Quasi-Newton Methods for Unstable Partitioned Fluid-Structure Interactions

Marcel Dominique Koch

Born 16th June 1992 in Gelsenkirchen, Germany

18th November 2016

Master's Thesis Mathematics

Advisor: Prof. Dr. Jochen Garcke

Second Advisor: Prof. Dr. Daniel Peterseim

Institute for Numerical Simulation

Mathematisch-Naturwissenschaftliche Fakultät der

Rheinischen Friedrich-Wilhelms-Universität Bonn

Contents

1 Introduction

The realistic modeling of many physical problems involves different physical processes acting and depending on each other. These kind of problems are called multifield or multiphysics problems. Fluid-structure interactions are a special case of multifield problems. The blood flow through an aorta or the airflow around a wind turbine are examples for this class of problems. The flow of the fluid conforms to the deformation of the structure, while the pressure within the fluid exerts a surface force on the structure. According to this force, the structure deforms into a new configuration, influencing the fluid again. This illustrates that these physical processes cannot be treated isolated.

The fluid and structure only interact on parts of their boundary, called the interface. This allows to partition the problem into a fluid and a structure sub problem, where the interaction is taken into account through the boundary conditions. Both sub problems can be modeled, discretized and solved separately, considering the different properties of the problems, leading to a highly flexible approach, called partitioned coupling. Furthermore, already existing and optimized software can be applied to the sub problems, only the interaction between the solvers has to be added.

The simplest method to incorporate the interaction is to exchange data on the interface between both sub problems once per time step. With the received data new boundary conditions are imposed. At least one solver must use data from the previous time step, resulting in a loss of accuracy. For compressible fluids this approach is still appropriate. However, solutions for incompressible fluids become unconditionally unstable if the ratio between the structural density and fluid density is close to one. This is caused by the so called added mass effect. For compressible fluids, this effect does not appear.

To circumvent this instability without changing the solvers for the sub problem, the solution in each time step is formulated as the solution to a fixpoint problem. Newton's method is not applicable, because the solver for both sub problems are handled as black boxes. It is assumed that no information except the output of a solver is known, which is often the case for commercial solvers. Especially the derivatives of the solvers are unknown. Currently, the most common solver for the fixpoint problem without use of derivatives is a under relaxation or Aitken's methods. Quasi-Newton methods had been considered only recently, most prominently in [14] and [5].

This thesis aims to bring these previously introduced quasi-Newton method into a more general context. More specifically, their relation to the generalized Broyden methods is shown. These methods build an approximative Jacobi matrix which minimizes the change regarding some matrix under the constraint of satisfying some secant equations. Through numerical tests the stability of a selection of quasi-Newton methods is displayed and influences of the added mass effect on the simulation are examined.

The second chapter gives an introduction into the mathematical models for the fluid and structure sub problem and their interaction. The fluid model encompasses incompressible fluids with moving boundary. The structural model allows for large deformation with small strains.

Discretization techniques for both models are discussed in chapter three. For the

fluid part the finite volume method is employed, while the finite element method is used for the structural part.

Chapter four introduces coupling methods for the partitioned approach, with a focus on the added mass effect.

In chapter five the generalized Broyden methods are derived and explained. Additionally, some details on the implementation of the methods is given. They are applied to numerical examples in chapter six, to examine the performance of different quasi-Newton methods. Furthermore, the impact of the added mass effect on implicit coupling is analysed.

The final chapter seven gives a summary of the thesis and an outlook for further investigations.

2 Mathematical Model for Fluid-Structure Interactions

This chapter aims to introduce the basic physical models, used for fluid-structure interactions. Figure 1 depicts the general geometry for such problems. The first part introduces some general equations governing the behavior of the material. In the second part, individual assumptions of the materials elaborate on these equations for the materials of interest in this thesis. More on the general theory of continuum mechanics can be found in [40] and [17].

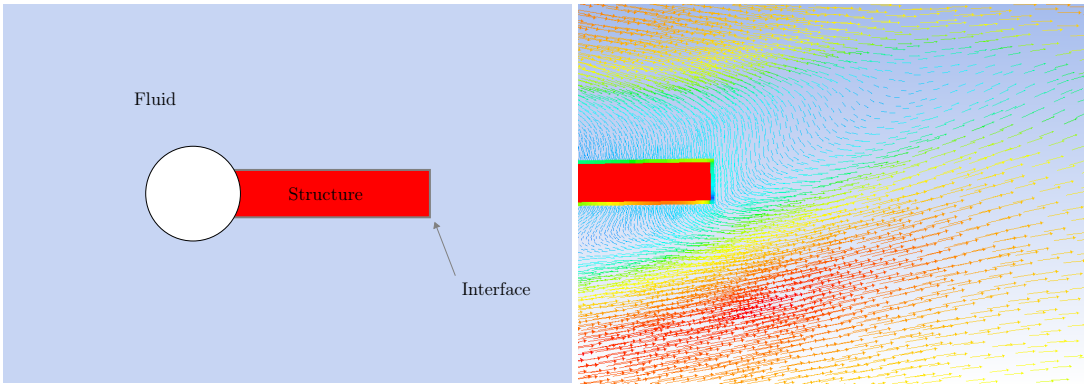


Figure 1: The left figure shows an abstract fluid-structure interaction problem. The right pictures an actual fluid-structure interaction simulation, appearing in section 6.2. It shows the velocity vectors, colored by magnitude, and the pressure at the interface.

Later on, the following material properties are assumed. The fluid material is incompressible and viscous, meaning the density does not change over time and internal friction hinders the flow. The structure is geometric nonlinear, allowing for larger movements. Furthermore its stress depends linear on the strain, leading to a materially linear structure. Both materials are isotherm, i.e. with constant temperature, and homogeneous, i.e. with constant density with respect to the space variable.

Generally valid models for the movement both of fluids and structure rely on the so-called conservation laws. These laws state that the following characteristics of a continuum

- mass,
- momentum and
- energy

do not change without external influences. Such external influences may be gravity, altering the momentum of a continuum, or radiation, increasing the energy of a continuum.

The conservation of mass and momentum are sufficient to describe the velocity of a flow and the pressure within. For the deformation of a structure, only the conservation momentum is needed. If, in addition, the temperature is variable, the conservation law

for energy is also needed. Since, both the fluid and the structure are considered to be isotherm, the following discussion omits the conservation of energy.

The first part of this section introduces some preliminary notions and theorems. Then the general conservation laws are derived and specified for fluid and structure models respectively.

2.1 Reynold's Transport Theorem

To derive the equations governing the conservation of the previous stated characteristics, the physical body is interpreted as a domain $\Omega \subset \mathbb{R}^d$. The domain Ω is open, connected and bounded. It is called the *reference configuration* and the points $\mathrm{X} \in \Omega$ are *material points*. This represents the body without forces acting on it.

Through time, the domain may undergo changes. For example external forces may deform the body. The map $\phi : [0, T] \times \Omega \mapsto \mathbb{R}^d$, called *motion*, describes these changes. The derivative $\mathrm{D}\phi$, with respect to X, is called *deformation gradient*. At time t, $\Omega_t := \phi(t, \Omega)$ denotes the current *configuration*. The derivation of the conservation laws needs some assumptions on ϕ:

A1 $\phi(0, \mathrm{X}) = \mathrm{X}$,

A2 ϕ is continuously differentiable in X and invertible for every time t,

A3 ϕ maintains its original orientation, i.e. $\mathrm{J}(t, \mathrm{X}) := \det(\mathrm{D}\phi(t, \mathrm{X}))$ is positive for all t.

For a quantity defined on the material there are two different ways to observe it:

- observe the same material point $\mathrm{X} \in \Omega$ at different times t, with different spatial points $\mathrm{x} \in \Omega_t$,
- observe the same spatial point $\mathrm{x} \in \phi(t, \Omega)$ at different times t, with different material points $\mathrm{X} \in \Omega$ passing by.

The first view describes the quantity in *Lagrangian coordinates* and uses upper case letters. The second one characterizes the quantity in *Eulerian coordinates*, using lower case letters. As an example

- $Q(t, \mathrm{X})$ denotes the value of a property at the material point $\mathrm{X} \in \Omega$ at time t, and
- $q(t, \mathrm{x})$ denotes the value of a property at the material point $\mathrm{X} \in \Omega$ which is at the spatial point $\mathrm{x} = \phi(t, X)$ at time t.

These two view are related by $q(t, \phi(t, \mathrm{X})) = Q(t, \mathrm{X})$.

Some of the following theorems depend on the velocity of the material points, defined by

- $V(t, \mathrm{X}) = \frac{\partial \phi}{\partial t}(t, \mathrm{X})$ in Lagrangian coordinates and

- $v(t, \mathrm{x}) = \frac{\partial \phi}{\partial t}(t, \phi^{-1}(t, \mathrm{x}))$ in Eulerian Coordinates.

With this definition, the partial time derivative of a quantity Q in Lagrangian coordinates can be written in terms of Eulerian coordinates $\mathrm{x} = \phi(t, \mathrm{X})$. This leads to the *material derivative* of q,

$$\begin{aligned} \mathrm{D}_t q(t, \mathrm{x}) : &= \frac{\partial}{\partial t} Q(t, \mathrm{X}) = \frac{\partial}{\partial t} q(t, \phi(t, \mathrm{X})) \\ &= \frac{\partial q}{\partial t}(t, \phi(t, \mathrm{X})) + \frac{\partial \phi}{\partial t}(t, \mathrm{X}) \cdot \nabla_{\mathrm{x}} q(t, \phi(t, \mathrm{X})) \\ &= \frac{\partial q}{\partial t}(t, \mathrm{x}) + v(t, \mathrm{x}) \cdot \nabla_{\mathrm{x}} q(t, \mathrm{x}), \end{aligned} \tag{2.1}$$

where ∇_{x} denotes the spatial derivative. The underscore is left out from here on. In addition to the change in time of q, the convective term $v \cdot \nabla q$ takes the movement of the material into consideration.

The following example underlines the significance of the convective term. Let Q describe the concentration of a material inside a flow with constant velocity v. Since the material is moving with the fluid, the concentration at each material point does not change in time, i.e. $\partial_t Q = 0$. In contrast, the time derivative of q at each spatial point x must satisfy the equation

$$\frac{\partial q}{\partial t} + v \cdot \nabla q = 0. \tag{2.2}$$

This is a simplified version of the transport equation, see [18].

Most quantities characterizing the movement of fluids or structures are proportional to the size of the continuum, called *extensive* quantities. Such quantities are mass, volume and energy. Pressure is an example for a quantity that is independent of the continuum's size.

An extensive quantity U is often characterized by some density function u_m, defined per mass. The value of $U(V)$, for a volume $V \subset \mathbb{R}^d$, is then given through integration

$$U(V) = \int_V \rho u_m \, \mathrm{d}x, \tag{2.3}$$

with mass density ρ of the material.

The conservation laws state that the considered quantities only change through external influences. Therefore the rate of change of quantities in the form of 2.3, with respect to time, is considered. First, examine the change of Ω. Application of the divergence theorem yields

$$\frac{d}{dt}|\Omega_t| = \frac{d}{dt} \int_{\Omega_t} \mathrm{dx} = \int_\Omega \frac{\partial \mathrm{J}}{\partial t}(t, \mathrm{X}) \, \mathrm{dX}, \tag{2.4}$$

with determinant of the configuration J, as stated in assumption A3. The time derivative of the determinant has an explicit formula, given by the following theorem.

Theorem 1. *If ϕ satisfies the assumptions A1-3 and has a continuous time derivative, then* $\mathrm{J}(t,\mathrm{X})$ *is continuously differentiable in time with*

$$\frac{\partial \mathrm{J}}{\partial t}(t,\mathrm{X}) = \nabla \cdot v(t,\mathrm{x})|_{\mathrm{x}=\phi(t,\mathrm{X})}\mathrm{J}(t,\mathrm{X}). \tag{2.5}$$

For a proof see [17].

Now, a general quantity with some density Q is examined. In Lagrangian coordinates the rate of change is simply

$$\frac{d}{dt}\int_\Omega Q(t,\mathrm{X})\,\mathrm{dX} = \int_\Omega \frac{\partial Q}{\partial t}(t,\mathrm{X})\,\mathrm{dX}, \tag{2.6}$$

since Q is defined on the reference configuration. An analog relation for Eulerian coordinate is more complicated. The quantity is defined on a moving domain Ω_t and therefore differentiation and integration cannot be interchanged easily. Instead the following Reynold's transport theorem gives the appropriate relation.

Theorem 2 (Reynold's Transport Theorem)**.** *The map ϕ satisfies A1-3 and the functions $v(t,\mathrm{x})$ and $q(t,\mathrm{x})$ are continuously differentiable. Then, for any connected subdomain $V \subset \Omega$ with configuration $V_t = \phi(t,V)$ it holds that*

$$\frac{d}{dt}\int_{V_t} q(t,\mathrm{x})\,\mathrm{dx} = \int_{V_t} \frac{\partial q}{\partial t}(t,\mathrm{x}) + \nabla\cdot(q(t,\mathrm{x})\cdot v(t,\mathrm{x}))\,\mathrm{dx}. \tag{2.7}$$

Proof. In the first step, transforming V_t into the reference configuration leads to

$$\frac{d}{dt}\int_{V_t} q(t,\mathrm{x})\,\mathrm{dx} = \frac{d}{dt}\int_V q(t,\phi(t,\mathrm{X}))J(t,\mathrm{X})\,\mathrm{dX}. \tag{2.8}$$

In the reference configuration, integration and differentiation can be interchanged. With application of the formula for the partial time derivative of J (2.5), the equation reads

$$\begin{aligned}\frac{d}{dt}\int_V q(t,\phi(t,\mathrm{X}))J(t,\mathrm{X})\,\mathrm{dX} = \int_V \Big(&\frac{\partial q}{\partial t}(t,\phi(t,\mathrm{X})) \\ &+ q(t,\phi(t,\mathrm{X}))\nabla\cdot v(t,\phi(t,\mathrm{X}))\Big)\,\mathrm{J}(t,\mathrm{X})\,\mathrm{dX}.\end{aligned}$$

The use of the material derivative $D_t q$ (2.1) along with the identity

$$\nabla\cdot(u\cdot w) = u(\nabla\cdot w) + \nabla u\cdot w \tag{2.9}$$

gives the final equation

$$\begin{aligned}\frac{d}{dt}\int_{V_t} q(t,\mathrm{x})\,\mathrm{dx} &= \int_{V_t} \mathrm{D}_t q(t,\mathrm{x}) + q(t,\mathrm{x})\nabla\cdot v(t,x)\,\mathrm{dx} \\ &= \int_{V_t} \frac{\partial q}{\partial t}(t,\mathrm{x}) + \nabla q(t,\mathrm{x})\cdot v(t,\mathrm{x}) + q(t,\mathrm{x})\nabla\cdot v(t,\mathrm{x})\,\mathrm{dx} \\ &= \int_{V_t} \frac{\partial q}{\partial t}(t,\mathrm{x}) + \nabla\cdot(q(t,\mathrm{x})\cdot v(t,\mathrm{x}))\,\mathrm{dx}.\end{aligned}$$

□

Remark. *The first term on the right hand side takes the rate of change of q into account, without considering the changes of V_t. The second term can be stated as an surface integral over V_t,*

$$\int_{V_t} \nabla \cdot (q \cdot v) \, \mathrm{d}x = \int_{\partial V_t} (q \cdot v) \cdot n \, \mathrm{dS}. \tag{2.10}$$

Therefore, this term describes the change over the boundary of V_t, while assuming no modification of q.

2.2 Conservation Laws

Reynold's Transport Theorem allows the derivation of the equations, governing the conservation of mass and momentum in Eulerian formulation. . The rest of this section uses the following functions,

- $\rho : [0,T] \times \Omega \mapsto \mathbb{R}$ the mass density of material,
- $v : [0,T] \times \Omega \mapsto \mathbb{R}^d$ the velocity of material,
- $t : [0,T] \times \partial\Omega \mapsto \mathbb{R}^d$ a force vector acting on the surface of the material,
- $f : [0,T] \times \Omega \mapsto \mathbb{R}^d$ a force vector acting on the volume of the material.

All functions are written in Eulerian coordinates.

The first equation to derive is the *continuity equation.* It is a consequence of the fact that mass is neither created nor destroyed. Therefore the mass of Ω_t must equal the mass of Ω, for all $t \geq 0$. With the mass density ρ, the mass of Ω is defined as

$$\mathrm{mass}(\Omega_t) = \int_{\Omega_t} \rho(t, \mathrm{x}) \, \mathrm{dx}. \tag{2.11}$$

In terms of its derivative the conservation of mass reads

$$\frac{d}{dt}\mathrm{mass}(\Omega_t) = \frac{d}{dt} \int_{\Omega_t} \rho(t, \mathrm{x}) \, \mathrm{dx} = 0. \tag{2.12}$$

The application of Reynold's Transport Theorem with $q = \rho$ leads to the continuity equation in integral form

$$\int_{\Omega_t} \frac{\partial \rho}{\partial t}(t, \mathrm{x}) + \nabla \cdot (\rho(t, \mathrm{x}) v(t, \mathrm{x})) \, \mathrm{dx} = 0. \tag{2.13}$$

Since the mass is maintained for every sub domain V of Ω, the integral form is true for every sub domain $V_t \subset \Omega_t$. This is only the case if the integrand is zero on Ω_t. With sufficient smoothness of ρ and v the continuity equation in differential form reads

$$\partial_t \rho + \nabla \cdot (\rho v) = 0 \tag{2.14}$$

Newton's second law of motion acts as basis for the derivation of the conservation of (linear) momentum. It states that the change of momentum $p = mv$ of a mass equals

the total force F acting on the mass. The total force divides into a surface force t only acting on the surface of the mass and a body force f acting on its volume. Gravity is an example for such a body force.

The integral formulation of Newton's second law reads, with normal vector n

$$\frac{d}{dt}\int_{\Omega_t} \rho v \,\mathrm{dx} = \int_{\Omega_t} \rho f \,\mathrm{d}x + \int_{\partial\Omega_t} t \,\mathrm{dS}. \tag{2.15}$$

With help from Reynold's Transport Theorem, the left hand side is formulated as an integral over Ω_t. Component wise it reads

$$\left(\frac{d}{dt}\int_{\Omega_t} \rho(t,\mathrm{x})v(t,\mathrm{x})\,\mathrm{dx}\right)_j = \int_{\Omega_t} \partial_t(\rho v_j) + \nabla\cdot(\rho v v_j)\,\mathrm{dx}. \tag{2.16}$$

To write surface integral of t over $\partial\Omega_t$ as a volume integral over Ω_t, more information on the surface force is needed. Under the assumption of Newton's second law, Cauchy's theorem [17] gives a linear dependency between the surface force t and the normal vector n, i.e.

$$t(\mathrm{x}, n) = \sigma(\mathrm{x})\cdot n(\mathrm{x}), \quad \sigma(\mathrm{x}) \in \mathbb{R}^{3\times 3}. \tag{2.17}$$

The matrix valued function σ, is called *Cauchy stress tensor*. It is a measurement for the forces acting on an area in the current configuration Ω_t.

The divergence theorem, applied to the surface integral over $\partial\Omega_t$, and the Reynold's Transport Theorem used for the left hand side of (2.16) now lead to the component wise integral formulation of the conservation of momentum

$$\int_{\Omega_t} \frac{\partial \rho v_j}{\partial t} + \nabla\cdot(\rho v v_j) - \rho f_j - (\nabla\cdot\sigma)_j \,\mathrm{dx} = 0. \tag{2.18}$$

Using the same argument as for the continuity equation (2.14), the first version of the differential form reads

$$\partial_t(\rho v_j) + \nabla\cdot(\rho v v_j) - \rho f_j - (\nabla\cdot\sigma)_j = 0. \tag{2.19}$$

The simplified conservation of momentum equation is achieved by plugging in the continuity equation (2.14) and the differentiable operator $v\cdot\nabla = \sum_{j=1}^{d} v_j\partial_{xj}$,

$$\rho(\partial_t v + (v\cdot\nabla)v) - \nabla\cdot\sigma = \rho f. \tag{2.20}$$

Summary

The conservation laws governing the behavior of both fluid and structure are

- the *continuity equation*:

$$\partial_t\rho + \nabla\cdot(\rho v) = 0 \quad \text{in } \Omega_t, \tag{2.21}$$

- and the *conservation of momentum equation*:

$$\rho\partial_t v + \rho(v \cdot \nabla)v - \nabla \cdot \sigma = \rho f \quad \text{in } \Omega_t. \tag{2.22}$$

The derivation of the stress tensor σ differs for each material, depending on its individual properties. Air, for example has little internal friction and uses therefore a different stress tensor than water. The following sections explain the stress tensors for the particular choices of material.

2.3 Model for Incompressible, Viscous Fluids

This section discusses the assumptions made on the fluid material and derives the *Navier-Stokes equation*. The incompressibility simplifies the continuity equation, and the viscosity defines the stress tensor. Furthermore a brief introduction into the *Arbitrary Lagrangian Eulerian* formulation is given. This plays a key role for the consideration of moving boundaries. The works [3] and [11] provide a detailed introduction into fluid dynamics. The Arbitrary Lagrangian Eulerian formulation is discussed in [16].

Flows are usually observed in some fixed domain $\Omega_f \subset \mathbb{R}^d$, $d = 2, 3$, with the fluid passing through it. Examples are gas flowing through a section of a pipe, or blood through an aorta. Both cases consider only a small part of a larger system, i.e. the circulatory system. As a result, the Eulerian formulation of the conservation equations has to be applied.

