

Adaptive time stepping for fluid-structure interaction solvers

M. Mayr^{a,b,1,*}, W. A. Wall^c, M. W. Gee^b

^aSandia National Laboratories, 7011 East Avenue, MS 9158, Livermore, CA 94550

^bMechanics & High Performance Computing Group, Technical University of Munich, Parkring 35, D-85748 Garching bei München, Germany

^cInstitute for Computational Mechanics, Technical University of Munich, Boltzmannstraße 15, D-85748 Garching bei München, Germany

Abstract

A novel adaptive time stepping scheme for fluid-structure interaction (FSI) problems is proposed that allows for controlling the accuracy of the time-discrete solution. Furthermore, it eases practical computations by providing an efficient and very robust time step size selection. This has proven to be very useful, especially when addressing new physical problems, where no educated guess for an appropriate time step size is available. The fluid and the structure field, but also the fluid-structure interface are taken into account for the purpose of *a posteriori* error estimation, rendering it easy to implement and only adding negligible additional cost. The adaptive time stepping scheme is incorporated into a monolithic solution framework, but can straightforwardly be applied to partitioned solvers as well. The basic idea can be extended to the coupling of an arbitrary number of physical models. Accuracy and efficiency of the proposed method are studied in a variety of numerical examples ranging from academic benchmark tests to complex biomedical applications like the pulsatile blood flow through an abdominal aortic aneurysm. The demonstrated accuracy of the time-discrete solution in combination with reduced computational cost make this algorithm very appealing in all kinds of FSI applications.

Keywords: fluid-structure interaction; time integration; error estimation; adaptivity

1. Introduction

Many scientific and engineering problems involve the coupling of several physical effects or models. One class of coupled problems is the interaction of fluid flow and solid bodies. Possible applications range from aero-elasticity over civil engineering to biomedical problems like the analysis of blood flow in the human vascular system. Application-wise and from a numerical point of view, the interaction of an incompressible fluid flow with solid bodies undergoing finite deformation is of particular interest. Although many researchers addressed this class of problems for decades, solving *fluid-structure interaction (FSI)* problems numerically still poses a challenging task.

Most problems in FSI are transient and, thus, involve temporal discretization and time integration. To allow for the computation of a temporally accurate solution while simultaneously limiting the computational cost, we propose an adaptive time stepping algorithm for FSI problems based on *a posteriori* error estimation. To the authors' best knowledge, such an approach is not described in literature yet, and we aim at closing this gap. In our FSI solver, we allow for the possibility of choosing time integration schemes for the solid and the fluid field independently and tailored to each field's needs as recently introduced by MAYR *et al.* [1] in the context of monolithic FSI solvers, but require the time step size to be the same in both fields. The error of the fully implicit marching time integration scheme is estimated with the help of an auxiliary explicit scheme. Due to the explicit character of the auxiliary scheme, computational cost associated with the error estimation is negligible. As FSI is a surface-coupled problem, the fluid field and the structure field, but also the fluid-structure interface are taken into account for error estimation. The estimated error is then used to adapt the time step size throughout the entire simulation such that the error does not exceed a user-given tolerance.

*Corresponding author (email: mayr.mt@tum.de)

Email addresses: mayr.mt@tum.de (M. Mayr), wall@lnm.mw.tum.de (W. A. Wall), gee@mhpc.mw.tum.de (M. W. Gee)

¹This work was performed while the author was affiliated with the Mechanics & High Performance Computing Group, Technical University of Munich, Parkring 35, D-85748 Garching bei München, Germany.

The algorithm ensures that the time step size is chosen such that every portion of the FSI domain satisfies its individual demand for accuracy. At the same time, huge savings of computational cost can be achieved w.r.t. two aspects. Firstly, wall clock time of a single run of a simulation can be significantly reduced compared to the case of non-adaptive time stepping. Secondly, finding a suitable time step size via an adaptive procedure is beneficial when one needs to simulate problems where no educated guess for an appropriate time step size is known. The adaptive scheme chooses a suitable time step size by itself and prevents the need of several trial runs with different values for the pure purpose of finding an appropriate value for the time step size. To the authors’ best knowledge, this is the first adaptive time stepping algorithm for nonlinear FSI problems undergoing large deformation.

In this contribution, we use the proposed adaptive time stepping scheme to study the interaction of an incompressible fluid flow in ALE description with a deformable solid body. We solve the problem with a fully implicit finite-element-based monolithic FSI solver as previously described in [1, 2], but we will also address the algorithmic layout in case of partitioned solution schemes. We base our approach on adaptive time stepping schemes for single-field problems that are well-known for decades, e.g. for solid dynamics [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] and fluid dynamics applications [14, 15, 16, 17, 18, 19, 20, 21].

Optimal design of time integration schemes is essential to guarantee stability and accuracy in FSI computations. Many efforts have been undertaken to develop stable time integration routines for ALE-based fluid computations on moving domains [22, 23, 24, 25, 26]. The possibility of choosing time integration schemes for the solid and the fluid field independently and tailored to each field’s needs has recently been introduced by MAYR *et al.* [1] in the context of monolithic FSI solvers.

Adaptive time stepping schemes for the solution of *ordinary differential equations (ODEs)* date back roughly a century. Based on an approximation of the error of the time-discrete solution, they adapt the time step size Δt_n to match a user-given level of accuracy. Schemes relying on *a posteriori* error estimation can be found in textbook literature [27, 28, 29, 30]. Alternative approaches based on control theory have been developed by GUSTAFSSON *et al.* [31] and SÖDERLIND [32, 33]. Such approaches are said to increase stability and to produce a smoother evolution of time step sizes. A detailed analysis of the analogy of these approaches to the more classical methods based on *a posteriori* error estimation is carried out in [28], for example. In this work, algorithms based on *a posteriori* error estimation produced satisfying results for FSI problems. Hence, approaches based on control theory are not considered in this manuscript, but can straightforwardly be incorporated into the presented algorithmic framework.

Recently, approaches for *goal-oriented error estimation* became very popular, where one aims at controlling the error in a user-chosen *quantity of interest*. Since these techniques usually require the solution of an adjoint problem, they become computationally and storage-wise very expensive in transient problems and, thus, are not considered in this work. However, some approaches addressing these issues are available in literature. For instance, CYR *et al.* [34] use data compression techniques to reduce the huge storage demands, while CAREY *et al.* [35] apply a block-wise adaptivity approach based on coarse scale adjoint solutions. Promising work based on modal analysis has been done by VERDUGO *et al.* [36, 37] for time-dependent solid mechanics problems.

The remainder of this manuscript is organized as follows: First, we briefly summarize the FSI problem and outline the monolithic solution procedure in Section 2. Some fundamentals of adaptive time stepping based on *a posteriori* error estimation are recalled in Section 3, before the adaptive time stepping procedure for FSI solvers is proposed in Section 4 including the discussion of important practical aspects. Section 5 presents numerical examples, that demonstrate and discuss features and properties of the proposed adaptive time stepping scheme. Finally, a summary is given in Section 6.

2. Fluid-Structure Interaction in a Nutshell

In the present contribution, we exemplarily study the interaction of an incompressible fluid flow with solid bodies undergoing finite deformation. We apply a monolithic solution scheme. A brief introduction to such FSI problems is given here, while a detailed description of the model, its discretization, and a thorough derivation of the monolithic solution method have been presented in MAYR *et al.* [1].

2.1. Physical Model

We couple two physical domains, namely a deformable fluid domain $\Omega^{\mathcal{F}}$ and a solid domain $\Omega^{\mathcal{S}}$, cf. Figure 1. To

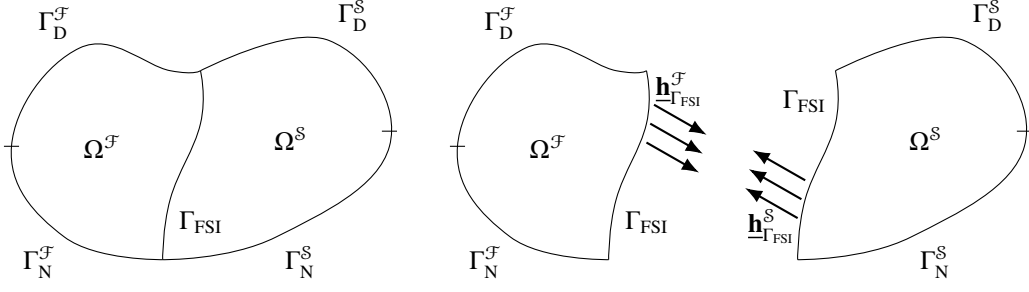


Figure 1: Problem statement adopted from [38] — *Left*: The domain Ω is subdivided into a fluid domain $\Omega^{\mathcal{F}}$ and a structural domain $\Omega^{\mathcal{S}}$ by the fluid-structure interface Γ_{FSI} . Both subdomains are bounded by DIRICHLET boundaries $\Gamma_{\text{D}}^{\mathcal{F}}$ and $\Gamma_{\text{D}}^{\mathcal{S}}$, NEUMANN boundaries $\Gamma_{\text{N}}^{\mathcal{F}}$ and $\Gamma_{\text{N}}^{\mathcal{S}}$, and the common fluid-structure interface Γ_{FSI} , respectively. *Right*: At the interface, kinematic continuity as well as equilibrium of interface traction fields $\underline{\mathbf{h}}_{\Gamma_{\text{FSI}}}^{\mathcal{F}}$ and $\underline{\mathbf{h}}_{\Gamma_{\text{FSI}}}^{\mathcal{S}}$ are required.

account for the moving fluid domain, an *arbitrary LAGRANGEAN–EULERIAN (ALE)* observer is used for the fluid field, while the solid body is described in a purely LAGRANGEAN fashion. The fluid field is governed by the incompressible NAVIER–STOKES equations

$$\begin{aligned} \rho^{\mathcal{F}} \frac{\partial \underline{\mathbf{u}}^{\mathcal{F}}}{\partial t} + \rho^{\mathcal{F}} (\underline{\mathbf{u}}^{\mathcal{F}} - \underline{\mathbf{u}}^{\mathcal{S}}) \cdot \nabla \underline{\mathbf{u}}^{\mathcal{F}} - 2\mu_{\text{dyn}}^{\mathcal{F}} \nabla \cdot \underline{\underline{\boldsymbol{\varepsilon}}}(\underline{\mathbf{u}}^{\mathcal{F}}) + \nabla p^{\mathcal{F}} = \rho^{\mathcal{F}} \underline{\mathbf{b}}^{\mathcal{F}}, \\ \nabla \cdot \underline{\mathbf{u}}^{\mathcal{F}} = 0, \end{aligned}$$

with the primary unknowns $\underline{\mathbf{u}}^{\mathcal{F}}$ and $p^{\mathcal{F}}$ being the fluid velocity and pressure field, respectively. The fluid density and dynamic viscosity are denoted by $\rho^{\mathcal{F}}$ and $\mu_{\text{dyn}}^{\mathcal{F}}$, respectively, while the strain rate tensor is computed as the symmetric gradient of the fluid velocity $\underline{\mathbf{u}}^{\mathcal{F}}$. Possible body forces in the fluid field are denoted by $\underline{\mathbf{b}}^{\mathcal{F}}$. As the fluid field is described in an ALE fashion, the grid velocity $\underline{\mathbf{u}}^{\mathcal{S}}$ needs to be computed from the grid displacement field $\underline{\mathbf{d}}^{\mathcal{S}}$. For moderately deforming fluid domains, the grid displacement field $\underline{\mathbf{d}}^{\mathcal{S}}$ is determined by harmonic extension whereas large deformations require the assumption that the ALE field behaves like a pseudo-elastic solid. The solid body with density $\rho^{\mathcal{S}}$ and body force $\underline{\mathbf{b}}_0^{\mathcal{S}}$ per undeformed unit volume is governed by the nonlinear elastodynamics equation

$$\rho_0^{\mathcal{S}} \frac{d^2 \underline{\mathbf{d}}^{\mathcal{S}}}{dt^2} = \nabla_0 \cdot \left(\underline{\underline{\mathbf{F}}} \underline{\underline{\mathbf{S}}} \right) + \underline{\mathbf{b}}_0^{\mathcal{S}},$$

where the displacement field $\underline{\mathbf{d}}^{\mathcal{S}}$ is the primary unknown. The deformation is fully characterized by the deformation gradient $\underline{\underline{\mathbf{F}}} = \nabla_0 \underline{\mathbf{x}}^{\mathcal{S}}$. For the compressible or nearly incompressible solid, we assume a hyperelastic strain energy function $\Psi^{\mathcal{S}}$ to compute the 2nd PIOLA–KIRCHHOFF stresses $\underline{\underline{\mathbf{S}}} = 2\partial\Psi^{\mathcal{S}}/\partial\underline{\underline{\mathbf{C}}}$ using the right CAUCHY–GREEN tensor $\underline{\underline{\mathbf{C}}} = \underline{\underline{\mathbf{F}}}^T \underline{\underline{\mathbf{F}}}$. At the fluid-structure interface Γ_{FSI} , we require kinematic continuity of fluid and solid velocity fields, i.e. $\underline{\mathbf{u}}_{\Gamma_{\text{FSI}}}^{\mathcal{F}}(\underline{\mathbf{x}}, t) = \partial \underline{\mathbf{d}}_{\Gamma_{\text{FSI}}}^{\mathcal{S}} / \partial t(\underline{\mathbf{X}}, t)$, as well as the equilibrium of interface traction fields $\underline{\mathbf{h}}_{\Gamma_{\text{FSI}}}^{\mathcal{F}}$ and $\underline{\mathbf{h}}_{\Gamma_{\text{FSI}}}^{\mathcal{S}}$. The kinematic constraint is enforced weakly via a LAGRANGE multiplier field $\underline{\lambda}$, which allows for an interpretation of the LAGRANGE multiplier field as the interface traction. Here, we make the arbitrary choice $\underline{\lambda} = \underline{\mathbf{h}}_{\Gamma_{\text{FSI}}}^{\mathcal{S}} = -\underline{\mathbf{h}}_{\Gamma_{\text{FSI}}}^{\mathcal{F}}$, i.e. the LAGRANGE multiplier field is seen as the interface traction acting onto the solid side of the interface.

2.2. The Monolithic Solution Method for FSI

To establish a monolithic solution method for the coupled FSI problem, where all equations are solved simultaneously, spatial and temporal discretization are performed field-wise before the final assembly of the monolithic system of equations. In this work, the spatial discretization of the solid and the fluid field is performed by the finite element method, but the proposed approach would also obviously work with other spatial discretization schemes. In the solid field, displacement-based linear finite elements are utilized, while techniques to deal with locking phenomena (like *enhanced assumed strains* or *F-bar elements* [39]) are applied whenever necessary. In the fluid field, equal-order interpolated linear finite elements are used, that require residual-based stabilization like *Streamline Upwind PETROV–GALERKIN*

(SUPG) [40], Pressure-Stabilized PETROV–GALERKIN (PSPG) [41], and a *grad-div* term [42]. The stabilization parameter follows the definition in [43]. The LAGRANGE multiplier field is discretized with a dual mortar method [1, 44, 45] resulting in mortar coupling matrices that allow for a cheap condensation of the LAGRANGE multiplier degrees of freedom from the monolithic system of equations. In the context of mortar methods, either the solid or the fluid field can be chosen as the master side, resulting in two distinct solver formulations referred to as *structure-handled interface motion* and *fluid-handled interface motion*, cf. [1] for details.

