatory boundary data. Despite the hyperbolic character of the solid problem that prevents the use of too coarse timesteps, parameter sets are found where the Parareal

FIGURE 9. Velocity error from the 3d case. We see the error between sequential (computed with the fine stepsize h_F) and Parareal solution with different coarse step sizes. They are calculated at the subinterval boundaries as the relative error in the Frobenius norm $\frac{\|v_p - v_s\|_F}{\|v_s\|_F}$. Where v_s is the sequential solution and v_p the Parareal solution. The bright grey line is the difference between the sequential solution and a reference solution.

method is both convergent and efficient. In the future it might be beneficial to combine the Parareal approach with multirate time stepping, where different step sizes are chosen for fluid and solid [45].

However, we also saw notable limitations of the algorithm when applied to configurations, where oscillations arise from the internal dynamics of the flow around an obstacle. The combination of different step sizes for coarse and fine discretization can result in a different dynamics and an offset between predictor and corrector. The fsi-3 benchmark problem by Hron and Turek [28] showed to be too challenging and it was not possible to pick discretization parameters that gave speedup and a stable solution at the same time.

FIGURE 10. 3d. Speedup of the Parareal algorithm on 20 cores. We show the speedup for the first 8 Parareal iterations. We mark the iteration, where the Parareal error is of the same order as the discretization error. On the right we show the raw data with all speedups.

Another issue is the severe nonlinearity of fluid-structure interactions. The design on nonlinear (and linear) solvers is already a challenge and coarse timesteps lead to an increased effort that limits the possible speedup.

The Parareal algorithm needs careful investigation of the problem before it can be successfully applied. Choosing the right time step sizes is crucial and even then it could still fail. If the problem is suitable on the other hand, it offers a simple way to speedup the solution.

Acknowledgements. The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany, grant number 05M16NMA as well as the GRK 2297 MathCoRe, funded by the Deutsche Forschungsgemeinschaft, grant number 314838170.

REFERENCES

- [1] G. Ariel, H. Nguyen and R. Tsai, θ -parareal schemes. Preprint <https://arxiv.org/abs/1704.06882> (2017).
- [2] E. Aubanel, Scheduling of tasks in the parareal algorithm. *Parallel Comput.* **37** (2011) 172–182.
- [3] E. Aulisa, S. Bna and G. Bornia, A monolithic ale Newton-Krylov solver with multigrid-Richardson-Schwarz preconditioning for incompressible fluid-structure interaction. *Comput. Fluids* **174** (2018) 213–228.
- [4] G. Avalos, I. Lasiecka and R. Triggiani, Higher regularity of a coupled parabolic hyperbolic fluid-structure interactive system. *Georgian Math. J.* **15** (2008) 402–437.
- [5] G. Avalos and R. Triggiani, Semigroup well-posedness in the energy space of a parabolic-hyperbolic coupled Stokes-Lamé PDE system of fluid-structure interaction. *Discrete Contin. Dyn. Syst. Ser. S* **2** (2009) 417–447.
- [6] G. Avalos and R. Triggiani, Fluid-structure interaction with and without internal dissipation of the structure: a contrast study in stability. *Evol. Equ. Control Theory* **2** (2013) 563–598.
- [7] A.-M. Baudron, J.-J. Lautard, Y. Maday, M. Riahi and J. Salomon, Parareal in time 3d numerical solver for the LWR benchmark neutron diffusion transient model. *J. Comput. Phys.* **279** (2014) 67–79.
- [8] R. Becker and M. Braack, A finite element pressure gradient stabilization for the Stokes equations based on local projections. *Calcolo* **38** (2001) 173–199.
- [9] R. Becker, M. Braack, D. Meidner, T. Richter and B. Vexler, The finite element toolkit GASCOIGNE. Available from: <https://gascoigne.math.uni-magdeburg.de/> (2020).
- [10] M. Besier and W. Wollner, On the pressure approximation in nonstationary incompressible flow simulations on dynamically varying spatial meshes. *Int. J. Numer. Math. Fluids* **69** (2012) 1054–1064.
- [11] A. Blouza, L. Boudin and S.M. Kaber, Parallel in time algorithms with reduction methods for solving chemical kinetics. *Commun. Appl. Math. Comput. Sci.* **5** (2011) 241–263.
- [12] P. Causin, J.F. Gerau and F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Comput. Methods Appl. Mech. Eng.* **194** (2005) 4506–4527.
- [13] R. Croce, D. Ruprecht and R. Krause, Parallel-in-space-and-time simulation of the three-dimensional, unsteady Navier-Stokes equations for incompressible flow. In *Modeling, Simulation and Optimization of Complex Processes-HPSC 2012*. Springer (2014) 13–23.
- [14] J. Donea, An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions. *Comput. Methods Appl. Mech. Eng.* **33** (1982) 689–723.