As stated in the beginning of the chapter, the fluid is incompressible. Therefore, the volume of any subset in the reference configuration $V \subset \Omega$ does not change in time. With the transformation determinant J and its time derivative 1 this leads to

$$\frac{dV_t}{dt} = \frac{d}{dt}\int_{V_t} \mathrm{dx} = \int_{V_t} \nabla \cdot v \,\mathrm{dx} = 0 \quad \forall\, V \subset \Omega. \tag{2.23}$$

Because of the arbitrary choice of V the divergence of v has to be zero in Ω_f. It follows now from the continuity equation that the mass density ρ must be constant, since

$$\partial_t \rho + \underbrace{\rho \nabla \cdot v}_{=0} = 0 \quad \text{in } \Omega_f. \tag{2.24}$$

Thus, the continuity equation can be reduced to

$$\nabla \cdot v = 0 \quad \text{in } \Omega_t. \tag{2.25}$$

Since the fluid is viscous, frictions within the fluids are significant. In addition to pressure based forces pI, the stress tensor also consists of viscous forces τ. To avoid confusion in the notation, σ_f denotes the stress tensor with respect to the fluid. Under the assumptions that τ only depends on the velocity differences Dv, the stress tensor has the following form,

$$\sigma_f = 2\nu\varepsilon(v) - pI. \tag{2.26}$$

Here, ν denotes the *dynamic viscosity* and

$$\varepsilon(v) = \frac{1}{2}(Dv + (Dv)^T) \tag{2.27}$$

the symmetric part of Dv, also known as the *rate-of-strain tensor*. The divergence of the stress tensor then takes the form

$$\nabla \cdot \sigma_f = \nu \Delta v - \nabla p. \tag{2.28}$$

Plugging in the divergence of σ_f into the conservation of momentum equation (2.22) together with the divergence free condition (2.25) leads to the Navier-Stokes equations

$$\begin{aligned} &\text{Continuity:} && \nabla \cdot v = 0 && \text{in } \Omega_f, && (2.29)\\ &\text{Momentum:} && \rho\, \partial_t v + \rho(v \cdot \nabla)v + \nabla p = \nu \Delta v + \rho f && \text{in } \Omega_f. && (2.30) \end{aligned}$$

The complete description a flow in a domain Ω_f over a time interval $[0, T]$ needs initial conditions and boundary conditions. For initial conditions the velocity is set to some value $v_0 : \Omega_f \mapsto \mathbb{R}^d$,

$$v(0, \cdot) = v_0(\cdot) \quad \text{on } \Omega_f. \tag{2.31}$$

The initial velocity must also satisfy the divergence free condition (2.25). On the boundary of Ω_f, combinations of Dirichlet and Neumann boundary conditions for v and p are possible. First, the boundary is divided into two disjoint sets $\partial\Omega_f = \Gamma_f^1 \cup \Gamma_f^2$. On the first set Γ_f^1 Dirichlet boundary conditions are imposed on the velocity

$$v = v_b \quad \text{on } (0, T] \times \Gamma_f^1. \tag{2.32}$$

The boundary condition on Γ_f^2 then provides Neumann condition for the stress tensor,

$$\sigma_f \cdot n = t \quad \text{on } (0, T] \times \Gamma_f^2. \tag{2.33}$$

Arbitrary Lagrangian-Eulerian Formulation

Neither the Eulerian formulation of the Navier-Stokes equations nor an Lagrangian formulation is well suited for fluid-structure interactions. In the Eulerian formulation, the domain Ω_f is fixed. However, fluid-structure interactions consider the effects of a moving structure on a fluid. Parts of the boundary of Ω_f must adapt to the motion of the structure. Therefore the domain cannot remain immobile. On the other hand, the Lagrangian formulation is not appropriate for large deformations, such as the motion of a fluid.

To contribute for the moving boundary the Arbitrary Lagrangian-Eulerian (ALE) formulation is introduced. This formulation uses a new reference domain $\Omega_{ALE} \subset \mathbb{R}^d$ with the continuously differentiable and invertible motion $\psi : [0, T] \times \Omega_{ALE} \mapsto \Omega_f$. In general, Ω_{ALE} and its motion are different from the reference configuration Ω and motion ϕ, introduced in the beginning of this chapter. The points $Y \in \Omega_{ALE}$ are the ALE-coordinates.

Section 3.1 defines a mesh for the discretization of the fluid equations. The ALE-coordinates represent these mesh points. With the motion ψ the mesh points can move independently of the material coordinates. Therefore, a displacement at the boundary can be transfered fairly easy to the mesh points.

To derive the ALE equations for the conservation of mass and momentum, a variant of the transport theorem 2 is applied. For any $V \subset \Omega_f$ and a scalar or vector valued function q it holds that

$$\frac{d}{dt}\int_V q \,\mathrm{dx} = \int_V \frac{\partial q}{\partial t} + \nabla \cdot (q \cdot \hat{v}) \,\mathrm{dx}, \tag{2.34}$$

where $\hat{v}(t, \mathrm{x}) = \frac{\partial \psi}{\partial t}(t, \psi^{-1}(t, \mathrm{x}))$ denotes the velocity of the ALE-coordinates. Hence, the integral formulation of the conservation of mass (2.13) reads in the ALE-formulation as

$$\frac{d}{dt}\int_{\Omega_f} \rho \,\mathrm{dx} + \int_{\Omega_f} \rho \nabla \cdot (v - \hat{v}) \,\mathrm{dx} = 0, \tag{2.35}$$

and the conservation of momentum in each component (2.18) as

$$\frac{d}{dt}\int_{\Omega_f} \rho v^i \,\mathrm{dx} + \int_{\Omega_f} \nabla \cdot (\rho v^i (v - \hat{v})) - (\nabla \cdot \sigma_f)^i \,\mathrm{dx} = \int_{\Omega} \rho f^i \,\mathrm{dx}. \tag{2.36}$$

These equations are the basis for the discretization techniques of section 3.1.

The motion ψ is not arbitrary, instead its velocity must satisfy the *geometric conservation law* (GCL)

$$\frac{d}{dt}\int_{\Omega_f} \mathrm{dx} - \int_{\Omega_f} \nabla \cdot \hat{v} \,\mathrm{dx} = 0. \tag{2.37}$$

This condition arises from the conservation laws for a constant fluid velocity $v = \text{const}$. With a reference velocity conforming to the GCL, the conservation of mass (2.35) is reduced to the divergence free condition for the fluid velocity.

2.4 Model for Nonlinear Structures

This section defines the structural model, used for this thesis. The model needs to enable potentially large and elastic deformations. However, only small strains will be examined. An extensive discussion of structural mechanics is found in [41] or [13].

Equations for the structural model usually employ the Lagrangian formulation. This is reasonable, because the deformed coordinates are the outcome of a computation, which needs to take place first. The use of the Lagrangian formulation also reduces the conservation equations, since it loses the material derivative (2.1). For this section, $\Omega_s \subset \mathbb{R}^d$, $d = 2, 3$, refers to the reference configuration.

Instead of the position in the current configuration, this section considers the displacement $d(t, \mathrm{X}) = \phi(t, \mathrm{X}) - \mathrm{X}$ of a material point X. Furthermore, $F := \mathrm{D}\phi$ denotes the deformation gradient. With this, the strain E is defined by

$$E = \frac{1}{2}(F^T F - I) = \frac{1}{2}(\nabla d + (\nabla d)^T + (\nabla d)^T \nabla d). \tag{2.38}$$

The stress tensor σ_s, introduced in 2.2, measures forces acting on the current configuration. Applying the transformation theorem for integrals leads to the *first Piola-Kirchhoff stress tensor* P, defined on the reference configuration,

$$P = \mathrm{J}\sigma_s F^{-T}. \tag{2.39}$$

Using this stress tensor, the conservation of momentum in the reference configuration reads

$$\rho\, \partial_{tt} d - \nabla \cdot P = f. \tag{2.40}$$

The 1. Piola-Kirchhoff stress tensor produces a stress vector $T = PN$, based at a point x in the current configuration. Pulling back this stress vector into the reference gives the *second Piola-Kirchhoff stress tensor*

$$S = F^{-1}P. \tag{2.41}$$

Furthermore, this stress tensor is symmetric, whereas the 1. Piola-Kirchhoff stress tensor P is not. Now, the conservation of momentum in the reference configuration, using the 2. Piola-Kirchhoff stress tensor reads as

$$\rho\, \partial_{tt} d - \nabla \cdot (F\, S) = \rho f \quad \text{in } \Omega_s. \tag{2.42}$$

Next, the relation between the geometric quantities, displacement d and strain E, and the 2. Piola-Kirchhoff stress is clarified. For this thesis, the St. Venant-Kirchhoff model provides the relation between strain E and stress S. Assuming this model is an appropriate choice, the stress tensor S has the form

$$S = \lambda\, \mathrm{tr}(E) I + 2\mu E, \tag{2.43}$$

with the two *Lamé constants* λ and μ. Instead of the Lamé constants the *Young's modulus* E and *Poisson's ratio* ν are often used. They are related to the Lamé constants via

$$\mu = \frac{E\nu}{(1+\nu)(1-2\nu)}, \quad \lambda = \frac{E}{2(1+v)}. \tag{2.44}$$

The St. Venant-Kirchhoff model implies a linear relation between strain E and stress S, which is suitable for small strains. For the modeling of large strains, other material models, for example the Mooney-Rivlin model, are a better choice.

For completeness, the conservation of momentum equation (2.42) requires additional initial and boundary conditions, to describe a physical problem. Since (2.42) is a hyperbolic differential eqation, it needs, together with an initial displacement d_0, an initial velocity field v_0, i.e.

$$d(0,\cdot) = d_0, \quad \partial_t d(0,\cdot) = v_0 \quad \text{on } \Omega_s. \tag{2.45}$$

Similar to the fluid model, the boundary of Ω_s is split into two parts, $\partial\Omega_s = \Gamma_s^1 \cup \Gamma_s^2$. On the first part, Dirichlet conditions are prescribed,

$$d = d_b \quad \text{on } (0,T] \times \Gamma_s^1, \tag{2.46}$$

and on the second part Neumann conditions are imposed,

$$FS \cdot n = s \quad \text{on } (0,T] \times \Gamma_s^2. \tag{2.47}$$

2.5 Model for Interactions between Fluids and Structures

This part examines the modeling of the influences and dependencies of fluids and structures, acting upon each other. The fluid and the structure share a common boundary $\Gamma(t) = \partial\Omega_f(t) \cap \partial\Omega_s(t)$, also called *interface*. Here $\Omega_f(t)$ and $\Omega_s(t)$ denote the current configuration of the fluid and structure domain respectively. On the interface, the solutions to each sub model must take the other solution into account respectively. For example a moving structure imposes the velocity of the fluid at the interface. On the other hand, the fluid exerts a force on the interface counteracting or aiding the movement of the structure.

To model the physical behavior, the following *interface constraints* on the structure and fluid model are introduced:

1. The geometries of both models must match at the interface. In particular, both configurations $\Omega_f(t)$ and $\Omega_s(t)$ may not overlap or seperate.

2. The velocities on the interface must be equal,

$$v(t, \mathrm{x}) = \partial_t d(t, \mathrm{x}) \quad \forall\, t \in [0, T] \text{ and } \mathrm{x} \in \Gamma(t). \tag{2.48}$$

3. The forces on the interface must be in equilibrium. Therefore, they must sum up to zero,

$$\sigma_f \cdot n_f + \sigma_s \cdot n_s = 0 \quad \forall\, t \in [0, T] \text{ and } \mathrm{x} \in \Gamma(t), \tag{2.49}$$

   where n_f and n_s are the outer normal vectors on the fluid and stucture domain, respectively.

The first two constraints are the kinematic constraints, the third one is the dynamic constraint. It applies the constraint on the stress tensor σ_s in the current configuration. However, the structure model uses the 2. Piola-Kirchhoff stress tensor S in the reference configuration. The two stress tensors are related through

$$\sigma_s = J^{-1} F S F^T, \tag{2.50}$$

which follows from the definitions of the 1. and 2. Piola-Kirchhoff stress tensors (2.39), (2.41).

These constraints prescribe boundary conditions on the interface. Kinematic constraints provide Dirichlet boundary conditions and dynamic constraints yield Neumann boundary conditions. Most commonly, the fluid model implements the kinematic interface constraints, while the structure model enforces the dynamic interface constraints.

3 Numerical Model of Fluid and Structure Problem

This chapter introduces general ideas on how to solve actual fluid-structure problems. The mathematical models described in section 2 have no analytic solution in general. There exist some special cases with analytic solutions, see [17], although these are not of use for practical applications. Instead, starting with the models from section 2, approximate versions, the numerical models, are derived. The aim in this chapter is to reduce the mathematical models, defined on an infinite number of points in space and time, to a finite number of equations.

The first section introduces the numerical fluid model, including mesh motion. The second part establishes the techniques for the structural model. The last part of this section discusses the discretized interface conditions. The different concepts used in the fluid and structure model are acknowledged there.

3.1 Discretized Fluid Model

This part gives a short overview of the numerical models used for computational fluid dynamics (CFD). First, the domain Ω is discretized with the *finite volume method* (FVM). Another possible spatial discretization, the *finite element method* (FEM), is as discussed as part of the numerical structure model. The second part illustrates methods employed for temporal discretization. The books [37], [58] and [56] give a detailed introduction into CFD and the finite volume method.

Spatial Discretization via FVM

At first, the domain Ω is divided into N^c control volumes C, also called cells. The cells have simple geometric shapes, for two dimensional applications triangles or rectangles, and for three dimensional applications tetrahedrons, hexahedrons, or prisms. Figure 2 show examples for different cell shapes. The division of the domain into cells is called mesh and the vertices of the cells are the mesh points. For a cell C, P_c denotes the midpoint of the cell.

(a) Triangular shaped cell (b) Hexahedral shaped cell

Figure 2: Examples for different cell shapes.

The idea of the finite volume method is that the conservation laws hold for each

control volume C. The integral form of the Navier-Stokes equations for a cell C reads

$$\int_C \nabla \cdot v \,\mathrm{dx} = 0, \tag{3.1}$$

$$\frac{d}{dt}\int_C \rho v^i \,\mathrm{dx} + \int_C \rho \nabla \cdot (v^i(v-\hat{v})) - \nu \Delta v^i \,\mathrm{dx} + \int_C \partial_{x_i} p \,\mathrm{dx} = \int_C \rho f^i \,\mathrm{dx}, \tag{3.2}$$

where v^i is the i-th velocity component. It is assumed the geometric conservation law is satisfied. The application of the divergence theorem transforms the volume integral over C into a boundary integral,

$$\int_{\partial C} v \cdot n \,\mathrm{dS} = 0, \tag{3.3}$$

$$\frac{d}{dt}\int_C \rho v^i \,\mathrm{dx} + \int_{\partial C} (\rho v^i (v-\hat{v}) - \nu \nabla v^i) \cdot n \,\mathrm{dS} + \int_{\partial C} p n^i \,\mathrm{dS} = \int_C \rho f \,\mathrm{dx}. \tag{3.4}$$

This formulation needs less regularity of the solution, since only first order derivatives under the integral appear.

To solve the equations (3.3) numerically, the integrals are discretized. A simple choice is the midpoint method, which takes the function value at the midpoint P_c as an approximation for the function on C, i.e.

$$\frac{d}{dt}\int_C \rho v(t,\mathrm{x}) \,\mathrm{dx} \approx \frac{d}{dt}|C| \rho v_{P_c}(t). \tag{3.5}$$

Here $v_{P_c}(t)$ denotes the approximation for the velocity at the midpoint of the cell at time t. The right hand side term is handled in a similar fashion.

For the integration over the surface, the boundary ∂C is split into its $d-1$ dimensional faces $S(C)$. Let M_f denote the midpoint of a face $f \in S(C)$, see figure 3. Then, the midpoint method for integrating the pressure part reads

$$\int_{\partial C} pn \,\mathrm{dS} = \sum_{f \in S(C)} \int_f pn \,\mathrm{dS} \approx \sum_{f \in S(C)} |f| p_{M_f} n_f. \tag{3.6}$$

The pressure at the face midpoint M_f is interpolated from the midpoints of the two cells sharing the face f, if f is not on the boundary of Ω. For faces on the boundary $\partial\Omega$, the boundary conditions are employed. The discretization of the continuity equations follows in the same manner.

The most straight forward interpolation scheme is the central difference scheme. For two cells C and D, sharing the face f, the central difference scheme defines the approximated value of $p(M_f)$ as the average of the values at the midpoints,

$$p_{M_f} = \frac{p_{P_c} + p_{P_d}}{2}. \tag{3.7}$$

This scheme has second order accuracy [37].

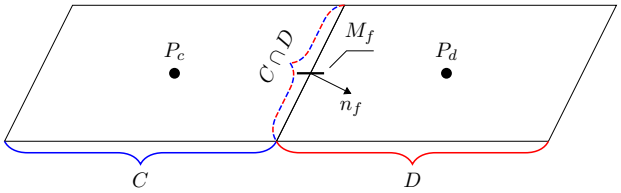


Figure 3: Two cells sharing one face. Midpoint and normal vector of the common face are displayed.

The computation of the convective term involves some extra work. The midpoint method for integration of the convective term, disregarding the constant density, results in

$$\int_{\partial C} v^i(v - \hat{v}) \cdot n \, \mathrm{dS} \approx \sum_{f \in S(C)} |f| v^i_{M_f} (v - \hat{v})_{M_f} n_f. \tag{3.8}$$

The choice of a suitable interpolation scheme for the midpoint velocities is crucial to the stability of the finite volume method. For small time steps the central difference scheme exhibits instabilities [37]. Therefore, other interpolation scheme are better suited.

The method of characteristics shows that for a linear hyperbolic equations, like (2.2), quantities travel along certain lines [18]. In the case of the Navier-Stokes equation the term $v - \hat{v}$ describes this direction. For a cell C with face f the flux

$$\Phi_f = |f|(v - \hat{v})_{M_f} \cdot n_f \tag{3.9}$$

indicates in which direction quantities flow. If Φ_f is positive, the angle between relative velocity and outer normal vector is less then 90°. Therefore, a quantity flows outward of the cell. For a negative flux Φ_f, the reversed relation holds.

The upwind scheme takes the direction of the flow into account. For a face f with the adjacent cells C and D, and normal vector in direction D, the upwind scheme reads

$$v(M_f)\Phi_f \approx \max(\Phi_f, 0)v(P_c) + \min(\Phi_f, 0)v(P_d). \tag{3.10}$$

The upwind scheme assigns the value of the cell, where the flow comes from, to the midpoint of f. This scheme has better stability than the central difference scheme. As a drawback, it is only first order convergent [37].

The gradient term ∇v^i leads to further difficulties. The discretization of gradient uses the midpoint method (3.6) for the integral combined with the central difference scheme (3.7) for the face midpoint value. This leads to

$$\int_{\partial C} \nabla v^i \, \mathrm{dx} \approx \sum_D |C \cap D| \frac{\nabla v^i_{P_c} + \nabla v^i_{P_d}}{2}, \tag{3.11}$$

which requires the evaluation of ∇v^i at the cell midpoints. The following method applies the divergence theorem to the volume integral of ∇v^i over the cell D. Using the midpoint

rule for the volume and boundary integral, the approximate gradient at cell midpoint P_d reads as

$$\nabla v^i_{P_d}(t) \approx \frac{1}{|D|}\int_D \nabla v^i \,\mathrm{dx} = \frac{1}{|D|}\int_{\partial D} v^i n \,\mathrm{dS} \tag{3.12}$$

$$\approx \frac{1}{|D|}\sum_{f\in S(D)} v^i_{M_f} n_f |f|. \tag{3.13}$$

Again, the midpoint values $v^i_{M_f}$ must be computed. The simplest approach is the central difference method, i.e. taking $v^i_{M_f}$ as the average of the two neighbouring cells. For irregular meshes, approaches using the average of values at the vertices of the face tend to be more accurate, although at a higher computational cost [30].

Using these discretization techniques leads to a set of ordinary differential equations,

$$\frac{d}{dt}\rho\mathbf{M}\mathbf{v}' + \mathbf{N}(\mathbf{v}) + \mathbf{G}\mathbf{p} = \mathbf{f} \tag{3.14}$$

$$\mathbf{D}\mathbf{v} = 0, \tag{3.15}$$

with the constraint that the velocity has to be divergence free. The vector $\mathbf{v}$ and $\mathbf{p}$ contain the velocity and pressure at the cell midpoints. The diagonal matrix $\mathbf{M}$ contains the volume of the cells. The nonlinear coefficients (3.8) and (3.11) are summarized in $\mathbf{N}$. The matrices $\mathbf{G}$ and $\mathbf{D}$ represent the discretization of the boundary integral of the pressure and the continuity equation (3.6). The right hand side $\mathbf{f}$ already incorporates the boundary conditions.

Time Discretization

The ordinary differential equation (3.14) can be formulated as

$$\frac{d}{dt}y(t) = f(y,t), \tag{3.16}$$

with $y(t)$ the vector of unknown quantities at cell midpoint and a nonlinear function f. To solve these kinds of ODEs, the time interval $[0,T]$ is divided into a number of time steps $0 = t_0 < t_1 < \cdots < t_N = T$. The numerical methods for the solution of the ODE (3.16) approximate the derivative of y, for example with a difference scheme like

$$\frac{d}{dt}y(t_{n+1}) \approx \frac{y_{n+1} - y_n}{\Delta t_{n+1}}, \tag{3.17}$$

with $\Delta t_{n+1} = t_{n+1} - t_n$ and y_{n+1} and y_n values of y at the time steps t_{n+1}, t_n.

So called explicit methods then use the value of y at time t_n for the computation of the right hand side. These methods are usually fast to compute, since they need only one evaluation of f. The approximation at the new time step is then simply a linear combination of, possible multiple, previous approximations y_n and $f(y_n, t_n)$. The drawback of explicit methods is that the time step size Δt must satisfy the CFL condition,

$\Delta t < \text{const}\, h$, where h is the maximal spatial discretization, $h = \max |C|$, and the constant factor depends on the velocity field. Therefore a fine spatial discretization implies an even finer time discretization.

Implicit methods use the value $f(y_{n+1}, t_{n+1})$ for their right hand side. The simplest implicit method is the implicit Euler method, which applies the backward differential quotient to the ODE (3.16). The update reads as follows

$$y_{n+1} = y_n + \Delta t_{n+1} f(y_{n+1}, t_{n+1}). \tag{3.18}$$

This method is convergent of order one and stable even for larger time step sizes [29]. But instead of a straightforward linear combination a possibly nonlinear system must be solved.

An implicit method, convergent of order two, is the backward differentiation formula (BDF2) [15]. Additionally, it takes the y value at time t_{n-1} into account. The update formula is given by

$$y_{n+1} = \frac{1}{4} y_n - \frac{1}{3} y_{n-1} + \frac{2}{3} \Delta t f(y_{n+1}, t_{n+1}). \tag{3.19}$$

As an example, choosing the implicit Euler method for the time discretization results in the nonlinear system

$$\rho \mathbf{v}^{n+1} = \mathbf{v}^n + \Delta t_{n+1} \left(\mathbf{f} - \mathbf{N}\left(\mathbf{v}^{n+1}\right) + \mathbf{G}\mathbf{p}^{n+1} \right), \tag{3.20}$$

$$\mathbf{D}\mathbf{v}^{n+1} = 0. \tag{3.21}$$

This system can be solved either in a coupled manner for both the velocity and the pressure or in a segregated manner. In both cases, application of Newton's method yields a linearized version the system. Chapter 5 discusses Newton's method in a general setting.