Temporal discretization is done by finite differencing. For time integration, we use fully implicit, single-step, single-stage time integration schemes. In the solid field, we employ the generalized- α method [46]. The fluid field either uses the generalized- α method [47] or the one-step- θ scheme [48]. Both generalized- α schemes [46, 47] are second order methods, while the accuracy of the one-step- θ scheme depends on the choice of the parameter θ . A temporal interpolation of the interface traction fields is introduced to allow for an independent choice of the time integration schemes in the solid and the fluid field in a temporally consistent manner [1].

Putting the residual expressions \mathbf{r}^S , \mathbf{r}^G and \mathbf{r}^F from the solid, the ALE, and the fluid field as well as the kinematic constraint \mathbf{r}^{kin} together yields the monolithic nonlinear residual vector $\mathbf{r}^{\text{FSI}^T} = \begin{bmatrix} \mathbf{r}^S & \mathbf{r}^G & \mathbf{r}^F & \mathbf{r}^{\text{kin}} \end{bmatrix}^T$ that needs to vanish in every time step. The nonlinearity is treated by a NEWTON–KRYLOV method with FSI-specific preconditioning as proposed by GEE *et al.* [2] or MAYR *et al.* [49]. After the assembly and the subsequent static condensation of the LAGRANGE multiplier and slave side interface degrees of freedom, the monolithic system of linear equations schematically reads

$$\begin{bmatrix} \mathcal{S} & & \mathcal{C}^{S\mathcal{F}} \\ & \mathcal{A} & \mathcal{C}^{G\mathcal{F}} \\ \mathcal{C}^{\mathcal{F}S} & \mathcal{C}^{\mathcal{F}G} & \mathcal{F} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}^S \\ \Delta \mathbf{x}^G \\ \Delta \mathbf{x}^F \end{bmatrix} = - \begin{bmatrix} \mathbf{r}^S \\ \mathbf{r}^G \\ \mathbf{r}^F \end{bmatrix}$$

where iteration and time step indices have been omitted to simplify the notation. The matrices \mathcal{S} , \mathcal{A} , and \mathcal{F} on the main diagonal reflect the solid, the ALE, and the fluid field residual linearizations, respectively. The coupling among the fields is represented by the off-diagonal blocks \mathcal{C}^{ij} , where superscripts $i, j \in \{S, G, F\}$ indicate the coupling between the fields.

2.3. Partitioned Solution Methods for FSI

In partitioned schemes, a sequence of single field solutions is required, where coupling information is exchanged between the fluid and the structure field which is often done in a DIRICHLET–NEUMANN partitioning [50, 51, 52, 53, 54] or by using ROBIN-type transmission conditions [55, 56, 57, 58]. In strongly coupled solvers, a fix-point iteration between the fields is performed until convergence, while in loosely coupled schemes this fix-point iteration is stopped after a couple or even after a single iteration.

Following the arguments in [44], the DIRICHLET partition coincides with the slave field, while the NEUMANN partition is analogue to the master field of a monolithic solver. In particular, we stress the analogy of DIRICHLET–NEUMANN partitioning and the monolithic solver with a structure-handled interface motion. This analogy will come into play when choosing the error norms that are required to assess the local discretization error of the coupled FSI problem.

3. Adaptive Time Stepping based on a *posteriori* Error Estimation

3.1. Adapting the Time Step Size

Primarily, adaptive time stepping algorithms aim at controlling the global error $\mathbf{g}_{\underline{\mathbf{x}}_{n+1}} = \mathbf{x}(t_{n+1}) - \underline{\mathbf{x}}_{n+1}$ [28, 29], such that $\|\mathbf{g}_{\underline{\mathbf{x}}_{n+1}}\|_{L_2} \leq \varepsilon_g^{\Delta t}$ holds in every time step with a user-given tolerance $\varepsilon_g^{\Delta t}$ for the global error and with the L_2 -norm defined as

$$\|(\bullet)\|_{L_2} = \sqrt{\int_{\Omega} (\bullet)^2}. \quad (1)$$

Since the global error is not accessible directly, its control is infeasible in practical applications. Defining the *local error* $\mathbf{l}_{\underline{\mathbf{x}}_{n+1}} = \mathbf{x}(t_{n+1}) - \underline{\mathbf{x}}_{n+1}$ as the error made within one time step when assuming to start from the exact solution $\underline{\mathbf{x}}_n =$

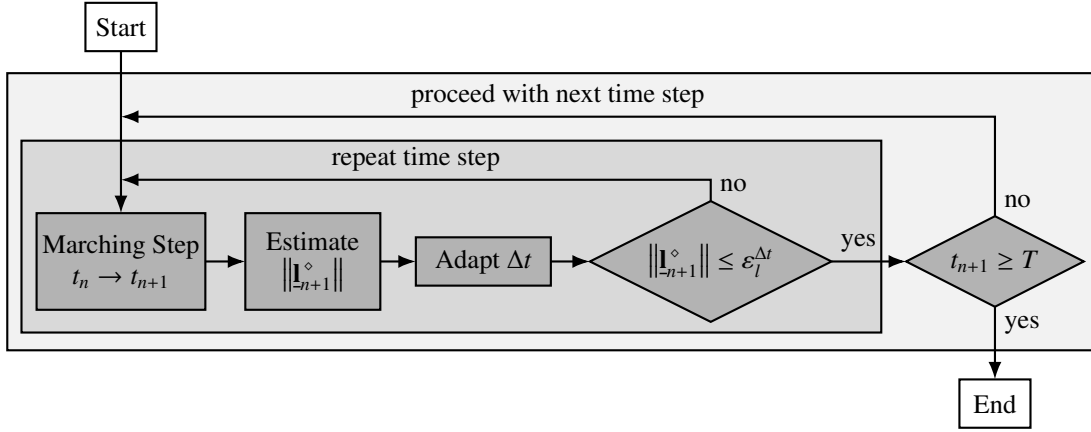


Figure 2: Adaptive time stepping algorithm based on *a posteriori* error estimation

$\underline{\mathbf{x}}(t_n)$ at the beginning of the time step, the alternative goal of keeping the local error below a user-given tolerance $\varepsilon_l^{\Delta t}$ is commonly pursued [28], reading

$$\|\underline{\mathbf{l}}_{n+1}\|_{L_2} \leq \varepsilon_l^{\Delta t}. \quad (2)$$

Applying standard manipulations and arguments and using p to denote the order of accuracy of the time integration scheme, an *optimized scaling factor*

$$\kappa^* \approx \sqrt[p+1]{\frac{\varepsilon_l^{\Delta t}}{\|\underline{\mathbf{l}}_{n+1}\|_{L_2}}} \quad (3)$$

can be computed, that is used to increase or decrease the time step size in order to satisfy (2). Since the exact local error $\|\underline{\mathbf{l}}_{n+1}\|_{L_2}$ is not available in (3), it has been replaced by its estimate $\|\underline{\mathbf{l}}_{n+1}^\diamond\|_{L_2}$ whose computation will be detailed in Section 3.2. Based on the current time step size Δt_n , the user-given set $[\kappa] = \{\kappa_{\min}, \kappa_{\max}, \kappa_s\}$ of algorithmic parameters as well as some user-given bounds on the time step size, an optimal time step size can be computed as

$$\Delta t^* = \min \{\Delta t_{\max}, \max \{\min \{\kappa_{\max}, \max \{\kappa_{\min}, \kappa_s \kappa^*\}\} \Delta t_n, \Delta t_{\min}\}\}. \quad (4)$$

In order to avoid too many repetitions of time steps, a *safety factor* $\kappa_s < 1$ is used, that helps to keep the local error below and away from the tolerance barrier [29]. The effective scaling factor $\kappa_s \kappa^*$ is limited by user-given factors κ_{\min} and κ_{\max} that denote the minimal and maximal ratio of time step size decrease or increase, respectively, such that $\kappa_{\min} \leq \kappa_s \kappa^* \leq \kappa_{\max}$. Finally, the time step size is limited by the lower and upper bounds Δt_{\min} and Δt_{\max} . Section 4.3 will provide more details on the choice of these algorithmic parameters as well as a brief discussion of the norm calculation in practical computer codes.

The algorithm is summarized in Figure 2. In each time step, one starts with performing the integration from t_n , where all values are known, to t_{n+1} using the fully implicit scheme as the marching time integration scheme. Afterwards, the error associated with the result of this marching step is assessed by means of error estimation, yielding $\|\underline{\mathbf{l}}_{n+1}^\diamond\|_{L_2}$. The estimated error is then used to compute a new time step size Δt^* using (4) in combination with (3). If the estimated error $\|\underline{\mathbf{l}}_{n+1}^\diamond\|_{L_2} > \varepsilon_l^{\Delta t}$, the accuracy demand (2) has been violated and, thus, the current time step needs to be repeated with a *reduced* time step size $\Delta t_n = \Delta t^*$ where (3) guarantees $\kappa^* < 1$ in (4). If $\|\underline{\mathbf{l}}_{n+1}^\diamond\|_{L_2} \leq \varepsilon_l^{\Delta t}$, one can directly proceed to the next time step with a new, potentially *increased*, time step size $\Delta t_{n+1} = \Delta t^*$.

In case of drastic changes of the transient behavior, especially if these changes occur suddenly like in the case of buckling events, it might be necessary to repeat a single time step $t_n \rightarrow t_{n+1}$ several times while reducing the time step size in each repetition. For practical reasons, the number of repetitions of a single time step may be limited. See Section 4.3 for further discussion.

To increase robustness of application codes, time step size reductions and time step repetitions may be necessary if the nonlinear solver fails to converge. Further details on such problems as well as a possible strategy to deal with them will be outlined in Section 4.3.

3.2. Estimation of the Local Discretization Error

For error estimation, it is common practice to compare two numerical solutions to each other, which are of different orders of accuracy. Although approaches like RICHARDSON *extrapolation* and *embedded methods* are very popular, error estimation in this work is mostly performed by comparison of two different schemes, namely the *marching* scheme of order p and the *auxiliary* scheme of order \hat{p} . To properly deal with the stiff nature of the coupled systems of PDEs, the marching scheme is always chosen as a fully implicit scheme while the auxiliary scheme is explicit for efficiency reasons. Denoting the solution at time t_{n+1} obtained with the marching and the auxiliary integrator by $\underline{\mathbf{x}}_{n+1}$ and $\hat{\underline{\mathbf{x}}}_{n+1}$, respectively, the local discretization error of the marching time integration scheme is estimated as

$$\mathbf{I}_{n+1}^\circ = \underline{\mathbf{x}}_{n+1} - \hat{\underline{\mathbf{x}}}_{n+1}, \quad (5)$$

which asymptotically is an estimate of the local discretization error of the lower order scheme of order p [29]. However in practical applications, one might choose the marching and the auxiliary scheme such that $p > \hat{p}$ or $p = \hat{p}$. Then, the concept of error estimation is abandoned and \mathbf{I}_{n+1}° is merely used for step size selection [29].

If the generalized- α time integration method is used in the solid field, error estimation is often based on TAYLOR series expansion. Following [59], the estimate is given as

$$\mathbf{I}_{n+1}^\circ \approx \left(\beta - \frac{1}{6} \right) \Delta t_n^2 \left(\ddot{\underline{\mathbf{x}}}_{n+1}^{\text{NM}} - \ddot{\underline{\mathbf{x}}}_n \right) \quad (6)$$

with the NEWMARK parameter β , the acceleration $\ddot{\underline{\mathbf{x}}}_{n+1}^{\text{NM}}$ at time t_{n+1} based on the NEWMARK update, and the known acceleration $\ddot{\underline{\mathbf{x}}}_n$ at time t_n .

4. The Adaptive Time Stepping Procedure for Fluid-Structure Interaction Solvers

We now present the adaptive time stepping procedure which is equally applicable to partitioned and monolithic FSI solvers. Here, we focus on monolithic solvers and exploit the analogy of partitioned and monolithic schemes as discussed in Section 2.3 whenever necessary.

We base the adaptive time stepping scheme on a *posteriori* error estimation of the temporal discretization errors in both the fluid and the structure field. Since dynamics and accuracy demands may differ between the fluid and the structure field, the temporal discretization errors are estimated separately in both fields. To account for the central role of the fluid-structure interface, additional attention is paid to its temporal discretization error. Although error estimation is performed field-wise and, thus, different suggestions for the new time step size can be calculated, we aim at a uniform time step size for all fields, since monolithic solution schemes do not easily allow for subcycling as it is possible in case of partitioned approaches [60, 61]. Due to the field-wise error estimation, errors and tolerances for both fields need to be distinguished, which is done by the superscripts $(\bullet)^{\mathcal{F}}$ and $(\bullet)^{\mathcal{S}}$ for fluid and solid quantities, respectively. Whenever the distinction is clear from the context or a statement applies equally to both fields, the superscript will be omitted for the sake of an uncluttered notation.

A typical outline of an adaptive time stepping algorithm has already been shown in Figure 2 for single-field applications. In case of the FSI problem, two modifications are necessary: First, the estimation of the local error is replaced by estimations of local errors in both the fluid and the structure field as well as at the fluid-structure interface (cf. Section 4.1). Second, during adaption of the time step size, a single value for the time step size needs to be selected, that is then used in all fields (cf. Section 4.2). Practical aspects, partitioned solvers, and the extensions to other coupled multiphysics phenomena will be discussed briefly in Sections 4.3, 4.4, and 4.5, respectively.

Table 1: Error norms e_β^α , $\alpha \in \{\mathcal{F}, \mathcal{S}\}$ and $\beta \in \{\Gamma \cup \text{I}, \Gamma, \text{I}\}$, for fluid-handled and structure-handled interface motion — On the master side, a separate interface error e_Γ^{ma} is estimated, while on the slave side only interior degrees of freedom contribute to the error norm e_1^{sl} . The set of estimated error norms for both choices of master and slave side are indicated by the symbol \times .

Interface motion handled by	master field	slave field	$e_{\Gamma \cup \text{I}}^{\mathcal{F}}$	$e_\Gamma^{\mathcal{F}}$	$e_1^{\mathcal{F}}$	$e_{\Gamma \cup \text{I}}^{\mathcal{S}}$	$e_\Gamma^{\mathcal{S}}$	$e_1^{\mathcal{S}}$
Fluid field	$\Omega^{\mathcal{F}}$	$\Omega^{\mathcal{S}}$	\times	\times	\times	\times	\times	\times
Structure field	$\Omega^{\mathcal{S}}$	$\Omega^{\mathcal{F}}$			\times	\times	\times	\times

4.1. Estimation of the Local Discretization Error

Since the solid and the fluid time integration schemes are chosen independently, the overall temporal discretization error of the coupled problem is not accessible. Hence, the temporal discretization error is estimated separately in both fields, while additional focus is put on the fluid-structure interface. The proposed framework is not limited to specific error estimation approaches, but rather can deal with a variety of methods. Exemplarily, error estimation approaches for solid and fluid dynamics are given in [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] and [14, 15, 16, 17, 18, 19, 20, 21], respectively, to name a few without claiming completeness of that list.

When accounting for the central role of the interface, the algorithmic decision of master and slave side comes into play. On the master side, whose interface degrees of freedom are used to express the interface motion, an interface error is deduced by extracting the interface degrees of freedom from the master side's vector of local error. In addition, the error associated with the interior degrees of freedom is considered in both master and slave field. The notation e_β^α is introduced to denote the norm of the estimated local error, where the superscript $\alpha \in \{\mathcal{F}, \mathcal{S}\}$ indicates fluid or structure field and the subscript $\beta \in \{\Gamma \cup \text{I}, \Gamma, \text{I}\}$ provides information on the involved degrees of freedom with Γ_{FSI} being the interface and I the interior of a field. The time step index has been dropped to ease notation. Table 1 provides an overview of the different sets of estimated error norms e_β^α in case of fluid-handled and structure-handled interface motion. The estimation of the error of the inner or interface subset of degrees of freedom does not lead to notable additional computational costs as it is performed by extracting the respective subsets of degrees of freedom from the full vector of estimated temporal discretization error.