- [15] L. Failer and T. Richter, A Newton multigrid framework for optimal control of fluid—structure interactions. To appear in: *Optim. Eng.* (2020). <https://doi.org/10.1007/s11081-020-09498-8>.
- [16] L. Failer and T. Richter, A parallel Newton multigrid framework for monolithic fluid—structure interactions. *J. Sci. Comput.* **82** (2020) 28.
- [17] M.A. Fernández and J.-F. Gerbeau, Algorithms for fluid—structure interaction problems. In *Cardiovascular Mathematics: Modeling and simulation of the circulatory system*, volume 1, edited by L. Formaggia, A. Quarteroni, and A. Veneziani. *MS & A. Springer* (2009) 307–346.
- [18] P.F. Fischer, F. Hecht and Y. Maday, A parareal in time semi-implicit approximation of the Navier-stokes equations. In *Domain decomposition methods in science and engineering*. Springer (2005) 433–440.
- [19] S. Frei and T. Richter, A locally modified parametric finite element method for interface problems. *SIAM J. Numer. Anal.* **52** (2014) 2315–2334.
- [20] M.J. Gander and S. Vandewalle, Analysis of the parareal time-parallel time-integration method. *SIAM J. Sci. Comput.* **29** (2007) 556–578.
- [21] M.W. Gee, U. Küttler and W.A. Wall, Truly monolithic algebraic multigrid for fluid—structure interaction. *Int. J. Numer. Meth. Eng.* **85** (2010) 987–1016.
- [22] D. Hartmann, C. Lessig, N. Margenberg and T. Richter, A neural network multigrid solver for the Navier-stokes equations. Preprint <https://arxiv.org/abs/2008.11520> (2020).
- [23] T. Haut and B. Wingate, An asymptotic parallel-in-time method for highly oscillatory PDEs. *SIAM J. Sci. Comput.* **36** (2014) A693–A713.
- [24] M. Heil, A.L. Hazel and J. Boyle, Solvers for large-displacement fluid—structure interaction problems: Segregated vs. monolithic approaches. *Comput. Mech.* **43** (2008) 91–101.
- [25] J.G. Heywood, R. Rannacher and S. Turek, Artificial boundaries and flux and pressure conditions for the incompressible Navier-Stokes equations. *Int. J. Numer. Math. Fluids.* **22** (1992) 325–352.
- [26] C.W. Hirt, A.A. Amsden and J.L. Cook, An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *J. Comp. Phys.* **14** (1974) 227–469.
- [27] J. Hron and S. Turek, A monolithic FEM/Multigrid solver for an ALE formulation of fluid—structure interaction with applications in biomechanics. In *Fluid-structure Interaction: Modeling, Simulation, Optimization*, edited by H.-J. Bungartz and M. Schäfer. *Lecture Notes in Computational Science and Engineering*. Springer (2006) 146–170.
- [28] J. Hron and S. Turek, Proposal for numerical benchmarking of fluid—structure interaction between an elastic object and laminar incompressible flow. In *Fluid-structure Interaction: Modeling, Simulation, Optimization*, edited by H.-J. Bungartz and M. Schäfer. *Lecture Notes in Computational Science and Engineering*. Springer (2006) 371–385.
- [29] T.J.R. Hughes, W.K. Liu and T.K. Zimmermann, Lagrangian-Eulerian finite element formulations for incompressible viscous flows. *Comput. Methods Appl. Mech. Eng.* **29** (1981) 329–349.
- [30] A. Kreienbuehl, A. Naegel, D. Ruprecht, R. Speck, G. Wittum and R. Krause, Numerical simulation of skin transport using parareal. *Comput. Visual. Sci.* **17** (2015) 99–108.
- [31] U. Langer and H. Yang, Recent development of robust monolithic fluid—structure interaction solvers. In *fluid—structure Interactions. Modeling, Adaptive Discretization and Solvers*. Vol. 20 of *Radon Series on Computational and Applied Mathematics*. de Gruyter (2017).