In the segregated case, at first, a prediction $\mathbf{v}^*$ of the new velocity is computed. This prediction applies the already calculated pressure form the current time step to the momentum equation. Usually, the divergence free condition is not satisfied by the velocity prediction. Therefore, a pressure correction method calculates the new pressure and velocity. The SIMPLE [45], [44] and the PISO [32] algorithm operate using this scheme, both with different pressure correction methods.

Mesh motion

The ALE formulation of the Navier-Stokes equations needs the velocity $\hat{v}$ of the reference system. The motion of the structure provides the displacement at the common interface. Through some mesh deforming algorithm, the finite volume mesh has to adjust itself to the displacement. Otherwise, the moving boundary leads to degenerated cells. The motion of the cell vertices then determines the approximate reference velocity $\hat{v}$.

There are different options on how to transfer the displacement of the interface nodes into the domain. A first idea is to associate each mesh point within Ω with one mesh point at the moving boundary. By means of interpolation, for example linear or quadratic, the

motion of the mesh point is then computed. For rectilinear meshes, it is fairly easy to decide which boundary point is associated with a mesh point. On general meshes the decision is not clear.

An alternative method interprets the mesh as a system of springs [4]. Then, Hook's law describes the motion of each mesh point. This leads to the linear system

$$\mathbf{K}\hat{d} = \mathbf{f}, \tag{3.22}$$

where $\hat{d}$ is the displacement of the mesh points.

The matrix $\mathbf{K}$ contains the stiffness k_{ij} of the imaginary spring between two points x_i and x_j. Usually, the stiffness is the inverse distance between the points, i.e.

$$k_{ij} = \frac{\kappa}{||x_i - x_j||}, \tag{3.23}$$

with some factor κ.

The force vector $\mathbf{f}$ is a linear transformation of the displacements at the interface, based on Hook's law. For cell shapes different from simplices, additional degrees of freedom can be added, for example by taking diagonals into consideration.

The reference velocity can be approximated using the newly computed displacements of the mesh points. To satisfy the geometric conservation law (2.37), the approximation must be exact of the same order as the time discretization, see [36] and [33]. For the implicit Euler method this results in

$$\hat{v} = \frac{\hat{d}^{n+1} - \hat{d}^n}{\Delta t}, \tag{3.24}$$

and for the BDF2 method the approximation reads

$$\hat{v} = \frac{3\hat{d}^{n+1} - 4\hat{d}^n + \hat{d}^{n-1}}{2\Delta t}. \tag{3.25}$$

3.2 Discretized Structure Model

In this part, a short introduction into the numerical solution of the structural is given. As before, firstly the spatial discretization of Ω, the finite element method is worked out. Because of the second time derivative, the temporal discretization varies from the one in the discretized fluid model. More information on the finite element method and its application to elasticity problems can be found in [6], [12], and [31].

Spatial Discretization via FEM

The finite element method uses the weak formulation of the structural equation (2.42) as a starting point. First, the equation is multiplied by a test function $v \in H_0^1(\Omega)$. The Sobolev space $H_0^1(\Omega)$ contains all weakly differentiable functions on Ω, fulfilling zero Dirichlet conditions on Γ_1. Integrating then leads to the equation

$$\rho \int_{\Omega_s} \partial_{tt} d \cdot v \, \mathrm{dX} - \int_{\Omega_s} (\nabla \cdot (FS)) \cdot v \, \mathrm{dX} = \rho \int_{\Omega_s} f \cdot v \, \mathrm{dX}. \tag{3.26}$$

The divergence theorem applied to the stress tensor term, together with the Neumann boundary condition $FS \cdot n = s \cdot n$ on Γ_2 and Dirichlet conditions on Γ_1, reduces the equation to

$$\rho\, \partial_{tt} \int_{\Omega_s} d \cdot v \,\mathrm{dX} + \int_{\Omega_s} FS : \nabla v \,\mathrm{dX} = \rho \int_{\Omega_s} f \cdot v \,\mathrm{dX} + \int_{\Gamma_s^2} v \cdot (s \cdot n) \,\mathrm{dS} \tag{3.27}$$

Here $A : B$ denotes the scalar product for matrices. This formulation requires only one weak derivative of d to be well defined.

The weak formulation of the structural problem now reads as follows: Find $d \in C^2([0,T], H_g^1)$ such that

$$\rho\, \partial_{tt} a(d(t), v) + k(d(t), v) = F(t)(v) \quad \forall\, v \in H_0^1, \tag{3.28}$$

where

$$a(u, v) := \int_\Omega u \cdot v \,\mathrm{dX} \tag{3.29}$$

is a bilinear form,

$$k(u, v) := \int_\Omega (FS)(u) : \nabla v \,\mathrm{dX} \tag{3.30}$$

is linear in its second component, but nonlinear in its first component and

$$F(t)(v) := \rho \int_\Omega f(t) \cdot v \,\mathrm{dX} + \int_{\Gamma^2} v \cdot (s \cdot n) \,\mathrm{dS} \tag{3.31}$$

is a time dependent linear form on H_0^1. The Sobolev space H_g^1 incorporates the Dirichlet boundary condition g on d.

The finite element method seeks an approximate solution d_h in some finite dimensional space $V_{h,g} \subset H_g^0$. Let $\{\phi_i,\, i = 1, \ldots, N^b\}$ be a basis of the space $V_{h,g}$. Every function in $V_{h,g}$ has a representation as linear combination of the basis functions ϕ_i. Thus, the approximation d_h has the expansion

$$d_h(t, \mathrm{X}) = \sum_{i=1}^{N^b} \mathbf{d}_i(t) \phi_i(\mathrm{X}), \tag{3.32}$$

with time dependent coefficients $\mathbf{d} : [0,T] \mapsto \mathbb{R}^d$.

In the same manner, the test space H_0^1 is discretized into the finite dimensional space V_h. The spaces V_h and $V_{h,g}$ differ only in the applied Dirichlet boundary conditions. For simplicity's sake, the Dirichlet condition for d is set to zero. Hence, ϕ is also a basis of V_h.

Since the weak formulation holds true for every $v \in H_0^1$, it must also hold for every basis function ϕ_i, leading to N^b equations. Using the representation (3.32) of d_h, the equations can be written compactly as,

$$\rho\, \mathbf{M}\, \mathbf{d}'' + \mathbf{K}(\mathbf{d}) = \mathbf{F}, \tag{3.33}$$

where $\mathbf{d}(t)$ are again the coefficients of d_h. The coefficients of the mass matrix M, the stiffness vector K, and the force vector F are

$$\mathbf{M}_{i,j} = a(\phi_i, \phi_j), \tag{3.34}$$
$$\mathbf{K}(\mathbf{d})_j = k(\mathbf{d}, \phi_j), \tag{3.35}$$
$$\mathbf{F}_j = F(t)(\phi_j). \tag{3.36}$$

Therefore, the approximation d_h is the solution to a nonlinear system of N^b ordinary differential equations. In linear elasticity theory, the nonlinear terms in the strain E are disregarded. This leads to a linear system for $\mathbf{d}$, because $k(\cdot, \cdot)$, as a consequence, is also a bilinear form.

The choice of basis functions ϕ_i is still left open. It should lead to a sparse matrix $\mathbf{M}$ and easy calculable integrals $\mathbf{K}(d)_j$ and $\mathbf{F}_j$. This is the case, if each function has only a limited support.

The domain Ω is again divided into N^e elements T_s. For two dimensional applications the elements are usually triangles and for three dimensional applications tetrahedrons or hexahedrons are a common choice. Figure 4 show an example triangulation of the unit square with missing square $[0.5, 1]^2$. The integrals appearing in equations (3.34) are split into integrals over each element, i.e. for some functions $u, v \in H_0^1$

$$\int_\Omega u \cdot v \, \mathrm{dX} = \sum_T \int_T u \cdot v \, \mathrm{dX}. \tag{3.37}$$

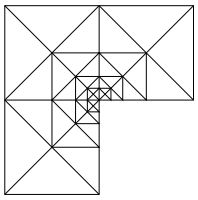


Figure 4: Example trianulation of a square with missing lower left sub square.

Therefore, it suffices to define the basis functions locally for each element T. On the element T the basis functions ϕ_i^T are polynomials and satisfies the condition

$$\phi_i^T(v_j) = \delta_{ij}, \tag{3.38}$$

with the nodes of the element v_j. For linear basis functions, only the vertices of the element are considered as nodes. Quadratic polynomials require additionally the edge midpoints. For 1D application, figure 5 sketches an example for piecewise linear basis functions.

Aggregating all local basis functions sharing the same node v_i results in the globally defined function ϕ_i. The globally defined function $\phi_i(x)$ is set to $\phi_i^T(x)$ if the point x lies within an element T sharing the node v_i. If the element containing x does not share this node, the function ϕ_i equals zero. Thus, the global functions are polynomial on each

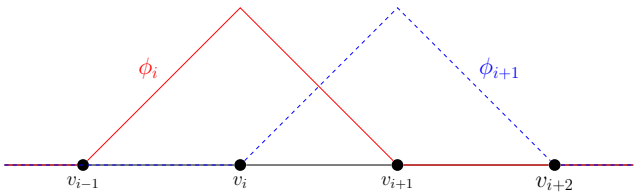


Figure 5: 1D example of global basis functions, satisfying $\phi_i(v_j) = \delta_{ij}$.

element and have limited support. It is also continuous on the whole domain, because of the condition $\phi_i(v_j) = \delta_{ij}$. Functions, which are piecewise smooth, and continuous on the whole domain, belong to $H^1(\Omega)$ [6]. Thus the functions ϕ_i are a basis of $V_{h,g}$.

To integrate over an element T, it is transformed into a reference element $\hat{T}$, via an affine linear function $G : \hat{T} \mapsto T$. This approach needs to define basis functions only on the reference element. Moreover, it is sufficient to compute the integrals $a(\cdot,\cdot)$, $k(\cdot,\cdot)$ and $f(\cdot)$ for the basis on the reference element exclusively. The integrals on other elements vary only by the constant $\det DG$. Integration on the reference element is done either exactly or using numerical quadrature rules.

Time Discretization

The software ABAQUS uses the Hilber-Hughes-Taylor α method (HHT-α) for time integration. This is an extension to the Newmark β method. Both methods operate directly on an ODE of second order. The implicit Euler method and the BDF2 method, described in the fluid discretization 3.1, expect a system of first order. Although every ODE of second order can be transformed into a system of first order ODE's, the application of the HHT-α method is more direct.

With the coefficients of the discretized acceleration $\mathbf{a}$ and velocity $\mathbf{w}$, the HHT-α method reads

$$\rho \mathbf{M} \mathbf{a}_{n+1} + \mathbf{K}(\mathbf{d}_{n+\alpha}) = \mathbf{F}_{n+\alpha}. \tag{3.39}$$

The values with subscript $n+\alpha$ are a convex combination of the values at time step $n+1$ and n, defined by

$$\mathbf{d}_{n+\alpha} = (1-\alpha)\mathbf{d}_{n+1} + \alpha \mathbf{d}_n, \tag{3.40}$$
$$\mathbf{w}_{n+\alpha} = (1-\alpha)\mathbf{w}_{n+1} + \alpha \mathbf{w}_n, \tag{3.41}$$
$$\mathbf{F}_{n+\alpha} = F((1-\alpha)t_{n+1} + \alpha t_n). \tag{3.42}$$

They are an approximation for the displacement, velocity and force vector at a intermediate time $t_{n+\alpha}$. The displacement and velocity at the new time step are given by

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{w}_n + \delta t^2 \left(\left(\frac{1}{2} - \beta \right) \mathbf{a}_n + \beta \mathbf{a}_{n+1} \right), \tag{3.43}$$
$$\mathbf{w}_{n+1} = \mathbf{w}_n + \Delta t \left((1-\gamma)\, \mathbf{a}_n + \gamma \mathbf{a}_{n+1} \right). \tag{3.44}$$

The time step size Δt is defined in the same manner as in time discretization of the fluid problem. The parameters α, β, and γ are characteristics of the HHT-α method. For a linearized stiffness matrix $\mathbf{K}$ the choice

$$\alpha \in \left[-\frac{1}{3}, 0\right], \ \beta = \frac{(1-\alpha)^2}{4}, \text{ and } \gamma = \frac{1-2\alpha}{2} \tag{3.45}$$

results in stable time stepping method, convergent of order two. The Newmark-β method arises from the choice $\alpha = 0$.

To solve the system (3.39)- (3.42), two approaches are possible. First, the equation (3.39) is solved in terms of $\mathbf{a}_{n+1}$, with the definition of $\mathbf{d}_{n+1}$ and $\mathbf{w}_{n+1}$ plugged in. The second possible approach is to rewrite $\mathbf{a}_{n+1}$ in terms of $\mathbf{d}_{n+1}$ using again the definition of $\mathbf{d}_{n+1}$ and $\mathbf{w}_{n+1}$. Then, the solution of equation (3.39) is computed with respect to $\mathbf{d}_{n+1}$. Since both approaches lead to a nonlinear system, the Newton method has to be applied. Chapter 5 discusses variants of the Newton method in detail.

3.3 Discretization of the Interface Constraints

This part discusses the discretization of the interface constraints. With the time and space discretized quantities, the conditions employed on the fluid and the structure are

$$v_{h,n+1} = w_{h,n+1}, \tag{3.46}$$

$$\sigma^f_{h,n+1} \cdot n + \sigma^s_{h,n+1} \cdot n = 0. \tag{3.47}$$

These constraints only hold at the mesh points of each part respectively, since in the discretization the boundary conditions only apply pointwise. In particular, they are not satisfied on the whole interface.

The partitioned approach in the solution of fluid-structure problems leads to different discretizations on both parts, exemplified in figure 6. Each part has a different mesh at the interface and thus the mesh points on the interface do not match. The quantities must be transfered to the mesh points of the other mesh accordingly.

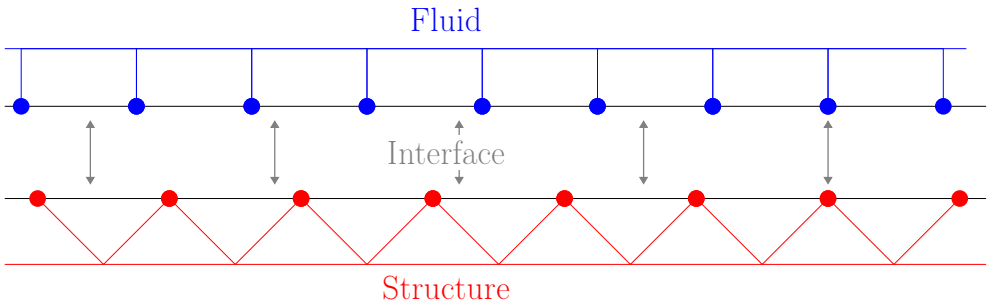


Figure 6: Non matching grids at the interface.

There are two possible approaches to transfer a quantity at a non matching mesh point x. On the one hand, only the direct neighbor elements or cells influence the value

at the mesh point x. This leads to local methods. On the other hand, the quantity at all elements or cells at the interface determines the quantity at x. Methods, relying on this approach, are called global.

A simple local method is to interpolate the quantity, see [10] and [21]. The interpolation takes place in the scope of the part, defining the quantity. For example, consider the displacement at a point x^f in the fluid mesh unknown. First, the point is projected onto the point x^S within the nearest element T of the structure mesh. Using the basis function ϕ_i^T defined on the element, the fluid displacement $\mathbf{d}_x$ is interpolated by

$$\mathbf{d}_x = \sum_{i=1}^{N_T} \phi_i^T(x^S)\mathbf{d}_i^T, \tag{3.48}$$

with the displacement of the nodes of the element $\mathbf{d}_i^T$. This method is called consistent interpolation, since it uses the basis functions of the displacement to interpolate the displacement in the fluid mesh. The interpolation of the force at structure node follows similar lines.

An example for the global method is the Mortar-element method [21] and [34]. This method is energy conserving, but requires the solution of a dense linear system. Interpolation methods are usually not conserving.

4 Coupling Methods

The solvers of each sub problem act mostly independent from each other, the interface constraints and the corresponding boundary conditions beeing the only intersection of the solvers. To couple both solvers, two different approaches are available. First, the interface quantities are exchanged only once between both solvers in each time step. This method is called *explicit* coupling or *weak* coupling. Enforcing the interface conditions in a nearly exact way requires more exchanges between both solvers. Methods, based on this, are called *implicit* coupling or *strong* coupling. The work of Felippa et al. [24] gives a detailed overview of coupling methods.

The first part of this chapter discusses explicit coupling schemes. For incompressible flows a special phenomena, the added-mass effect, occurs. A derivation of this effect is presented in the second part. Lastly, implicit methods are shortly described.

To simplify the notation for the fluid-structure interaction, solution operators for both sub parts are introduced. The solution operator

$$\mathcal{S}^{n+1} \,:\, \mathbf{f} \mapsto \mathbf{d}^{n+1} \tag{4.1}$$

maps a discretized interface force $\mathbf{f}$ to a new displacement $\mathbf{d}^{n+1}$ at time step t_{n+1}, using for example the discretized structure equations. Boundary conditions on other parts of the boundary than the interface are already included. Analogous, the solution operator

$$\mathcal{F}^{n+1} \,:\, \mathbf{d} \mapsto \mathbf{f}^{n+1} \tag{4.2}$$

computes the interface force $\mathbf{f}^{n+1}$, resulting from the solution to the Navier-Stokes equation at time step t_{n+1} with interface displacement $\mathbf{d}$. Since both operators are defined on different meshes, the inputs are interpolated, to match the mesh of each operator respectively. If there is no confusion about the time step, the superscript on the operator is dropped.

4.1 Explicit Coupling

An explicit coupling scheme uses the solution value at the current time step t_n as input for at least one operator. Therefore, the next time step can be computed with only one application of each solver. This shows the main advantage of the weak coupling schemes; they have minimal computational cost. The disadvantage of weak coupling schemes is that the interface constraints are not satisfied, since at least one solver uses old inputs.

The most common explicit method is the *staggered* coupling scheme, see [23]. With this method, the operators are executed in a sequential manner, for example

$$\mathcal{F}(\mathbf{d}^n) = \mathbf{f}^{n+1}, \tag{4.3}$$

$$\mathcal{S}(\mathbf{f}^{n+1}) = \mathbf{d}^{n+1}. \tag{4.4}$$

This sequence leads to satisfied dynamic interface constraints. However, the forces $\mathbf{f}^{n+1}$ are not the 'true' interface forces for the time step t_{n+1}, since they were computed on the

old domain Ω_f^n. Moreover, the kinematic interface constraints are not met. Thus, this sequence of solvers is called space discontinuous method. The data flow for this coupling method is depicted in figure 7.

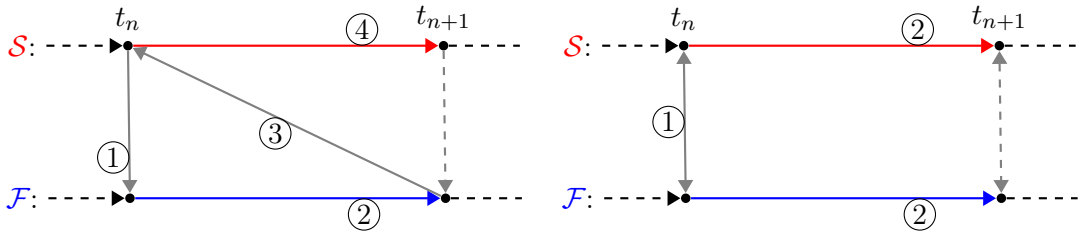
Swapping the order of execution, results in a space continuous method. This sequence satisfies the kinematic interface constraints, but not the dynamic constraints. Usually, it shows a greater instability [46], and therefore it is the worse choice.

To fully utilize parallel computing, the operators can be coupled in the following manner,

$$\mathcal{F}(\mathbf{d}^n) = \mathbf{f}^{n+1}, \tag{4.5}$$

$$\mathcal{S}(\mathbf{f}^n) = \mathbf{d}^{n+1}. \tag{4.6}$$

This is called *parallel* coupling, displayed idealized in figure 7. The disadvantage is that this scheme obviously satisfies no interface constraint. An additional exchange at the midpoint of the time step can improve the stability of the parallel coupling scheme [22].



(a) Staggered coupling scheme.

(b) Parallel coupling scheme.

Figure 7: Data flow for one time step, showing different coupling schemes. The red arrow implies one evaluation of the structure solver and the blue arrows one evaluation of the fluid solver. Gray arrows stand for data transfer.

All explicit schemes have in common that they violate at least one interface constraint. Therefore, the explicit coupling is not absolutely stable, even if each sub solver is absolutely stable respectively, see [59], [47]. To achieve a stable simulation, a small time step size has to be used. Moreover, there are two techniques to enhance the stability and accuracy of the explicit coupling schemes.

Subcycling [48] enables one solver to use a smaller time step size than the other solver. Often, fluid problems need to satisfy a small time step size, while larger sizes are possible for the structure problem. Instead of setting the structure time step size to the smaller one, the fluid solver computes several smaller time steps with out coupling with structure. The coupling between the two solvers happens only at the larger time step. If the structural solver needs a smaller time step, the roles are reversed.

Another stabilizing technique is the use of a structural predictor. Instead of the displacement of the current time step, a extrapolation of the displacement $\tilde{\mathbf{d}}$ is used as input of the fluid solver. The choice

$$\tilde{\mathbf{d}}^{n+1} = \mathbf{d}^n \tag{4.7}$$

gives a predictor of zeroth order. A first order accurate extrapolation reads as follows

$$\tilde{\mathbf{d}}^{n+1} = \mathbf{d}^n + \Delta t \mathbf{w}^n. \tag{4.8}$$

Higher order predictors are also possible [46].

4.2 Added-Mass Effect

Of special interest are explicit coupling strategies applied to fluid-structure interactions with incompressible flows. For these cases, the inexact interface constraints result in instabilities, which are known as the added-mass effect. Descriptions are found in [35] and [57]. For compressible fluid, these instabilities do not occure, if the time step size is sufficiently small, see [55].