We use the notation $[\bullet]$ to denote a set of quantities (\bullet) of the same type but related to different subdomains or portions of degrees of freedom. As given in Table 1, the set of estimated error norms in case of fluid-handled interface motion is denoted by

$$[e]^{\mathcal{F}} = \{e_{\Gamma \cup \text{I}}^{\mathcal{F}}, e_\Gamma^{\mathcal{F}}, e_1^{\mathcal{F}}, e_1^{\mathcal{S}}\},$$

where the superscript \mathcal{F} at $[e]^{\mathcal{F}}$ indicates the fact that the fluid field has been chosen as master field. Accordingly, the set of estimated error norms $[e]^{\mathcal{S}}$ for the case of structure-handled interface motion is given as

$$[e]^{\mathcal{S}} = \{e_{\Gamma \cup \text{I}}^{\mathcal{S}}, e_\Gamma^{\mathcal{S}}, e_1^{\mathcal{S}}, e_1^{\mathcal{F}}\}. \quad (7)$$

Details on the computation of the error norms e_β^α , $\alpha \in \{\mathcal{F}, \mathcal{S}\}$ and $\beta \in \{\Gamma \cup \text{I}, \Gamma, \text{I}\}$, that are necessary to provide a scalar value for each estimated error, will be given in Section 4.3.1. Note that degrees of freedom that are subject to DIRICHLET boundary conditions are excluded from the error estimation, since their values are prescribed and, thus, exact.

4.2. Adapting the Time Step Size

Every error norm in the set $[e]^{\text{ma}}$, $\text{ma} \in \{\mathcal{F}, \mathcal{S}\}$, needs to be taken into account to compute the new time step size for the coupled problem. In order to transform the set of error norms to a single value for the optimal time step size Δt^* , the following steps are performed: First, a set of optimal scaling factors $[\kappa^*]$ is computed by applying (3) to each estimated error norm in the set of errors $[e]^{\text{ma}}$. To account for the accuracy demands in the single fields, the respective tolerance $\varepsilon_1^{\mathcal{S}, \Delta t}$ or $\varepsilon_1^{\mathcal{F}, \Delta t}$ is used. Secondly, a set $[\Delta t^*]$ of time step size suggestions is calculated based on the set $[\kappa^*]$ of optimal scaling factors as well as the algorithmic parameters $[\kappa]$. Finally, the optimal time step size Δt^* of the coupled problem is determined as

$$\Delta t^* = \min \{[\Delta t^*]\}. \quad (8)$$

Equation (8) guarantees that the time step size is governed by the subset of degrees of freedom, that is most critical in terms of achieving the desired tolerance $\varepsilon_l^{S,\Delta t}$ or $\varepsilon_l^{F,\Delta t}$. Doing so, the accuracy demand can be satisfied globally in the entire computational domain.

4.3. Practical Considerations

4.3.1. Computation of Norms

In many practical computer codes, the computation of the L_2 -norm as defined in (1) is replaced by EUCLIDIAN vector norms $\|(\bullet)\|_2$, which are recommended in the context of time step size adaptivity due to their smoothness properties [28]. To account for the size of the spatial discretization and to allow comparability in the case of mesh refinement, a length scaling is introduced such that

$$\|(\bullet)\|_2 = \sqrt{\frac{\sum_{m=1}^M (\bullet)_m^2}{M}}$$

with M being the number of entries in the respective vector (\bullet) .

4.3.2. Averaging of Increasing Time Step Sizes

As given in (4), the increase of the time step size is limited by the user-given factor κ_{\max} . However, this still may lead to *overshooting*, i.e. to situations where the time step size is increased so much that the subsequent time step requires a time step size reduction which finally yields a non-smooth and wiggly evolution of the time step size. As a remedy, we propose an averaging procedure in the event of an increasing time step size. The new time step size Δt_{n+1} is computed as a linear combination of the newly determined optimal time step size Δt^* and M previous time step sizes via

$$\Delta t_{n+1} = \gamma_{n+1} \Delta t^* + \sum_{m=0}^{M-1} \gamma_{n-m} \Delta t_{n-m} = \gamma_{n+1} \Delta t^* + \gamma_n \Delta t_n + \gamma_{n-1} \Delta t_{n-1} + \dots \quad (9)$$

with user-chosen weights γ_i satisfying $\sum_{m=0}^M \gamma_{n-m+1} = 1$ and $\gamma_{n-m+1} = 0 \forall m > M$. This can effectively reduce the occurrence of overshooting events. Furthermore, it fosters a smooth evolution of the time step size, which seems to be beneficial when having in mind the time step size dependency of the fluid stabilization or stability and accuracy properties of the time integration schemes, that are only guaranteed if the ratio $\Delta t_{n+1}/\Delta t_n$ is not too large [30].

Since a decrease of the time step size is always triggered by a violation of the accuracy demand, i.e. $\|\mathbb{I}_{n+1}^\circ\|_{L_2} > \varepsilon_l^{\Delta t}$, it is crucial to allow for the desired decrease immediately in order to satisfy the accuracy demand. Henceforth, the time step size Δt_n of the time step, that needs to be repeated, is set to the decreased value Δt^* immediately without any further modifications.

4.3.3. Choice of Algorithmic Parameters

The upper time step size limit Δt_{\max} can often be chosen based on the desired temporal resolution, meaning that the time step size may not exceed a certain value in order to be able to capture the transient behavior of the problem at hand.

The lower time step size limit Δt_{\min} should be chosen small enough such that the algorithm is able to satisfy the accuracy demand, but not too small. For example, numerical results for a STOKES problem reported in [62] indicate that the stabilizing effect of the fluid stabilization deteriorates if the time step size is too small in comparison to the spatial grid resolution. A detailed analysis of this issue has been performed by [63], which can be seen as providing a rule of thumb to choose the lower time step size limit Δt_{\min} given a specific stabilization technique and a certain grid resolution.

We start every simulation using the lower bound Δt_{\min} as initial time step size, i.e. $\Delta t_0 = \Delta t_{\min}$ to ensure the satisfaction of the accuracy demand (2) right from the beginning of the simulation. Since the initial time step size is chosen as the lower bound, a rapid increase is expected during the first couple of time steps. If the time step size does not increase at the beginning of the simulation, the lower bound Δt_{\min} is not low enough to achieve the desired error level $\varepsilon_l^{\Delta t}$. Either the tolerance $\varepsilon_l^{\Delta t}$ needs to be reviewed or Δt_{\min} has to be decreased to allow for satisfying the accuracy

demand (2) throughout the entire simulation. If the time step size hits the lower bound Δt_{\min} during the simulation, at least a warning should be issued to the user.

The parameters κ_{\min} and κ_{\max} are chosen in the ranges of 0.1 – 0.5 and 1.5 – 5 [29], respectively. Depending on the time integration scheme, these limitations have to be chosen even tighter [29]. Typical values for the safety factor κ_s are in the range of 0.8 – 0.95 [29]. Note that choosing κ_s strictly smaller than 1 speeds up the computation by avoiding many time step repetitions

If a massive reduction of Δt_n is required, multiple repetitions of the same time step $t_n \rightarrow t_{n+1}$ may be necessary. The number of repetitions is usually limited to avoid stalling of the simulation. Furthermore, a large number of repetitions may indicate that the time step size has been far too large just before the transient event and, thus, a tighter tolerance should be used. In case the algorithm reaches the maximally allowed number of time step repetitions, the simulation is aborted with an error. It is advised to rerun the simulation with a decreased tolerance $\varepsilon_l^{\Delta t}$ and a decreased upper limit Δt_{\max} of the time step size. Usually, allowing for five repetitions of a single time step is sufficient even in very demanding situations, cf. numerical examples in Section 5.

4.3.4. Choice of Error Tolerances

The tolerance for the local discretization error is usually related to a characteristic quantity $(\bullet)_{char}$ of the solution, i.e. a characteristic velocity magnitude $u_{char}^{\mathcal{F}}$ of the fluid flow field or a characteristic displacement magnitude d_{char}^S of the solid body, by a relative tolerance $\varepsilon_{rel}^{\Delta t}$. The tolerance for the local error is then given as

$$\varepsilon_l^{\Delta t} = \varepsilon_{rel}^{\Delta t} (\bullet)_{char}$$

with $(\bullet)_{char}$ being the characteristic quantity of the respective field. Typically, values of $\varepsilon_{rel}^{\Delta t}$ are in the range of $10^{-3} - 10^{-4}$. When choosing the tolerance $\varepsilon_l^{\Delta t}$ for the local error somewhat smaller than the desired tolerance for the global error, the global error will be comparable to the tolerance of the local error [28, 64].

4.3.5. Handling of Convergence Issues of the Nonlinear Solver

In the presence of an exceptionally strong nonlinearity, the nonlinear solver might fail to converge. In such an event, the solution at time t_{n+1} is not available and, thus, cannot be assessed by means of error estimation. To avoid abortion of the simulation in such cases, the current time step is repeated with an heuristically reduced time step size $\kappa_{nl}\Delta t_n$ with the user-given factor $\kappa_{nl} \in]0, 1[$ usually chosen in the range of 0.5 – 0.8. As soon as the nonlinear solver converged successfully with the decreased time step size, further increase or decrease of Δt_n is again based on error estimation.

It is stressed that this strategy does not replace time step size adaptivity based on error estimation. It acts on top of the underlying adaptivity approach and can rather be seen as an *additional* safety measure to avoid breakdown of the simulation.

4.3.6. Computational Efficiency and Savings

The primary reason to apply error estimation-based adaptive time stepping is to control the accuracy of the numerical solution. In practical applications, computational savings often come along with adaptive time stepping. In theory, comparisons of wall clock time between adaptive and non-adaptive simulations provide the actual computational savings, but are infeasible in practice.

Assuming that each time step is associated with roughly the same cost, an alternative and much cheaper efficiency measure can be based on the number of time steps, that can be saved by the adaptive scheme. Therefore, we solve the problem with the adaptive time stepping scheme and a given tolerance for the local error and identify the smallest time step size $\min\{\Delta t_n\}$ that occurred during the simulation. It is then used to compute a virtual number

$$N^{\text{const}} = \frac{T}{\min\{\Delta t_n\}} \quad (10)$$

of time steps that would be necessary in a non-adaptive scheme with the same level of accuracy. Note that the denominator in (10) should only include time step sizes that have been achieved by time step size reduction caused by the algorithm to exclude the small values from the starting phase of the simulation where initially $\Delta t_0 = \Delta t_{\min}$. Now, relative savings in the number of computed time steps δN^{save} are given as

$$\delta N^{\text{save}} = 100\% - \frac{N^{\text{ada}}}{N^{\text{const}}} \quad (11)$$

with N^{ada} and N^{const} being the number of time steps performed by the adaptive and non-adaptive scheme, respectively. To account for the extra effort of possible time step size repetitions, repeated time steps have to be included into N^{ada} .

4.4. Possible Enhancements in Case of Partitioned Solvers

When applying partitioned solution techniques, time discretization of the solid and the fluid field can either be conforming, i.e. Δt is the same in both fields, or subcycling [51, 60, 61] can be employed, where one field uses a *large* time step Δt and the other field uses m *small* time steps of size $\Delta t^{(m)} = \frac{1}{m}\Delta t$ to advance from t_n to $t_{n+1} = t_n + \Delta t$. Both fields exchange information at time instances that coincide with multiples of Δt and use linear interpolations in between.

In the case of a conforming time discretization, the adaptive time stepping scheme can be applied as described above. Furthermore, it can be easily adjusted to subcycling schemes by ensuring $\Delta t_n = \sum_m \Delta t_n^{(m)}$ with $\Delta t_n^{(m)}$ itself being determined by an adaptive procedure.

4.5. Extension to the Coupling of Multiple Physical Models

This manuscript focuses on adaptive time stepping for FSI problems. However, extension to a other classes of coupled problems involving an arbitrary number of physical models or other types of couplings like volume-coupling is straightforward. The time stepping algorithm as sketched in Figure 2 is still valid. Assuming the existence of an error estimator for each physical field, time step size suggestions for each field can be computed via (4), while coupling effects should be included into the error computations and the time step size calculations. An overall time step size for the entire coupled problem is then given by (8).

5. Numerical Examples

Three numerical examples are used to demonstrate the accuracy and the efficiency of the proposed method. They range from academic benchmark tests to complex biomedical applications. In all the examples, the user-given level of accuracy could be achieved, while computational cost was reduced tremendously compared to the case with non-adaptive time stepping.

5.1. An Elastic Wall in a Channel Flow

To analyze the behavior of the fluid error estimator as well as its influence on the evolution of the time step size, an elastic wall is put into a channel flow. Figure 3 depicts the domain of interest by means of different views including all geometric features and boundary conditions. The solid is modelled with a NEO-HOOKEAN material with YOUNG'S modulus $E^{\text{S}} = 500.0$, Poisson's ratio $\nu^{\text{S}} = 0.0$, and density $\rho^{\text{S}} = 1.0$. The fluid is assumed to be NEWTONIAN with dynamic viscosity $\mu_{\text{dyn}}^{\text{F}} = 0.01$ and density $\rho^{\text{F}} = 1.0$. At $x = 0$, the x -component $\bar{u}_x^{\text{F}}(y, z, t)$ of the inflow velocity follows the parabolic profile

$$\bar{u}_x^{\text{F}}(y, z, t) = \hat{u}^{\text{F}}(t) \left(1 - \frac{4y^2}{(h^{\text{F}})^2}\right) \left(1 - \frac{4z^2}{(b^{\text{F}})^2}\right)$$

with the time-dependent peak value

$$\hat{u}^{\text{F}}(t) = \begin{cases} \hat{u}^{\text{F}} \cdot 0.5 \left(1 - \cos\left(\frac{\pi}{6}t\right)\right) & \text{if } 0 < t \leq 6 \\ \hat{u}^{\text{F}} & \text{if } 6 < t \leq 20 \\ \hat{u}^{\text{F}} \left(1 + \cos\left(\frac{\pi}{10}(t - 20)\right)\right) & \text{if } 20 < t \leq 60 \\ \hat{u}^{\text{F}} & \text{if } 60 < t \leq 80 \\ \hat{u}^{\text{F}} \left(1 + \cos\left(\frac{\pi}{10}(t - 80)\right)\right) & \text{if } 80 < t \leq 120 \end{cases}$$

with $\hat{u}^{\text{F}} = 0.1$, cf. Figure 4(a). The components in y - and z -direction are set to zero. Top and bottom walls as well as lateral walls are subject to no-slip boundary conditions. At the outflow area, a zero-traction boundary condition is applied. Due to the symmetry of the problem, only the portion $z \leq 0$ of the channel is modelled. Appropriate symmetry conditions are applied to the xy -plane.