- [32] M. Lenoir, Optimal isoparametric finite elements and error estimates for domains involving curved boundaries. *SIAM J. Numer. Anal.* **23** (1986) 562–580.
- [33] J.-L. Lions, Y. Maday and G. Turinici, Résolution d’edp par un schéma en temps. *C. R. Acad. Sci. Math.* **332** (2001) 661–668.
- [34] M. Luskin and R. Rannacher, On the smoothing property of the Crank-Nicolson scheme. *Appl. Anal.* **14** (1982) 117–135.
- [35] M. Molnar, *Stabilisierte Finite Elemente für Strömungsprobleme auf bewegten Gebieten*. Master’s thesis, Universität Heidelberg (2015).
- [36] R. Rannacher, Finite element solution of diffusion problems with irregular data. *Numer. Math.* **43** (1984) 309–327.
- [37] R. Rannacher, On the stabilization of the Crank-Nicolson scheme for long time calculations. In preparation (1986).
- [38] T. Richter, A monolithic geometric multigrid solver for fluid—structure interactions in ALE formulation. *Int. J. Numer. Meth. Eng.* **104** (2015) 372–390.
- [39] T. Richter, Fluid—structure Interactions. Models, Analysis and Finite Elements. Vol. 118 of *Lecture notes in computational science and engineering*. Springer (2017).
- [40] T. Richter and T. Wick, Finite elements for fluid—structure interaction in ALE and Fully Eulerian coordinates. *Comput. Methods Appl. Mech. Eng.* **199** (2010) 2633–2642.
- [41] T. Richter and T. Wick, On time discretizations of fluid—structure interactions. In *Multiple Shooting and Time Domain Decomposition Methods*, edited by T. Carraro, M. Geiger, S. Körkel, and R. Rannacher. Vol. 9 of *Contributions in Mathematical and Computational Science*. Springer (2015) 377–400.
- [42] T. Richter and S. Frei, Second order time-stepping for parabolic interface problems with moving interfaces. *Model. Math. Anal. Numer.* **51** (2017) 1539–1560.
- [43] D. Ruprecht, Wave propagation characteristics of parareal. *Comput. Visual. Sci.* **19** (2018) 1–17.
- [44] D. Samaddar, D.E. Newman and R. Sánchez, Parallelization in time of numerical simulations of fully-developed plasma turbulence using the parareal algorithm. *J. Comput. Phys.* **229** (2010) 6558–6573.
- [45] M. Soszynska and T. Richter, Adaptive time-step control for a monolithic multirate scheme coupling the heat and wave equation. Preprint <https://arxiv.org/abs/2007.05372> (2020).

- [46] J. Steiner, D. Ruprecht, R. Speck and R. Krause, Convergence of parareal for the Navier-Stokes equations depending on the Reynolds number. In Numerical Mathematics and Advanced Applications - *ENUMATH 2013*. Edited by Assyr Abdulle, Simone Deparis, Daniel Kressner, Fabio Nobile, and Marco Picasso. Springer International Publishing, Cham (2015) 195–202.
- [47] J.M.F. Trindade and J.C.F. Pereira, Parallel-in-time simulation of the unsteady Navier–stokes equations for incompressible flow. *Int. J. Numer. Methods Fluids* **45** (2004) 1123–1136.
- [48] S. Turek, J. Hron, M. Madlik, M. Razzaq, H. Wobker and J. Acker, Numerical simulation and benchmarking of a monolithic multigrid solver for fluid–structure interaction problems with application to hemodynamics. Technical report, Fakultät für Mathematik, TU Dortmund. Ergebnisberichte des Instituts für Angewandte Mathematik, Num. 403 (2010).
- [49] S. Turek, L. Rivkind, J. Hron and R. Glowinski, Numerical study of a modified time–stepping theta–scheme for incompressible flow simulations. *J. Sci. Comput.* **28** (2006) 533–547.