The numerical implications of the added-mass effect are well understood. For density ratios ρ_s/ρ_f close to 1 the time stepping methods become unstable. Attempts to avoid these instabilities by reducing the time step size result in an earlier occurrence of the instabilities. Furthermore, the structural predictor (4.8) decreases the stability of the explicit coupling scheme, instead of improving it, as it would be the case for compressible flows.

The reasoning for the added-mass effect mostly relies on heuristics. Approaches to derive a theoretical background were made in the publications of Causin et al. [9] and Förster et al. [26], [25]. Both methods use simplified models to obtain a mathematical foundation for the added-mass effect.

Causin's at al. work focuses on the special case of a two dimensional flexible tube. There, they develop the so called added-mass operator, based on the weak formulation of the fluid-structure interaction problem. They then go on to show that, regardless of the actual spatial discretization, the chosen time stepping method for the structure is unconditionally unstable if the ratio of the two densities is close to one.

The work by Förster et al. uses the spatial discretized equations for a general fluid-structure interaction model. Therefore, they derive a discretized version of the added-mass operator. Since this derivation is applicable to more general problems than the on in Causin's publication, it is considered here.

To simplify the notation for the analysis, the spatial discretization is done via FEM for the structure and the fluid problem. The FEM for the Navier-Stokes equation is explained in [27] or [16]. Moreover the meshes of both sub problems coincide on the interface. Therefore, the exchange between the two solvers needs no interpolation of interface quantities. The basic equations for this part read as follows

$$\text{Fluid:} \qquad \rho^f \mathbf{M}^{\mathbf{f}} \mathbf{v}' + \mathbf{N}(\mathbf{u}) + \mathbf{G}\mathbf{p} = \mathbf{f}_\Gamma^f, \tag{4.9}$$

$$\mathbf{G}^T \mathbf{u} = 0, \tag{4.10}$$

$$\text{Structure:} \qquad \rho^s \mathbf{M}^{\mathbf{s}} \mathbf{d}'' + \mathbf{K}(\mathbf{d}) = \mathbf{f}_\Gamma^s, \tag{4.11}$$

$$\text{Interface:} \quad \mathbf{d}' = \mathbf{v}, \text{ and } \mathbf{f}_\Gamma^f + \mathbf{f}_\Gamma^s = 0. \tag{4.12}$$

The finite volume discretization leads to a similar problem, except for the continuity equation.

The derivation of the added-mass operator requires further assumptions on the discretized models:

1. The mesh motion has only small a impact on the fluid equations. This leads to time independent matrices. For early time steps this simplification is reasonable.

2. There are no external forces, except the interface forces, acting on the structure.

3. The fluid and structure stiffness vectors $\mathbf{N}$ and $\mathbf{K}$ can be omitted. This is justified for smaller time steps sizes, where the added-mass effect is stronger, because the mass matrices $\mathbf{M}^f$ and $\mathbf{M}^s$ dominate the equations.

These assumptions mostly lead to simplified equations and ease the notation later on. The removing of nonlinear factors appears rather strict, but, especially at earlier time steps, the acceleration terms determine the solution. Numerical examples show that the results still hold with all nonlinearities taken into consideration.

The sequentially staggered coupling (4.3) is chosen as coupling scheme. Other explicit coupling methods have worse stability, thus they are disregarded for the analysis. The aim is to rewrite the coupling scheme as only one time step of the structural solver. Therefore, the fluid force $\mathbf{f}^f$ is computed in terms of the interface acceleration.

Assumption 1 allows to apply the time derivative on the divergence of the velocity,

$$\frac{d\mathbf{G}^T\mathbf{v}}{dt} = \mathbf{G}^T\mathbf{v}'. \tag{4.13}$$

With the separation of the fluid equations the inner degrees of freedom $\mathbf{v}'_I$ and the degrees of freedom on the interface $\mathbf{v}'_\Gamma$, equation(4.9) now reads as

$$\begin{bmatrix} \rho^f \mathbf{M}^f_{\mathrm{II}} & \rho^f \mathbf{M}^f_{\mathrm{I}\Gamma} & \mathbf{G}_\mathrm{I} \\ \rho^f \mathbf{M}^f_{\Gamma\mathrm{I}} & \rho^f \mathbf{M}^f_{\Gamma\Gamma} & \mathbf{G}_\Gamma \\ \mathbf{G}^T_\mathrm{I} & \mathbf{G}^T_\Gamma & 0 \end{bmatrix} \begin{bmatrix} \mathbf{v}'_\mathrm{I} \\ \mathbf{v}'_\Gamma \\ p \end{bmatrix} = \begin{bmatrix} 0 \\ f^f_\Gamma \\ 0 \end{bmatrix} \tag{4.14}$$

The last block equation yields

$$\mathbf{G}^T_\mathrm{I}\mathbf{v}'_\mathrm{I} = -\mathbf{G}^T_\Gamma\mathbf{v}'_\Gamma, \tag{4.15}$$

and the first block equation is

$$\mathbf{v}'_\mathrm{I} = -\frac{1}{\rho^f}(\mathbf{M}^f_{\mathrm{II}})^{-1}(\rho^f\mathbf{M}^f_{\mathrm{I}\Gamma}\mathbf{v}'_\Gamma + \mathbf{G}_\mathrm{I}\mathbf{p}). \tag{4.16}$$

By inserting the equation of the velocity on the inner degrees of freedom into the last block equation

$$\frac{1}{\rho^f}\mathbf{G}^T_\mathrm{I}(\mathbf{M}^f_{\mathrm{II}})^{-1}(\mathbf{M}^f_{\mathrm{I}\Gamma}\mathbf{v}'_\Gamma + \mathbf{G}_\mathrm{I}\mathbf{p}) = \mathbf{G}^T_\Gamma\mathbf{v}'_\Gamma, \tag{4.17}$$

the pressure, expressed in terms of the fluid interface velocity, is

$$\mathbf{p} = \rho^f(\mathbf{G}^T_\mathrm{I}(\mathbf{M}^f_{\mathrm{II}})^{-1}\mathbf{G}_\mathrm{I})^{-1}(\mathbf{G}^T_\Gamma - \mathbf{G}^T_\mathrm{I}(\mathbf{M}^f_{\mathrm{II}})^{-1}\mathbf{M}^f_{\mathrm{I}\Gamma})\mathbf{v}'_\Gamma. \tag{4.18}$$

Rewriting the velocity inside the domain with the pressure equation above results in

$$\mathbf{v}'_{\mathrm{I}} = -\Big\{(\mathbf{M}^f_{\mathrm{II}})^{-1}\mathbf{M}^f_{\mathrm{II}\Gamma} + \mathbf{G}_{\mathrm{I}}(\mathbf{G}_{\mathrm{I}}^T(\mathbf{M}^f_{\mathrm{II}})^{-1}\mathbf{G}_{\mathrm{I}})^{-1} \cdot(\mathbf{G}_\Gamma^T - \mathbf{G}_{\mathrm{I}}^T(\mathbf{M}^f_{\mathrm{II}})^{-1}\mathbf{M}^f_{\mathrm{II}\Gamma}\Big\}\,\mathbf{v}'_\Gamma. \tag{4.19}$$

The middle block equation gives the following relation for the fluid interface force and fluid interface acceleration

$$\begin{aligned}\mathbf{f}^f_\Gamma &= \rho^f \mathbf{M}^f_{\Gamma\mathrm{I}}\mathbf{v}'_{\mathrm{I}} + \rho^f \mathbf{M}^f_{\Gamma\Gamma}\mathbf{v}'_\Gamma + \mathbf{G}_\Gamma \mathbf{p} \\ &= \rho^f \mathcal{M}_A \mathbf{v}'_\Gamma,\end{aligned} \tag{4.20}$$

with the added-mass operator

$$\begin{aligned}\mathcal{M}_A = &\left(\mathbf{M}^f_{\Gamma\mathrm{I}}\left(\mathbf{M}^f_{\mathrm{II}}\right)^{-1}\mathbf{G}_{\mathrm{I}} - \mathbf{G}_\Gamma\right)\left(\mathbf{G}_{\mathrm{I}}^T\left(\mathbf{M}^f_{\mathrm{II}}\right)^{-1}\mathbf{G}_{\mathrm{I}}\right)^{-1} \\ &\cdot\left(\mathbf{G}_{\mathrm{I}}^T\left(\mathbf{M}^f_{\mathrm{II}}\right)^{-1}\mathbf{M}^f_{\mathrm{II}\Gamma} - \mathbf{G}_\Gamma^T\right) + \mathbf{M}^f_{\Gamma\Gamma} - \mathbf{M}^f_{\Gamma\mathrm{I}}\left(\mathbf{M}^f_{\mathrm{II}}\right)^{-1}\mathbf{M}^f_{\mathrm{II}\Gamma}.\end{aligned} \tag{4.21}$$

The term $(\mathbf{G}_{\mathrm{I}}^T(\mathbf{M}^f_{\mathrm{II}})^{-1}\mathbf{G}_{\mathrm{I}})^{-1}$ is positive definite, since $\mathbf{M}^f_{\mathrm{II}}$ is positive definite and $\mathbf{G}_{\mathrm{I}}$ has full rank. With $\mathbf{M}^f_{\Gamma\mathrm{I}} = (\mathbf{M}^f_{\mathrm{II}\Gamma})^T$ it follows that the first term is also positive. The remainig terms are the Schur complement of the fluid mass matrix $\mathbf{M}^f$ and therefore positive. Thus, the operator $\mathcal{M}_A$ is positive and additionally symmetric. It represents the condensed solution operator of the fluid problem. The finite volume discretization would result in a similar linear relation between the fluid interface acceleration and force, under the assumptions above. For consistent meshes, the force equilibrium leads to

$$\mathbf{f}^s_\Gamma = -\rho^f\,\mathcal{M}_A\,\mathbf{v}'_\Gamma. \tag{4.22}$$

Separating the degrees of freedom for the structural equation, similar to the approach above, gives the reduced structural equation

$$\rho^s \begin{bmatrix}\mathbf{M}^s_{II} & \mathbf{M}^s_{I\Gamma}\\ \mathbf{M}^s_{\Gamma I} & \mathbf{M}^s_{\Gamma\Gamma}\end{bmatrix}\begin{bmatrix}\mathbf{d}''_I\\ \mathbf{d}''_\Gamma\end{bmatrix} = \begin{bmatrix}0\\ -\rho^f\,\mathcal{M}_A\,\mathbf{v}'_\Gamma\end{bmatrix}. \tag{4.23}$$

Now, the staggered coupling scheme consists only of the structural equation, with the solution of the fluid problem hidden in the added-mass operator. In this formulation it becomes evident, why the phenomena is called added-mass effect. Since the fluid interface acceleration is an approximation of the structural acceleration, the equation shows that the fluid solution acts on the structural acceleration at the interface additionally through the operator $\mathcal{M}_A$.

The ordinary differential equation is discretized in time with the HHT-α method (3.39). For $\alpha = 0$, the acceleration reads as

$$\mathbf{d}'' = \frac{1}{\Delta t^2}\left(2\mathbf{d}^{n+1} - 5\mathbf{d}^n + 4\mathbf{d}^{n-1} - \mathbf{d}^{n-2}\right). \tag{4.24}$$

The computation of interface acceleration $\mathbf{v}_\Gamma'^{n+1}$ needs to be formulated in terms of the displacements $\mathbf{d}$ of the interface. Applying the backwards Euler method to the interface acceleration results in

$$\mathbf{v}_\Gamma'^{n+1} = \frac{\hat{v}_\Gamma^{n+1} - \hat{v}_\Gamma^n}{\Delta t}, \tag{4.25}$$

where $\hat{v}_\Gamma^{n+1}$ is the ALE reference velocity at the interface. With the definition of the reference velocity $\hat{v}$ (3.24) and use of the staggered coupling scheme, the interface acceleration is given by

$$\mathbf{v}_\Gamma'^{n+1} = \frac{\mathbf{d}_\Gamma^n - 2\mathbf{d}_\Gamma^{n-1} + \mathbf{d}_\Gamma^{n-2}}{\Delta t^2}, \tag{4.26}$$

since the fluid solver can employ only the previous displacements. Overall, the time stepping method is formulated as

$$\frac{1}{\Delta t^2}\rho^s \mathbf{M}^s(2\mathbf{d}^{n+1} - 5\mathbf{d}^n + 4\mathbf{d}^{n-1} - \mathbf{d}^{n-2}) = \mathbf{f}^{n+1}, \tag{4.27}$$

with

$$\mathbf{f}^{n+1} = \begin{bmatrix} 0 \\ -\frac{1}{\Delta t^2}\rho^f \mathcal{M}_A \left(\mathbf{d}_\Gamma^n - 2\mathbf{d}_\Gamma^{n-1} + \mathbf{d}_\Gamma^{n-2}\right) \end{bmatrix}. \tag{4.28}$$

Lumping the structural mass matrix into a diagonal matrix $\mathbf{M}_l$ and limiting the equation to the degrees of freedom on the interface, simplifies the equation to,

$$\begin{aligned} &\rho^s \mathbf{M}_l \left(2\mathbf{d}_\Gamma^{n+1} - 5\mathbf{d}_\Gamma^n + 4\mathbf{d}_\Gamma^{n-1} - \mathbf{d}_\Gamma^{n-2}\right) \\ &+ \rho^f \mathcal{M}_A \left(\mathbf{d}_\Gamma^n - 2\mathbf{d}_\Gamma^{n-1} + \mathbf{d}_\Gamma^{n-2}\right) = 0. \end{aligned} \tag{4.29}$$

Since $\mathcal{M}_A$ is positive, each vector has a representation in terms of the eigenvectors $\mathbf{w}_i$,

$$\mathbf{d}_\Gamma^l = \sum_i d_i^l \mathbf{w}_i. \tag{4.30}$$

Therefore, the coefficient d_i need to solve

$$\begin{aligned} &\rho^s m_i \left(2\mathbf{d}_i^{n+1} - 5\mathbf{d}_i^n + 4\mathbf{d}_i^{n-1} - \mathbf{d}_i^{n-2}\right) \\ &+ \rho^f \mu_i \left(\mathbf{d}_i^n - 2\mathbf{d}_i^{n-1} + \mathbf{d}_i^{n-2}\right) = 0, \end{aligned} \tag{4.31}$$

where m_i is the mass assigned to the interface node i, and μ_i is the eigenvalue corresponding to the eigenvector $\mathbf{w}_i$. The eigenvalues are not considered in detail. For the model case examined by Causin et al., they give a derivation for the eigenvalues of the continuous added mass operator, indicating a dependency on the geometry.

This defines a characteristic polynomial for the multistep method (4.31),

$$p_i(\lambda) = \rho^s m_i(2\lambda^3 - 5\lambda^2 + 4\lambda - 1) + \rho^f \mu_i(\lambda^2 - 2\lambda + 1) = 0. \tag{4.32}$$

The multistep method is stable, if and only if the roots of each polynomial have an absolute value of less than 1, see for example [15]. If the value at -1 is greater than zero,

the polynomial has a root strictly less then -1, since it is of third degree. This leads to the unstability condition

$$p_i(-1) = -12\rho^s m_i + 4\rho^f \mu_i > 0 \quad \Leftrightarrow \quad \frac{\rho^s}{\rho^f} < \frac{\mu_i}{3m_i}. \tag{4.33}$$

Thus, for small enough density ratios the staggered coupling scheme described above is unstable. Förster et al. show the a more general result for arbitrary multistep methods used for the time discretization of (4.23).

Theorem 3 (Förster)**.** *Under the assumptions 1-3, every sequentially staggered scheme for the solution of* (4.9) *has a mass ration* ρ^s/ρ^f *at which the overall algorithm becomes unstable.*

The derivation for the added-mass effect presented here, relies heavily on the assumptions 1-3. Without the simplifications it seems quite difficult to obtain a similar result. Numerical examples, such as in chapter 6, show that the result still holds for the general fluid-structure interaction model.

Influences of other parameters are not captured by this approach. By omitting nonlinear effects of other material parameters, such as the fluid viscosity or the Young's modulus and Poisson's ratio of the structure, their impact on the stability cannot be measured here. Förster et al. note that the structural stiffness has a stabilizing effect, while the fluid viscosity acts destabilizing.

4.3 Implicit Coupling

The advantage of explicit coupling schemes stems from their minimal computational cost. For some cases, especially for incompressible flows combined with low structural density, as demonstrated by the previous section, explicit coupling is not a viable choice. In those cases the instability of explicit methods outweighs the low computational cost.

The aim now is to enforce exact interface constraints, up to some precision, through the equilibrium

$$\begin{aligned} \mathcal{F}\left(\mathbf{d}_*^{n+1}\right) &= \mathbf{f}_*^{n+1}, \\ \mathcal{S}\left(\mathbf{f}_*^{n+1}\right) &= \mathbf{d}_*^{n+1}. \end{aligned} \tag{4.34}$$

With these interface values, the stability analysis in 4.2 does not hold any more, because the exact interface constraints result in both terms of the characteristic polynomial (4.32) having the same degree. This problem can be reformulated as one of the following equivalent fixpoint systems,

(a) $\mathcal{S}\left(\mathcal{F}(\mathbf{d}^{n+1})\right) \overset{!}{=} \mathbf{d}^{n+1}$,

(b) $\mathcal{F}\left(\mathcal{S}(\mathbf{f}^{n+1})\right) \overset{!}{=} \mathbf{f}^{n+1}$,

(c) $\begin{bmatrix} 0 & \mathcal{F} \\ \mathcal{S} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{f}^{n+1} \\ \mathbf{d}^{n+1} \end{bmatrix} \overset{!}{=} \begin{bmatrix} \mathbf{f}^{n+1} \\ \mathbf{d}^{n+1} \end{bmatrix}$.

The equilibrium is achived by coupling the solvers multiple times within one time step. Without advancing in time, the output of the solvers serves as the input for the next iterations. These repeated application of the solvers, within one time step, are called inner iterations. If the inner iterations converge, the interface constraints are satisfied, up to the precision criteria.

The inner iterations can consist of sequentially or parallel information exchange between the two solvers, depending on which fixpoint formulation (a)-(c) is considered. At time step t^{n+1} the sequential scheme for fixpoint system (a) leads to the inner iteration

$$\begin{aligned} \mathcal{F}(\mathbf{d}_k) &= \mathbf{f}_{k+1}, \\ \mathcal{S}(\mathbf{f}_{k+1}) &= \mathbf{d}_{k+1}, \end{aligned} \tag{4.35}$$

where the time super script of the quantities has been dropped. Figure 8 depicts a schematic representation of this coupling method. A predictor gives the initial input $\mathbf{d}_0 = \tilde{\mathbf{d}}^{n+1}$. Usually, the convergence criteria

$$\frac{||\mathbf{d}_{k+1} - \mathbf{d}_k||}{||\mathbf{d}_{k+1}|| + ||\mathbf{d}_k||} < \varepsilon \tag{4.36}$$

is enforced, with some norm $||\cdot||$ and a desired precision ε. Alternatively, both interface quantities must meet this criteria simultaneously. This is one step of the Gauß-Seidel method, therefore the sequentially staggered scheme is called Gauß-Seidel iteration. Similarly, the parallel coupling scheme results in a Jacobi method and this scheme is called Jacobi iteration.

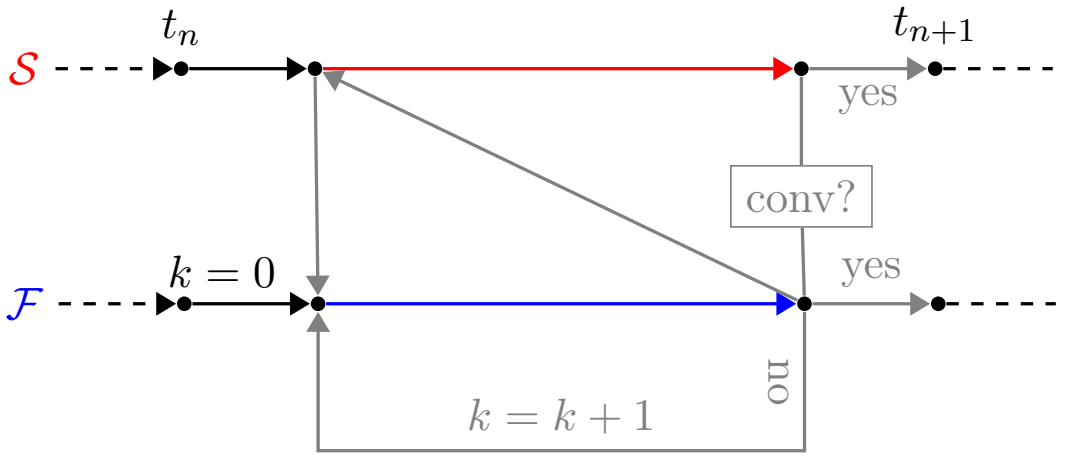


Figure 8: Schematic view of implicit coupling with fixpoint formulation (a). The inner iterations are counted by k.

Iterative methods satisfy the interface constraints up to the precision ε. These implicit methods suffer from the disadvantage that they need multiple solver applications to achieve convergence. Thus, the computational cost is several factors higher than for the

explicit methods. A deeper analysis on the convergence of the Gauß-Seidel and Jacobi method can be found in [50].

A common method to reduce the number of inner iterations is the under- or over-relaxation, see for example [43]. Instead of using the output $\mathbf{d}_{k+1}^{out}$ of the Gauß-Seidel iteration (4.35) as new iterate for the next inner iteration, a linear combination of the new output and the previous iterate is applied,

$$\mathbf{d}_{k+1} = \omega_{k+1}\mathbf{d}_{k+1}^{out} + (1 - \omega_{k+1})\mathbf{d}_k, \tag{4.37}$$

with $\omega_k \in (0, 2)$. The right choice of ω_k is deeply problem dependent. Simply setting the relaxation parameter to constant $\omega_k = \omega$ can lead to a high number of iterations or no convergence at all, if the constant is inferior, see the numerical examples in [43].