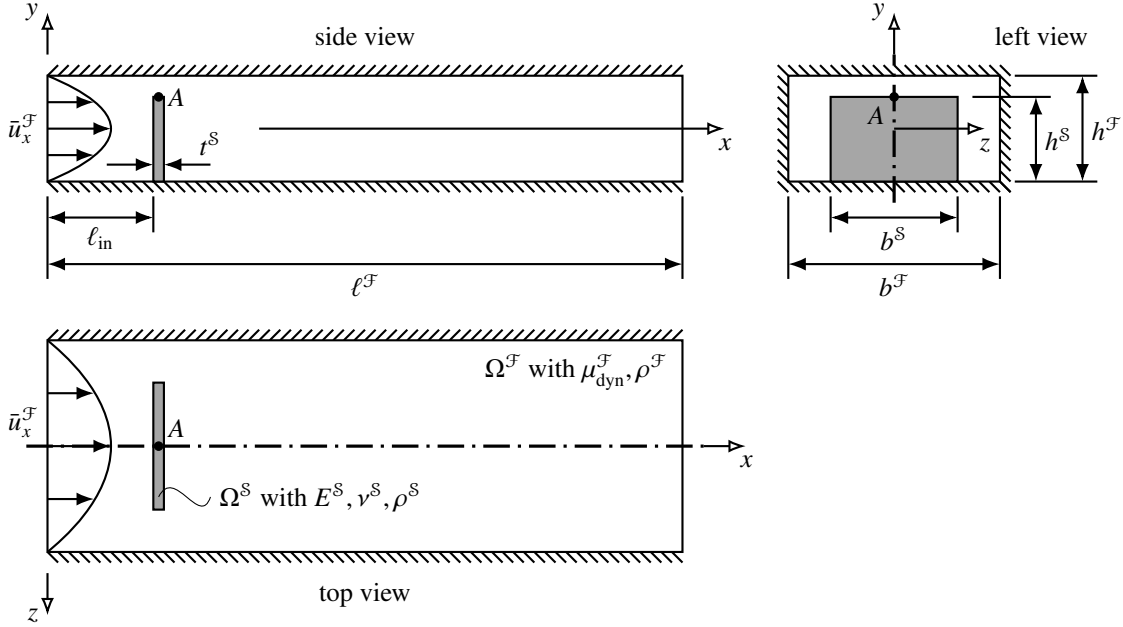


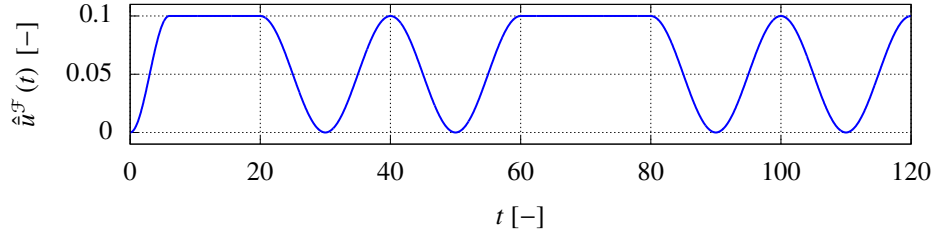
Figure 3: Geometry and boundary conditions of an elastic wall in a channel flow — An elastic wall Ω^S ($b^S = 0.6, h^S = 0.4, t^S = 0.05$) is put into a fluid channel Ω^F ($l^F = 3.0, b^F = 1.0, h^F = 0.5$) at $\ell_{in} = 0.5$ and clamped at its bottom. A time-dependent parabolic inflow profile is prescribed, while the outlet is traction-free. The channel walls are subject to no-slip boundary conditions. The top, center point of the elastic wall is denoted by A .

The structural domain is discretized with 2125 nodes grouped to $4 \times 24 \times 16$ Hex8 F-Bar finite elements [39], the fluid domain with 26017 nodes forming 22704 equal-order interpolated Hex8 finite elements using residual-based stabilization as detailed in Section 2.2. We note that the application of the F-bar finite element technology is not necessary in this example due to Poisson's ratio $\nu^S = 0$, but it does not add significant computational effort and does not alter the results in the context of time step size adaptivity either.

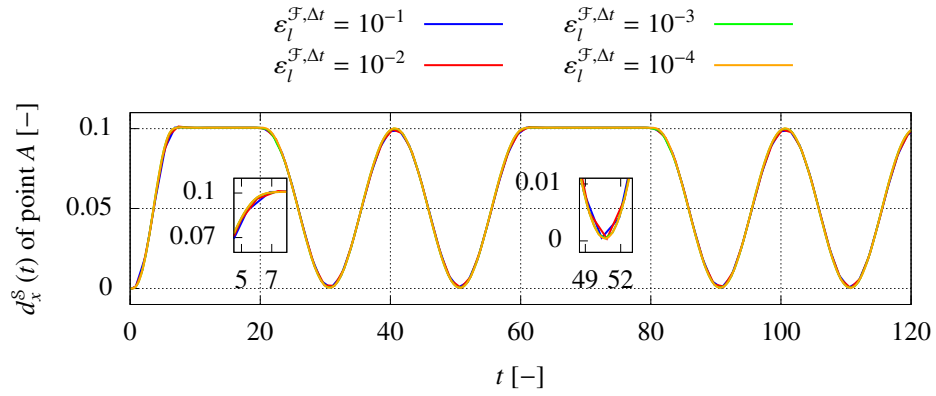
Temporal discretization is done with generalized- α time integration in structure and fluid field. The spectral radii are chosen as $\rho_\infty^S = 0.8$ and $\rho_\infty^F = 0.5$. Different tolerances $\varepsilon_l^{\mathcal{F}, \Delta t} \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ are used for adapting the time step size based on the estimation of the temporal discretization error in the fluid field, which is performed by comparison of the implicit solution to an explicit one obtained with an ADAMS-BASHFORTH-2 scheme. The setup of the example is such that the transient behavior of solid and fluid field coincide, allowing to estimate the error in the fluid field only with without expecting changes to the results. Furthermore, the solid field is not included into the error estimation to demonstrate the flexibility and possibility of user-given configurations of the proposed algorithm. The remaining algorithmic parameters required to evaluate (4) are globally chosen as follows: $\kappa_{min} = 1/3, \kappa_{max} = 3.0, \kappa_s = 0.9, \Delta t_{min} = 10^{-5}, \Delta t_{max} = 1.5$. The number of time step repetitions is limited to five. No averaging of increasing time step sizes is applied.

The x -displacement of the center point A at the top of the elastic wall, cf. Figure 3, is depicted in Figure 4(b). Due to the low inflow velocity, the flow does not detach from the solid and, thus, the solid deformation is able to follow the prescribed inflow velocity shown in Figure 4(a) very closely. Figure 5 reports a snapshot at the state of maximum deflection at $t = 16.7$, which is within the first stationary period of the inflow velocity, cf. Figure 4(a).

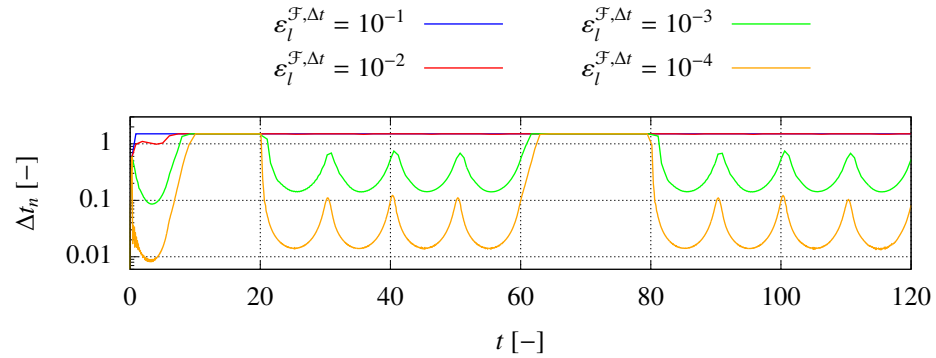
The evolution of the time step size Δt_n over time for all chosen tolerances is reported in Figure 4(c). In case of the loosest tolerance $\varepsilon_l^{\mathcal{F}, \Delta t} = 10^{-1}$, the time step size starts with the initial value $\Delta t_0 = \Delta t_{min}$ and increases monotonically until it reaches the maximum bound Δt_{max} . Since the demanded accuracy is quite low, the simulation does not require any reduction of Δt_n throughout the entire simulation. For $\varepsilon_l^{\mathcal{F}, \Delta t} = 10^{-2}$, a minor reduction is necessary only during the initial phase of the simulation. During the transient phases, the time step size needs to be reduced slightly once in a while, taking the smallest value $\Delta t_n = 1.47161 < \Delta t_{max} = 1.5$, which is a reduction by less than 2%. Remarkable



(a) Prescribed inflow velocity — Starting with a value of zero, the inflow velocity is smoothly increased to its maximum value $\hat{u}^{\mathcal{F}} = 0.1$. Portions with constant and varying inflow velocity alternate in order to trigger the adaptive time stepping algorithm.



(b) Tip displacement of the elastic wall — The deformation of the elastic wall follows the prescribed inflow velocity very closely. Results differ only slightly for different tolerances, cf. zooms around $t = 6$ and $t = 50$. The location of the monitored point A is specified in Figure 3.



(c) Evolution of time step size Δt_n — Four different tolerances $\varepsilon_l^{\mathcal{F},\Delta t}$ are used. The choice $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-1}$ results in an immediate increase of Δt_n to Δt_{\max} and uses that value throughout the entire simulation. Lower tolerances use Δt_{\max} only during the stationary phases, while they require a reduction of Δt_n in transient phases. In general, cases with a tighter tolerance result in smaller time step sizes as expected.

Figure 4: The prescribed inflow velocity, the resulting tip displacement of point A, and the evolution of the time step size Δt_n for the elastic wall in a channel flow correspond to each other very well.

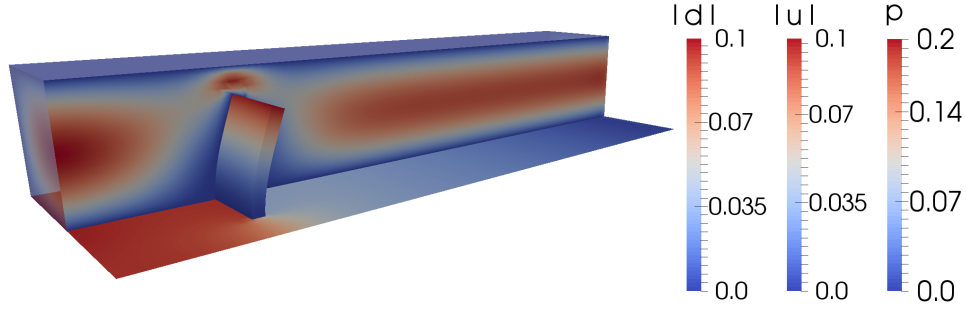


Figure 5: State of maximum deformation of an elastic wall in a channel flow — The solution obtained with $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-4}$ is depicted during a phase of constant inflow velocity at time $t = 16.7$. The elastic wall is colored according to its displacement magnitude denoted by $|d|$. The vertical fluid plane shows a contour plot of the velocity magnitude denoted by $|u|$, while the bottom plane illustrates the pressure field denoted by p .

Table 2: Computational savings for an elastic wall in a channel flow — Based on N^{ada} , the number of time steps of the adaptive simulation, computational savings δN^{save} are estimated using (11). Computational savings are more pronounced in case of smaller tolerances.

$\varepsilon_l^{\mathcal{F},\Delta t}$	N^{ada}	N^{const}	δN^{save}
10^{-1}	91	80	-13.8%
10^{-2}	93	122	23.8%
10^{-3}	456	1400	67.4%
10^{-4}	4225	14116	70.1%

changes of the time step size occur for $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-3}$ and $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-4}$. During the stationary phases, the time step size approaches the maximum bound $\Delta t_{\text{max}} = 1.5$, but during the transient inflow the time step size is reduced massively. As expected, lower values for Δt_n are obtained for the lower tolerance $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-4}$. Common to the cases $\varepsilon_l^{\mathcal{F},\Delta t} = \{10^{-2}, 10^{-3}, 10^{-4}\}$ is the fact, that the minimal time step size resulting from time step size reductions occurs in the initial phase of the simulation. As the simulation starts with a zero initial flow field, the increasing inflow velocity needs to be propagated through the entire domain in the initial phase of the simulation. Later, when the flow field has been established and just needs to change its velocity according to the prescribed inflow velocity, the time step size reduction to slightly larger values is sufficient to achieve the desired level of accuracy.

The maximally allowed number of repetitions of time steps due to violations of the accuracy demand (2) is limited to five. No repetitions occur for the cases $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-1}$ and $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-2}$, where no or only very little reductions of Δt_n are necessary. For the smaller tolerances $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-3}$ and $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-4}$, the number of actual time step repetitions during reductions of Δt_n ranges between 1 and 4. The occurrence of time step repetitions coincides with massive reductions of the time step size Δt_n at times $t \in \{0, 20, 30, 40, 50, 80, 90, 100, 110\}$, cf. Figure 4(c). More repetitions are necessary for $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-4}$, since the time step size needs to be reduced by two orders of magnitude instead of only one for $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-3}$.

Computational savings δN^{save} in terms of a reduced number of time steps to be computed are reported in Table 2. Obviously, the computational savings δN^{save} as defined in (11) depend on the tolerance $\varepsilon_l^{\mathcal{F},\Delta t}$. If a very accurate solution is required, i.e. a low tolerance $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-4}$ is chosen, the adaptive time stepping enables huge computational savings up to $\delta N^{\text{save}} = 70.1\%$ by increasing the time step size during the stationary phases. As already seen in the evolution of the time step size, cf. Figure 4(c), the quite loose tolerance $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-2}$ does not lead to significant reductions of the time step size Δt_n . Only a tiny reduction is required during the starting phase of the simulation. Hence, computational savings are rather small for this case. Only for the loosest choice $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-1}$, the adaptive time stepping seems to be inefficient, since it needs more time steps. This is only due to the initial phase of the simulation, which is necessary to increase the time step from its initial value $\Delta t_0 = \Delta t_{\text{min}}$ up to a suitable value. For the loosest tolerance, this suitable value is limited by Δt_{max} . Since the time step size is never decreased throughout the simulation, the additional steps

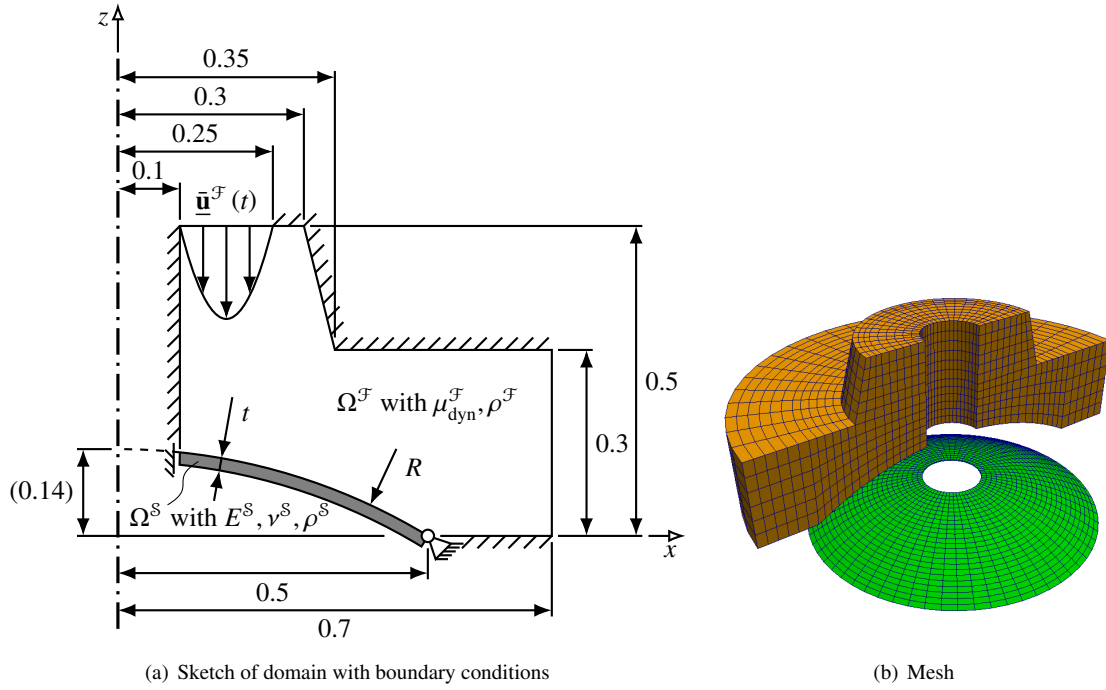


Figure 6: Geometry and boundary conditions of gasket-like example — *Left*: The fluid domain $\Omega^{\mathcal{F}}$ is enclosed by the inflow boundary with the prescribed inflow velocity profile $\hat{\mathbf{u}}^{\mathcal{F}}(t)$, the outflow boundary with a zero-traction boundary condition, walls with no-slip conditions and the fluid-structure interface. The structural domain $\Omega^{\mathcal{S}}$ is a segment of a thin-walled sphere. It can slide along z -direction at its inner radius and rotate around its circumferential direction at its outer radius. The solid shell is characterized by its radius $R = 0.96$ and its thickness $t = 0.0032$ resulting in a slenderness ratio $R/t = 300$. The problem exhibits rotational symmetry w.r.t. the z -axis. *Right*: Purely hexahedral structured mesh.

from the initial increase of the time step size are seen as additional cost by (11), resulting in additional 13.8% time steps to be computed.