With the aid of *Aitken's method*, a faster convergence can be accomplished. Rather than setting ω_k to a constant value, the parameter is adjusted in each iteration. First, an Aitken factor μ_{k+1} is computed by

$$\mu_{k+1} = \mu_k + (\mu_k - 1)\frac{(\Delta\mathbf{d}_k - \Delta\mathbf{d}_{k+1})^T \cdot \Delta\mathbf{d}_{k+1}}{||\Delta\mathbf{d}_k - \Delta\mathbf{d}_{k+1}||_2^2}, \tag{4.38}$$

with the difference

$$\Delta\mathbf{d}_i = \mathbf{d}_{i-1} - \mathbf{d}_i^{out}. \tag{4.39}$$

Then, the relaxation parameter in(4.37) reads as follows

$$\omega_{k+1} = 1 - \mu_{k+1}. \tag{4.40}$$

The advantage of this method is that it automatically chooses an appropriate ω_{k+1} and numerical examination indicates that the convergence is faster than the constant relaxation, see [43].

5 Quasi-Newton Methods

This chapter introduces quasi-Newton methods, to accelerate the convergence of the inner iterations of the implicit coupling schemes. To bring the problem of section 4.3 into more general terms, the aim is to solve the fixpoint equation

$$H(x) = x, \tag{5.1}$$

with some nonlinear function $H : D \subset \mathbb{R}^N \mapsto \mathbb{R}^N$. The dimension N of the domain depends on the discretization of the interface and the coupling scheme for the inner iterations. In the case of the sequentially staggered scheme of (4.35) the fixed point problem H is given by

$$H(\mathbf{d}) = \mathcal{S}\left(\mathcal{F}\left(\mathbf{d}\right)\right) \tag{5.2}$$

and therefore the dimension of the domain of H is the number of degrees of freedom for the interface displacement. Other coupling choices for the inner iterations result in different functions H. Unrelated to the particular selection of coupling schemes, function evaluations of H are expensive, since it computes one time step of each sub problem. Furthermore, the function H is assumed to act as a black box. This means, no further information than the output to a specific input is available, especially no derivatives of H.

The problem (5.1) is equivalent to solving the nonlinear system

$$R(x) := H(x) - x = 0. \tag{5.3}$$

Before formulating a method to solve this nonlinear system, the residuum needs some regularities. The function H needs to be continuously differentiable and there exists exactly one fixpoint x^* of H in the domain D. Furthermore, the Jacobian matrices of H and R must be invertible and bounded for all $x \in D$.

The most widely used solver for the nonlinear system (5.3), Newton's Method, seeks a new iteration improving on the previous

$$x_{k+1} = x_k + p. \tag{5.4}$$

The new iterate would be ideal, if the increment satisfies $R(x_k + p) = 0$. To find an approximate increment, the truncated Taylor series

$$R(x_k + p) \approx R(x_k) + \mathrm{D}R(x_k)p \tag{5.5}$$

gives the relation

$$\mathrm{D}R(x_k)p = -R(x_k). \tag{5.6}$$

This results in the iteration

$$x_{k+1} = x_k - \mathrm{D}R(x_k)^{-1}R(x_k). \tag{5.7}$$

Under the assumptions above, convergence of this sequence is given by the following theorem, if the initial vector x_0 is close to the fixpoint x^*.

Theorem 4. *Let $D \subset \mathbb{R}^N$ be open and convex. The function $R : D \mapsto \mathbb{R}^N$ is continuously differentiable with invertible Jacobi matrix $\mathrm{D}R(x)$ and*

$$||\mathrm{D}R(x)^{-1}||_2 \leq \alpha \quad \forall\, x \in D. \tag{5.8}$$

Additionally, $\mathrm{D}R(x)$ is Lipschitz continuous on D with constant β. The point x^* is root of R and x_0 is the initial vector with $x_0 \in B(x^*, \gamma)$, where γ is small enough, such that $B(x^*, \gamma) \subset D$ and $0 < \gamma < \frac{2}{\alpha\beta}$. Then, the sequence defined by (5.7) stays in the ball $B(x^*, \gamma)$ and converges quadratically to x^*, i.e.

$$||x_{k+1} - x^*||_2 \leq \frac{\alpha\beta}{2}||x_k - x^*||_2^2 \quad \forall\, k \in \mathbb{N}. \tag{5.9}$$

A prove can be found in [29]. Generally, the algorithms presented in the following sections do not have convergence results. Therefore, this theorem, especially its assumptions, acts as a surrogate.

Usually, the nonlinear equations of the discretized fluid and structure model in section 3.1 and 3.2 are solved with Newton's method. The Jacobi matrix is computed analytically, depending on the discretization technique. In contrast, partitioned fluid-structure interactions use both solvers as black boxes. Therefore, implicit coupling schemes cannot rely on any derivatives of the sub problem solvers. Attempts to approximate the Jacobian through finite differences can be found in [42]. This approximative version of Newton's method has the same convergence properties, but one Newton iteration needs at least 10 solutions to each sub problem. The methods described in the next part do not converge quadratically, but they require only one application of both sub problem solvers for one iteration.

5.1 Generalized Broyden Method

This part derives a quasi-Newton variant, the generalized Broyden method, which approximates the Jacobi matrix with only one solution to each sub problem per inner iteration. It is an extension to the classical Broyden method [7], and was introduced in [20]. For $m+1$ iterates $x_{k-m}, \ldots, x_k$ the Taylor series expansion leads to the following *secant equations* for the Jacobian of the residuum

$$\mathrm{D}R(x_k)\Delta x_i = \Delta R_i \quad \forall\, i = k-m, \ldots, k-1, \tag{5.10}$$

with an error of order $\mathcal{O}(||x_i - x_k||^2)$. The differences Δx_i, ΔR_i are defined by

$$\begin{aligned} \Delta x_i &= x_i - x_k, \\ \Delta R_i &= R(x_i) - R(x_k). \end{aligned} \tag{5.11}$$

Therefore, an approximation G_k of $\mathrm{D}R(x_k)$ should satisfy the same secant equations

$$G_k\Delta x_i = \Delta R_i \quad \forall\, i = k-m, \ldots, k-1, \tag{5.12}$$

and an approximation J_k of $\mathrm{D}R(x_k)^{-1}$ the equations

$$J_k \Delta R_i = \Delta x_i \quad \forall\, i = k-m, \ldots, k-1. \tag{5.13}$$

For $m < N$, both sets of equations result in underdetermined linear systems

$$G_k W = V, \text{ and } J_k V = W, \tag{5.14}$$

with the difference matrices

$$\begin{aligned} V &= [\Delta R_{k-m}, \ldots, \Delta R_k] \in \mathbb{R}^{d \times k}, \\ W &= [\Delta x_{k-m}, \ldots, \Delta x_k] \in \mathbb{R}^{d \times k}. \end{aligned} \tag{5.15}$$

It is always assumed that both matrices have full rank. If $m = N$ and $k > N+1$, the approximation of $\mathrm{D}R(x_k)$ is given by

$$G_k = V\, W^{-1}. \tag{5.16}$$

Also, the relation $G_k^{-1} = J_k$ holds. Since the dimension of the nonlinear system R is associated with the degrees of freedom on the interface of the fluid-structure problem, it is not feasible to have n evaluations of R.

To derive suitable matrices G_k or J_k in the case $m < N$, a second condition must be imposed. First, only the matrix J_k is considered. It should give the same outcome as J_{k-m} for a direction q if and only if the direction is orthogonal to $\mathrm{Im}(W)$ or $\mathrm{Im}(V)$ respectively. In the following, only J_k is considered, since the derivation for G_k is analogous. For J_k this no-change condition reads

$$(J_k - J_{k-m})q = 0 \quad \Leftrightarrow \quad \forall\, q \perp \mathrm{Im}(V). \tag{5.17}$$

With the identity $\mathrm{Ker}(J_k - J_{k-m})^{\perp} = \mathrm{Im}((J_k - J_{k-m})^T)$, this shows that $\mathrm{Im}(V) = \mathrm{Im}((J_k - J_{k-m})^T)$. Thus there exists a matrix $Z \in \mathbb{R}^{m \times N}$ such that

$$(J_k - J_{k-m})^T = V\, Z. \tag{5.18}$$

Transposing and multiplying with V from right leads to

$$(J_k - J_{k-m})V = Z^T V^T V. \tag{5.19}$$

Therefore, the matrix Z can be computed by

$$Z^T = (J_k - J_{k-m})V(V^T V)^{-1} = (W - J_{k-m}V)(V^T V)^{-1}, \tag{5.20}$$

using the secant equation (5.13) for J_k. Plugging the formulation for Z into (5.18) yields the update formula for the approximate inverse Jacobian

$$J_k = J_{k-m} + (W - J_{k-m}V)(V^T V)^{-1} V^T. \tag{5.21}$$

The new iterate x_{k+1} is now obtained through

$$\begin{aligned} x_{k+1} &= x_k - J_k R(x_k) \\ &= x_k - J_{k-m} R(x_k) - (W - J_{k-m}V)(V^T V)^{-1} V^T R(x_k). \end{aligned} \tag{5.22}$$

An alternative derivation [19] of this update formula seeks J_k as a minimizer of

$$||J_k - J_{k-m}||_F^2 \tag{5.23}$$

under the constraint that J_k satisfies the secant equation (5.13).

The approximation of the Jacobi matrix $\mathrm{D}R(x_k)$ follows the same procedure. Additionally, the Sherman-Morrison-Woodbury formula provides an explicit formulation of the inverse of G_k,

$$G_k^{-1} = G_{k-m}^{-1} + (W - G_{k-m}^{-1}V)(WG_{k-m}^{-1}V)^{-1} WG_{k-m}^{-1}. \tag{5.24}$$

Therefore, the update scheme involving the approximate Jacobian requires only one matrix vector multiplication with G_k^{-1} instead of solving a system of linear equations with G_k.

The algorithm 1 summarizes the generalized Broyden method, resulting from choosing the inverse Jacobi matrix as a starting point. It already contains the additional practical considerations from section 5.3. Currently, there is no result on the convergence of the algorithm for an arbitrary number of secant equations m. The following two sections consider special cases of m, 1 and ∞, and their resulting algorithms. Other possible choices are not examined, since finding the ideal number of secant equations is highly problem dependent, see [20].

Algorithm 1: Generalized Broyden Method

Input : initial guesses x_0 and J_0, number of secant equations m
Output: converged solution x_*, approximative Jacobian J_*

for $k = 0, 1, \ldots$ **until** *converged* **do**
- $y_k = H(x_k)$
- $R_k = y_k - x_k$
- $V = [\Delta R_{k-m}, \ldots, \Delta R_{k-1}]$, with $\Delta R_i = R_i - R_k$
- $W = [\Delta x_{k-m}, \ldots, \Delta x_{k-1}]$, with $\Delta x_i = x_i - x_k$
- $\alpha = \arg\min_{\beta \in \mathbb{R}^d} ||V\beta + R_k||$
- $x_{k+1} = x_k - J_{k-m}R_k - (W - J_{k-m}V)\alpha$
- **if** $k > m$ **then**
  - $J_{k+1} = J_{k-m} + (W - J_{k-m}V)(V^T V)^{-1} V^T$

Remark. *It is possible to construct the difference matrices V and W in a different manner. For that, the differences Δx_i and ΔR_i are defined by*

$$\begin{aligned} \Delta x_i &= x_{i+1} - x_i \\ \Delta R_i &= R(x_{i+1}) - R(x_i) \end{aligned}. \tag{5.25}$$

Then, the matrices V and W are not computed anew each iteration. Instead, the current secant equation $\mathrm{D}R(x_k)\Delta x_k \approx \Delta R_k$ is appended to V and W. The disadvantage of this approach is that the Jacobian does not satisfies $\mathrm{D}R(x_k)W \approx V$ anymore. In the works [20] and [19] it is shown numerically that the generalized Broyden method still converges. Since it reduces the computational cost of the difference matrices to a constant, this approach is used for the implementation of these methods.

Case $m = 1$

The choice $m = 1$ leads to the well known Broyden method. Although this is one of the standard method for the solution of nonlinear equations, the Broyden method has not been applied to implicit coupling of fluid-structure interactions until recently, see [5]. The Broyden variant related to the approximation of the inverse Jacobian is known as Broyden's 'bad' method, while the other variant is called Broyden's 'good' method. In spite of this notation, there is no general evidence in favor of one or the other method.

Broyden's bad method satisfies only one secant equation $J_k\Delta R_{k-1} = \Delta x_{k-1}$, while minimizing $||J_k - J_{k-1}||_F^2$. This leads to the update formula

$$\begin{aligned} J_k &= J_{k-1} + \frac{\Delta x_{k-1} - J_{k-1}\Delta R_{k-1}}{||\Delta R_{k-1}||_2^2}\Delta R_{k-1}^T \\ x_{k+1} &= x_k - J_k R(x_k). \end{aligned} \tag{5.26}$$

Noting that $J_{k-1}\Delta R_{k-1} = J_{k-1}R(x_k) + \Delta x_{k-1}$ reduces the rank one update to

$$J_k = J_{k-1} - \frac{J_{k-1}R(x_k)}{||\Delta R_{k-1}||_2^2}\Delta R_{k-1}^T. \tag{5.27}$$

For any initial guess J_0 close to $\mathrm{D}R(x_*)^{-1}$ this method converges superlinearly, see [8].

Case $m = \infty$

Choosing $m = \infty$, results in a method, satisfying always the maximal amount of secant equations. The appropriate minimizer is $||J_k - J_0||_F^2$. The approximation J_{k+1} is never computed explicitly, since it only depends on the initial guess matrix and the difference matrices V and W. Therefore, the corresponding update formula reads as follows

$$x_{k+1} = x_k - J_0R(x_k) - (W - J_0V)(V^TV)^{-1}V^TR(x_k). \tag{5.28}$$

This method depends greatly on the choice of the initial guess J_0. In case of black box coupling, no further information on the solvers than their output is available. Thus, no educated guess of the initial matrix J_0 based on properties of the sub problems is possible. Instead, a reasonable guess is a scaled unity matrix, i.e. $J_0 = -\beta I$, $\beta > 0$. The Anderson mixing scheme, introduced in [1], is equivalent to the generalized Broyden method with this choice of m, see [20]. Recently, the r-linear convergence of the Anderson mixing has been shown, see [51], under the additional assumption that the function H is a contraction.

The fixpoint problem is part of a time stepping method. At time step t_{n+1} the an approximation J_*^n from the previous time step has already been computed. By using the last approximation J_*^n as initial guess, information from the previous time step is available for the generalized Broyden method. This can lead to a faster convergence, since, heuristically speaking, the fixpoint problems for two consecutive time steps are closely related, especially for small time steps. The effect of the initial guess is examined in the numerical example 6.

5.2 Inverse Generalized Broyden Method

The publications [14] and [5] introduce iterative methods to solve the fixed point problem (5.1), occurring in fluid-structure interactions. Although they can be classified as quasi-Newton methods, see [38], their connection to other quasi-Newton methods like the generalized Broyden methods is difficult to recognize. These methods solve a different nonlinear system, resulting in slightly adjusted methods, which is introduced in the following.

Instead of the residuum $R(x)$, the 'inverse' residuum

$$\tilde{R}(y) = y - H^{-1}(y) \tag{5.29}$$

is considered. The inverse function of H exist in a neighborhood around x_*, because H is continuously differentiable with invertible Jacobian. Since H and H^{-1} have the same fixed points, both problems are equivalent in the domain D. Applying the generalized Broyden method, described above, leads to the iteration scheme

$$y_{k+1} = y_k - J_k \tilde{R}(y_k), \tag{5.30}$$

where J_k minimizes $||J_k - J_0||_F^2$ with an appropriate choice for the initial guess.

It is possible to choose J_k as minimizer of $||J_k - J_{k-m}||$ satisfying $m < \infty$ secant equations. Similarly to the generalized Broyden case, the only variant of interest for this type is the choice $m = 1$. In every test case considered, this inverse generalized Broyden method diverged and therefore it is not examined further. Currently, it is not clear if this behavior holds true for all $m < \infty$.

The inverse function H is not given explicitly. Therefore, the inverse residuum cannot be evaluated for arbitrary $y \in \mathbb{R}^N$. Instead, computing an intermediate update x_{k+1} with the update formula above (5.30) and then applying the function H, i.e.

$$\begin{aligned} x_{k+1} &= y_k - J_k \tilde{R}(y_k), \\ y_{k+1} &= H(x_{k+1}), \end{aligned} \tag{5.31}$$

gives a new iterate, with an easy to determine residuum. In this case, the inverse residuum is simply

$$\tilde{R}(y_{k+1}) = H(x_{k+1}) - x_{k+1} = R(x_{k+1}). \tag{5.32}$$

Thus, the quasi-Newton iterations needs only one calculation of H, the same amount as for the generalized Broyden method. In the following, this quasi-Newton method is denoted as *inverse generalized Broyden method*. Algorithm 2 displays the full method.

Algorithm 2: Inverse Generalized Broyden Method

Input : initial guesses x_0 and J_0

Output: converged solution x_*, approximative Jacobian J_*

for $k = 0, 1, \ldots$ **until** *converged* **do**

- $y_k = H(x_k)$
- $R_k = y_k - x_k$
- $V = [\Delta R_{k-m}, \ldots, \Delta R_{k-1}]$, with $\Delta R_i = R_i - R_k$
- $W = [\Delta y_{k-m}, \ldots, \Delta y_{k-1}]$, with $\Delta y_i = y_i - y_k$
- $\alpha = \arg\min_{\beta \in \mathbb{R}^d} ||V\beta + R_k||$
- $x_{k+1} = y_k - J_0 R_k - (W - J_0 V)\alpha$

The selection of J_0 characterizes the method further. The same choices as for the generalized Broyden method are possible. The algorithm, obtained by choosing $J_0 = -\beta I$, is closely related to the Anderson mixing method.

Bogaers introduced the *multi-vector update quasi-Newton* (MVQN) method in [5] using the same update formula (5.31) with the Jacobi approximate defined by

$$J_k = J_*^n + (W - J_*^n V) \left(V^T V\right)^{-1} V^T. \tag{5.33}$$

Carrying over the approximation of the fixpoint problem from the previous time step in the inverse generalized Broyden's method, i.e. $J_0 = J_*^n$, results in the same update method for J_k. Thus, the MVQN method is a special case of the inverse generalized Broden's method.

If no previous approximation J_*^n exist, for example in the very first time step of a coupled simulation, Bogaers suggested to use $J_0 = 0$ as an initial guess. In the following another approach is used. As in the case of Anderson mixing, the initial guess is set to $J_0 = -\beta I$. Since the Jacobian is invertible it seems more reasonable to start with an approximation which is also invertible.

The inverse generalized Broyden method allows for a further alternative to the aforementioned initial matrices. With $J_0 = 0$, the method is equal to the *quasi-Newton least square* (QNLS) method, described in [14]. Again, the author employed the update formula (5.31) and for the approximation they used

$$J_k = W \left(V^T V\right)^{-1} V^T. \tag{5.34}$$

This shows that the LSQN method is also a special case of the inverse generalized Broyden's method.

The corresponding generalized Broyden method creates iterates x_k only within the two dimensional space $\text{span}\{x_0, x_1\}$. Therefore it cannot converge for any x_* not in the linear span. This follows from induction.

Let x_0 and x_1 be the first two iterates of the generalized Broyden algorithm 1 with $m = \infty$ and $J_0 = 0$. Then, for the second iteration it holds that

$$x_2 = x_1 + W\alpha = x_1 + \alpha(x_1 - x_0) \in \text{span}\{x_0, x_1\}, \tag{5.35}$$

with some scalar $\alpha \in \mathbb{R}^N$. For $k > 2$ assume that $x_i \in \text{span}\{x_0, x_1\}$, $i = 0, \ldots, k$. The image of W is a subspace of $\text{span}\{x_0, x_1\}$, since Δx_i, $i = 0, \ldots, k-1$, are a linear combination of x_0 and x_1. Thus, the new iteration

$$x_{k+1} = x_k + \underbrace{W\alpha}_{\in \text{span}\{x_0, x_1\}}, \tag{5.36}$$

with arbitrary $\alpha \in \mathbb{R}^k$, also lies in the linear span of x_0 and x_1.

The inverse generalized Broyden method does not suffer from this deficiency. The intermediate value x_{k+1}, computed by (5.30), is always in the linear span of $y_0, \ldots, y_k$ but the new iterate $y_{k+1} = H(x_{k+1})$ is not generally in this subspace. Therefore, the search space does not stagnate in each iteration.

5.3 Implementation Details

This final section gives some notes on the actual implementation of the methods presented above. The generalized Broyden method with $m = 1$ can be simplified as seen before. For the case $m = \infty$, simplification are also possible.

The first thing to note is that the methods with $m = \infty$ do not explicitly require the matrix J_k at any point. The matrix vector product $J_k R(x_k)$ or $J_k \tilde{R}(x_k)$ is completely defined by the initial guess J_0 and the difference matrices V and W, compare (5.21).

In the generalized Broyden method and the inverse generalized Broyden method , the matrix vector multiplication involves the product

$$\alpha = (V^T V)^{-1} V^T z, \quad z \in \mathbb{R}^N, \tag{5.37}$$

with $V \in \mathbb{R}^{N \times k}$ and $\alpha \in \mathbb{R}^k$, which is the solution of the normal equation to $V\alpha = z$. This is a least square problem

$$||V\alpha - z||_2 \to \min \tag{5.38}$$

for $k < N$ and the most common solution technique for it is the column pivoting QR decomposition.

Near the fixpoint x_* the sequence x_k becomes almost linear dependent. Thus, the least square problem is ill conditioned for growing k. To compute a stable solution, the QR decomposition is regularized. The decomposition

$$VP = Q \cdot \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix} \tag{5.39}$$

is adjusted, such that R_{11} is the largest upper triangular block matrix with a condition number below a certain threshold. Then, the block R_{22} is set to zero. With an additionally orthogonal transformation

$$(R_{11}, R_{12})\tilde{Q} = (\tilde{R}_{11} 0), \tag{5.40}$$

the solution to the least square system (5.38) is

$$\alpha = P\tilde{Q} \begin{bmatrix} \tilde{R}_{11}^{-1} Q_1^T z \\ 0 \end{bmatrix}, \tag{5.41}$$

where Q_1 consists of the columns of Q corresponding to R_{11}.

With the solution to the least square problem α, the matrix vector multiplication with J_k reduces to

$$J_k R(x_k) = J_0 R(x_k) + (W - J_0 V)\alpha. \tag{5.42}$$

The computation of α needs $\mathcal{O}(k^2 d)$ operations and in the case $J_0 \in \{0, -\beta I\}$ the product of α and $(W - J_0 V)$ is done in $\mathcal{O}(kN)$. Thus, the overall complexity for one iteration of the quasi-Newton methods with $m = \infty$ and $J_0 \in \{0, -\beta\}$ is $\mathcal{O}(k^2 N)$. If the matrices V and W are defined as in the remark above, the QR decomposition from the previous quasi-Newton iteration can be updated with only one Givens rotation, see [28]. This results in a complexity of $\mathcal{O}(kN)$ for the methods mentioned above.

The choice $J_0 = J_*^n$ brings some additional cost. The matrix vector multiplication $(W - J_*^n V)\alpha$ and $J_*^n R(x_k)$ now require $\mathcal{O}(N^2)$ operations, since the matrix J_*^n is dense. Moreover, at the beginning of a new time step or at the end of the current time step, the approximated inverse Jacobian J_*^n must be computed explicitly. In this instance, the matrix product

$$Z = (V^T V)^{-1} V^T \tag{5.43}$$

cannot be considered as a least square problem. Instead the linear system

$$(V^T V) Z = V^T \tag{5.44}$$

needs to be solved. Since the matrix $V^T V$ is symmetric and positive definite if V has full rank, the Cholesky decomposition of V can be aplied. The decomposition needs $\mathcal{O}(k^3)$ operations, which is feasible for small $k << m$. The overall complexity for the solution is $\mathcal{O}(k^3 + k^2 N)$. An alternative solution approach for the linear system is to use the QR decomposition of V again, which exhibited stability issues over the course of this thesis and therefore was disregarded.

Convergence Improvements

After a large number of time steps, the matrix J_*^n tends to be a poor initial guess. It incorporates information from all previous time steps, which can become inconsistent. The quasi-Newton methods without $J_0 = J_*^n$ can exhibit the same problem, if the number of quasi-Newton iterations grows large. In such cases the quasi-Newton method needs to restart.

It is difficult to predict, when a restart is necessary. If, within one time step, the residuum $R(x_k)$ increases by a factor of more than r, where $r > 0$ is a restart parameter, i.e.

$$||R_k|| > r\,||R_{k-1}||\,, \tag{5.45}$$

this indicates that the approximation J_k is not suitable anymore. The parameter is chosen large enough, that restarts only happen rarely. Furthermore, if the residuum decreases in every inner iteration for a small amount of time steps, a second restart criteria can be implemented. In this case, after two consecutive iterations without decreasing residuum, i.e.

$$\min\{||R_k||\,, ||R_{k-1}||\} > ||R_{k-2}||\,, \tag{5.46}$$

a restart is appropriate. If the convergence is rather slow and the residuum is not decreasing for each iteration even for a low number of time steps, this restarting criteria is unsuitable.

For a restart, all iterates are discarded, except either iterates $k-1$ or $k-2$ respectively, the last iterations with a reasonable decrease of the residuum. Then, this iterate is the new initial vector for the quasi-Newton method. Additionally, the initial guess is set to $-\beta I$ or 0 again.

Another convergence enhancing technique is the damping of the new search direction $-J_k R(x_k)$. This can even guarantee global convergence. The cost for that are several evaluations of H to find a fitting damping parameter. Therefore, these methods are not considered further.

5.4 Integration into MpCCI

To apply these quasi-Newton methods to fluid-structure interactions, they must be implemented into a coupling software package. MpCCI offers the possibility to couple, explicitly or implicitly, a wide range of commercial and non-commercial software packages. The architecture of MpCCI consists of one coupling server and two separate code adapters, for each solver individually. This is depicted in figure 9. The code adapters communicate with the solver and then send data to or receive data from the server. The server interpolates received data to make it compatible with the mesh of the target solver.

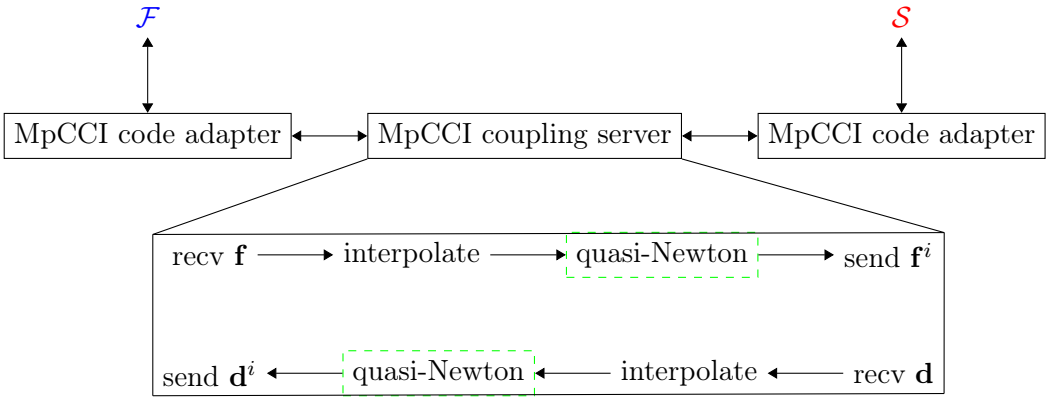


Figure 9: Simplified architecture of MpCCI with newly implemented quasi-Newton step.

The quasi-Newton methods implemented into the coupling server of MpCCI over the course of this thesis are summarized in table 1. After the interpolation of the received data, one quasi-Newton step is performed. In the case $m = \infty$, this quasi-Newton step consists of the following parts:

1. if $J_0 = J_*^n$, compute $J_*^n = J_*^{n-1} + (W - J_*^{n-1}V)(V^TV)^{-1}V^T$ at the start of a new time step,

2. save the current value of H, for example $y_k = \mathbf{f}^i$, and compute $R_k = x_k - y_k$,

3. update the difference matrices V and W according to the remark in section 5.1,

4. solve the least square problem $||V\alpha + R_k||_2$ and

5. compute the new step $x_{k+1} = x_k - J_0 R_k - (W - J_0 V)\alpha$ or
   $x_{k+1} = y_k - J_0 R_k - (W - J_0 V)\alpha$ for inverse methods.

The least square problem $||V\alpha + R_K||$ and the linear system $(V^T V)^{-1} V^T$ are solved using the techniques described in section 5.3. For the case $m = 1$, the update for x_{k+1} and J_k simplifies to (5.26), as mentioned earlier. The concrete implementation of the linear algebra methods uses the LAPACK library, version 3.6.0 [2].

	GB(J_k)	GB($-\beta I$)	I-GB($-\beta I$)	I-GB(0)	GB(J_*^n)	I-GB(J_*^n)
m	1	∞	∞	∞	∞	∞
J_0	J_*^n	$-\beta I$	$-\beta I$	0	J_*^n	J_*^n
Fixpoint problem	H	H	H^{-1}	H^{-1}	H	H^{-1}

Table 1: Comparison of implemented quasi-Newton methods.

Currently, the quasi-Newton methods can only handle inner iterations using the sequential coupling schemes. The parallel coupling scheme needs both quantities available in the same place to compute the residuum. This is not possible in the current version of MpCCI, since between the interpolation and the send command of one quantity the other one is not needed. In [54] and [38] the authors showed an clear advantage of the sequential coupling methods compared to the parallel ones. Thus, enabling the possibility of parallel coupling was not a priority.

6 Numerical Investigation

This chapter applies the quasi-Newton methods from the previous chapter applied to fluid-structure interactions. The fluid problems are solved by the commercial software package FLUENT. For the solution of the structural problems, the commercial software package ABAQUS is employed. Both solvers are coupled by the software package MpCCI. The parameter β, appearing in the definition of the initial matrix, is always set to 0.1. A in detail analysis of this parameter is omitted, since it would entail testing the numerical examples with a large number of possible choices. The quasi-Newton methods use the fixpoint function $H = \mathcal{F} \circ \mathcal{S}$.

The convergence of the quasi-Newton methods above is examined. The quasi-Newton methods use exactly one solution to both sub problems in one iteration, thus its performance determines the computational effort of the whole simulation. Furthermore, it is compared with Aitken's method, the solver for implicit coupling currently used by MpCCI. The first example additionally explores the effects of the added mass on the implicit coupling.

6.1 Driven Cavity with Moving Bottom

The first example is an extension of the classic 2D driven cavity model, where the bottom is not fixed. The oscillating movement of the top induces currents within the fluid, which in turns results in over- or underpressure on the bottom. This creates the motion of the bottom. Figure 10 shows the geometry for the example. The edges are 1m long and the slits are 0.2m wide.

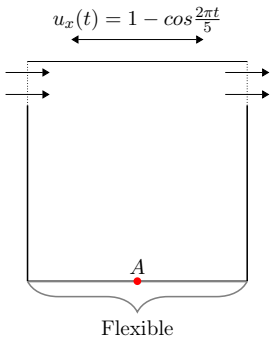


Figure 10: Geometry of the driven cavity example with flexible bottom.

No slip boundary conditions are imposed on the vertical walls for the fluid. The motion of the bottom changes the area of the geometry and therefore a pressure inlet and outlet near the top are needed, since the fluid is incompressible. The structure is fixed on both ends with zero stress on the downward facing boundary.

The fluid density is set to $\rho_f = 1\text{kg/m}^3$ and the viscosity to $\nu_f = 0.01\text{kg/ms}$. The structural density varies from $\rho_s = 25\text{kg/m}^2$ to 500kg/m^3 with a Young's modulus $E =$

250N/m^2 and Poisson's ratio $\nu_s = 0$. The low structural stiffness leads to a strong added mass effect.

The fluid domain is discretized with 24×24 rectilinear cells. The Navier-Stokes equations (3.14) are solved with the backwards Euler formula in a coupled manner. The structure discretization uses three layers of 80 bilinear elements each. For time integration, the HHT-α method with $\alpha = -0.005$ is used.

Stability Results

To illustrate the instabilities stemming from the added mass effect, the fluid and structure solvers are coupled explicitly, with the Gauß-Seidel scheme $H = \mathcal{F} \circ \mathcal{S}$. Figure 11 shows the displacement of the reference node A. The simulation is stable on the interval $[0, 40\text{s}]$ with structural density $\rho_s = 500\text{kg/m}^3$ and time step size $\Delta t = 0.1\text{s}$. Reducing the density to $\rho_s = 250\text{kg/m}^2$ results in a diverging simulation after about 10s. The computation is not stabilised by reducing the time step size to $\Delta t = 0.05\text{s}$, instead with this time step the previous stable simulation becomes unstable after 20s.

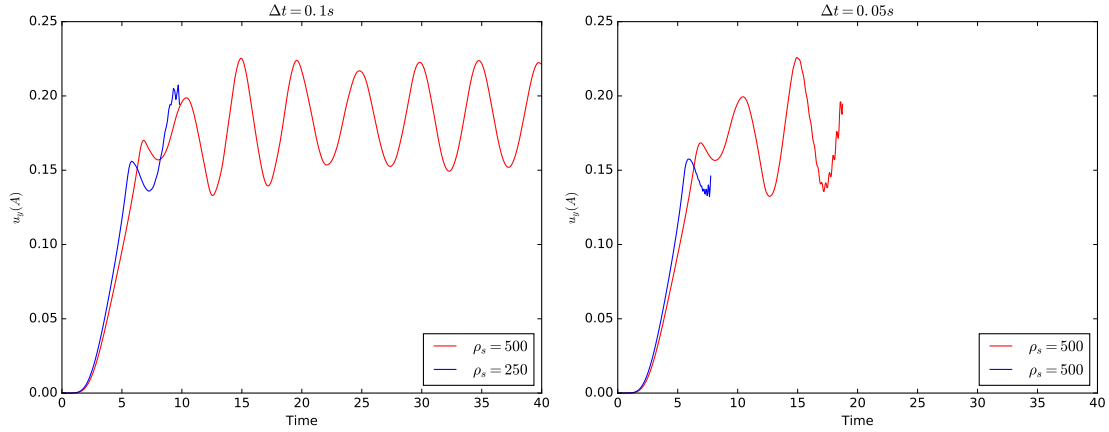
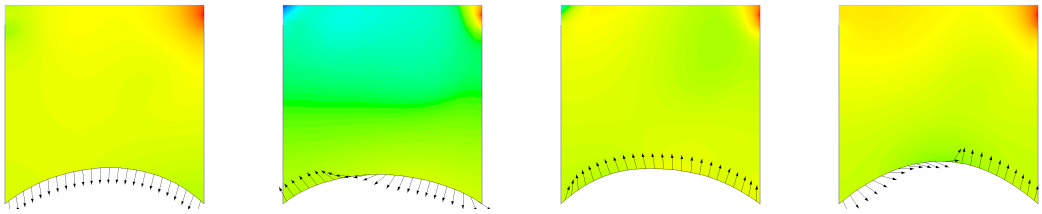


Figure 11: Displacement of the point A computed with explicit coupling and different parameters.

In contrast, implicit coupling, with any of the quasi-Newton methods named above, is stable for the previous stated combinations of structural density and time step size. The figure 12 depicts the solution for the parameters $\rho_s = 250\text{kg/m}^3$ and $\Delta t = 0.05\text{s}$, computed with the quasi-Newton method I-GB(J_*^n). This shows that the application of implicit coupling avoids the instabilities of the added mass effect. The disadvantage of these implicit coupling schemes is the higher computational cost. About six iterations of the I-GB(J_*^n) method are needed to achieve a relative error less than 10^{-7}, i.e.

$$\frac{||R_k||_\infty}{||x_k||_\infty + ||H(x_k)||_\infty} < 10^{-7}, \tag{6.1}$$

within each time step. Therefore, the quasi-Newton method is six times as expensive as a stable explicit coupling would be.



(a) $t = 30.9$s (b) $t = 32.15$s (c) $t = 33.4$s (d) $t = 34.65$s

Figure 12: Pressure and velocity direction at the bottom of the solution with density $\rho_s = 250\text{kg/m}^3$ and $\Delta t = 0.05$s, computed by GB(J_*^n), at different times of one oscilation.

Although the low structural density does not result in divergence anymore, c.f. figure 12, it has a negative effect on the convergence of the quasi-Newton methods. The left graph in figure 13 shows the impact lower structural density has on the convergence of different quasi-Newton methods with time step size $\Delta t = 0.1$s. The mean number of iterations to achieve the convergence criteria above rises with decreasing density ratio. Aitken's method is affected the most, its mean number of iterations is more than doubled for the lowest density $\rho_s = 25\text{kg/m}^3$. The quasi-Newton methods require around $1.6 - 1.7$ times as many iterations on the lowest density setting compared to the simulation with the highest density. This indicates that the regularities of the fixpoint problem are reduced by decreasing the density ratio.

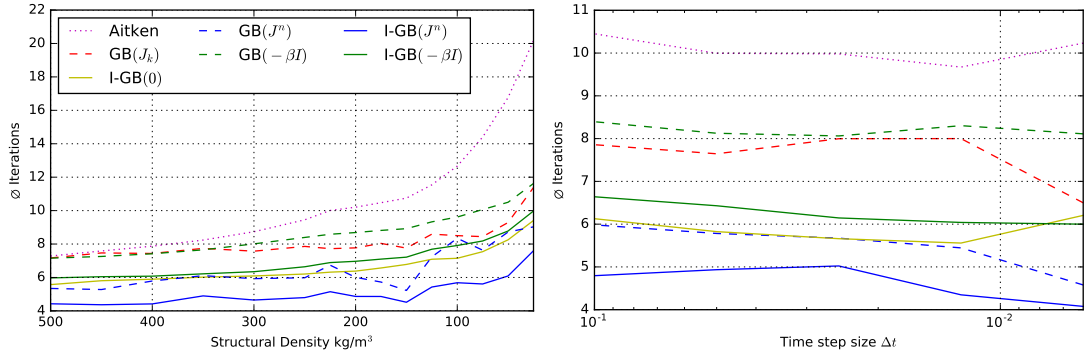


Figure 13: Mean number of iterations until convergence for varying structural densities with constant time step $\Delta t = 0.1$s on the left. The right shows the mean number of iterations for variing time step sizes with density set to $\rho_s = 250\text{kg/m}^3$.

As seen in figure 11, lowering the time step size has a destabilizing effect on the explicit simulation. This does not hold for implicit coupling. Furthermore, the time step size has only negligible influence on the convergence of the implicit coupling schemes, as shown in the right graph in figure 13. The only negative effect of a small step size is the larger amount of time steps needed to complete the simulation, which increases the probability of restarting the quasi-Newton methods. In the second example 6.2 the effects of restarting a quasi-Newton iteration are discussed in more detail.

Performance Results

The first graph in figure 13 shows a clear advantage of the quasi-Newton methods over the Aitken's method with respect to their performance. For every considered density, all quasi-Newton methods are faster than Aitken's method on average, with a speed up of $1.7 - 2.6$ for the lowest structural density. The inverse generalized Broyden method I-GB(J_*^n) with initial guess $J_0 = J_*^n$ and $m = \infty$ shows the best convergence overall.

Next, the influences of the different parameters of the quasi-Newton method, the initial guess J_0, the number of secant equations m and the choice of residuum, R or $\tilde{R}$, are examined. For this purpose, the simulation with structural density $\rho_s = 150\text{kg/m}^3$ and $\Delta t = 0.1\text{s}$ is looked at more thoroughly.

To show the impact of the initial matrix, Broyden's method with $J_0 = -\beta I$ and $J_0 = J_*^n$ are compared. The number of iterations to achieve convergence (6.1) for each time step is displayed in figure 14. The convergence speed of the Broyden method without reuse of information from earlier time steps is independent of the current time step. This method usually requires $12 - 13$ iteration until convergence. In contrast, the reuse of the approximated inverse Jacobi matrix leads to convergence within 10 iterations or less after 100 time steps. The number of iterations decreases even further such that after 200 time steps only $5 - 6$ iterations are sufficient.

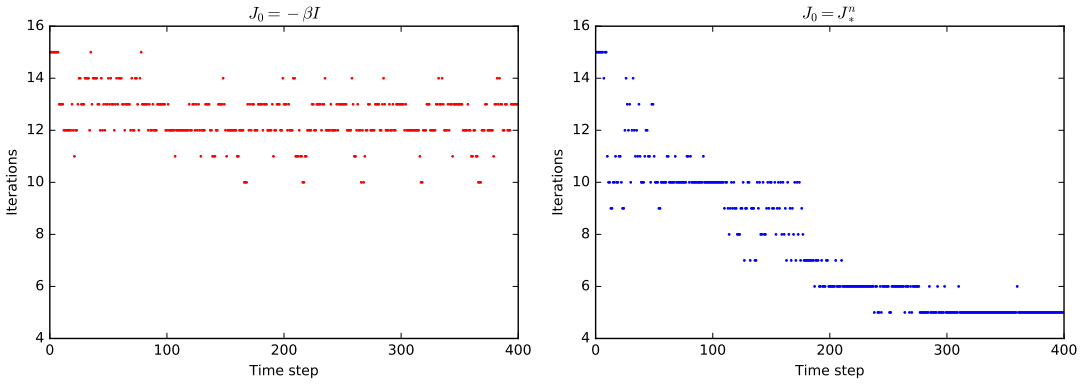


Figure 14: Number of iterations until convergence within each time step for the simulation with density $\rho_s = 150\text{kg/m}^3$ and time step $\Delta t = 0.1\text{s}$. Both methods use the standard residuum and only one secant equations. The graph on the left uses the method with $J_0 = -\beta I$, while on the right the method reuses information through $J_0 = J_*^n$.

This shows that with the reuse of information from previous time steps a speed-up of factor 2 can be achieved, after enough time steps. A further comparison between GB($-\beta I$) and GB(J_*^n) and between their inverse variants I-GB($-\beta I$) and I-GB(J_*^n), as seen in figure 13, support this notion. In all cases the version with initial guess $J_0 = J_*^n$ is $1.3 - 1.7$ times faster than the version with $J_0 = -\beta I$. It should be noted that a too large amount of time steps can slow down convergence. This can be seen in the later example, which uses a large amount of time steps.

Comparing Broyden's method with $J_0 = -\beta I$ and GB($-\beta I$) highlights the influence

of the number of secant equations m. Broyden's method uses one secant equation, while Anderson's mixing incorporates all available equations. Figure 15 shows that the method using all secant equations needs $3-4$ less iterations to converge, which amounts to a 1.4 times faster simulation. The comparison of GB(J_k) and GB(J_*^n) in figure 13 reinforce this result with a speed-up of $1.1-1.4$. Since, in all cases, the number of iterations until convergence is relatively small, a different behavior might be observed for a larger number of quasi-Newton iterations, see [20].

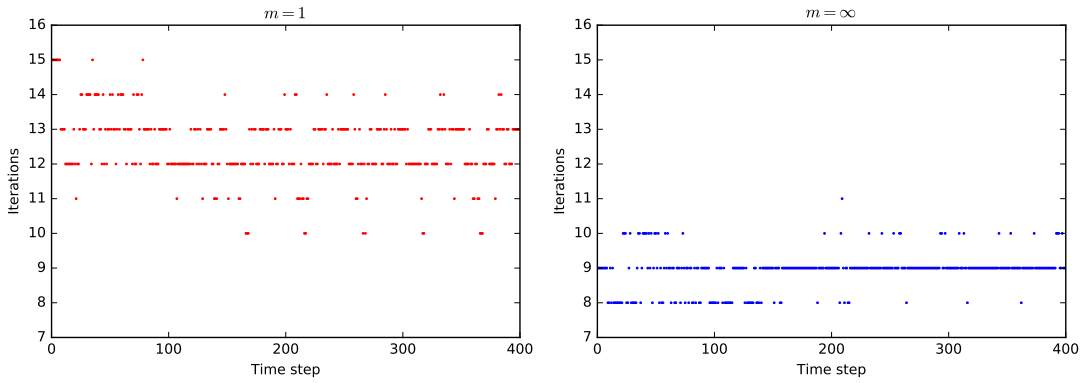


Figure 15: Number of iterations until convergence within each time step for the simulation with density $\rho_s = 150\text{kg/m}^3$ and time step $\Delta t = 0.1\text{s}$. Both methods use the standard residuum and the initial guess $J_0 = -\beta I$. The graph on the left uses the method with only one secant equation, while on the right the method uses all available secant equations.

Lastly, the impact of the choice of residuum, $R(x) = H(x) - x$ or $\tilde{R}(y) = y - H^{-1}(y)$, is considered in figure 16. Both methods, GB(J_*^n) and I-GB(J_*^n), accelerate their convergence over multiple time steps until 4 quasi-Newton iterations are sufficient. The inverse method achieves this about 100 time steps earlier than the generalized Broyden method. Additionally, within the first 100 time step I-GB(J_*^n) converges considerably faster than GB(J_*^n). Overall, the inverse generalized Broyden method is around 1.3 times faster, which can also be seen from comparing GB($-\beta I$) and I-GB($-\beta I$) in figure 13.