5.2. Snap-Through of a Gasket

Now, error estimation in both the fluid and the structure field are used to adapt the time step size in the simulation of the snap-through of a gasket. The geometry mimics a machine part with rotational symmetry. It is inspired by an example in [65]. A detailed sketch including all geometric dimensions can be found in Figure 6. At the top, fluid inflow is prescribed, while fluid outflow is in radial direction. Moreover, the fluid domain is confined by a rigid casing and a thin-walled solid cap mimicking a rubber-like gasket. At the walls of the rigid casing and at the FSI interface, a no-slip condition is assumed. The thin-walled spherical solid shell can slide along the z -direction at its inner radius, while it can rotate around the circumferential direction at its outer radius. The solid is modelled with a Neo-Hookean material with Young's modulus $E^{\mathcal{S}} = 10^5$, Poisson's ratio $\nu^{\mathcal{S}} = 0.3$, and density $\rho^{\mathcal{S}} = 100$. The fluid is assumed to be Newtonian with dynamic viscosity $\mu_{\text{dyn}}^{\mathcal{F}} = 0.1$ and density $\rho^{\mathcal{F}} = 100.0$. A time-dependent parabolic inflow profile $\hat{\mathbf{u}}^{\mathcal{F}}(x, y, z, t)$ is prescribed. Its x - and y -components are set to zero, while its z -component follows the spatial profile

$$\hat{u}_z^{\mathcal{F}}(x, y) = -\frac{4}{(0.1 - 0.25)^2} \left(\sqrt{x^2 + y^2} - 0.1 \right) \left(\sqrt{x^2 + y^2} - 0.25 \right) \hat{u}^{\mathcal{F}}(t)$$

Table 3: Mesh overview and computational resources for the gasket-like example — Two meshes are used, a *coarse* and a *fine* one. Number of unknowns per field as well as total number of unknowns are listed. Computational resources are detailed.

Mesh ID	$n^{\mathcal{S},\text{dof}}$	$n^{\mathcal{F},\text{dof}}$	$n^{\mathcal{S},\text{dof}}$	$n_{\text{total}}^{\text{dof}}$	n^{proc}	$n^{\text{dof}}/\text{core}$
<i>coarse</i>	9600	41088	30816	81504	8	10188
<i>fine</i>	37632	302208	226656	566496	64	8851.5

with the time-dependent peak value

$$\hat{u}^{\mathcal{F}}(t) = \begin{cases} 0.1 \cdot \hat{u}^{\mathcal{F}} \cdot \frac{1}{2} (1 - \cos \frac{\pi t}{2}) & \text{if } 0 < t \leq 2 \\ 0.1 \cdot \hat{u}^{\mathcal{F}} & \text{if } 2 < t \leq 5 \\ \hat{u}^{\mathcal{F}} (0.1 + 0.9 \cdot \frac{1}{2} (1 - \cos \pi(t - 2))) & \text{if } 5 < t \leq 6 \\ \hat{u}^{\mathcal{F}} & \text{if } t > 6 \end{cases}$$

with $\hat{u}^{\mathcal{F}} = 1.0$. A zero-traction boundary condition is prescribed at the fluid outflow where an additional NEUMANN *inflow boundary condition* [66] is applied to safely account for possible backflow. Starting from an initially resting configuration, the inflow velocity is smoothly increased to $\hat{u}^{\mathcal{F}}(t = 2) = 0.1$ and then kept constant until $t = 5$, such that a stationary state can be reached. Under these flow conditions, the load exerted onto the structure is quite small, such that the structure’s deformations remain small and the shell is able to maintain its shape. At $t = 5$, the inflow velocity is smoothly, but rapidly increased to $\hat{u}^{\mathcal{F}}(t = 6) = 1.0$ and then kept constant again. This increase of the fluid inflow velocity results in an increase of the load onto the structure above a critical value, such that the thin-walled solid will start buckling. The solid will snap through and finally find a stable equilibrium configuration, such that a stationary flow field can be re-established.

The solid field is discretized with Hex8 finite elements with *enhanced assumed strains (EAS)*, while the fluid uses equal-order interpolated linear finite elements with residual-based stabilization as detailed in Section 2.2. To demonstrate mesh independency of the adaptive time stepping scheme, two structured and purely hexahedral meshes are used, namely a *coarse* and a *fine* one. The *coarse* one is depicted in Figure 6(b). Details on the number of unknowns per field and in total are given in Table 3. Simulations have been carried out on the *Xeon* partition² of an in-house cluster, where 8 and 64 cores have been used for the *coarse* and the *fine* mesh, respectively.

Both the fluid and the structure field use generalized- α time integration schemes with spectral radii chosen as $\rho_{\infty}^{\mathcal{F}} = 0.5$ and $\rho_{\infty}^{\mathcal{S}} = 0.8$, respectively. Both fields contribute to the error estimation and the time step size adaption. In the fluid field, comparison to an auxiliary ADAMS-BASHFORTH-2 scheme with a tolerance $\varepsilon_l^{\mathcal{F},\Delta t}$ is used, while the structure field employs the ZIENKIEWICZ-XIE error estimator [59] with a tolerance $\varepsilon_l^{\mathcal{S},\Delta t}$. The solid’s local error tolerance is varied as $\varepsilon_l^{\mathcal{S},\Delta t} \in \{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\}$, while the fluid’s one is chosen as $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-3}$. The remaining algorithmic parameters required to evaluate (4) are globally chosen as follows: $\kappa_{\min} = 0.1$, $\kappa_{\max} = 2.0$, $\kappa_s = 0.9$, $\Delta t_{\min} = 10^{-4}$, $\Delta t_{\max} = 0.2$. The number of time step repetitions is limited to five. In case of an increasing time step size, $\gamma_{n+1} = 0.3$ and $\gamma_n = 0.7$ are used as weights for the averaging procedure (9) based on experience from other examples [38].

Figure 7 shows a series of snapshots to illustrate the transient solution. Displacement and velocity of a node on the inner radius of the solid are depicted in Figure 8 for the *fine* mesh and tolerances $\varepsilon_l^{\mathcal{S},\Delta t} = 5 \cdot 10^{-4}$ and $\varepsilon_l^{\mathcal{F},\Delta t} = 10^{-3}$. Until $t = 5$, the spherical shell is able to maintain an equilibrium state close to its initial configuration, i.e. displacement and velocity are close to zero. After increasing the inflow velocity at $t = 5$, the fluid load onto the structure exceeds a critical value, such that the shell cannot maintain its shape and starts buckling. Hence, large displacements and velocities occur while the shell is transformed to another equilibrium configuration. Due to the viscous damping of the fluid and since the inflow velocity is kept constant for $t \geq 6$, oscillations are damped and a stationary flow field can be established. In this state, the solid maintains a *stable* equilibrium configuration, i.e. the displacement is constant and the velocity vanishes. We note, that the transient behaviour in the present scenario is less dramatic in comparison to buckling phenomena in pure structural mechanics due to the interaction of the shell with the fluid field.

²Intel Xeon E5-2670 SandyBridge, 2 octocore CPUs per node, nominal frequency 2.6 GHz, 32 GB memory per node, Infiniband network Mellanox ConnexX, 40 GBit/s

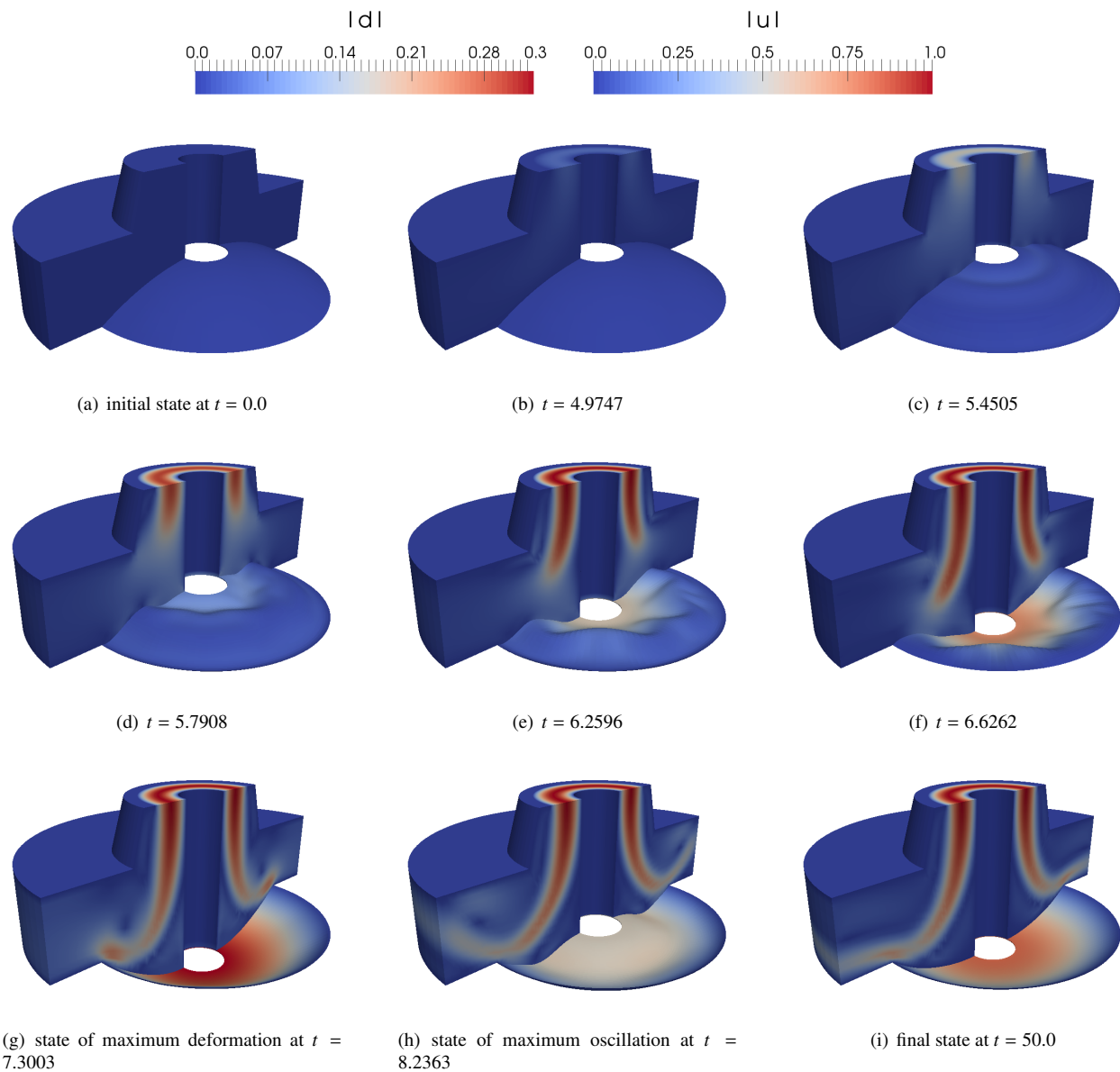


Figure 7: Snapshots of the solution of a gasket at selected time steps using the *fine* mesh — The fluid field is colored according to the magnitude of its velocity field denoted by $|u|$ while the structure field's color is encoded with the displacement field magnitude denoted by $|d|$. Color scales are calibrated at maximum deformation at $t = 7.30$. Simulation has been performed with $\varepsilon_l^{S,\Delta t} = 10^{-3}$ and $\varepsilon_l^{F,\Delta t} = 10^{-3}$.

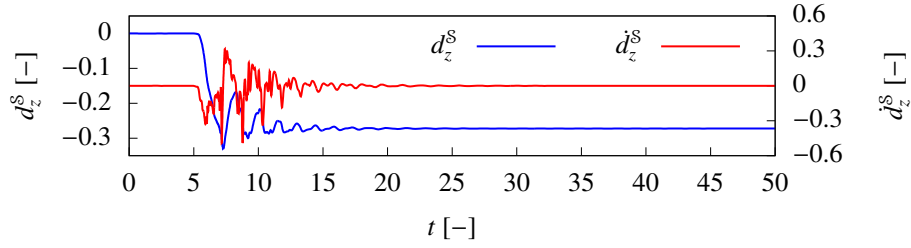


Figure 8: Displacement and velocity in the gasket-like example — Starting from resting conditions, large amplitudes in displacement and velocity occur as soon as the buckling starts. Afterwards, the solid settles in a stable equilibrium configuration. Results are obtained with $\varepsilon_l^{S,\Delta t} = 5 \cdot 10^{-4}$ and $\varepsilon_l^{F,\Delta t} = 10^{-3}$.

Figure 9 reports the evolution of time step size for both meshes and all tolerances. Both the coarse and the fine mesh show a similar behavior regarding the evolution of the time step size. In the initial phase, i.e. prior to $t = 5$, the time step size approaches its maximum bound $\Delta t_{\max} = 0.2$. Only the most accurate case with $\varepsilon_l^{S,\Delta t} = 10^{-4}$ requires a slight reduction of Δt_n during this initial phase, all coarser tolerances allow for a monotonic increase. As the buckling starts, all choices for $\varepsilon_l^{S,\Delta t}$ require a massive reduction of the time step size. As the final steady state is approached, the time step size is increased until it reaches Δt_{\max} for all $\varepsilon_l^{S,\Delta t}$. As expected, smaller tolerances result in smaller values of the time step size Δt_n . Time step repetitions are only performed at the beginning of the buckling event, where some steps require one or two repetitions. All other portions of the simulation get along without time step repetitions.