6.2 FSI Benchmark by Turek

The second numerical example is the benchmark proposed by Turek et al. [52], which is used to evaluate different fluid-structure interactions solvers. A wide range of different solution techniques has been applied to this problem, see [53]. This benchmark allows an examination of the performance of the quasi-Newton methods for a greater number of time steps than the driven cavity example. Figure 17 sketches the geometry of the model.

The channel is 2.5m long and 0.41m high. The cylinder with radius 0.05m is positioned around the point $(0.2m, 0.2m)$. The flexible beam has a length of 0.35m and a height of 0.02m.

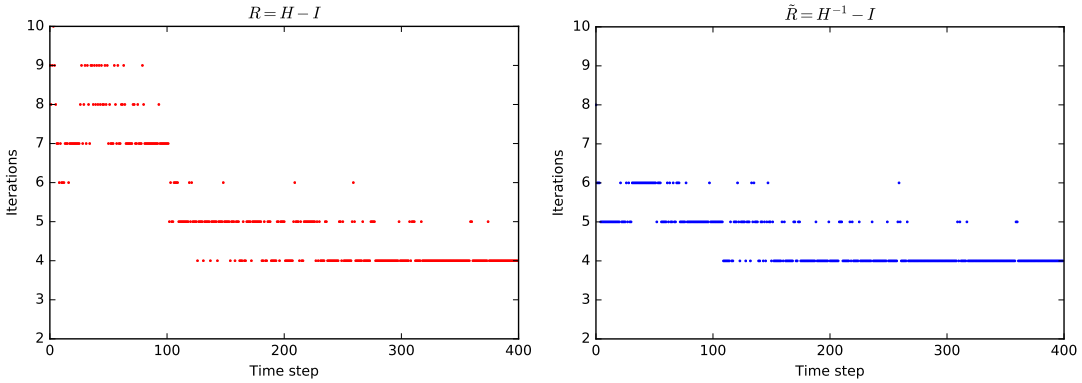


Figure 16: Number of iterations until convergence within each time step for the simulation with density $\rho_s = 150\text{kg/m}^3$ and time step $\Delta t = 0.1\text{s}$. Both methods use the inital guess $J_0 = J_*^n$ and all available secant equations. The graph on the left uses the method with the standard residuum $R = H - I$, while on the right the method employes the inverse residuum $\tilde{R} = I - H^{-1}$.

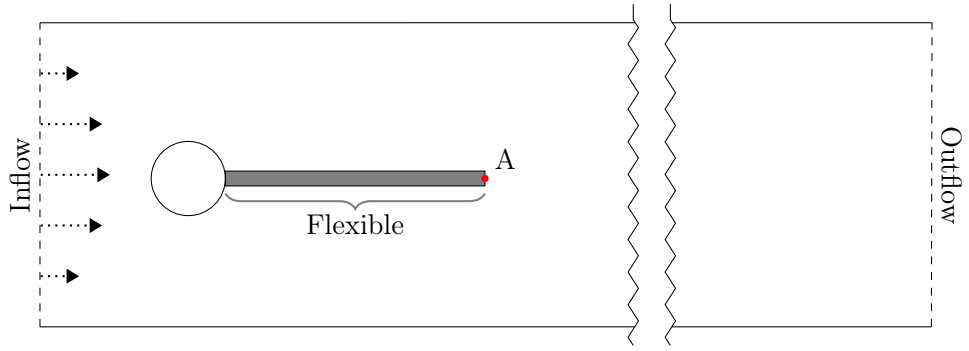


Figure 17: Geometry of the benchmark propsed by Turek [52].

The fluid model is incompressible and viscous with density $\rho_f = 1000\text{kg/m}^3$ and $\nu_f = 10^{-3}\text{kg/ms}$. The structure model uses a linear elastic material model and allows for large deformation. The Young's modulus is set to $E = 5.6 \cdot 10^6\text{kg/ms}^2$, the Poisson's ratio to $\nu_s = 0.4$ and the structural density to $\rho_s = 1000\text{kg/m}^3$. This leads to a strong added mass effect, although weakened by the high structural stiffness.

On the left channel side a velocity inlet with parabolic profile

$$v_p(0, y) = \frac{12}{0.41^2} y(0.41 - y) \tag{6.2}$$

is prescribed. The inlet velocity slowly increses until $t = 2\text{s}$, leading to the boundary condition

$$v_b(t, 0, y) = \begin{cases} \frac{1-\cos\left(\frac{\pi}{2}t\right)}{2} v_p(0, y) & \text{if } t < 2, \\ v_p(0, y) & \text{otherwise.} \end{cases} \tag{6.3}$$

On the right channel side outflow conditions are imposed and no slip conditions on the top and bottom boundary parts. The structure is fixed on the left side to the cylinder.

The fluid solver uses an unstructured mesh with 27361 cells. The time stepping method is again the backwards Euler method and the discretized continuity and conservation of momentum equation are solved by the SIMPLE algorithm. The structural domain is discretized with 5 layers of 70 bilinear elements, resulting in 568 nodes. The HHT-α method with $\alpha = -0.005$ is applied. For both sub problems the time step size is $\Delta t = 0.001$s.

Explicit coupling methods break down after around 20 time steps, see [39]. Implicit coupling with $H = \mathcal{F} \circ \mathcal{S}$ is stable over the interval $[0s, 10s]$ for any quasi-Newton method. The periodic currents in the fluid induce a periodic motion of the structure after around 4s. The motion of the structure is measured by the displacement of the point A, c.f. figure 17. Table 2 shows the oscillating movement of the point A computed by the quasi-Newton methods and Aitken's method and additionally the reference solution given by Turek et al. [52]. The first value in the second and third column display the mean value of the x and y displacement respectively and the second value shows the amplitude. The last two columns show the frequencies of the motions.

Additionally, figure 18 shows the x and y displacement for the last 0.5 seconds of the simulation, computed by GB(J_*^n). In figure 19 the magnitude of the velocity on the whole domain is given on the interval $[9.604s, 9.744s]$, showing one oscilation of the y motion.

Method	d_x(A) $\times 10^{-3}$	d_y(A) $\times 10^{-3}$	f_x	f_y
Reference	-2.68 ± 2.53	1.48 ± 34.38	10.9	5.3
Aitken	-2.07 ± 1.96	1.87 ± 29.56	10.8	5.4
GB($-\beta I$)	-2.05 ± 2.00	1.70 ± 30.55	10.8	5.4
GB(J^n)	-2.05 ± 2.00	1.70 ± 30.55	10.8	5.4
GB(J_k)	-2.05 ± 2.00	1.70 ± 30.55	10.8	5.4
I-GB($-\beta I$)	-2.05 ± 2.00	1.70 ± 30.55	10.8	5.4
I-GB(J^n)	-2.05 ± 2.00	1.70 ± 30.55	10.8	5.4
I-GB(0)	-2.05 ± 2.00	1.70 ± 30.55	10.8	5.4

Table 2: Comparison of the oscillating motion of the control point A. In the first and second column, the values $m \pm a$ denote the mean value m and the amplitude a of the oscilation. The frequencies are found in the last two columns

All quasi-Newton methods result in the same motion of A, nearly identical to the simulation with Aitken's method. This indicates that the quasi-Newton methods work correctly. The here computed simulations show a lower amplitude than the reference solution, which most likely stems from the spatial and time discretization of the fluid and structure. Because of temporal constraints, a simulation with finer meshes or time step sizes has not been done.

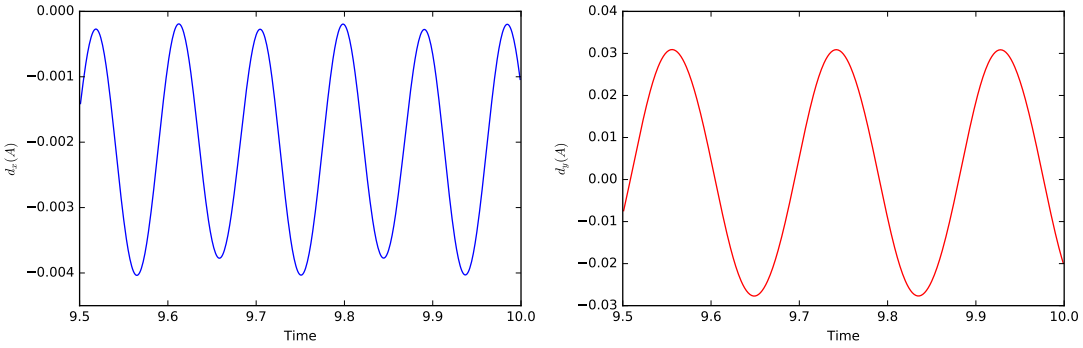


Figure 18: Displacement in x and y direction of the control point A, computed by I-GB(J_*^n)

Performance Results

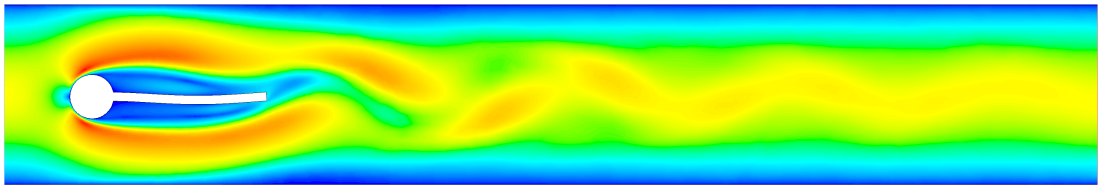
This part examines the performance of the quasi-Newton methods in detail. Table 3 gives an overview of the convergence of each method. The minimal, maximal and mean number of iterations to achieve convergence refer to the simulation after the oscillation is fully developed. The last column considers the whole simulation.

Method	MIN	MAX	MEAN	SUM
Aitken	19	50	25.86	251641
GB($-\beta I$)	13	19	15.88	147024
GB(J_*^n)	9	27	12.33	114834
GB(J_k)	7	39	9.52	87698
I-GB($-\beta I$)	13	15	14.18	132679
I-GB(J_*^n)	6	19	7.21	66254
I-GB(0)	12	13	12.77	119129

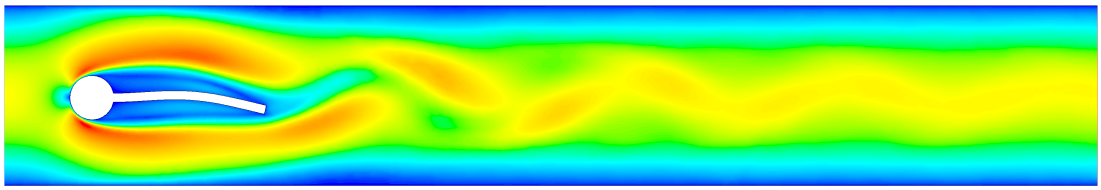
Table 3: Minimal, maximal and mean number of iterations until convergence within one time step for each method as well as the number of iteration for the whole simulation. The minimal, maximal and mean values are restricted to the time steps after the oscilation is fully developed.

All quasi-Newton are substantially faster than Aitken's method. The slowest quasi-Newton method GB($-\beta I$) requires ten iterations less than Aitken's method, on average. This leads to a speed-up of 1.7. Again, the best performance is achieved by I-GB(J_*^n), resulting in a speed-up of 3.7 compared to Aitken's method.

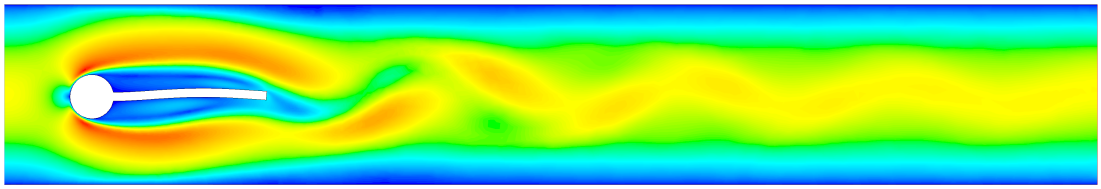
Broyden's method is the second fastest variant, while for the previous example it is the second slowest quasi-Newton method. It profits from the larger number of time steps, as already suggested. Furthermore, Broyden's method is faster than the GB(J_*^n) method, indicating an advantage of a lower number of secant equations. This conclusion is not as definitive as in the driven cavity example, since no other method with a low



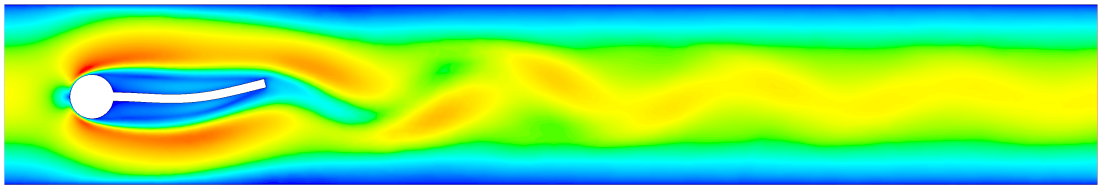
(a) $t = 9.604$s



(b) $t = 9.651$s



(c) $t = 9.697$s



(d) $t = 9.744$s

Figure 19: Velocity magnitude of the solution computed by I-GB(J_*^n) at different times of one y oscilation.

number of secant equation is considered.

Figure 23 allows for a closer comparison of all quasi-Newton methods. The methods in the left column use the standard residuum $R(x) = H(x) - x$ and the ones on the right use the inverse residuum $\tilde{R}(y) = y - H^{-1}(y)$. It becomes evident that the inverse generalized Broyden's methods converge faster than their generalized Broyden's counterparts.

The quasi-Newton methods without reusing information from earlier time steps show nearly no variation in the speed of convergence after the oscillation is fully developed. The least deviation is displayed by the GB(0) method.

In contrast, the methods with $J_0 = J_*^n$ exhibit clear peaks in their number of iterations until convergence. This is a crucial disadvantage of reusing information. Incorporating information of a too large number of time steps leads to an unsuitable initial guess. Thus the convergence speed decreases after a certain number of time steps. In the previous driven cavity example this is not a problem, since only 400 time steps are considered. Publications testing the quasi-Newton method I-GB(J_*^n) for similar problems do not report such behavior, see for example [49] or [5]. In these works, the I-GB(J_*^n) method shows a deviation from the mean value comparable to the deviation of the I-GB(0) method.

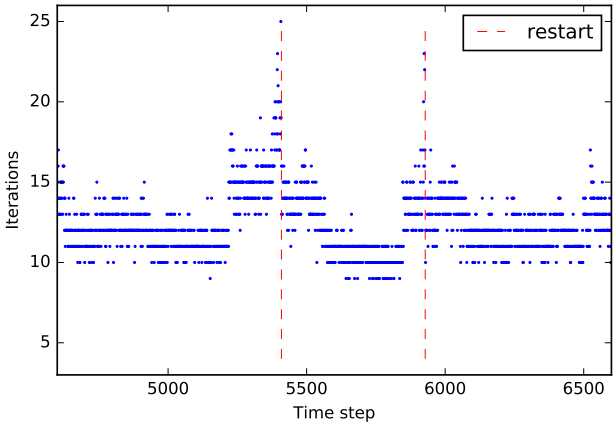
To obtain the same convergence speed as before, the quasi-Newton iterations are restarted, as explained in section 5.3. Figure 20 illustrates the effect of the restarts. It shows the convergence behavior for GB(J_k), GB(J_*^n) and I-GB(J_*^n) in detail for the time frame $[4.6s, 6.6s]$, marking time steps where a restart occurred. The required amount of iterations increases abruptly for all methods. This triggers the restarts, since either the residuum increases drastically or the method is not converging.

The I-GB(J_*^n) methods always restarts because of the not decreasing criteria (5.46). Both other methods restart because of an sudden increase in the residuum (5.45). As can be seen from the behavior of the GB(J_*^n) method in figure 20a, the criterions are not optimal. The number of iteration increases well before the criteria triggers the restart, worsening the performance.

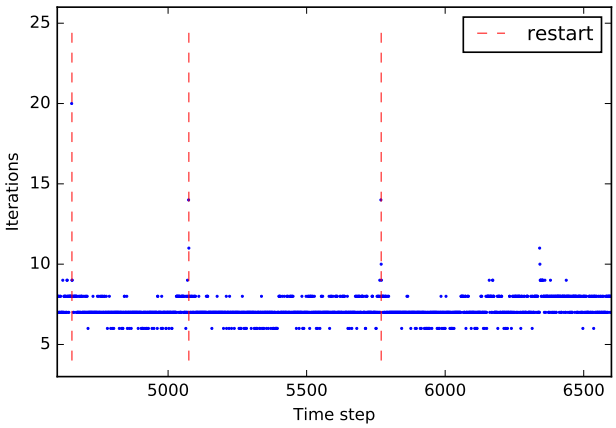
After the restarts, the convergence speed increases again and some time steps later, depending on the method, the convergence is as good as before. The inverse generalized Broyden method I-GB(J_*^n) has the same speed as before nearly immediately after restart, while the standard version GB(J_*^n) needs a couple of hundreds time steps to regain its convergence speed. The restarts appear to occur periodically with roughly 500 time steps between two restarts, although the exact distance depends on the method.

The number of time steps between two consecutive restarts is displayed in table 4, after the oscillation is build up. The shown distances lie in the range between $400 - 700$ time steps with some outliers, but they do not seem to follow any clear pattern. It is currently uncertain, if the distance between restarts depends solely on the method used, or additionally on the examined problem. Further examples with a large enough sample seize of time steps are needed to get a conclusive answer.

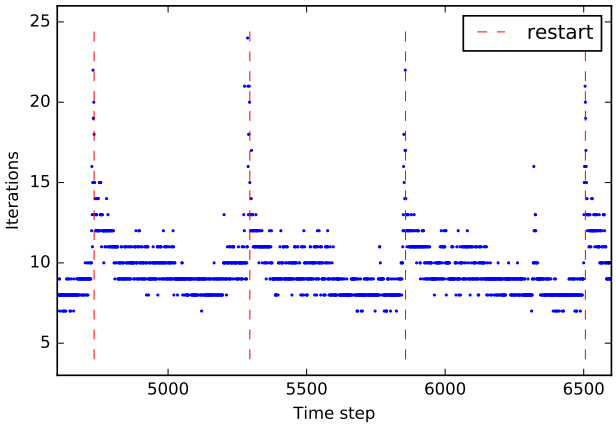
Lastly, table 5 shows the computational time needed for each quasi-Newton method. The MIN,MAX, and MEAN column refer to the time with the oscillating movement of the structure. Additionally the MAX column contains the iteration with maximal computa-



(a) GB(J_*^n)



(b) I-GB(J_*^n)



(c) GB(J_k)

Figure 20: Number of iterations until convergence on the interval $[4.6s, 6.6s]$ for the quasi-Newton methods GB(J_k), GB(J_*^n) and I-GB(J_*^n). Additionally, time steps with a restart of the iterations are marked.

	Distance between restarts								
GB(J_k)	562	562	649	738	708	600	477		
GB(J_*^n)	932	519	1026	1	48	428	114	618	546
I-GB(J_*^n)	422	694	1221	736	562	638	648	387	

Table 4: The number of time steps between two consecutive restarts, after the oscillation builds up.

tional time. The last column gives the overall computational cost of the quasi-Newton methods for the whole simulation. All times disregard the time for the evaluation of the sub solver.

Method	MIN	MAX $[k]$	MEAN	SUM
GB($-\beta I$)	0.07ms	0.48ms [14]	0.19ms	26.99s
GB(J_*^n)	0.32ms	0.90ms [24]	0.44ms	61.64s
GB(J_k)	0.51ms	0.68ms [8]	0.61ms	53.48s
I-GB($-\beta I$)	0.07ms	0.43ms [14]	0.17ms	21.49s
I-GB(J_*^n)	0.34ms	0.56ms [7]	0.37ms	24.86s
I-GB(0)	0.07ms	0.39ms [12]	0.15ms	17.67s

Table 5: Minimal, maximal and mean time consumption for one iteration within a time step for each method. The column with the maximal time consumption also displays which iteration took the longest. In addition the last column shows the whole time spent on each method for the whole simulation. Again, the minimal, maximal and mean values are restricted to the time steps after the oscilation is fully developed.

Broyden's method, which considers only one secant equation, has little variation in its computational cost. In contrast, the methods using all available secant equations increase their computational cost with each iteration. Since the number of iteration until convergence is quite small, the computation time stays below 1ms.

Matrix free methods, i.e. $J_0 \in \{0, -\beta I\}$, need the least amount of time, because they do not require any operation with a cost of $\mathcal{O}(N^2)$. The overall lowest computational cost has the I-GB(0) method, which converges faster and uses two vector additions less than the other matrix free methods.

The quasi-Newton methods with $J_0 = J_*^n$ have a higher base cost of around 0.3ms, since they always need two matrix vector multiplications with $N \times N$ matrices. The additional computational time from the increasing number of secant equations is comparatively low. It amounts only to a three times higher cost at worst. In contrast, matrix free methods increase their time until they need around five to seven times longer than the first iteration within a time step.

The computational cost for every quasi-Newton methods is minimal compared to the overall simulation. At worst, the quasi-Newton method needs around 1min for all iterations of the simulation combined, while the whole simulation time, with evaluation of both sub solvers, takes around 1.5 days to complete. This difference is due to the

fact that the interface is small compared to at least one sub problem. For problems with larger interfaces, in comparison to both sub problems, the computational cost of the quasi-Newton methods can become considerable. In this case, especially the methods with $J_0 = J_*^n$ are at disadvantage, since their cost depends quadratically on the degrees of freedom on the interface. This is explored in the following example.

6.3 Pressure Wave within Flexible Tube

For the third example, a 3D case with high discretization is examined. On the left side of a tube pressure is applied, leading to a wave traveling through the tube. The solution does not exhibit any oscillation like in the examples before. This example is described, for example, in [5].

The fluid is incompressible with density $\rho_f = 1000\text{kg/m}^3$ and viscosity $\nu_f = 6 \cdot 10^{-6}\text{kg/ms}$. The structure model uses a linear elastic material model with geometric nonlinearities. The density is set to $\rho_s = 1200\text{kg/m}^3$, the Young's modulus to $E = 3 \cdot 10^6\text{kg/ms}^2$ and the Possion's ratio to $\nu_s = 0.3$.