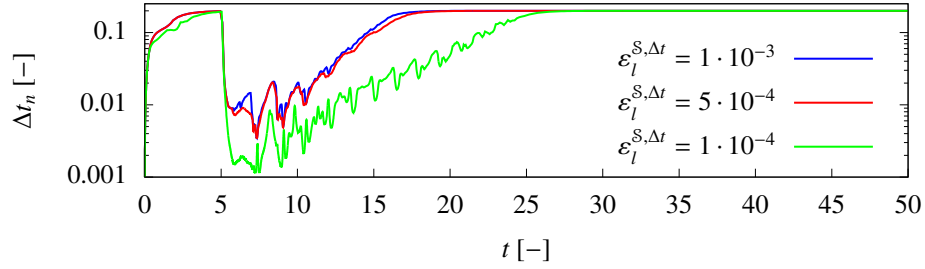
The influence of the spatial discretization on the evolution of the time step size is studied as well. Figure 10 reports the evolution of Δt_n over t for the setting $\varepsilon_l^{S,\Delta t} = 5 \cdot 10^{-4}$ and $\varepsilon_l^{F,\Delta t} = 10^{-3}$ for the coarse and the fine mesh. Both curves deviate only very little from each other. Due to the different spatial refinements, both meshes results in different buckling patterns, while the finer mesh includes more localized buckling phenomena. Thus, more localized changes of the dynamic behavior occur during the simulation resulting in slightly smaller time step sizes during the phase of increasing time step sizes $t \in [10, 25]$. Besides these local phenomena, the spatial discretization does not influence the adaptive time stepping as expected.

5.3. Pulsatile Blood Flow through a Patient-specific Abdominal Aortic Aneurysm

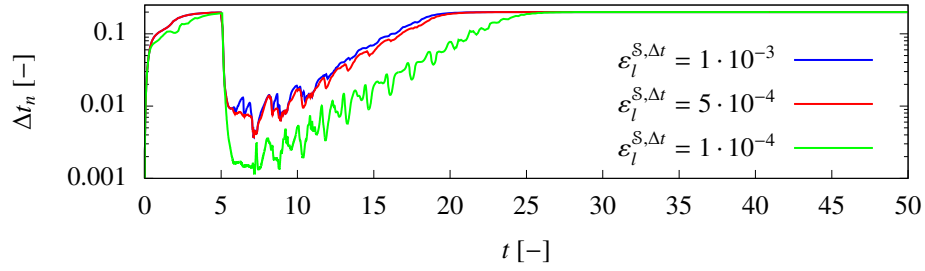
To demonstrate the applicability of the proposed method to complex examples, the pulsatile blood flow through a patient-specific *Abdominal Aortic Aneurysm* (AAA) with an outer diameter of 7.5 cm is considered. Model generation starts from *computer tomography* (CT) images and follows the procedure outlined in [67], whereas meshing is done with Cubit V13.2. The solid phase consists of the *intraluminal thrombus* (ILT) and the *arterial wall*, while the fluid domain covers the *lumen*. The reconstructed geometry is depicted in Figure 11.

The arterial wall utilizes the hyperelastic material model by RAGHAVAN and VORP [68] with constitutive parameters $\alpha_{\text{wall}}^S = 0.174$ MPa, $\beta_{\text{wall}}^S = 1.881$ MPa, and POISSON'S ratio $\nu_{\text{wall}}^S = 0.49$. The ILT employs a coupled form of the compressible NEO-HOOKEAN model [69] with the YOUNG'S modulus $E_{\text{ilt}}^S = 0.1044$ MPa and POISSON'S ratio $\nu_{\text{ilt}}^S = 0.45$. Density of both arterial wall and ILT is chosen as $\rho^S = 1.0$ g/cm³. All values of the solid's constitutive parameters are taken from [67]. Blood uses the modelling assumption of a NEWTONIAN fluid with dynamic viscosity $\mu_{\text{dyn}}^F = 4 \cdot 10^{-3}$ Pa · s and density $\rho^F = 1.0$ g/cm³ [70].

ILT and arterial wall are clamped at inflow and outflow cross sections. A visco-elastic embedding is used to mimic the surrounding tissue. It is realized with springs and dashpots attached normal to the wall surface in material configuration. Spring stiffness and dashpot viscosity are chosen uniformly as $k_{\text{emb}} = 10$ g/(mm² · s²) and $c_{\text{emb}} = 100$ g/(mm² · s), respectively, taken from [71]. At the fluid inflow area $A_{\text{in}}^F = 267,018$ mm², a time-dependent parabolic velocity profile is prescribed. The time curve is based on measurement data [72] and results in a flow rate $Q_{\text{in}} = 3.01$ /min, which seems to be in the physiological range. It is depicted in Figure 12. At the beginning of the heart cycle, the inflow velocity is immediately increased to its peak value at time $t \approx 0.09$ s. In this short period of time, almost all transport of blood volume happens. This peak is followed by a brief plateau at low inflow velocities. Then, the inflow



(a) Coarse mesh



(b) Fine mesh

Figure 9: Evolution of the time step size Δt_n for the gasket-like example — The time step size is increased in the initial phase of the simulation. When the buckling starts, a massive reduction of Δt_n is performed, while it can increase again towards the end of the simulation. Behavior for the coarse mesh (*top*) and the fine mesh (*bottom*) are similar.

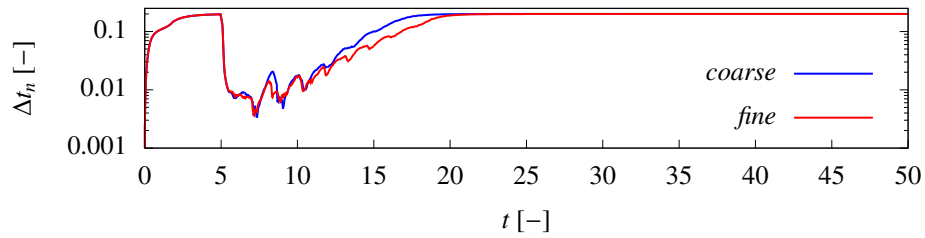


Figure 10: Influence of the mesh on the evolution of the time step size Δt_n for the gasket-like example — The time step size evolves very similarly for both meshes. The finer mesh sometimes requires slightly smaller Δt_n due to localized dynamic effects that come along with the better resolved buckling patterns.

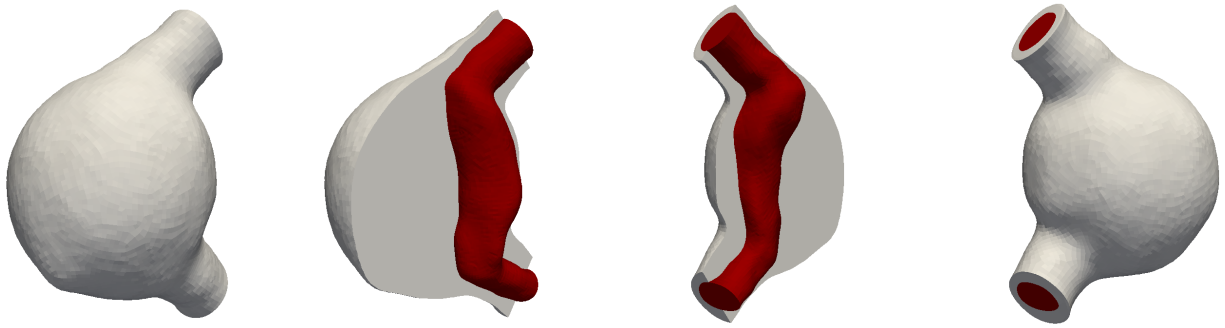


Figure 11: Geometry of a patient-specific AAA — The solid phase consists of the *intraluminal thrombus (ILT)* and the *arterial wall*. It is colored in gray. The fluid domain covers the *lumen* and is colored in red. Pulsatile fluid inflow is prescribed at the top, while the outflow boundary is at the bottom.

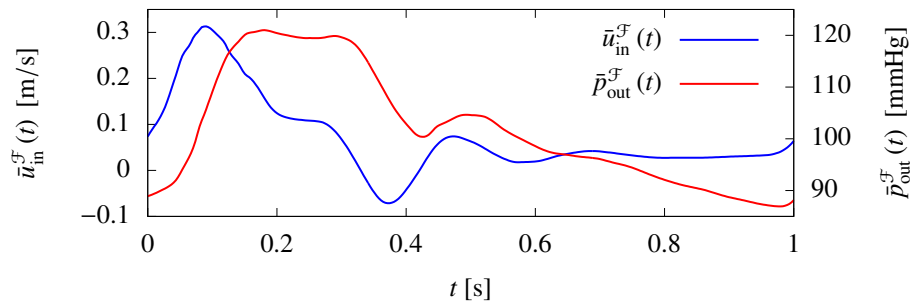


Figure 12: Time curves for physiological fluid boundary conditions of AAA— The inflow velocity time curve $\bar{u}_{in}^F(t)$ is based on measurement results published in [72] and on the computational example in [70]. The temporal evolution $\bar{p}_{out}^F(t)$ of the pressure level at the outflow boundary is analog to numerical results in [70]. Both curves are periodic in time with a cycle time of $T_c = 1.0$ s.

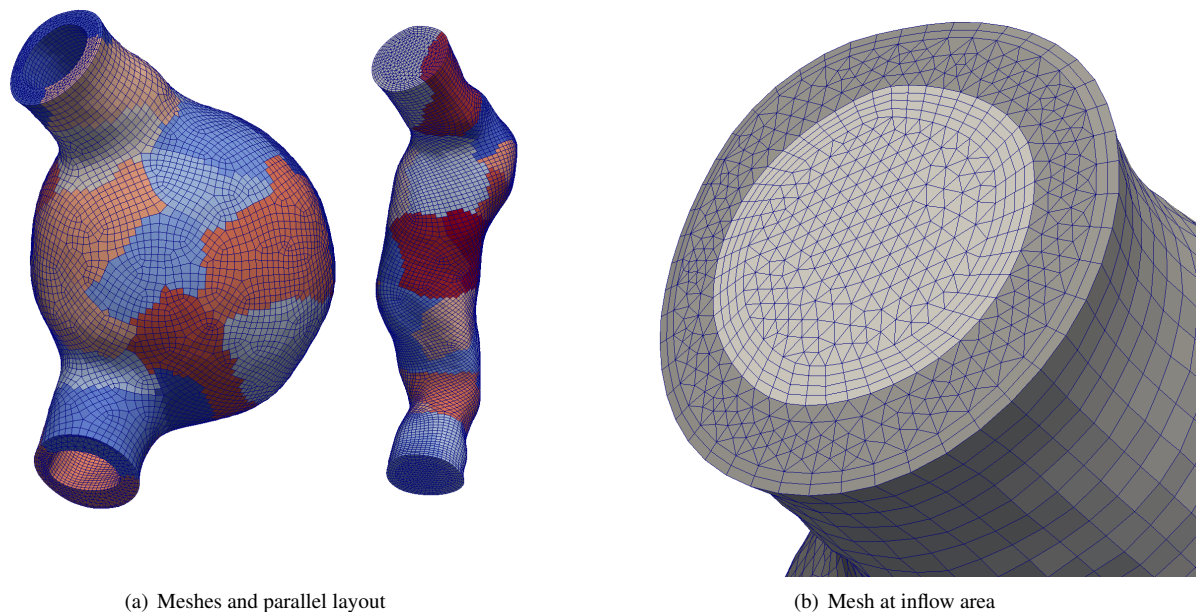


Figure 13: Mesh of a patient-specific AAA — *Left*: Meshes of solid and fluid domain are shown. Colors indicate the domain decomposition among 28 processors. *Right*: Close-up of the mesh at the inflow.

direction is even reverted during a short period of prescribed backflow with its peak at time $t \approx 0.37$ s. Starting with approximately $t > 0.6$ s only very low and constant inflow velocities are prescribed. Although the downstream portion of the vascular system is not resolved geometrically, it is included into the model via a traction boundary condition at the fluid outflow area. The prescribed time-dependent traction value varies between the diastolic level $p_{\text{dia}}^{\mathcal{F}} = 86.9$ mmHg and the systolic level $p_{\text{sys}}^{\mathcal{F}} = 121$ mmHg. Its time curve is based on the results in [70] and is also depicted in Figure 12. Its peak plateau is in the range of approximately $0.15 \text{ s} \leq t \leq 0.31 \text{ s}$, i.e. it is slightly delayed to the velocity curve. Its peak value is located at $t = 0.18$ s. Both velocity and pressure curves are periodic in time with a cycle time of $T_c = 1.0$ s.

To account for the fact, that the stack of CT images has been recorded *in vivo*, i.e. in the presence of blood pressure, a prestressing phase up to the diastolic blood pressure $p_{\text{dia}}^{\mathcal{F}}$ using a *modified updated LAGRANGEAN formulation (MULF)* [73] is prefixed to the actual transient simulation. During this prestressing phase, the fluid inflow and outflow boundary conditions are smoothly increased to their initial values at the beginning of the heart cycle, while the solid is treated with the MULF algorithm.

The mesh is depicted in Figure 13. The FSI coupling surface and wall-sided surface of the *intraluminal thrombus (ILT)* are meshed with quadrilaterals prior to filling ILT and lumen volumina with linear tetrahedra. For the transition of quadrilateral surface meshes to tetrahedral volume meshes, a layer of linear pyramid elements is inserted. The arterial wall as well as the fluid boundary layer are meshed with 8-noded hexahedra only. In the arterial wall, Hex8 F-Bar finite elements [39] are used, while the ILT utilizes displacement-based linear pyramid and tetrahedral finite elements. The fluid is resolved using equal-order interpolated linear finite elements with residual-based stabilization as detailed in Section 2.2. Finally, the discretization consists of 211827 solid, 328548 fluid, and 246411 ALE degrees of freedom, yielding a total number of unknowns of $n_{\text{total}}^{\text{dof}} = 786786$. The problem is ran on 28 cores of SuperMUC's *Phase 2* partition³ at the Leibniz Supercomputing Centre, Garching, Germany.

Time integration is performed by means of the generalized- α method with spectral radii $\rho_{\infty}^{\mathcal{S}} = 0.8$ and $\rho_{\infty}^{\mathcal{F}} = 0.5$

³Haswell Xeon Processor E5-2697 v3, 2 processors with 14 cores each per node, nominal frequency 2.6 GHz, 2.3 GB memory per core, Infiniband FDR14 network, <https://www.lrz.de/services/compute/supermuc/systemdescription/>, visited on June 1st 2016

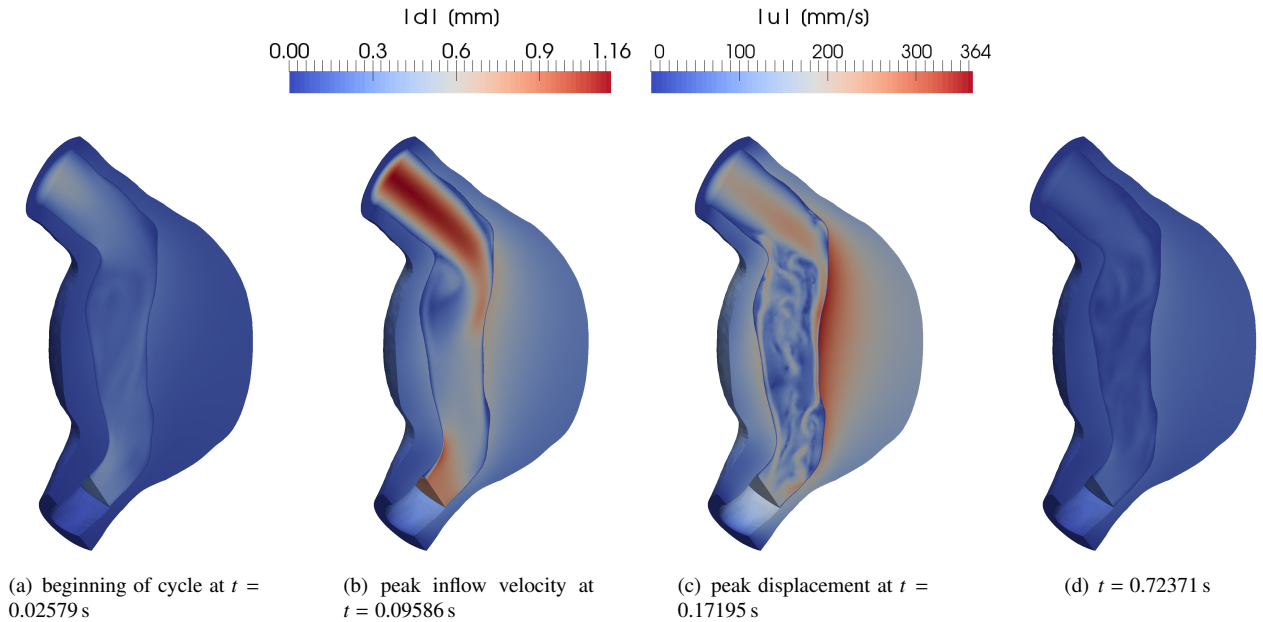


Figure 14: Snapshots of the solution of a pulsatile blood flow through an AAA — The fluid field is colored according to the magnitude of its velocity field denoted by $|u|$ while the structure field's color is encoded with the displacement field magnitude denoted by $|d|$. The fluid color scale is calibrated at maximum inflow velocity at $t = 0.09586$ s, while the solid color scale is based on maximum deformation at $t = 0.17195$ s. The pulsatile motion of ILT and arterial wall is further detailed in Figure 15.

in solid and fluid field, respectively. Adaptive time stepping is applied based on error estimation in the fluid field only, using the comparison to an auxiliary ADAMS–BASHFORTH-2 scheme. The solid is not taken into account for error estimation since only very little deformation is expected. Moreover, dynamics of the problem are solely driven by the fluid flow. The local error tolerance is related to the peak fluid inflow velocity as $\varepsilon_l^{\mathcal{F},\Delta t} = 2.0 \approx 0.5\% \cdot \hat{u}_{in}^{\mathcal{F}}$ as suggested in Section 4.3.4. The time step size is limited to the range $\Delta t_{\min} = 10^{-5}$ s $\leq \Delta t_n \leq 0.1$ s $= \Delta t_{\max}$. The remaining algorithmic parameters related to adaptive time stepping have been chosen as $\kappa_{\min} = 0.1$, $\kappa_{\max} = 2.0$, and $\kappa_s = 0.9$. Averaging of increasing time step sizes (9) is applied using $\gamma_{n+1} = 0.3$ and $\gamma_n = 0.7$. The number of time step repetitions is limited to five.

For preconditioning of the monolithic system of equations, the fully coupled algebraic multigrid preconditioner for FSI problems proposed by GEE *et al.* [2] is applied.

Figure 14 shows cuts through the AAA to illustrate flow patterns and internal deformation. The lumen is cut prior to the outlet for the purpose of visualization only. Starting from an initial state with only little flow and almost no deformation, cf. Figure 14(a), the state with the peak inflow velocity is reached quickly at time $t = 0.09586$ s, cf. Figure 14(b). The prescribed inflow can be seen clearly, which is then deflected by the ILT. Due to a larger diameter of the lumen inside the AAA, velocities are lower there and increase only towards the narrowed outlet. The maximum deformation of ILT and arterial wall is reached at time $t = 0.17195$ s, cf. Figure 14(c), where the largest displacements occur directly at the fluid-structure interface. However, deformation of the wall is visible as well, even if it takes smaller values than in the ILT due to its higher stiffness. Towards the end of the heart cycle, as only a low and almost constant inflow velocity is prescribed, fluid flow in the domain almost vanishes, cf. Figure 14(d). Additionally, only small displacements occur due to the fluid pressure being closer to the diastolic than the systolic pressure level. Overall, a periodic state has been reached.

The pulsatile motion of ILT and arterial wall can be seen in Figure 15, where the absolute displacements of two exemplary solid nodes are related to the prescribed pressure at the outflow cross section. Both nodes are picked at about half length of the aneurysm in regions, where large deformations occur. One is located on the fluid-structure interface and the other one on the outer surface of the arterial wall. Temporal evolution of their displacement magnitudes follows

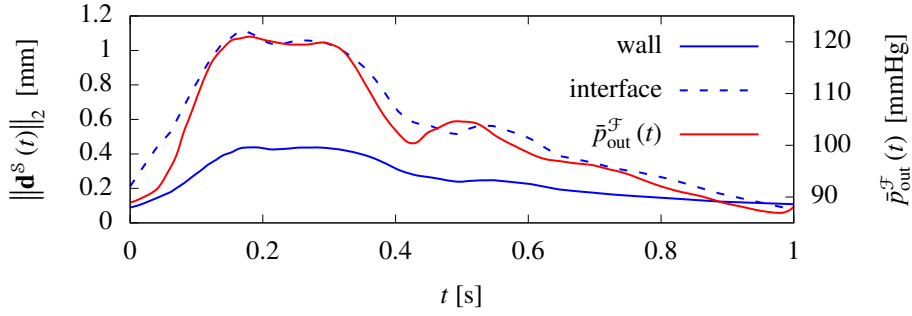


Figure 15: Displacement of and pressure inside the AAA throughout the heart beat

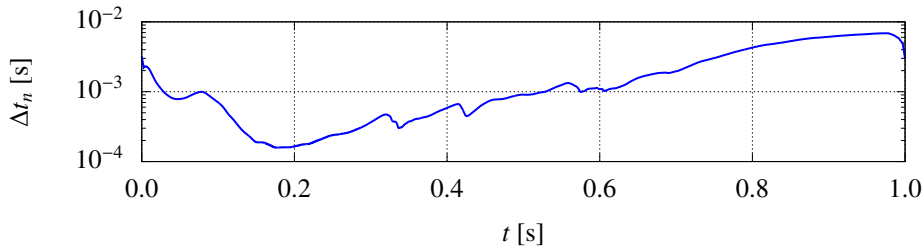


Figure 16: Evolution of time step size Δt_n for pulsatile flow through AAA

the time curve of the prescribed outflow pressure $\bar{p}_{\text{out}}^f(t)$. Solid deformation obviously seems to be governed by the fluid's pressure rather than by the fluid's velocity field. This fact stresses the importance of prescribing physiologically correct outflow boundary conditions on the fluid domain, which in turn requires the application of a prestressing technique in the solid.

The evolution of the times step size Δt_n is reported in Figure 16. A variation of the time step size can be clearly observed. At the beginning of the heart cycle, the time step size is reduced due to the fast velocity changes in the flow field stemming from the peak of the inflow velocity. Towards the end of the heart cycle, i.e. $t > 0.6$ s, a low and almost constant inflow velocity is prescribed. Additionally, dynamic behavior from the initial velocity peak already vanished. Thus, an increase of the time step size by roughly an order of magnitude is possible. Repetition of time steps is necessary only for the last five time steps of the heart cycle where the inflow velocity already increases a little bit, cf. Figure 12. All other reductions of Δt_n can be handled by the safety factor κ_s without the need of time step repetitions. As the beating motion of the heart and, thus, the pulsatile blood flow exhibit periodic behavior, such a periodicity can also be observed for the evolution of the time step size.

To assess computational efficiency, computational savings compared to the case with constant time step size and the same accuracy requirements are quantified using (11). Taking into account the smallest time step size used during the computation, $\Delta t_n = 1.5873 \cdot 10^{-4}$ s, and the number of adaptive time steps $n_{\text{ada}} = 1616$ needed for one heart cycle, the computational savings evaluate to 74.5% which is a tremendous gain in efficiency.

6. Concluding remarks

To provide an efficient and very robust time stepping scheme for transient simulation of FSI problems that can meet application-driven and user-given accuracy requirements, we proposed an *adaptive* time stepping scheme for FSI solvers which is equally applicable to monolithic and partitioned solution schemes. It is based on *a posteriori* error estimation in both the fluid and the structure field as well as at the fluid-structure interface. Based on the estimated error, the time step size is adapted to the current needs of the simulation, which allows to keep the local error below

a user-given tolerance while at the same time limiting the computational cost to the amount necessary. The proposed method is readily extendible to other, possibly more sophisticated error estimators. Furthermore, the extension to other classes of coupled problems has been sketched briefly.

The proposed method has been applied to a variety of numerical examples. In academic benchmarks tests as well as in practical applications in the field of biomechanics, this approach allows for ensuring accuracy of the time-discrete solution. Furthermore, tremendous savings and speed-up compared to the case with non-adaptive time stepping have been demonstrated. On the one hand, the number of time step sizes to be computed has been reduced significantly, which yields huge savings in terms of wall clock time. On the other hand, time-consuming trial runs of the simulations to find an appropriate time step size have not been necessary since the adaptive algorithm finds an optimal time step size by itself. Overall, the proposed method is considered very attractive since it enables the desired accuracy control via efficient error estimation techniques without adding computational effort while simultaneously saving computational cost and strengthening the robustness of the overall simulation procedure.

Acknowledgement

We gratefully acknowledge the support by Sebastian Kehl⁴ during model generation and meshing of the AAA geometry utilized in Section 5.3.

References

- [1] Mayr M, Klöppel T, Wall WA, Gee MW. A Temporal Consistent Monolithic Approach to Fluid–Structure Interaction Enabling Single Field Predictors. *SIAM Journal on Scientific Computing* 2015; **37**(1):B30–B59, doi:10.1137/140953253. URL <http://dx.doi.org/10.1137/140953253>.
- [2] Gee MW, Küttler U, Wall WA. Truly monolithic algebraic multigrid for fluid–structure interaction. *International Journal for Numerical Methods in Engineering* 2011; **85**(8):987–1016. URL <http://dx.doi.org/10.1002/nme.3001>.
- [3] Choi CK, Chung HJ. Error estimates and adaptive time stepping for various direct time integration methods. *Computers & Structures* 1996; **60**(6):923–944, doi:http://dx.doi.org/10.1016/0045-7949(95)00452-1. URL <http://www.sciencedirect.com/science/article/pii/S0045794995004521>.
- [4] Chung J, Cho EH, Choi K. A priori error estimator of the generalized- α method for structural dynamics. *International Journal for Numerical Methods in Engineering* 2003; **57**(4):537–554. URL <http://dx.doi.org/10.1002/nme.688>.
- [5] Hulbert GM, Jang I. Automatic time step control algorithms for structural dynamics. *Computer Methods in Applied Mechanics and Engineering* 1995; **126**(1–2):155–178. URL <http://www.sciencedirect.com/science/article/pii/S004578259500791X>.
- [6] Li XD, Zeng LF, Wiberg NE. A simple local error estimator and an adaptive time-stepping procedure for direct integration method in dynamic analysis. *Communications in Numerical Methods in Engineering* 1993; **9**(4):273–292, doi:10.1002/cnm.1640090402. URL <http://dx.doi.org/10.1002/cnm.1640090402>.
- [7] Li XD, Wiberg NE. Implementation and adaptivity of a space-time finite element method for structural dynamics. *Computer Methods in Applied Mechanics and Engineering* 1998; **156**(1–4):211–229, doi:http://dx.doi.org/10.1016/S0045-7825(97)00207-7. URL <http://www.sciencedirect.com/science/article/pii/S0045782597002077>.
- [8] Park K, Underwood P. A variable-step central difference method for structural dynamics analysis – part 1. Theoretical aspects. *Computer Methods in Applied Mechanics and Engineering* 1980; **22**(2):241–258. URL <http://www.sciencedirect.com/science/article/pii/S0045782580900870>.
- [9] Romero I, Lacoma LM. A methodology for the formulation of error estimators for time integration in linear solid and structural dynamics. *International Journal for Numerical Methods in Engineering* 2006; **66**(4):635–660. URL <http://dx.doi.org/10.1002/nme.1568>.
- [10] Schleupen A, Ramm E. Local and global error estimations in linear structural dynamics. *Computers & Structures* 2000; **76**(6):741–756, doi:http://dx.doi.org/10.1016/S0045-7949(99)00186-8. URL <http://www.sciencedirect.com/science/article/pii/S0045794999001868>.
- [11] Underwood P, Park K. A variable-step central difference method for structural dynamics analysis – part 2. Implementation and performance evaluation. *Computer Methods in Applied Mechanics and Engineering* 1980; **23**(3):259–279. URL <http://www.sciencedirect.com/science/article/pii/S0045782580900092>.

⁴Mechanics & High Performance Computing Group, Technical University of Munich, Parkring 35, D-85748 Garching b. München, Germany