The tube is 50mm long in z direction with an inner radius of 5mm and an outer radius of 6mm. On the left inlet, a smooth pressure impuls is applied, i.e.

$$p_i(t) = \begin{cases} \frac{1-\cos\left(\frac{2\pi}{0.003}t\right)}{2} 1333 \cdot 10^4 & \text{if } t < 0.003, \\ 0 & \text{otherwise,} \end{cases} \tag{6.4}$$

and a pressure outlet is prescribed on the right end. The structure is fixed only at the inlet and outlet. Figure 21 shows the geometry of the tube.

Figure 21: Geometry of the pressure wave example.

The fluid discretization uses $200,000$ hexahedral cells. The structure domain is discretized with 60480 hexahedral 8-node elements, leading to 76230 degrees of freedom. The simulation is conducted over the interval $[0s, 0.01s]$. For both problems, the time step size $\Delta t = 10^{-4}$ is chosen. The same time stepping methods as in the previous examples are used. Since the structural discretization has 15246 nodes on the interface, the dimension of the fixpoint problem for this example is quite large.

Explicit coupling cannot produce a stable simulation for 100 time steps. All implemented quasi-Newton methods lead to a stable simulation. Figure 22 shows a cross section view of the pressure within the tube, computed by I-GB($-\beta I$).

Table 6 summarizes the performance of the quasi-Newton methods. Because of time constraints, Aitken's acceleration was not applied to the simulation. The table shows the minimal, maximal and mean number of iteration to achieve a relative residual less than 10^{-4}. These values do not consider the first eight time steps, since the methods converged considerably worse in these time steps, with some methods not converging within 25 iterations. Additionally, the number of quasi-Newton iterations needed for the whole simulation is displayed.

	MIN	MAX	MEAN	SUM
GB($-\beta I$)	10	19	12.09	1112
GB(J_*^n)	7	17	11.62	1069
GB(J_k)	9	18	11.80	1086
I-GB($-\beta I$)	11	16	11.93	1098
I-GB(J_*^n)	8	16	11.00	1012
I-GB(0)	11	16	11.64	1071

Table 6: Minimal, maximal and mean number of iterations until convergence within one time step for each method as well as the number of iteration for the whole simulation. The minimal, maximal and mean values disregard the first eight time steps, because all methods showed considerably worse convergence in those time steps.

All methods need nearly the same number of iterations until convergence. Over the whole simulation the differences accumulate only to 100 iterations between the fastest and the slowest method. Therefore, this example is not suitable to judge the performance of this method, although it still aligns with the results from the two previous examples. More time steps, or a higher proposed accuracy could lead to a more decisive comparison for this case.

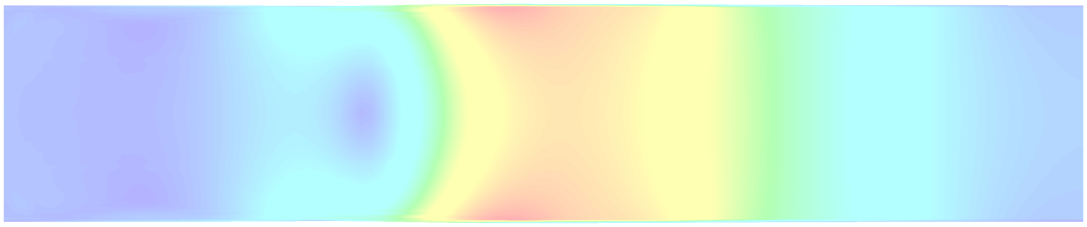
Even though the convergence results are not as informative as in the previous sections, this example illustrates clearly the advantage of the matrix free methods, their low computational cost. In table 7, the minimal, maximal and mean computation time for one iteration can be found. Again, the first eight time steps are disregarded, since all methods exhibit worse convergence in those time steps, and therefore they are not representative. Additionally, the time spent on quasi-Newton methods is displayed in the last column.

The matrix free method, i.e. methods with $J_0 \in \{0, -\beta I\}$, are at least about 50 times faster than the methods with $J_0 = J_*^n$ in their computation. Over the course of the simulation, the quasi-Newton methods GB($-\beta I$), I-GB($-\beta I$), and I-GB(0) need $4-5$ seconds for the computation of the quasi-Newton steps. The other methods require at least 10 minutes computation time.

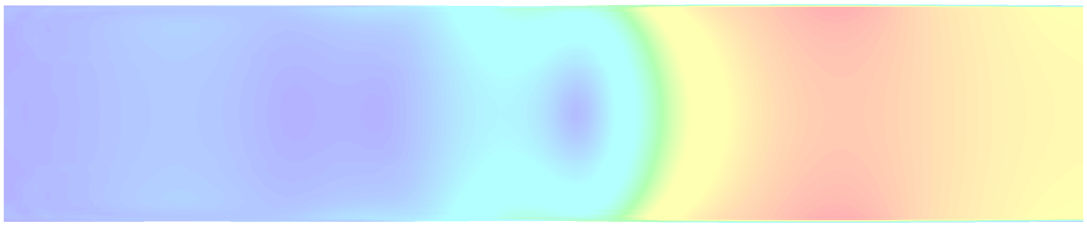
This significant difference is due to the matrix vector multiplication with a dense $N \times N$ matrix, needed by GB(J_k), GB(J_*^n), and I-GB(J_*^n). The matrix free methods

(a) $t = 0.0025$s

(b) $t = 0.005$s



(c) $t = 0.0075$s



(d) $t = 0.01$s

Figure 22: Cross section view of the pressure within the tube.

Method	MIN	MAX $[k]$	MEAN	SUM
GB($-\beta I$)	0.13ms	11.03ms [18]	2.85ms	4.91s
GB(J_*^n)	452.33ms	478.71ms [11]	459.89ms	680.21s
GB(J_k)	962.89ms	1209.61ms [3]	977.99ms	1147.50s
I-GB($-\beta I$)	0.13ms	8.70ms [15]	2.77ms	4.32s
I-GB(J_*^n)	452.15ms	484.81ms [2]	459.89ms	651.14s
I-GB(0)	0.12ms	8.68ms [15]	2.66ms	4.29s

Table 7: Minimal, maximal and mean time consumption for one iteration within a time step for each method. The column with the maximal time consumption also displays which iteration took the longest. In addition the last column shows the whole time spent on each method for the whole simulation. Again, the minimal, maximal and mean values disregard the first eight time steps, because all methods showed considerably worse convergence in those time steps.

use only $N \times k$ matrices, where k is the number of iterations, resulting in a much faster computation for larger N. Broyden's method GB(J_k) needs twice as long as GB(J_*^n) and I-GB(J_*^n), because of an inefficient implementation. Since the whole simulation needs about 21 hours to complete, the $10 - 20$ minutes spent on the quasi-Newton methods is still negligible.

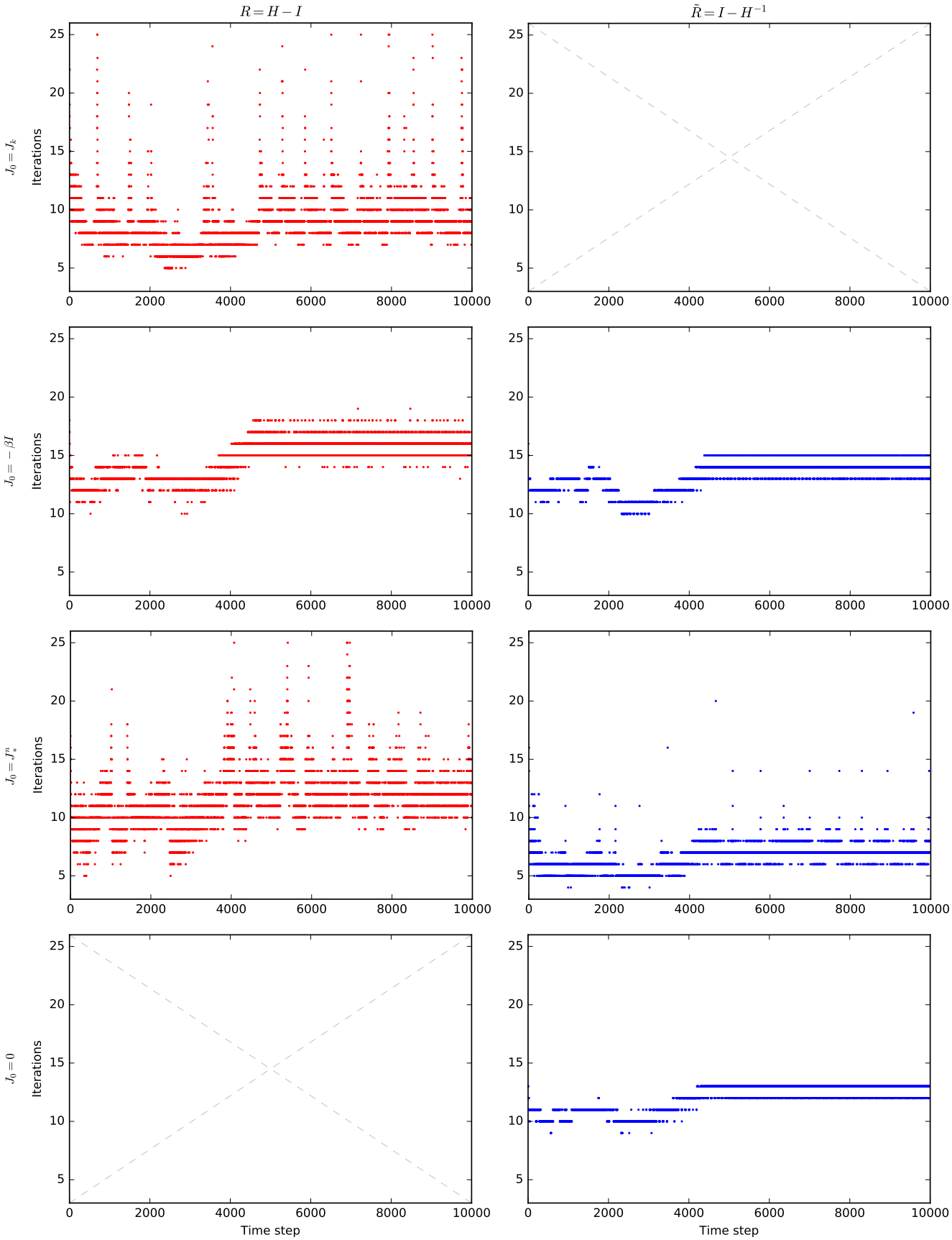


Figure 23: Number of iterations until convergence for each time steps and all considered quasi-Newton methods.

7 Conclusion

This work was aimed to solve fluid-structure interactions efficiently. The partitioned approach split the problem into a fluid and structure sub problem, where the interaction is modeled through appropriate boundary conditions. This allowed to employ already existing solution techniques to each sub problem separately.

The solver for each sub problem could be coupled explicitly, only one data exchange per time step, or implicitly, multiple data exchanges for one time step. The explicit coupling method was simpler and more efficient, but led to unstable simulations for incompressible fluids. These instabilities were caused by the added mass effect. A stable simulation was achieved by using implicit coupling, at a higher computational cost.

To reduce the cost of the implicit coupling, generalized Broyden's methods were examined. Furthermore, this work introduced the 'inverse' generalized Broyden's method based on the inverse of the fixpoint equation appearing in implicit coupling. The known quasi-Newton methods for implicit coupling, least square quasi-Newton and multi-vector update quasi-Newton, were shown to be special cases of the inverse generalized Broyden's method, allowing for comparison with different cases of generalized Broyden's method.

A numerical example showed that the implicit coupling with quasi-Newton methods still has remnants from the added mass effect. Although the computation were stable, decreasing the density ratio led to slower convergence, irregardless of the quasi-Newton method employed. However, reducing the time step size had no negative effect on the convergence.

Furthermore, the performance of the selected quasi-Newton methods was examined. Overall, the inverse generalized Broyden's methods exhibited better convergence than their standard counterparts. Additionally, the reuse of information from previous time steps, through the initial guess for the approximation of the Jacobian, proved to be advantageous. For simulations with a large number of time steps, this reuse must be treated carefully, by eventually restarting the time step, to avoid divergence.

The introduction of the inverse generalized Broyden's method leaves some questions open. The approximate Jacobian was constructed incorporating all available secant equations within one time step. Other choices for the number of secant equation were not considered, leaving the possibility open for better convergence with a certain number of secant equations. Since this number is most likely problem dependent, considering methods to adaptively choose the number of secant equations may be worthwhile, even for the generalized Broyden's method.

The initial guess for the approximate Jacobian plays an integral role in the convergence of the quasi-Newton methods. Therefore, new ways to deduce a initial matrix could lead to a better convergence. For example, variables like displacement or velocity are often extrapolated for the next time step, and thus similar approaches to the approximate Jacobi matrix could be possible.

As a final remark, the established quasi-Newton methods implement only indirect the fact that the fixpoint problem is part of a time stepping method. Thus, more sophisticated ways to integrate the time dependency would be an interesting field of research.

References

[1] Donald G Anderson. Iterative procedures for nonlinear integral equations. *Journal of the ACM (JACM)*, 12(4):547–560, 1965.

[2] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999.

[3] George Keith Batchelor. *An introduction to fluid dynamics*. Cambridge university press, 2000.

[4] John T Batina. Unsteady euler airfoil solutions using unstructured dynamic meshes. *AIAA journal*, 28(8):1381–1388, 1990.

[5] Alfred Edward Jules Bogaers, S Kok, BD Reddy, and T Franz. Quasi-newton methods for implicit black-box fsi coupling. *Computer Methods in Applied Mechanics and Engineering*, 279:113–132, 2014.

[6] Dietrich Braess. *Finite elements: Theory, fast solvers, and applications in solid mechanics*. Cambridge University Press, 2007.

[7] Charles G Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of computation*, 19(92):577–593, 1965.

[8] Charles George Broyden, John E Dennis, and Jorge J Moré. On the local and superlinear convergence of quasi-newton methods. *IMA Journal of Applied Mathematics*, 12(3):223–245, 1973.

[9] Paola Causin, Jean-Frédéric Gerbeau, and Fabio Nobile. Added-mass effect in the design of partitioned algorithms for fluid–structure problems. *Computer methods in applied mechanics and engineering*, 194(42):4506–4527, 2005.

[10] Juan Raul Cebral and Rainald Lohner. Conservative load projection and tracking for fluid-structure problems. *AIAA journal*, 35(4):687–692, 1997.

[11] Alexandre Joel Chorin, Jerrold E Marsden, and Jerrold E Marsden. *A mathematical introduction to fluid mechanics*, volume 3. Springer, 1990.

[12] Philippe G. Ciarlet. The finite element method for elliptic problems. Studies in Mathematics and its Applications. Vol. 4. Amsterdam - New York - Oxford: North-Holland Publishing Company. XIX, 580 p.; (1978)., 1978.

[13] Philippe G. Ciarlet. *Mathematical elasticity. Volume I: Three-dimensional elasticity.* Amsterdam etc.: North-Holland, 1988.

[14] Joris Degroote, Klaus-Jürgen Bathe, and Jan Vierendeels. Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. *Computers & Structures*, 87(11):793–801, 2009.

[15] Peter Deuflhard and Folkmar Bornemann. *Gewöhnliche Differentialgleichungen.* Walter de Gruyter, 2013.

[16] Jean Donea and Antonio Huerta. *Finite element methods for flow problems.* John Wiley & Sons, 2003.

[17] Christof Eck, Harald Garcke, and Peter Knabner. *Mathematische Modellierung.* Springer-Verlag, 2008.

[18] Lawrence C. Evans. *Partial differential equations. 2nd ed.* Providence, RI: American Mathematical Society (AMS), 2nd ed. edition, 2010.

[19] V Eyert. A comparative study on methods for convergence acceleration of iterative vector sequences. *Journal of Computational Physics*, 124(2):271–285, 1996.

[20] Haw-ren Fang and Yousef Saad. Two classes of multisecant methods for nonlinear acceleration. *Numerical Linear Algebra with Applications*, 16(3):197–221, 2009.

[21] Charbel Farhat, Michael Lesoinne, and P Le Tallec. 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*, 157(1):95–114, 1998.

[22] Charbel Farhat, Michel Lesoinne, and Nathan Maman. Mixed explicit/implicit time integration of coupled aeroelastic problems: Three-field formulation, geometric conservation and distributed solution. *International Journal for Numerical Methods in Fluids*, 21(10):807–835, 1995.

[23] CA Felippa and KC Park. Staggered transient analysis procedures for coupled mechanical systems: formulation. *Computer Methods in Applied Mechanics and Engineering*, 24(1):61–111, 1980.

[24] Carlos A Felippa, KC Park, and Charbel Farhat. Partitioned analysis of coupled mechanical systems. *Computer methods in applied mechanics and engineering*, 190(24):3247–3270, 2001.

[25] Christiane Förster. Robust methods for fluid-structure interaction with stabilised finite elements. 2007.

[26] Christiane Förster, Wolfgang A Wall, and Ekkehard Ramm. Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows. *Computer methods in applied mechanics and engineering*, 196(7):1278–1293, 2007.

[27] Max D Gunzburger. *Finite element methods for viscous incompressible flows: a guide to theory, practice, and algorithms*. Elsevier, 2012.

[28] Sven Hammarling and Craig Lucas. Updating the qr factorization and the least squares problem. 2008.

[29] Martin Hanke-Bourgeois. *Grundlagen der numerischen Mathematik und des wissenschaftlichen Rechnens*. Springer, 2009.

[30] DG Holmes and SD Connell. *Solution of the 2D Navier-Stokes equations on unstructured adaptive grids*. American Institute of Aeronautics and Astronautics, 1989.

[31] Thomas JR Hughes. *The finite element method: linear static and dynamic finite element analysis*. Courier Corporation, 2012.

[32] Raad I Issa, AD Gosman, and AP Watkins. The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme. *Journal of Computational Physics*, 62(1):66–82, 1986.

[33] Bruno Koobus and Charbel Farhat. Second-order time-accurate and geometrically conservative implicit schemes for flow computations on unstructured dynamic meshes. *Computer Methods in Applied Mechanics and Engineering*, 170(1):103–129, 1999.

[34] Catherine Lacour and Yvon Maday. Two different approaches for matching nonconforming grids: the mortar element method and the feti method. *BIT Numerical Mathematics*, 37(3):720–738, 1997.

[35] Patrick Le Tallec and Jean Mouro. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering*, 190(24):3039–3067, 2001.

[36] Michel Lesoinne and Charbel Farhat. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Computer methods in applied mechanics and engineering*, 134(1):71–90, 1996.

[37] Randall J. Leveque. *Finite volume methods for hyperbolic problems*. Cambridge: Cambridge University Press, 2002.

[38] Florian Lindner, Miriam Mehl, Klaudius Scheufele, and Benjamin Uekermann. A comparison of various quasi-newton schemes for partitioned fluid-structure interaction. In *Proceedings of 6th International Conference on Computational Methods for Coupled Problems in Science and Engineering*, pages 1–12, 2015.

[39] Felix Lippold. Zur simulation von fluid-struktur-wechselwirkungen mit flexiblen kopplungsverfahren. 2010.

[40] Lawrence E Malvern. *Introduction to the Mechanics of a Continuous Medium*. Number Monograph. 1969.

[41] Jerrold E Marsden and Thomas JR Hughes. *Mathematical foundations of elasticity*. Courier Corporation, 1994.

[42] Hermann G Matthies and Jan Steindorf. Partitioned strong coupling algorithms for fluid–structure interaction. *Computers & Structures*, 81(8):805–812, 2003.

[43] Daniel Pinyen Mok. *Partitionierte Lösungsansätze in der Strukturdynamik und der Fluid-Struktur-Interaktion*. 2001.

[44] Suhas Patankar. *Numerical heat transfer and fluid flow*. CRC press, 1980.

[45] Suhas V Patankar and D Brian Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International journal of heat and mass transfer*, 15(10):1787–1806, 1972.

[46] Serge Piperno. 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*, 25(10):1207–1226, 1997.

[47] Serge Piperno and Charbel Farhat. *Design and evaluation of staggered partitioned procedures for fluid-structure interaction simulations*. PhD thesis, INRIA, 1997.

[48] Serge Piperno, Charbel Farhat, and Bernard Larrouturou. Partitioned procedures for the transient solution of coupled aroelastic problems part i: Model problem, theory and two-dimensional application. *Computer methods in applied mechanics and engineering*, 124(1):79–112, 1995.

[49] Klaudius Scheufele. Robust quasi-newton methods for partitioned fluid-structure simulations. 2015.

[50] Jan Steindorf. *Partitionierte Verfahren für Probleme der Fluid-Struktur-Wechselwirkung*. Mechanik-Zentrum, 2002.

[51] Alex Toth and CT Kelley. Convergence analysis for anderson acceleration. *SIAM Journal on Numerical Analysis*, 53(2):805–819, 2015.

[52] Stefan Turek and Jaroslav Hron. Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. In *Fluid-structure interaction*, pages 371–385. Springer Berlin Heidelberg, 2006.

[53] Stefan Turek, Jaroslav Hron, Mudassar Razzaq, Hilmar Wobker, and Michael Schäfer. Numerical benchmarking of fluid-structure interaction: A comparison of different discretization and solution approaches. In *Fluid Structure Interaction II*, pages 413–424. Springer Berlin Heidelberg, 2011.

[54] Benjamin Uekermann, Hans-Joachim Bungartz, Bernhard Gatzhammer, and Miriam Mehl. A parallel, black-box coupling algorithm for fluid-structure interaction. In *Proceedings of 5th International Conference on Computational Methods for Coupled Problems in Science and Engineering*, pages 1–12, 2013.

[55] EH Van Brummelen. Added mass effects of compressible and incompressible flows in fluid-structure interaction. *Journal of Applied mechanics*, 76(2):021206, 2009.

[56] Henk Kaarle Versteeg and Weeratunge Malalasekera. *An introduction to computational fluid dynamics: the finite volume method*. Pearson Education, 2007.

[57] Wolfgang A Wall. Fluid-struktur-interaktion mit stabilisierten finiten elementen. 1999.

[58] Pieter Wesseling. *Principles of computational fluid dynamics*, volume 29. Springer Science & Business Media, 2009.

[59] OC Zienkiewicz and AHC Chan. Coupled problems and their numerical solution. In *Advances in Computational Nonlinear Mechanics*, pages 139–176. Springer, 1989.