- [12] Wiberg NE, Li XD. A post-processing technique and an a posteriori error estimate for the newmark method in dynamic analysis. *Earthquake Engineering & Structural Dynamics* 1993; **22**(6):465–489, doi:10.1002/eqe.4290220602. URL <http://dx.doi.org/10.1002/eqe.4290220602>.
- [13] Zeng LF, Wiberg NE, Li XD, Xie YM. A posteriori local error estimation and adaptive time-stepping for newmark integration in dynamic analysis. *Earthquake Engineering & Structural Dynamics* 1992; **21**(7):555–571. URL <http://dx.doi.org/10.1002/eqe.4290210701>.
- [14] Donea J, Huerta A. *Finite Element Methods for Flow Problems*. Wiley, 2003.
- [15] Gresho PM, Sani RL. *Incompressible Flow and the Finite Element Method*. Wiley, 2000.
- [16] Gresho PM, Griffiths DF, Silvester DJ. Adaptive time-stepping for incompressible flow. Part I: Scalar Advection-Diffusion. *SIAM Journal on Scientific Computing* 2008; **30**(4):2018–2054, doi:http://dx.doi.org/10.1137/070688018.
- [17] Johnson C. Error Estimates and Adaptive Time-Step Control for a Class of One-Step Methods for Stiff Ordinary Differential Equations. *SIAM Journal on Numerical Analysis* 1988; **25**(4):908–926, doi:10.1137/0725051. URL <http://dx.doi.org/10.1137/0725051>.
- [18] Johnson C, Rannacher R, Boman M. Numerics and Hydrodynamic Stability: Toward Error Control in Computational Fluid Dynamics. *SIAM Journal on Numerical Analysis* 1995; **32**(4):1058–1079, doi:10.1137/0732048. URL <http://dx.doi.org/10.1137/0732048>.
- [19] Kay D, Gresho P, Griffiths DF, Silvester D. Adaptive time-stepping for incompressible flow. Part II: Navier-Stokes equations. *SIAM Journal on Scientific Computing* 2010; **32**(1):111–128. © 2008 Society for Industrial and Applied Mathematics.
- [20] Saleri F, Veneziani A. Pressure Correction Algebraic Splitting Methods for the Incompressible Navier–Stokes Equations. *SIAM Journal on Numerical Analysis* 2005; **43**(1):174–194, doi:10.1137/S0036142903435429. URL <http://dx.doi.org/10.1137/S0036142903435429>.
- [21] Veneziani A, Villa U. ALADINS: An ALgebraic splitting time ADaptive solver for the Incompressible Navier–Stokes equations. *Journal of Computational Physics* 2013; **238**:359–375, doi:http://dx.doi.org/10.1016/j.jcp.2012.11.049. URL <http://www.sciencedirect.com/science/article/pii/S0021999112007310>.
- [22] Boffi D, Gastaldi L. Stability and geometric conservation laws for ALE formulations. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(42–44):4717–4739. URL <http://www.sciencedirect.com/science/article/pii/S0045782504002336>.
- [23] Förster C, Wall WA, Ramm E. On the geometric conservation law in transient flow calculations on deforming domains. *International Journal for Numerical Methods in Fluids* 2006; **50**(12):1369–1379. URL <http://dx.doi.org/10.1002/flid.1093>.
- [24] Formaggia L, Nobile F. A Stability Analysis for the Arbitrary Lagrangian Eulerian Formulation with Finite Elements. *East-West Journal of Numerical Mathematics* 1999; **7**:105–132.
- [25] Formaggia L, Nobile F. Stability analysis of second-order time accurate schemes for ALE–FEM. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(39–41):4097–4116. URL <http://www.sciencedirect.com/science/article/pii/S0045782504002142>.
- [26] Hay A, Yu K, Etienne S, Garon A, Pelletier D. High-order temporal accuracy for 3D finite-element ALE flow simulations. *Computers & Fluids* 2014; **100**:204–217, doi:http://dx.doi.org/10.1016/j.compfluid.2014.04.036. URL <http://www.sciencedirect.com/science/article/pii/S004579301400187X>.
- [27] Butcher JC. *Numerical Methods for Ordinary Differential Equations*. 2nd edn., John Wiley & Sons, Ltd, 2008, doi:10.1002/9780470753767. URL <http://dx.doi.org/10.1002/9780470753767>.
- [28] Deuffhard P, Bornemann F. *Numerische Mathematik 2: Gewöhnliche Differentialgleichungen*, vol. 2. 3rd edn., De Gruyter: Boston, 2008.
- [29] Hairer E, Nørsett SP, Wanner G. *Solving Ordinary Differential Equations I*. 2nd edn., Springer Series in Computational Mathematics, Springer Berlin Heidelberg, 1993, doi:10.1007/978-3-540-78862-1. URL <http://link.springer.com/book/10.1007/978-3-540-78862-1>.
- [30] Hairer E, Wanner G. *Solving Ordinary Differential Equations II*. 2nd edn., Springer Series in Computational Mathematics, Springer Berlin Heidelberg, 1996, doi:10.1007/978-3-642-05221-7. URL <http://link.springer.com/book/10.1007/2F978-3-642-05221-7>.
- [31] Gustafsson K, Lundh M, Söderlind G. A PI Step-size Control for the Numerical Solution of Ordinary Differential Equations. *BIT* 1988; **28**(2):270–287, doi:10.1007/BF01934091. URL <http://dx.doi.org/10.1007/BF01934091>.
- [32] Söderlind G. Automatic Control and Adaptive Time-Stepping. *Numerical Algorithms* 2002; **31**(1–4):281–310, doi:10.1023/A:1021160023092.
- [33] Söderlind G. Time-step selection algorithms: Adaptivity, control, and signal processing. *Applied Numerical Mathematics* 2006; **56**(3–4):488–502. URL <http://www.sciencedirect.com/science/article/pii/S0168927405000954>.
- [34] Cyr E, Shadid J, Wildey T. Towards efficient backward-in-time adjoint computations using data compression techniques. *Computer Methods in Applied Mechanics and Engineering* 2015; **288**:24–44, doi:http://dx.doi.org/10.1016/j.cma.2014.12.001. URL <http://www.sciencedirect.com/science/article/pii/S0045782514004800>.

- [35] Carey V, Estep D, Johansson A, Larson M, Tavener S. Blockwise Adaptivity for Time Dependent Problems Based on Coarse Scale Adjoint Solutions. *SIAM Journal on Scientific Computing* 2010; **32**(4):2121–2145, doi:10.1137/090753826. URL <http://dx.doi.org/10.1137/090753826>.
- [36] Verdugo F, Parés N, Díez P. Modal-based goal-oriented error assessment for timeline-dependent quantities in transient dynamics. *International Journal for Numerical Methods in Engineering* 2013; **95**(8):685–720, doi:10.1002/nme.4538. URL <http://dx.doi.org/10.1002/nme.4538>.
- [37] Verdugo F, Parés N, Díez P. Goal-oriented space-time adaptivity for transient dynamics using a modal description of the adjoint solution. *Computational Mechanics* 2014; **54**(2):331–352, doi:10.1007/s00466-014-0988-2. URL <http://dx.doi.org/10.1007/s00466-014-0988-2>.
- [38] Mayr M. A Monolithic Solver for Fluid-Structure Interaction with Adaptive Time Stepping and a Hybrid Preconditioner. PhD Thesis, Technische Universität München 2016.
- [39] de Souza Neto EA, Perić D, Dutko M, Owen DRJ. Design of simple low order finite elements for large strain analysis of nearly incompressible solids. *International Journal of Solids and Structures* 1996; **33**(20–22):3277–3296, doi:http://dx.doi.org/10.1016/0020-7683(95)00259-6. URL <http://www.sciencedirect.com/science/article/pii/S0020768395002596>.
- [40] Brooks AN, Hughes TJR. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**(1):199–259, doi:http://dx.doi.org/10.1016/0045-7825(82)90071-8. URL <http://www.sciencedirect.com/science/article/pii/S0045782582900718>.
- [41] Hughes TJ, Franca LP, Balestra M. A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: a stable Petrov-Galerkin formulation of the stokes problem accommodating equal-order interpolations. *Computer Methods in Applied Mechanics and Engineering* 1986; **59**(1):85–99, doi:http://dx.doi.org/10.1016/0045-7825(86)90025-3. URL <http://www.sciencedirect.com/science/article/pii/S0045782586900253>.
- [42] de Mulder T. The role of bulk viscosity in stabilized finite element formulations for incompressible flow: A review. *Computer Methods in Applied Mechanics and Engineering* 1998; **163**(1–4):1–10, doi:http://dx.doi.org/10.1016/S0045-7825(98)00015-2. URL <http://www.sciencedirect.com/science/article/pii/S0045782598000152>.
- [43] Gravemeier V, Gee MW, Kronbichler M, Wall WA. An algebraic variational multiscale–multigrid method for large eddy simulation of turbulent flow. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(13–16):853–864, doi:http://dx.doi.org/10.1016/j.cma.2009.05.017. URL <http://www.sciencedirect.com/science/article/pii/S0045782509002059>.
- [44] Klöppel T, Popp A, Küttler U, Wall WA. Fluid–structure interaction for non-conforming interfaces based on a dual mortar formulation. *Computer Methods in Applied Mechanics and Engineering* 2011; **200**(45–46):3111–3126. URL <http://www.sciencedirect.com/science/article/pii/S0045782511002246>.
- [45] Wohlmuth BI. A Mortar Finite Element Method Using Dual Spaces for the Lagrange Multiplier. *SIAM Journal on Numerical Analysis* 2000; **38**(3):989–1012, doi:10.1137/S0036142999350929. URL <http://epubs.siam.org/doi/abs/10.1137/S0036142999350929>.
- [46] Chung J, Hulbert G. A Time Integration Algorithm for Structural Dynamics With Improved Numerical Dissipation: The Generalized- α Method. *Journal of Applied Mechanics* 1993; **60**(2):371–375.
- [47] Jansen KE, Whiting CH, Hulbert GM. A generalized- α method for integrating the filtered Navier–Stokes equations with a stabilized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2000; **190**(3–4):305–319. URL <http://www.sciencedirect.com/science/article/pii/S0045782500002036>.
- [48] Faires JD, Burden RL. *Numerical Methods*. 4th edn., Brooks/Cole: Pacific Grove, CA, USA, 2012.
- [49] Mayr M, Noll MH, Gee MW. Hybrid Additive/Multiplicative Schwarz Preconditioning for Monolithic Solvers for Coupled Multi-Physics Problems. Submitted.
- [50] Farhat C, Lesoinne M, Le Tallec P. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering* 1998; **157**(1–2):95–114. URL <http://www.sciencedirect.com/science/article/pii/S0045782597002168>.
- [51] Farhat C, Lesoinne M. Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems. *Computer Methods in Applied Mechanics and Engineering* 2000; **182**(3–4):499–515, doi:http://dx.doi.org/10.1016/S0045-7825(99)00206-6. URL <http://www.sciencedirect.com/science/article/pii/S0045782599002066>.
- [52] Felippa CA, Park KC. Staggered transient analysis procedures for coupled mechanical systems: Formulation. *Computer Methods in Applied Mechanics and Engineering* 1980; **24**(1):61–111, doi:http://dx.doi.org/10.1016/0045-7825(80)90040-7. URL <http://www.sciencedirect.com/science/article/pii/S0045782580900407>.
- [53] Küttler U, Wall WA. Fixed-point fluid–structure interaction solvers with dynamic relaxation. *Computational Mechanics* 2008; **43**(1):61–72. URL <http://dx.doi.org/10.1007/s00466-008-0255-5>.

- [54] Yu Y, Baek H, Karniadakis GEE. Generalized fictitious methods for fluid–structure interactions: Analysis and simulations. *Journal of Computational Physics* 2013; **245**:317–346, doi:<https://doi.org/10.1016/j.jcp.2013.03.025>.
- [55] Badia S, Nobile F, Vergara C. Fluid–structure partitioned procedures based on Robin transmission conditions. *Journal of Computational Physics* 2008; **227**(14):7027–7051. URL <http://www.sciencedirect.com/science/article/pii/S0021999108002192>.
- [56] Burman E, Fernández MA. Stabilization of explicit coupling in fluid–structure interaction involving fluid incompressibility. *Computer Methods in Applied Mechanics and Engineering* 2009; **198**(5–8):766–784, doi:<http://dx.doi.org/10.1016/j.cma.2008.10.012>.
- [57] Astorino M, Chouly F, Fernández MA. Robin Based Semi-Implicit Coupling in Fluid-Structure Interaction: Stability Analysis and Numerics. *SIAM Journal on Scientific Computing* 2010; **31**(6):4041–4065, doi:10.1137/090749694.
- [58] Banks JW, Henshaw WD, Schwendeman DW. An analysis of a new stable partitioned algorithm for FSI problems. Part I: Incompressible flow and elastic solids. *Journal of Computational Physics* 2014; **269**:108–137, doi:<http://dx.doi.org/10.1016/j.jcp.2014.03.006>.
- [59] Zienkiewicz OC, Xie YM. A simple error estimator and adaptive time stepping procedure for dynamic analysis. *Earthquake Engineering & Structural Dynamics* 1991; **20**(9):871–887. URL <http://dx.doi.org/10.1002/eqe.4290200907>.
- [60] Belytschko T, Yen HJ, Mullen R. Mixed methods for time integration. *Computer Methods in Applied Mechanics and Engineering* 1979; **17–18**:259–275, doi:[http://dx.doi.org/10.1016/0045-7825\(79\)90022-7](http://dx.doi.org/10.1016/0045-7825(79)90022-7). URL <http://www.sciencedirect.com/science/article/pii/0045782579900227>.
- [61] Piperno S. Explicit/implicit fluid/structure staggered procedures with a structural predictor and fluid subcycling for 2d inviscid aeroelastic simulations. *International Journal for Numerical Methods in Fluids* 1997; **25**(10):1207–1226, doi:10.1002/(SICI)1097-0363(19971130)25:10<1207::AID-FLD616>3.0.CO;2-R. URL [http://dx.doi.org/10.1002/\(SICI\)1097-0363\(19971130\)25:10<1207::AID-FLD616>3.0.CO;2-R](http://dx.doi.org/10.1002/(SICI)1097-0363(19971130)25:10<1207::AID-FLD616>3.0.CO;2-R).
- [62] Bochev PB, Gunzburger MD, Shadid JN. On inf–sup stabilized finite element methods for transient problems. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**(15–16):1471–1489. URL <http://www.sciencedirect.com/science/article/pii/S0045782504000453>.
- [63] Bochev PB, Gunzburger MD, Lehoucq RB. On stabilized finite element methods for the Stokes problem in the small time step limit. *International Journal for Numerical Methods in Fluids* 2007; **53**(4):573–597. URL <http://dx.doi.org/10.1002/flid.1295>.
- [64] Shampine LF. Local error control in codes for ordinary differential equations. *Applied Mathematics and Computation* 1977; **3**(3):189–210, doi:[http://dx.doi.org/10.1016/0096-3003\(77\)90001-7](http://dx.doi.org/10.1016/0096-3003(77)90001-7). URL <http://www.sciencedirect.com/science/article/pii/0096300377900017>.
- [65] Ramm E, Wall WA. Shell structures—a sensitive interrelation between physics and numerics. *International Journal for Numerical Methods in Engineering* 2004; **60**(1):381–427, doi:10.1002/nme.967. URL <http://dx.doi.org/10.1002/nme.967>.
- [66] Gravemeier V, Comerford A, Yoshihara L, Ismail M, Wall WA. A novel formulation for Neumann inflow boundary conditions in biomechanics. *International Journal for Numerical Methods in Biomedical Engineering* 2012; **28**(5):560–573, doi:10.1002/cnm.1490. URL <http://dx.doi.org/10.1002/cnm.1490>.
- [67] Maier A, Gee MW, Reeps C, Pongratz J, Eckstein HH, Wall WA. A Comparison of Diameter, Wall Stress, and Rupture Potential Index for Abdominal Aortic Aneurysm Rupture Risk Prediction. *Annals of Biomedical Engineering* 2010; **38**(10):3124–3134, doi:10.1007/s10439-010-0067-6. URL <http://dx.doi.org/10.1007/s10439-010-0067-6>.
- [68] Raghavan M, Vorp DA. Toward a biomechanical tool to evaluate rupture potential of abdominal aortic aneurysm: identification of a finite strain constitutive model and evaluation of its applicability. *Journal of Biomechanics* 2000; **33**(4):475–482, doi:[http://dx.doi.org/10.1016/S0021-9290\(99\)00201-8](http://dx.doi.org/10.1016/S0021-9290(99)00201-8). URL <http://www.sciencedirect.com/science/article/pii/S0021929099002018>.
- [69] Holzapfel GA. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. Wiley, 2000.
- [70] Ismail M, Wall WA, Gee MW. Adjoint-based inverse analysis of windkessel parameters for patient-specific vascular models. *Journal of Computational Physics* 2013; **244**:113–130, doi:<http://dx.doi.org/10.1016/j.jcp.2012.10.028>. URL <http://www.sciencedirect.com/science/article/pii/S0021999112006298>.
- [71] Moireau P, Xiao N, Astorino M, Figueroa CA, Chapelle D, Taylor CA, Gerbeau JF. External tissue support and fluid–structure simulation in blood flows. *Biomechanics and Modeling in Mechanobiology* 2012; **11**(1):1–18, doi:10.1007/s10237-011-0289-z. URL <http://dx.doi.org/10.1007/s10237-011-0289-z>.
- [72] Swillens A, Lanoye L, De Backer J, Stergiopoulos N, Verdonck P, Vermassen F, Segers P. Effect of an Abdominal Aortic Aneurysm on Wave Reflection in the Aorta. *IEEE Transactions on Biomedical Engineering* 2008; **55**(5):1602–1611, doi:10.1109/TBME.2007.913994.
- [73] Gee MW, Förster C, Wall WA. A computational strategy for prestressing patient-specific biomechanical problems under finite deformation. *International Journal for Numerical Methods in Biomedical Engineering* 2010; **26**(1):52–72, doi:10.1002/cnm.1236. URL <http://dx.doi.org/10.1002/cnm.1236>.